



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE

COMUNICACIÓN

TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE

INGENIERÍA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN

TEMA

SISTEMA BIOFEEDBACK DE SENSORES DE DETECCIÓN DE ESTRÉS

MEDIANTE ALGORITMOS DE APRENDIZAJE DE MÁQUINAS EN

ESTUDIANTES UNIVERSITARIOS

AUTOR: SAMIA CRISTINA TUTILLO ALCOCER

DIRECTOR: MSC. JAIME ROBERTO MICHILENA CALDERÓN

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	100398373-9
Apellidos y nombres	Tutillo Alcocer Samia Cristina
Dirección	Alpachaca, final de la calle Tungurahua.
E-mail	sctutilloa@utn.edu.ec
Teléfono móvil	0961423053
DATOS DE LA OBRA	
TÍTULO:	SISTEMA BIOFEEDBACK DE SENSORES DE DETECCIÓN DE ESTRÉS MEDIANTE ALGORITMOS DE APRENDIZAJE DE MÁQUINAS EN ESTUDIANTES UNIVERSITARIOS.
AUTOR:	Tutillo Alcocer Samia Cristina
FECHA	25 de octubre de 2021
SOLO PARA TRABAJOS DE GRADO	

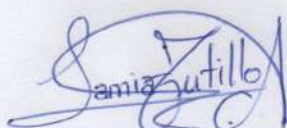
PROGRAMA	PREGRADO	POSGRADO
TITULO	Ingeniero en Electrónica y Redes de Comunicación	
DIRECTOR	Ing. Jaime Michilena Calderón, Msc.	

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 sume la responsabilidad sobre contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, 25 de octubre de 2021

EL AUTOR:



Samia Cristina Tutillo Alcocer

100398373-9



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

Mac. JAIME MICHILENA CALDERÓN, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA:

Que, el presente trabajo de Titulación "SISTEMA BIOFEEDBACK DE SENSORES DE DETECCIÓN DE ESTRÉS MEDIANTE ALGORITMOS DE APRENDIZAJE DE MÁQUINAS EN ESTUDIANTES UNIVERSITARIOS". Ha sido desarrollado por la Señorita Samia Cristina Tutillo Alcocer, bajo mi supervisión.

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

Msc. Jaime Michilena

CC: 100219843-8

DIRECTOR



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

El presente trabajo de titulación dedico principalmente a Dios quien guía cada uno de mis pasos y me ha dado la fuerza y sabiduría para culminar esta etapa de mi vida.

A mis padres Giovany y Leonor quienes han sido unos padres luchadores para brindarme lo que necesito y por ser un pilar fundamental en mi vida ya que siempre me han apoyado a cumplir con lo que me propongo brindándome sus consejos y sobre todo por la paciencia que han tenido conmigo.

A mi hermana Margarita por ser mi ejemplo, guiarme por el buen camino y por su ayuda incondicional al momento de realizar este trabajo de grado.

A mi hermano Sebastián para que vea el esfuerzo realizado y utilice el mejor ejemplo de sus hermanas y luche por sus sueños.

A mis tías Lulú y Carmita por su ayuda a lo largo de toda mi carrera universitaria ya que de una u otra forma cuando más necesitaba su apoyo siempre estuvieron para mí.

A ustedes y al resto de mi familia, les dedico con mucho cariño mi trabajo.

Samia Cristina Tutillo Alcocer.



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTOS

Quiero agradecer a Dios por su protección y ayuda para poder superar cada obstáculo que se me presenta en la vida y sobre todo por la fortaleza que me brindó en este camino. Agradezco a mis padres Giovany y Leonor por sus palabras de apoyo a cada instante, por su motivación para salir adelante y no dejarme vencer por las dificultades a lo largo de la carrera.

Gracias a mis hermanos Margarita y Sebastián por ser mis mejores amigos, alegrar mi día, sacarme una sonrisa en los momentos de aflicción y por su apoyo incondicional en todas mis decisiones.

A ti Jefferson por tu ayuda a culminar esta etapa, por estar pendiente de mí y por tus consejos que me ayudan a elegir el camino correcto.

Gracias a mis amigos y compañeros a lo largo de mi carrera por permitirme compartir valiosos momentos llenos de alegrías y apoyarnos en las buenas y en las malas.

Al Ingeniero Jaime Michilena quien fue mi director de tesis, por brindarme su tiempo y sus conocimientos para desarrollar de mejor manera este trabajo conjuntamente con los ingenieros Carlos Vásquez, Luis Suárez y Paul Rosero. A la Universidad Técnica del Norte simultáneamente con la carrera de Electrónica y Redes de comunicación por abrirme las puertas y permitirme formar ética y profesionalmente.

Samia Cristina Tujillo Aleccer.

RESUMEN

En este trabajo de titulación se desarrolla el sistema BIOSTRESS, este es un sistema *Biofeedback* de sensores que predice el nivel de estrés mediante algoritmos de aprendizaje de máquinas, el objetivo de este sistema es detectar estos niveles y ayudar a controlarlo mediante un posible tratamiento, sirviendo de apoyo para el trabajo que realizan los psicólogos.

El diseño del sistema se basó en la metodología denominada Modelo en V y se encuentra conformado principalmente por la placa *MySignals* ya que sirve de ayuda para desarrollar sistemas *e-Health*. Además, cuenta con sensores biomédicos para detectar los cambios fisiológicos de los estudiantes y así obtener su información, con ello, se continúa al procesamiento de los datos para determinar el nivel de estrés que posee el paciente, pudiendo visualizar mediante la interfaz de una aplicación desarrollada en el *software Processing*.

Finalmente, la funcionalidad del sistema es analizado mediante pruebas a un grupo de 10 estudiantes de diferentes carreras entre hombres y mujeres, lo que ayudó a lograr la factibilidad del sistema, puesto que se obtuvo diferentes resultados entre los cuales el 70% de los sujetos de prueba presentan un nivel de estrés diferente del normal y con la ayuda del *Biofeedback* estos niveles se pueden normalizar. BIOSTRESS sería de gran utilidad a los profesionales en el tema para que el diagnóstico de las personas sea mucho más preciso.

ABSTRACT

In this degree work the BIOSTRESS system is developed, this is a Biofeedback system of sensors that predicts the level of stress through machine learning algorithms, the objective of this system is to detect these levels and help control it through a possible treatment, serving support for the work carried out by psychologists.

The design of the system was based on the methodology called V-Model and is mainly made up of the MySignals board as it helps to develop e-Health systems. In addition, it has biomedical sensors to detect the physiological changes of the students and thus obtain their information, with this, the data processing is continued to determine the level of stress that the patient has, being able to visualize through the interface of a developed application in the Processing software.

Finally, the functionality of the system is analyzed through tests on a group of 10 students of different careers between men and women, which helped to achieve the feasibility of the system, since different results were obtained, among which 70% of the subjects of test present a different stress level than normal and with the help of Biofeedback these levels can be normalized. BIOSTRESS would be very useful to professionals in the field so that the diagnosis of people is much more accurate.

ÍNDICE DE CONTENIDOS

IDENTIFICACIÓN DE LA OBRA	II
CONSTANCIA	III
CERTIFICACIÓN.....	IV
DEDICATORIA.....	V
AGRADECIMIENTOS.....	VI
RESUMEN	VII
ABSTRACT	VIII
ÍNDICE DE CONTENIDOS.....	1
ÍNDICE DE FIGURAS	5
ÍNDICE DE TABLAS.....	10
1. CAPÍTULO I.....	11
1.1. Tema	11
1.2. Problema	11
1.3. Objetivos.....	13
1.3.1. Objetivo general	13
1.3.2. Objetivos específicos.....	13
1.4. Alcance	14
1.5. Justificación	16
2. CAPÍTULO II.....	18
2.1. Estrés.....	18
2.1.1. Causas.....	18
2.1.2. Efectos de Salud	19
2.1.3. Variables Fisiológicas	21
2.1.4. Estrés Estudiantil.....	22
2.2. Sistemas Embebidos	26

2.2.1.	Introducción.....	26
2.2.2.	Tipos	28
2.2.3.	Sensores	33
2.2.4.	MySignals.....	41
2.3.	Diseño del Sistema Electrónico	42
2.3.1.	Modelo en V	43
2.3.2.	IEEE 29148	44
2.3.3.	Diagramas de Bloque y Flujo	45
2.3.4.	Benchmark.....	47
2.4.	Biofeedback	47
2.4.1.	Introducción.....	47
2.4.2.	Funcionamiento	47
2.4.3.	Tipos	49
2.5.	Machine Learning	50
2.5.1.	Aprendizaje Supervisado	50
2.5.2.	Aprendizaje No Supervisado	51
2.5.3.	Aprendizaje Por Refuerzo	51
3.	CAPÍTULO III	52
3.1.	Metodología.....	52
3.2.	Análisis	52
3.2.1.	Situación Actual.	52
3.2.2.	Técnicas de investigación.....	54
3.3.	Requerimientos	57
3.3.1.	Stakeholders.	57
3.3.2.	Requerimientos de los Stakeholders.....	58
3.3.3.	Requerimientos del sistema.....	59
3.3.4.	Requerimientos de arquitectura.....	61

3.4.	Elección del Software y Hardware para el diseño	63
3.4.1.	Elección del Hardware.....	63
3.4.2.	Elección del Software.....	66
3.5.	Recursos.....	67
3.5.1.	Recursos Humanos.....	68
3.5.2.	Recursos Económicos.....	68
3.5.3.	Recursos Tecnológicos.....	69
3.6.	Diseño del sistema	69
3.6.1.	Descripción general del funcionamiento del sistema BIOSTRESS.....	70
3.6.2.	Arquitectura.....	70
3.7.	Diseño detallado del Sistema BIOSTRESS.....	75
3.7.1.	Diseño del bloque de alimentación.....	75
3.7.2.	Diseño del bloque de sensores.....	76
3.7.3.	Diseño del bloque de toma de datos y bloque de procesamiento.....	83
3.7.4.	Diseño del bloque de visualización.....	91
3.8.	Implementación	95
3.8.1.	Implementación del hardware.....	95
3.8.2.	Implementación del software.....	96
4.	CAPÍTULO IV	106
4.1.	Test unitarios.....	106
4.2.	Integración del sistema BIOSTRESS	108
4.3.	Pruebas.....	108
4.3.1.	Pruebas de hardware.....	109
4.3.2.	Pruebas con los usuarios.....	109
4.3.3.	Discusión de resultados.....	119
4.4.	Análisis del nivel de estrés en estudiantes	120
5.	CAPÍTULO V	122

5.1. Conclusiones	122
5.2. Recomendaciones	123
6. BIBLIOGRAFÍA	125
7. GLOSARIO	133
8. ANEXOS	135
ANEXO 1. Formato de Encuestas con Tabulación de los Resultados y Entrevista.	135
ANEXO 2. Programación de BIOSTRESS	148
ANEXO 3. Tabla del conjunto de datos de entrenamiento.	182
ANEXO 4. Fotografías del trabajo.	183
ANEXO 5. Manual de Usuario.....	185

ÍNDICE DE FIGURAS

Figura 1. Estructura de un sistema embebido. Adaptado de: (Wei & Wang, 2007)	28
Figura 2. Vista del sistema de tiempo real. Adaptado de: (Qing & Yao, 2013).....	30
Figura 3. Sistemas embebidos en tiempo real. Adaptado de: (Michel et al., 2016)	31
Figura 4. Estructura de un sistema de tiempo real. Adaptado de: (Bogdan et al., 2016) 32	
Figura 5. Sistema DSP. Adaptado de: (Zahradnik & Šimák, 2012).....	33
Figura 6. Sensor. Adaptado de: (Wilson, 2008)	34
Figura 7. Sensor análogo. Adaptado de: (Gorgues, 2017).....	34
Figura 8. Sensor digital. Adaptado de: (Gorgues, 2017).....	35
Figura 9. Gráfica de la señal $x(n)$	39
Figura 10. Gráfica de la señal $h(n)$	39
Figura 11. Convolución de las dos señales. Fuente: (Arar, 2017).....	39
Figura 12. Diagrama de flujo de los pasos para el diseño de sistemas integrados. Adaptado de: (Jansson & Schwabe, 2010)	40
Figura 13. Tarjeta de MySignals. Fuente: (Sarraiya et al., 2019)	42
Figura 14. Fases del modelo en V. Adaptado de: (Planning, 2017)	43
Figura 15. Ejemplo de un diagrama de flujo. Adaptado de: (Batovski, 2014).....	46
Figura 16. Ejemplo de un diagrama de bloques. Adaptado de: (Batovski, 2014).....	46
Figura 17. Funcionamiento del Biofeedback. Adaptado de: (Marín Agudelo & Vinaccia Alpi, 2005).....	48
Figura 18. Proceso de Biofeedback. Adaptado de: (Carrobbles, 2016)	48
Figura 19. Placa MySignals con sensores seleccionados. Fuente: Autoría basado en (Sarraiya et al., 2019)	66
Figura 20. Diagrama de Bloques de BIOSTRESS. Fuente: Autoría.	71
Figura 21. Diagrama de Flujo del proceso de BIOSTRESS. Fuente: Autoría.	74
Figura 22. Diseño de conexión del circuito de encendido del computador hacia la placa Arduino. Fuente: Autoría basado en (Sarraiya et al., 2019)	75
Figura 23. Diagrama de conexión de las placas. Fuente: Autoría basado en (Sarraiya et al., 2019).....	76
Figura 24. Diagrama de flujo del bloque de sensores. Fuente: Autoría.	77
Figura 25. Sensor de Flujo de aire. Fuente: (Sarraiya et al., 2019)	78
Figura 26. Posición adecuada del sensor Airflow. Fuente: (Sarraiya et al., 2019).....	78
Figura 27. Conexión del sensor Airflow. Fuente: Autoría basado en (Sarraiya et al., 2019).....	79

Figura 28. Sensor ECG. Fuente: Autoría.....	79
Figura 29. Posición adecuada del sensor ECG. Fuente: (Sarraipa et al., 2019)	80
Figura 30. Conexión del sensor ECG. Fuente: Autoría basado en (Sarraipa et al., 2019)	81
Figura 31. Sensor GSR. Fuente: (Libelium Comunicaciones, 2017)	81
Figura 32. Posición adecuada del sensor GSR. Fuente: (Libelium Comunicaciones, 2017).....	82
Figura 33. Conexión del sensor GSR. Fuente: Autoría basado en (Libelium Comunicaciones, 2017)	82
Figura 34. Esquema de conexión del sistema. Fuente: Autoría basado en (Sarraipa et al., 2019).....	83
Figura 35. Diagrama de flujo de la toma de datos del sistema BIOSTRESS. Fuente: Autoría.....	85
Figura 36. Diagrama de flujo del proceso del algoritmo KNN. Fuente: Autoría.....	87
Figura 37. Diagrama de bloques del algoritmo KNN. Fuente: Autoría.	88
Figura 38. Algoritmo KNN en Arduino. Fuente: Autoría.	89
Figura 39. Diagrama de bloques de la fase de visualización. Fuente: Autoría.....	91
Figura 40. Diagrama de flujo de la fase de visualización. Fuente: Autoría.	93
Figura 41. Proceso del Biofeedback. Fuente: Autoría.....	94
Figura 42. Ejercicio de respiración. Fuente: (Relaxing white noise, 2018)	95
Figura 43. Case del sistema BIOSTRESS. Fuente: Autoría.....	96
Figura 44. Implementación del Hardware. Fuente: Autoría.....	96
Figura 45. Código de la toma de datos del sistema BIOSTRESS. Fuente: Autoría.....	97
Figura 46. Obtención de datos. Fuente: Autoría.....	97
Figura 47. Correcta ubicación de los sensores al paciente. Fuente: Autoría.	99
Figura 48. Interfaz inicial de BIOSTRESS. Fuente: Autoría.	103
Figura 49. Interfaz de las mediciones de BIOSTRESS. Fuente: Autoría.....	104
Figura 50. Interfaz del estado del paciente. Fuente: Autoría.....	104
Figura 51. Interfaz del Biofeedback. Fuente: Autoría.	105
Figura 52. Prueba del sensor ECG. Fuente: Autoría.	106
Figura 53. Calibración de los sensores. Fuente: Autoría.....	107
Figura 54. Prueba de Funcionamiento de la interfaz. Fuente: Autoría.....	107
Figura 55. Vista externa del sistema BIOSTRESS. Fuente: Autoría.	108
Figura 56. Vista interna del sistema BIOSTRESS. Fuente: Autoría.	108

Figura 57. Colocación de los sensores al paciente. Fuente: Autoría.	111
Figura 58. Ubicación de los sensores al sistema. Fuente: Autoría.	111
Figura 59. Interfaz de la toma de datos del estado normal. Fuente: Autoría.	112
Figura 60. Nivel de estrés normal. Fuente: Autoría.	112
Figura 61. BDD del archivo csv del estado Normal. Fuente: Autoría.	113
Figura 62. Someter al paciente a estresores. Fuente: Autoría.	113
Figura 63. Interfaz de la toma de datos del estado algo elevado. Fuente: Autoría.	114
Figura 64. BDD del archivo csv del estado algo elevado. Fuente: Autoría.	114
Figura 65. Nivel de estrés algo elevado. Fuente: Autoría.	115
Figura 66. Interfaz del Biofeedback. Fuente: Autoría.	116
Figura 67. Ejecución del Biofeedback por parte del paciente. Fuente: Autoría.	116
Figura 68. Interfaz de la toma de datos del estado alto. Fuente: Autoría.	117
Figura 69. BDD del archivo csv del estado alto. Fuente: Autoría.	117
Figura 70. Nivel de estrés alto. Fuente: Autoría.	118
Figura 71. Interfaz del Biofeedback de nivel de estrés alto. Fuente: Autoría.	118
Figura 72. Datos de la Frecuencia Respiratoria. Fuente: Autoría.	120
Figura 73. Datos de la Frecuencia Cardíaca. Fuente: Autoría.	121
Figura 74. Datos de la Conductancia de la piel. Fuente: Autoría.	121

Figuras de los Anexos

Anexo 1

Figura. 1. Gráfica del Análisis - Pregunta 1	137
Figura. 2. Gráfica del Análisis - Pregunta 2	138
Figura. 3. Gráfica del Análisis - Pregunta 3	138
Figura. 4. Gráfica del Análisis - Pregunta 4	139
Figura. 5. Gráfica del Análisis - Pregunta 5	139
Figura. 6. Gráfica del Análisis - Pregunta 6	140
Figura. 7. Gráfica del Análisis - Pregunta 7	140
Figura. 8. Gráfica del Análisis - Pregunta 8	141
Figura. 9. Gráfica del Análisis - Pregunta 9	141
Figura. 10. Gráfica del Análisis - Pregunta 10	142
Figura. 11. Gráfica del Análisis - Pregunta 11	142

Figura. 12. Gráfica del Análisis - Pregunta 12	143
Figura. 13. Gráfica del Análisis - Pregunta 1-Encuesta 2	144
Figura. 14. Gráfica del Análisis - Pregunta 2-Encuesta 2	144
Figura. 15. Gráfica del Análisis - Pregunta 3-Encuesta 2	145
Figura. 16. Gráfica del Análisis - Pregunta 4-Encuesta 2	145
Figura. 17. Entrevista virtual con el psicólogo.....	147
Figura. 18. Desarrollo de la Encuesta online.....	183
Figura. 19. Respuesta de la Encuesta.	183
Figura. 20. Construcción de BIOSTRESS.....	184

Anexo 4

Ilustración 1. Conexión de BIOSTRESS.....	187
Ilustración 2. Administrador de dispositivos.....	188
Ilustración 3. Puertos COM y LPT	188
Ilustración 4. Interfaz de BIOSTRESS.....	189
Ilustración 5. Interfaz de toma de datos.....	189
Ilustración 6. Ubicación de los sensores.....	190
Ilustración 7. Inicio del sensor ECG.....	190
Ilustración 8. Conexión del sensor ECG.	190
Ilustración 9. Conectar el sensor ECG.	191
Ilustración 10. BPM obtenidos.	191
Ilustración 11. Conexión del sensor Airflow.....	191
Ilustración 12. Iniciar el sensor Airflow.....	192
Ilustración 13. RPM obtenidos.	192
Ilustración 14. GSR obtenido.	192
Ilustración 15. Comenzar resultado.	193
Ilustración 16. Resultados del test.	193
Ilustración 17. Iniciar Biofeedback.	194
Ilustración 18. Mensaje del ejercicio del Biofeedback.....	194
Ilustración 19. Biofeedback.....	194
Ilustración 20. Archivo csv.....	195
Ilustración 21. Datos guardados.	195
Ilustración 22. Versiones de BIOSTRESS.	196

Ilustración 23. Carpeta de la versión de 64 bits.....	196
Ilustración 24. Creación de acceso directo de BIOSTRESS.	196
Ilustración 25. Icono de BIOSTRESS en el escritorio.	197

ÍNDICE DE TABLAS

Tabla 1. Tipos de Biofeedback.....	49
Tabla 2. Stakeholders del Sistema.....	57
Tabla 3. Requerimientos de Stakeholders	58
Tabla 4. Requerimientos del Sistema	59
Tabla 5. Requerimientos de Arquitectura.....	61
Tabla 6. Elección del Hardware	64
Tabla 7. Especificaciones Técnicas de Arduino UNO	64
Tabla 8. Especificaciones Técnicas de MySignals.....	65
Tabla 9. Elección del Software.....	67
Tabla 10. Recursos Humanos	68
Tabla 11. Recursos Económicos.....	68
Tabla 12. Recursos Tecnológicos.....	69
Tabla 13. Parámetros del sensor Airflow	78
Tabla 14. Parámetros del sensor ECG.....	80
Tabla 15. Parámetros del sensor GSR	81
Tabla 16. Estructura de la BDD inicial.....	86
Tabla 17. Estructura de la BDD inicial más dos parámetros.....	87
Tabla 18. Rangos de los niveles de estrés.	88
Tabla 19. Casos Críticos.....	90
Tabla 20. Estructura de la matriz de confusión.	90
Tabla 21. Extracto del conjunto de datos de entrenamiento.....	100
Tabla 22. Clasificación de los niveles de estrés.	101
Tabla 23. Valores para cada caso crítico.	102
Tabla 24. Matriz de confusión.....	102
Tabla 25. Parámetros de las pruebas del Hardware.....	109
Tabla 26. Sujetos de prueba.....	110
Tabla 27. Nivel de estrés de los sujetos de prueba.	119

1. CAPÍTULO I

ANTECEDENTES

En este capítulo se encuentra detallada las bases del desarrollo del presente trabajo de titulación, siendo éstos: el tema, el problema, los objetivos, el alcance y la justificación, que intervienen en el desarrollo del presente proyecto, sobre qué está sustentado, al igual que sus limitaciones, con la finalidad de diseñar un sistema *biofeedback* de sensores de detección de estrés mediante algoritmos de aprendizaje de máquinas en estudiantes universitarios.

1.1. Tema

SISTEMA BIOFEEDBACK DE SENSORES DE DETECCIÓN DE ESTRÉS MEDIANTE ALGORITMOS DE APRENDIZAJE DE MÁQUINAS EN ESTUDIANTES UNIVERSITARIOS.

1.2. Problema

El estrés es un problema que se produce cuando ciertas circunstancias externas o internas provocan que un individuo se sienta amenazado (Flores Ortega, 2006). De tal manera que, si la cantidad o la intensidad de los estímulos estresantes se mantienen en el tiempo, puede producir una desorganización o desequilibrio en el organismo que puede llegar a sobrecargar o a bloquear su capacidad de reacción (Mario & Director, 2017). El estrés producido por los estudiantes es un problema de salud que debe ser estudiado con frecuencia (SOUZA, 2016), puesto que es un estado común en los seres humanos para afrontar diversas situaciones de la vida cotidiana (Bakker, Pechenizkiy, & Sidorova, 2011). Sin embargo, cuando es prolongado o es de gran intensidad puede provocar afecciones, como son presión arterial alta, diabetes, trastornos neuro-psiquiátricos,

depresión, entre otros (Acevedo, M., Acevedo, M., 2001). Por lo tanto, es de vital importancia poder medir y prevenir la ocurrencia continua y prolongada del estrés.

En los estudiantes universitarios, el estrés académico es sentido hoy en día como respuesta a diferentes factores como son: la competitividad grupal, realización de un examen, sobrecargas de tareas, exposición de trabajos en clase, exceso de responsabilidad entre otras posibles causas (Rivera & Cruz, 2018) (Widanti, Sumanto, Rosa, & Fathur Miftahudin, 2015). Esto puede provocar la falta de concentración, fatiga intelectual, descenso del rendimiento intelectual o físico, depresión, tristeza, nerviosismo, ansiedad, angustia, impaciencia, dolores de espalda y cuello, entre otros (Oweis, Quteishat, Zgoul, & Haddad, 2018).

Para medir el nivel de estrés académico se puede evaluar al estudiante a partir de diferentes factores como las bioseñales producidas por los organismos del ser humano. Algunos de los parámetros que se pueden medir son: la aceleración del ritmo respiratorio, producción de diferentes tipos de hormonas que se incorporan al sistema circulatorio e inhibición de otras (hormonas sexuales o del crecimiento), aumento de la actividad cardiovascular, cierre del tracto digestivo, entre otras (Healey, 2011). Se propone un sistema de *biofeedback* basado en tres sensores que midan la frecuencia cardíaca (FC), la conductancia de la piel (GSR) y la frecuencia respiratoria (FR) para detectar mediante algoritmos de aprendizaje el nivel de estrés de los estudiantes universitarios.

Con lo propuesto se pretende detectar mediante sensores el nivel de estrés académico en un grupo de estudiantes determinado, con los cuales al momento de manifestar mediante el dispositivo su condición en tiempo real, una aplicación dará a conocer el procedimiento para que la condición de los estudiantes no empeore y cause futuras afecciones.

1.3. Objetivos

1.3.1. Objetivo general

Diseñar un sistema *biofeedback* de sensores de detección de estrés mediante algoritmos de aprendizaje de máquinas en estudiantes universitarios.

1.3.2. Objetivos específicos

- Analizar la información sobre el estrés académico y sus variables con la finalidad de adquirir requerimientos que serán de ayuda para el desarrollo del sistema.
- Realizar el diseño del sistema mediante sensores que a través de algoritmos de aprendizaje de máquinas determinen los parámetros para la detección del estrés.
- Desarrollar la aplicación que dé a conocer en tiempo real a los usuarios el procedimiento para que su condición no empeore.
- Comprobar el funcionamiento del sistema mediante pruebas de *software/hardware* en un grupo específico de estudiantes.

1.4. Alcance

Para la elaboración del proyecto de pregrado se tratarán algunos conceptos sobre una enfermedad que sufren los estudiantes sin darse cuenta denominado estrés, para poder analizar el estrés mediante un sistema *biofeedback* se realizará un dispositivo electrónico que permita medir los niveles de estrés académico. El sistema propuesto tiene como objetivo predecir el nivel de estrés en estudiantes durante las diferentes circunstancias del ciclo académico.

El proyecto se desarrollará en base al modelo en V, para el diseño del dispositivo electrónico se empleará el estándar IEEE-29148, se seleccionará el sistema embebido más adecuado, este tendría que ser potente, con extensas posibilidades de diseño y que tenga mayor flexibilidad para trabajar en casi todas las plataformas informáticas. Adicionalmente, que cuente con gran cantidad de proyectos de *Open Source* en cuanto al *software* y *hardware* disponibles, para que permita ampliar la capacidad de desarrollar y construir nuevos sistemas inteligentes en poco tiempo para realizar pruebas.

Las señales obtenidas van a ser evaluadas mediante algoritmos de decisión supervisados, con lo cual el sistema electrónico será capaz de predecir los niveles de estrés en los estudiantes. Para la elección del algoritmo de decisión supervisada, se realizarán una serie de pruebas y finalmente se determinará el algoritmo ideal con el cual se trabajará.

El *software* que se va a incorporar en el sistema debe ser desarrollado utilizando la instrumentación que ayude a capturar y validar los datos obtenidos por los sensores, mediante una interfaz gráfica interactiva con el usuario se podrá monitorizar el estrés. Para la realización del *biofeedback* se implementará una aplicación la cual indique al usuario en tiempo real la condición que posea y para que esta condición no se descontrolé

el usuario deberá realizar un procedimiento que previamente será dictado por un profesional (Psicólogo).

En cuanto al *hardware* del sistema estará constituido por sensores de frecuencia cardíaca, conductancia de la piel y frecuencia respiratoria, que mediante un análisis se pueda determinar los niveles de estrés a través de algoritmos de aprendizaje de máquina. Para la calibración y correcta lectura de los sensores se utilizará dispositivos electrónicos pasivos, para no producir amplificación y controlar la electricidad colaborando al mejor funcionamiento de los sensores y evitando saltos desmesurados en la obtención de datos.

Se realizará las pruebas de funcionamiento del sistema *biofeedback* de sensores en un grupo determinado de estudiantes que se seleccionará gracias a la información brindada por el departamento de bienestar universitario y por la carrera de Psicología General donde se verificaría la funcionalidad del sistema en corto tiempo y se tendría que corregir los errores que se encuentren.

1.5. Justificación

Este proyecto se realiza con la finalidad de aplicar los conocimientos adquiridos en el transcurso de la formación estudiantil en aspectos teóricos y prácticos. Además, de contribuir con el cuidado de la salud de los usuarios, acorde a los requerimientos de cada uno de ellos.

La evolución de la tecnología se convierte en una herramienta fundamental en el campo de la psicología para generar beneficios a los ciudadanos mediante diferentes técnicas de análisis, evaluación y modificación de la conducta de las personas. Mediante aparatos electrónicos se puede medir las funciones orgánicas y señales fisiológicas de los ciudadanos dando una precisión en el diagnóstico y en los resultados (Botella et al., n.d.). El estudio de los datos ha sido posible gracias a la tecnología, electrónica y comunicación para la creación y desarrollo de dispositivos electrónicos que contribuyan en la salud con el diagnóstico, seguimiento o tratamiento de enfermedades (“Tecnología al servicio de la salud | Innovación | Portafolio,” n.d.). En este caso se utiliza la tecnología y el análisis de datos para predecir los niveles de estrés en estudiantes.

Se observa que en la actualidad existe en todo el mundo una amplia gama de tratamientos para determinar el estrés, sin embargo, la mayoría de estos son ambiguos ya que solo se realizan a base de test psicológicos, y, los *biofeedback* que se encuentran vigentes tienen un costo realmente elevado para controlar todo tipo de funciones del cuerpo humano no específicamente del estrés. Es por ello que este dispositivo busca dar a conocer el diagnóstico del nivel de estrés en estudiantes y a su vez pueda servir de ayuda en las terapias brindadas por los psicólogos.

Este sistema está encaminado hacia el desarrollo humano y ayuda a lo mencionado en el objetivo 1 inciso 1.6 del Plan de Desarrollo donde se señala que el país debe garantizar el derecho a la salud, la educación y al cuidado integral durante el ciclo de vida,

bajo criterios de accesibilidad, calidad y pertinencia territorial y cultural. Adicionalmente contribuye a lo establecido en el objetivo 5 inciso 5.6 en el cual el país debe promover la investigación, la formación, la capacitación, el desarrollo y la transferencia tecnológica, la innovación y el emprendimiento, la protección de la propiedad intelectual, para impulsar el cambio de la matriz productiva mediante la vinculación entre el sector público, productivo y las universidades (Yasuní, n.d.).

Como el estrés es una respuesta del cuerpo a condiciones externas viendo afectado el rendimiento físico y académico de los estudiantes que con el paso del tiempo va generando enfermedades más graves, se pretende medirlo mediante un *biofeedback* para predecir el nivel de estrés de los estudiantes mediante las bioseñales producidas por el organismo de estos.

2. CAPÍTULO II

REVISIÓN BIBLIOGRÁFICA

En esta sección se encuentran los conceptos afines con el estudio del estrés donde se tiene desde los conceptos generales, qué es el estrés, estrés estudiantil con sus causas y efectos. Además, se encuentran conceptos de sensores, sistemas embebidos y *Biofeedback*.

2.1. Estrés

Según (Widanti et al., 2015) lo define como una forma que utiliza el cuerpo humano para manifestar una alteración que proceda del interior y del exterior del cuerpo. Éste es una respuesta ante una situación inesperada o un evento que altere la paz de los individuos. Cada persona tiene una reacción diferente, es decir, una situación puede ser relajante para una persona mientras que para otra puede resultar completamente indeseable.

Este sentimiento de tensión puede disminuir el sistema inmunológico del ser humano, así como aumentar la presión arterial y frecuencia cardíaca que provoca diferentes comportamientos del cuerpo al realizar actividades generales. (Acevedo, M., Acevedo, M., 2001)

2.1.1. Causas

Se puede producir por diferentes causas las cuales se debe tener en cuenta para que pueda ser controlado a tiempo. La mayoría de situaciones y problemas en la vida de las personas que llegan a causar estrés se las denomina estresores, estos son todo lo que suponga demandas altas. Las causas excesivas dependen de la percepción de cada ser humano, (Martínez Díaz, 2016). Pueden existir las causas comunes de estrés externo e interno que se detallan a continuación (Robinson, 2017):

En este contexto, las causas comunes internas pueden ser:

- Pesimismo
- Perfeccionismo
- Pensamiento rígido
- Actitud de todo o nada
- Preocupación crónica

Las causas comunes externas pueden ser:

- Factores personales
- Trabajo o estudios
- Cambios significativos en la vida
- Dificultades en las relaciones
- Problemas económicos

2.1.2. Efectos de Salud

El estrés puede producir graves daños en la salud y afectar el bienestar en general, los síntomas que se pueden presentar son: irritabilidad, insomnio, depresión, ansiedad y dolores de cabeza (Timothy J., 2017). Nuestro organismo se encuentra equipado para soportar este sentimiento de presión, pero en pequeñas dosis y cuando este se vuelve crónico o prolongado puede tener efectos dañinos para el cuerpo. A continuación, se podrá observar los efectos en cada sistema de nuestro cuerpo:

2.1.2.1. Sistema Musculoquelético. La tensión muscular se produce al momento de poseer estrés, es allí cuando los músculos del cuerpo se encuentran en un estado de protección constante, donde a menudo, puede haber una lesión que desencadena el estado doloroso crónico. El dolor musculoquelético en la parte baja de la espalda y las extremidades superiores también se ve relacionado con este, especialmente con el estrés laboral. Pueden existir técnicas de relajación y terapias que lo pueden aliviar, estas

reducen efectivamente la tensión muscular y trastornos como el dolor de cabeza para aumentar la sensación de bienestar en el cuerpo. (Shaw, 2018)

2.1.2.2. Sistema Respiratorio. Las emociones fuertes y el estrés se pueden presentar con síntomas respiratorios tales como dificultad para respirar o respiración rápida. Los estresores psicológicos pueden agravar estos problemas para las personas con enfermedades respiratorias ya existentes como el asma, enfisema y bronquitis crónica. Se puede desarrollar la relajación y respiración con ayuda de un psicólogo para ayudar a reducir los mismos. (Labott-Stewart, 2018)

2.1.2.3. Sistema Cardiovascular. El estrés constante experimentado durante un período prolongado de tiempo afecta al sistema cardiovascular, ya que, el aumento constante y continuo de la frecuencia cardíaca y los niveles de hormonas del estrés y de la presión arterial puede aumentar el riesgo de hipertensión, ataque cardíaco o un accidente cardiovascular. Este puede contribuir a la inflamación del sistema circulatorio, principalmente a las arterias coronarias, puesto que esta es una vía que lo vincula con el ataque cardíaco. También la forma en la que las personas responden al estrés afecta a los niveles de colesterol. (Burg, 2018)

2.1.2.4. Sistema Endócrino. El cortisol es importante para la regulación del sistema inmunitario, esto es valioso durante situaciones que produzcan estrés donde las lesiones provocan una mayor activación de este sistema, pero, si es prolongado puede resultar en una comunicación deteriorada entre éste y el eje hipotalámico-hipofisario-adrenal (HPA), esta comunicación deteriorada se ha relacionado con el desarrollo de afecciones a la salud física y mental que incluyen fatiga, trastornos metabólicos (diabetes, obesidad), depresión y trastornos inmunes perjudiciales para nuestro cuerpo. (Alen & Hostinar, 2018)

2.1.2.5. Sistema Gastrointestinal. El cerebro posee millones de neuronas que están en constante comunicación con el intestino, el estrés puede afectar esta comunicación entre intestino y cerebro lo que provoca que el dolor, la hinchazón y otros malestares intestinales se sientan con mayor facilidad. Se encuentra asociado con cambios en las bacterias intestinales que pueden influir en el estado de ánimo, por lo tanto, estos microorganismos y los nervios influyen en el cerebro. Estos cambios pueden aumentar el riesgo de enfermedades gastrointestinales posteriores o disfuncionales. (Tilburg, 2018)

2.1.2.6. Sistema Nervioso. Cuando el cuerpo se encuentra estresado el sistema nervioso simpático (SNS) contribuye a que el cuerpo cambie los recursos energéticos para luchar contra una amenaza de vida o huir de un enemigo. El sistema nervioso central es importante para desencadenar respuestas a un sentimiento de presión, puesto que regula el sistema nervioso autónomo y desempeña un papel central en la interpretación de contextos como amenazantes. El estrés no afecta en gran cantidad al sistema nervioso, sino que la activación continua de este hace que otros sistemas corporales se vuelvan problemáticos. (Berntson, 2018)

2.1.2.7. Sistema Reproductor Masculino. El estrés continuo puede afectar a la producción de testosterona que provoca la disminución del deseo sexual e incluso causar disfunción eréctil o impotencia. También afecta a la producción y maduración de espermatozoides causando dificultades en parejas que intentan concebir. (Tovian, 2018)

2.1.3. Variables Fisiológicas

Sin importar la situación o tipo de estrés, el cerebro es el encargado de enviar señales bioquímicas que activan la secreción de hormonas, estas inician una reacción sucesiva en el organismo. El cerebro se manifiesta activando el sistema nervioso simpático, con lo cual, el ritmo cardíaco y la respiración se aceleran, la presión arterial aumenta, la sangre deja las venas superficiales de la piel dirigiéndose hacia los músculos

que provee una mayor cantidad de oxígeno, el estado de insulina comienza a elevarse lo que permite que el cuerpo metabolice más energía. Con lo mencionado, el organismo se encarga de responder a la emergencia ya sea combatiendo o desertando de ella, los diferentes estresores presentados en las personas originan una alteración de las funciones inmunológicas. (De Luca et al., 2004)

2.1.3.1. Parámetros fisiológicos para determinar estrés. Los parámetros fisiológicos se producen en el organismo ante los diversos estresores, estos se presentan de mayor a menor intensidad, algunos ejemplos son los siguientes:

- Aumento de la frecuencia cardíaca.
- Se libera gran cantidad de glóbulos rojos debido a la contracción del bazo.
- Se produce una redistribución de la sangre, abandona los puntos menos importantes como la piel y viseras intestinales para acudir a los músculos, cerebro y corazón.
- Aumento de la capacidad respiratoria.
- Se produce dilatación de las pupilas.
- Aumento de la temperatura cutánea.
- Variación de la actividad electrodérmica, puesto que las propiedades eléctricas de la piel se ven afectadas por los incrementos de sudoración que son controlados por el sistema nervioso simpático. (Garzón, 2017)

2.1.4. Estrés Estudiantil

Los estudiantes cuentan con un único grupo de experiencias que pueden resultar estresantes y aunque su rendimiento académico se vea alterado por muchos factores en su vida, según (Essel & Owusu, 2017) el rendimiento académico puede verse afectado por el estrés. Los eventos que pueden resultar más estresantes para los estudiantes pueden

ser: desafíos financieros complicados, gran obligación académica y responsabilidades domésticas y de trabajo si cuentan con uno.

2.1.4.1. Causas. Existen diferentes estresores que provocan el estrés académico en los estudiantes, estos se deben conocer para poder controlarlo y que no se vuelva algo peligroso que afecte a la vida de los estudiantes y de las personas en general. Las causas más comunes pueden ser:

2.1.4.1.1. Las Relaciones. El vínculo que se crea entre las personas puede ser un factor que produce una carga para un estudiante si se crea un abuso que puede ser relacionado con el dinero, sexual o física y puede provocar peligros, aislamiento y terror. Esto hace que se vuelvan factores estresantes para el estudiante puesto que comienza a pensar las formas de resolver los problemas y que conlleva a la distracción de la parte académica. (Essel & Owusu, 2017)

2.1.4.1.2. Factores Personales. Los factores personales pueden causar este sentimiento de presión en los estudiantes y varían de persona a persona que resulta un conjunto diferente de percepciones, actitudes y comportamientos. (Sánchez, 2019).

Algunos de los factores personales pueden ser:

- Cambio en el ambiente de vida
- Cambio en los hábitos de sueño
- Nuevas responsabilidades
- Dificultades financieras
- Combinación del trabajo con los estudios
- Problemas de salud
- Mal hábito alimenticio

2.1.4.1.3. Factores Académicos. Existen diversos factores académicos que tienen lugar en las actividades de los estudiantes y pueden producir estrés, entre los cuales están:

- Aumento de deberes enviados en cada clase
- Tomar varios cursos
- Obtener malas calificaciones
- Exceso en las horas de estudio
- Dificultades del lenguaje
- Postergar tareas enviadas
- Exámenes
- No poder asistir a clases
- No comprender la clase impartida por los docentes. (Sánchez, 2019)

2.1.4.1.4. Factores ambientales. Algunos estudiantes se pueden adaptar a un nuevo entorno, mientras que, para otros un cambio de entorno les resulta incómodo lo que les genera estrés. Algunos de los factores ambientales son los siguientes:

- Falta de vacaciones
- Falta de habilidades para usar un computador
- Malas condiciones de vida
- Acontecimientos importantes (divorcio entre padres)
- Estar en situaciones desconocidas
- Miedo a fracasar y a hablar en público
- Preocupaciones futuras
- Expectativas irrealistas. (Sánchez, 2019)

2.1.4.2. Efectos. Los estudiantes universitarios en la actualidad perciben altos niveles de estrés debido a diferentes causas mencionadas anteriormente, por ejemplo, los horarios de estudio ya que son muy desafiantes y pueden provocar agotamiento, depresión, y otras enfermedades. El alto nivel de este en los estudiantes universitarios puede llegar hasta el límite de suicidarse. El efecto que tiene en los estudiantes

universitarios puede verse de diferentes maneras ya que impacta en la parte académica, social y físicamente. (Campo et al., 2015)

2.1.4.2.1. Efecto En La Parte Académica. El estrés estudiantil puede llevar a los estudiantes universitarios a poseer un mal desempeño en el trabajo académico. Interfiere en la concentración de los estudiantes puesto que, si los niveles no son altos mejora la concentración a corto plazo ya que los estudiantes se concentran para tratar de resolver un problema en cuestión, sin embargo, si estos niveles son demasiado altos afecta la concentración a largo plazo de los estudiantes. Además, puede afectar al rendimiento que obtienen ya que cuando poseen estrés no dan lo mejor de sí en hacer los trabajos y los resultados se manifiestan en las calificaciones, tampoco hacen la preparación adecuada para desarrollar los exámenes. Incluso, a largo plazo puede afectar a los planes que tengan en el futuro. Este hace que los estudiantes puedan sufrir de amnesia puesto que olvidan fácilmente lo que les enseñan los docentes debido a que sus mentes se encuentran agobiadas por los estresores. (Campo et al., 2015)

2.1.4.2.2. Efectos En La Sociedad. Los estudiantes pueden iniciar aislamientos sociales cuando tienen estrés, es decir, si los estudiantes poseen muchos factores estresantes necesitan pensar y prefieren estar solos y lejos de sus amigos para reflexionar y tratar de solucionar sus problemas. Cuando los estudiantes están con este sentimiento de presión fácilmente poseen problemas de ira ya que se irritan con las pequeñas cosas que realizan las demás personas en su entorno y esto hace que las personas se alejen incluido a sus amigos. También puede provocar un menor deseo sexual porque el estudiante se encuentra cada vez más cansado y esto hace que afecte a sus matrimonios o relaciones que posean ya que el sexo es un factor vital en su vida. (Campo et al., 2015)

2.1.4.2.3. Efecto En La Parte Física. El efecto físico en los estudiantes se ve reflejado en su salud ya que puede hacer respirar incluso más fuerte de lo normal, esto

para la mayoría de personas no es un problema grave, sin embargo, para las personas que sufren una enfermedad pulmonar como asma les resulta más difícil obtener el oxígeno que necesitan para poder respirar. La hiperventilación o respiración rápida puede provocar un ataque de pánico en algunas personas. Además, hace que los músculos se tensen, en altos niveles los músculos del cuerpo se encuentran en un estado de protección constante. El estrés puede inducir entre los estudiantes al aumento de consumo de alcohol, drogas y tabaco, ya que, algunos consideran que estas sustancias pueden calmarlos en situaciones estresantes, a la larga puede provocar implicaciones graves en su salud. Por último, puede producir fatiga en los estudiantes puesto que pasan la mayoría del tiempo cansados ya que gastan todas sus energías pensando en los factores estresantes de su vida y cómo pueden solucionarlos. (Campo et al., 2015)

2.1.4.3. Métodos de Detección. Existen diferentes métodos para detectar el estrés, como encuestas, tests o en consultas con especialistas, pero no son muy eficientes debido a que los estudiantes al momento de realizar estos métodos para detectar este problema pueden mentir sobre cómo se encuentran para no seguir un debido tratamiento o por miedo. Los especialistas han optado por obtener dispositivos electrónicos que sean capaces de detectar los cambios que sufren los órganos del cuerpo para divisar diferentes enfermedades y brindar un tratamiento más adecuado. En la actualidad existen algunos dispositivos que sirven para detectar síntomas del estrés, pero este está integrado para detectar conjuntamente con el estrés otras enfermedades como la ansiedad, depresión, entre otros. Este dispositivo es de gran tamaño y los estudiantes no podrían acceder a adquirir uno de estos dispositivos. (Hasanbasic et al., 2019)

2.2. Sistemas Embebidos

2.2.1. Introducción

Como lo dice (Wei & Wang, 2007) un sistema embebido significa que está unido a otra cosa, es decir que se puede considerar como un sistema de *hardware* de computadora basado en un microprocesador con *software* diseñado para que cumpla una función determinada, ya sea como un sistema independiente o como parte de un grande sistema. En el núcleo se puede encontrar un circuito integrado que está diseñado para realizar cómputos para operaciones en tiempo real. (Quespaz, 2019). Los sistemas embebidos cuentan con características tales como:

- Única función: por lo general solo se realiza una operación especializada la cual la realiza repetidamente.
- Estrechamente restringido: los sistemas embebidos son estrictamente restringidos en las métricas de diseño, estas son una medida de las características de una implementación, como puede ser el costo, tamaño, rendimiento y la potencia. El tamaño adecuado debe ser para un solo chip, el funcionamiento tiene que ser adecuadamente rápido para que procese los datos en tiempo real y pueda consumir la potencia mínima para extender la vida útil de la batería.
- Reactivo y en tiempo real: los sistemas embebidos continuamente deben reaccionar a los cambios en el entorno del sistema y también calcular ciertos resultados en tiempo real sin ninguna demora.
- Basado en microprocesadores: se encuentran basados en un microcontrolador o microprocesador para que cumpla las funciones que se le asigna.
- Memoria: su *software* por lo general se integra a la ROM, es por eso que posee una memoria.
- Conectado: para conectar dispositivos de entrada y salida cuenta con periféricos conectados en su estructura.

- Sistemas HW-SW: se utiliza el *Hardware* para el rendimiento y seguridad, y el *Software* para para poder obtener más funciones. (Lin et al., 2016)

En la siguiente figura se puede observar la estructura básica de un sistema embebido

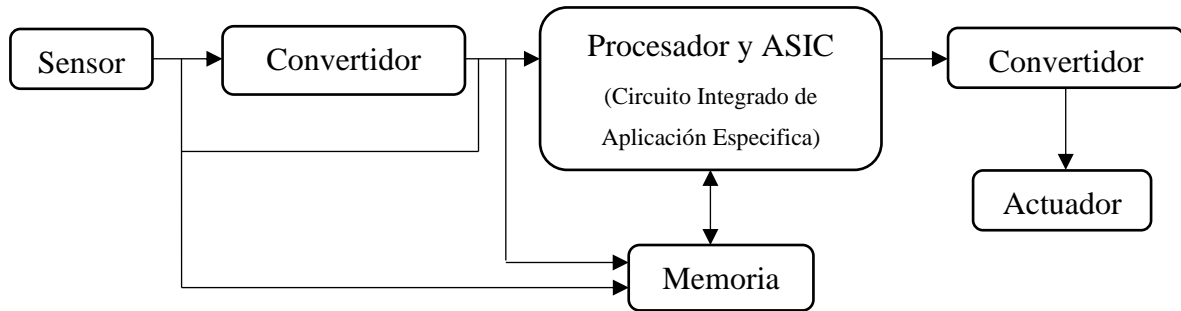


Figura 1. Estructura de un sistema embebido. Adaptado de: (Wei & Wang, 2007)

Los sistemas embebidos son fácilmente personalizables, consumen poca energía, tienen bajo costo y un rendimiento mejorado, sin embargo, tienen un alto esfuerzo de desarrollo y mayor tiempo para ser comercializados. Si bien, se espera que los sistemas embebidos continúen creciendo rápidamente para impulsar el desarrollo continuo de la inteligencia artificial, realidad virtual, realidad aumentada, aprendizaje automático e internet de las cosas. (Wei & Wang, 2007)

2.2.2. Tipos

2.2.2.1. Redes de Sensores Inalámbricas (WSN). *Wireless Sensor Networks*, es una red de sensores que se basa en dispositivos de consumo energético y costo muy bajos, estas redes no son idóneas para adquirir datos de su entorno para después procesar la información local y comunicarse de manera remota con el nodo central. (Burbano, 2019)

Las redes de sensores cuentan con un conjunto de sensores, estos poseen diferentes capacidades sensitivas y de comunicación inalámbrica que les permite formar redes *ad-hoc* sin una administración central. Esta red no cuenta con un nodo central

debido a que todos los dispositivos de la red tienen igualdad de condiciones. (Singh & Ranjan, 2011)

Según menciona (Sathish Kannan & Thilagavathi, 2013), estas redes inalámbricas tienen la facilidad de despliegue puesto que son autoconfigurables, también cuentan con una gestión eficaz de energía la cual las convierte en redes de sensores inalámbricas plenamente operativas.

En las redes de sensores inalámbricas se puede encontrar la red *ad-hoc* la cual cumple con las siguientes características:

- Movilidad: se despliegan muy rápido sin formar grupos lo que quiere decir que cada nodo puede ser solvente individualmente.
- Seguridad: estas redes pueden ser vulnerables a los diferentes ataques ya que se puede emular un nodo legítimo y capturar los distintos paquetes para robar información.
- Conservación de energía: cuentan con batería limitada y es importante diseñarlos para que sean eficientes y así tener un mejor rendimiento y prolongar la vida útil de las baterías.
- Escalabilidad: los nodos pueden crecer mucho debido que no existe un *access point* determinado como central y la incorporación y descarte de nodos es sencillo.

(Fernández Martínez et al., 2009)

Se puede encontrar dos tipos de WSN las cuales son multisalto y monosalto. En las redes multisalto la información se retransmite por varios nodos para llegar a la estación base, lo contrario de las redes monosalto que como su nombre lo indica son redes que realizan un solo salto ya que se conectan directamente con la estación base. (Shrivastava et al., 2019)

2.2.2.2. Reloj de Tiempo Real (RTC). Los sistemas de tiempo real son aquellos sistemas que responden a eventos externos de manera oportuna. Esto se puede observar en la figura 2 donde el tiempo de respuesta está garantizado (Qing & Yao, 2013).

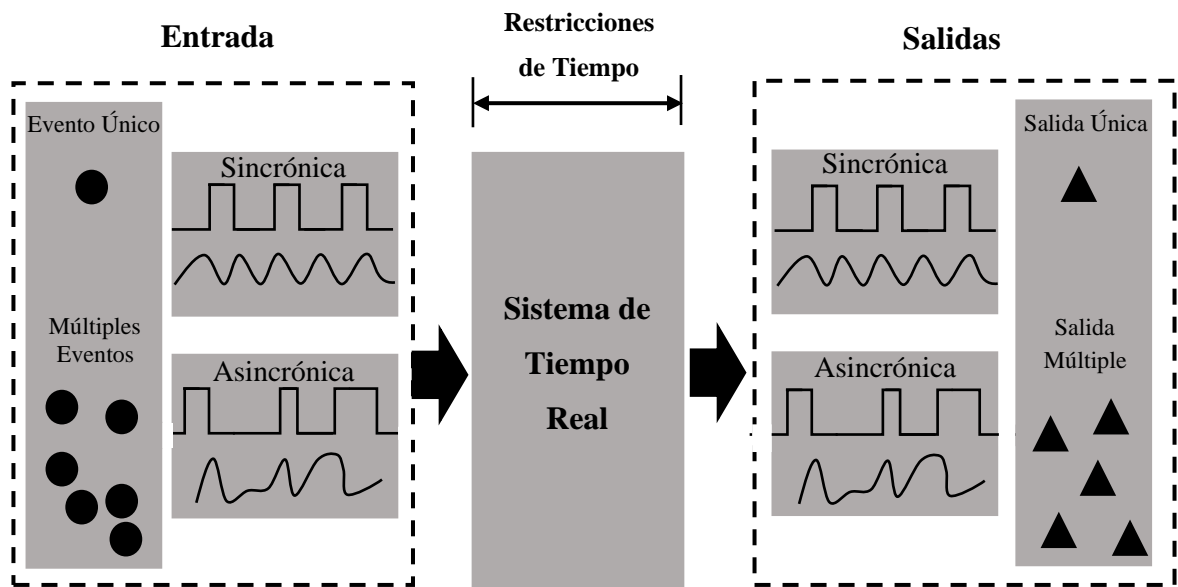


Figura 2. Vista del sistema de tiempo real. Adaptado de: (Qing & Yao, 2013)

Los eventos externos tienen características asincrónicas o sincrónicas, responder a eventos externos contiene reconocer cuando sucede un evento, establecer el procesamiento requerido y generar los resultados dentro de una restricción dada de tiempo es decir en tiempo real que incluyen el tiempo de inicio y el tiempo final. En la figura 3 se puede observar que no todos los sistemas embebidos presentan comportamientos en tiempo real ni todos los sistemas en tiempo real están integrados. Sin embargo, estos sistemas no son excluyentes y se crea la combinación de los sistemas conocidos como sistemas integrados en tiempo real. (Michel et al., 2016)

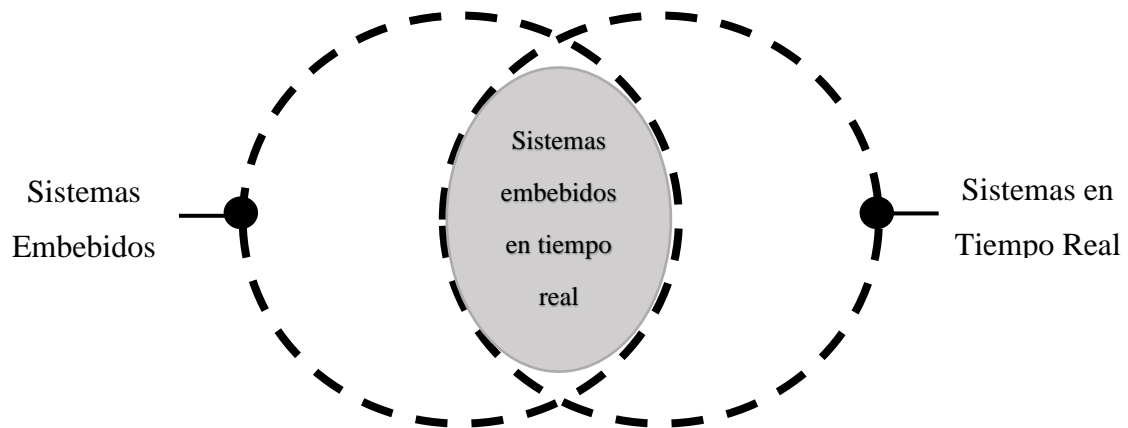


Figura 3. Sistemas embebidos en tiempo real. Adaptado de: (Michel et al., 2016)

En la figura 4 se puede ver la estructura de un sistema en tiempo real que es un sistema de control y un sistema controlado. El sistema de control tiene una interacción periódica con el sistema controlado donde la comunicación comienza desde el sistema de control hacia el sistema controlado, esta comunicación se produce a intervalos que se encuentran predefinidos. También la interacción puede ser aperiódica, en la cual se inicia la comunicación desde el sistema controlado hacia el sistema de control donde la comunicación es determinada por las salidas aleatorias de los eventos externos en un entorno controlado. Por último, la comunicación puede ser una combinación de los dos tipos donde el sistema de control procesa y responde la información generada por el sistema controlado en un lapso de tiempo que es garantizado. (Bogdan et al., 2016)

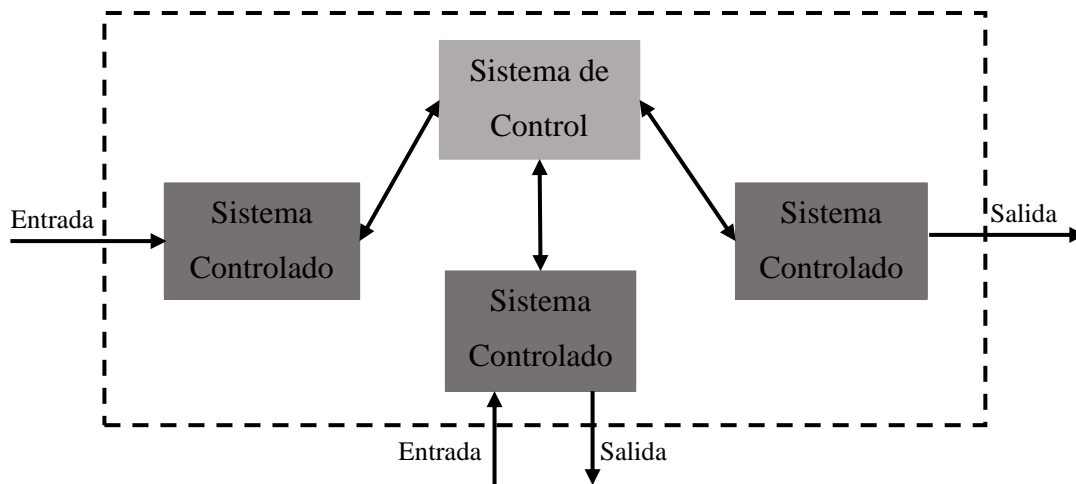


Figura 4. Estructura de un sistema de tiempo real. Adaptado de: (Bogdan et al., 2016)

Las características que poseen los sistemas en tiempo real pueden ser las siguientes:

- Poseen gran tamaño y son más complejos puesto que cuentan con millones de líneas de código.
- Tienen simultaneidad de acciones que quiere decir que trabajan al mismo tiempo, las tareas operan concurrentemente.
- Cuentan con dispositivos de entrada y salida especiales que forman parte del *software* de aplicación. (Aloisio et al., 2015)

2.2.2.3. Procesador Digital de Señal (DSP). El Procesador digital de señal (DSP) cuenta con un propósito especial el cual sirve para la ejecución de algoritmos de procesamiento de la señal digital. La gran parte de estos algoritmos tienen la misma operación como núcleo por ejemplo la transformada discreta de Fourier. (Zahradnik & Šimák, 2012)

Según menciona (Tan & Jiang, 2019), el DSP es un método para procesar señales y datos que las mejoran o modifican con el fin de determinar el contenido de la información concreta. En la figura 5 se observa el sistema de un DSP el cual consiste en

un procesador y un *hardware* que se utilizan para convertir señales analógicas externas en señales digitales y nuevamente en señales analógicas.



Figura 5. Sistema DSP. Adaptado de: (Zahradnik & Šimák, 2012)

2.2.2.4. Wearables. Como lo dice (Luque Ordóñez, 2016), la tecnología *wearable* es la electrónica diseñada para ser vestida ya sea como complemento o siendo parte de algún material que se use en la ropa. Los *wearables* pueden desempeñar algunas tareas de los computadores o los dispositivos móviles, también son más sofisticados puesto que cuentan con sensores que interactúan con el cuerpo y el entorno.

Los *wearables* tienen como característica principal la capacidad de la conectividad inalámbrica ya que permite que el portador pueda acceder a la información en tiempo real. Existen diversos tipos de dispositivos *wearables* en el mundo que tienen diferentes propósitos, los *smartphones* no son considerados dispositivos *wearables* debido a que no son pensados como sensores, aunque cuenten con alguno de ellos como el GPS. (Kumar & Julian, 2015).

2.2.3. Sensores

Los sensores se pueden definir según (Bossche, 1996), como un dispositivo de entrada que proporciona una salida que sería una señal con respecto a una cantidad física específica definida como entrada. El sensor adquiere la cantidad física la cual la convierte en una señal para ser debidamente procesada. Los sensores convierten la medición de algunos fenómenos físicos en una señal eléctrica. (Wilson, 2008)

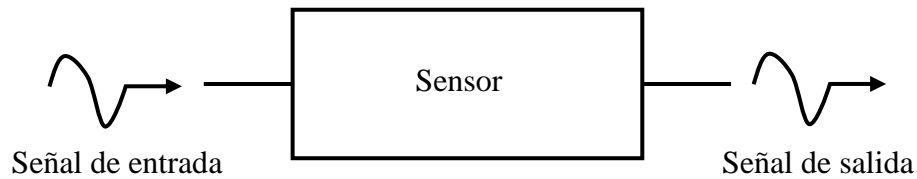


Figura 6. Sensor. Adaptado de: (Wilson, 2008)

2.2.3.1. Tipos.

2.2.3.1.1. Análogos. Entre los tipos de sensores se encuentran los sensores análogos los cuales según (Lee et al., 2007) son los que producen una señal de voltaje o salida continua que por lo general es proporcional a la cantidad que se está midiendo. Detecta los parámetros externos como los producidos por la naturaleza y genera un voltaje analógico como salida. El rango del voltaje de salida esta entre 0 y 5V. En la imagen a continuación se puede observar la temperatura de un líquido que puede ser medido usando un termómetro y el diagrama de la señal que genera. (Gorgues, 2017)

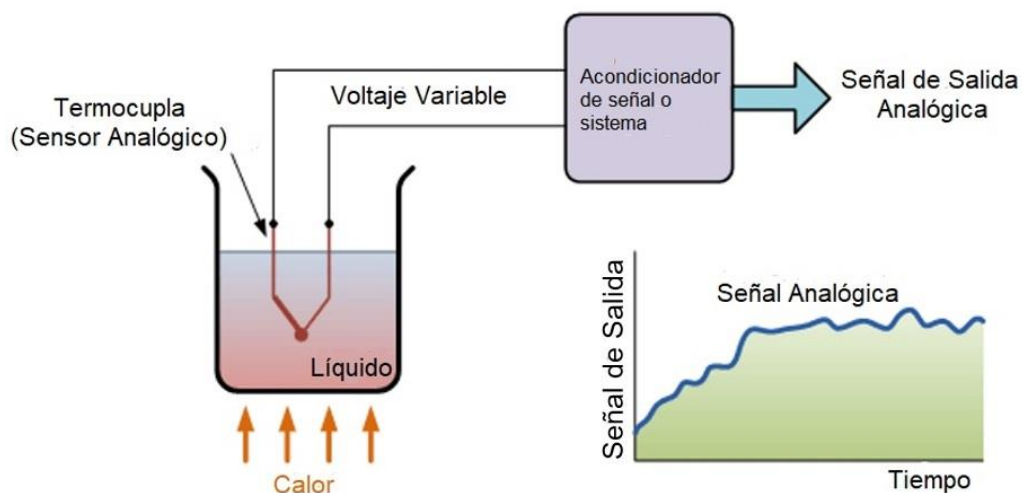


Figura 7. Sensor análogo. Adaptado de: (Gorgues, 2017)

2.2.3.1.2. Digitales. Los sensores digitales son los que producen señales de salida digitales discretas o voltajes que son una representación de lo que se está midiendo. Producen una señal de salida binaria en forma de un "0" lógico o un "1" lógico, es decir

que la señal digital solo puede originar valores no continuos que se emiten como un solo bit. (Miyara, 2004). En la figura 8 se puede observar el sensor digital con su respectivo diagrama de señal que se genera y se puede ver la diferencia con el diagrama de la señal del sensor análogo.

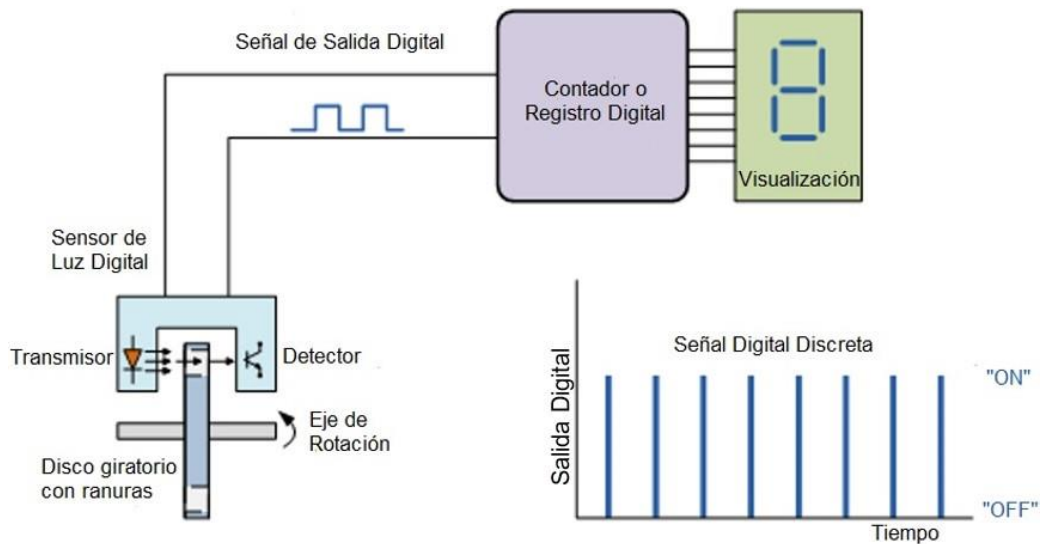


Figura 8. Sensor digital. Adaptado de: (Gorgues, 2017)

2.2.3.1.3. Tren de Pulsos. Como lo menciona (Jiménez & Ángeles, 2015), un microcontrolador de gama baja puede ofrecer un recurso versátil para convertir lecturas de voltaje de un sensor analógico en un tren de pulsos. El tren de pulsos se refiere a una secuencia de pulsos continuos en un intervalo de tiempo, como es repetitivo cuenta con dos factores que es su frecuencia de repetición y su nivel. Cuando se presentan diversos circuitos se necesita un tren continuo de pulsos para que se pueda controlar el momento adecuado de una secuencia y poder elegir el mejor circuito. (Coen & Haelterman, 2000)

2.2.3.2. Acoplamiento.

2.2.3.2.1. Filtros Hardware. Los filtros de *hardware* son filtros que operan de manera directa con las entradas analógicas de los componentes electrónicos como por ejemplo las resistencias, condensadores e inductores. (Torgaev et al., 2015)

El amplificador operacional es un dispositivo amplificador de voltaje que se diseña para ser utilizado con componentes como resistencias y condensadores en sus terminales de entrada y salida. Este dispositivo puede realizar diferentes operaciones como resistividad, capacidad o ambas, es por ello que toma el nombre de amplificador operacional. (Carter & Huelsman, 2001)

Según (Chaniotakis, 2006), los amplificadores operacionales se pueden considerar como uno de los componentes básicos de un diseño lineal, puesto que consta de dos terminales de entrada, una de ellas invierte la fase de la señal y la otra entrada conserva la fase, además consta de un terminal de salida. (Carter & Huelsman, 2001)

Las características con las que cuenta un amplificador operacional son las siguientes:

- Consta de una impedancia de entrada infinita
- Posee un ancho de banda infinito
- Tiene ganancia infinita
- En la salida la impedancia es cero
- El consumo de energía es nulo. (Zumbahlen, 2008)

El funcionamiento básico de un amplificador operacional se puede resumir fácilmente. Como primer punto, se supone que existe una parte de la salida que se retroalimenta al terminal inversor, lo que hace que se establezca la ganancia fija para el amplificador, esto se denomina retroalimentación negativa. Con esto se obtiene que cualquier voltaje diferencial que entra mediante los terminales de entrada se multiplica por la ganancia de bucle abierto del amplificador. Si este voltaje diferencial tiene una magnitud más positiva en el terminal inversor que en el terminal no inversor, la salida será más negativa. Si la magnitud del voltaje diferencial es más positiva en el terminal no inversor que en el terminal inversor su voltaje de salida es más positivo. La ganancia de

bucle abierto del amplificador pretenderá forzar que el voltaje diferencial sea cero. Mientras que la entrada y la salida se hallen en el rango operativo del amplificador se mantendrá el voltaje diferencial en cero y la salida será el voltaje de entrada que se multiplica por la ganancia determinada por la retroalimentación. Para todo lo explicado, las entradas responden al modo diferencial no al voltaje de entrada en modo común. (Torgaev et al., 2015)

Los amplificadores operacionales solo pueden responder a la diferencia entre dos voltajes sin importar los valores individuales de las entradas. Los condensadores o resistencias externos algunas veces se conectan al amplificador operacional de diferentes maneras para constituir circuitos. (Carter & Huelsman, 2001)

2.2.3.2.2. Filtros Software. EL filtro que es más común se denomina filtro de *software*. Este filtro puede leer y manipular los datos que se obtiene de los sensores para ajustarlos a un patrón diferente de salida o también puede eliminar los datos obtenidos que no son necesarios (Hernandez & Jacinto, 2009). Según (Torgaev et al., 2015), los filtros de *software* que se encuentran en un procesador de señal digital, se encargan de leer las muestras que entran de un convertidor A/D, puede realizar diferentes manipulaciones matemáticas que son dictadas por la transformada de Fourier y una vez terminado la modificación se genera el resultado a través de un convertidos D/A.

Los datos se pueden presentar como funciones de tiempo o espacio, para transformarles en componentes de frecuencia se utiliza la transformada de Fourier. Esta transformada es una de las operaciones primordiales en el procesamiento de señales digitales. (Kovacs & Kollar, 2017). Fourier dedujo que una señal aperiódica se puede considerar como una señal periódica que tiene un período infinito, a medida que el período se vuelve infinito en la señal, se forma un continuo de los componentes de frecuencia dando como resultado una suma de la serie de Fourier lo que se llega a

convertir una integral. Esta integral se puede observar en la figura a continuación donde para cualquier función $f(t)$ la transformada de Fourier se expresa $F(\omega)$, se tiene que t es una medida del tiempo y ω corresponde a la señal del dominio de la frecuencia. (Marshall & Sidorov, 2018)

$$\mathcal{F}(f(t)) = F(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-j\omega t} dt, \quad \forall -\infty < \omega < +\infty$$

Ecuación 1. Integral de la Serie de Fourier. Fuente: (Marshall & Sidorov, 2018)

La transformada discreta de Fourier (DFT) es una parte importante para el procesamiento de señales digitales la cual permite encontrar el espectro de una señal de duración finita. Esta DFT por lo general se calcula utilizando la Transformada Rápida de Fourier (FFT). La DFT es el equivalente de la transformada de Fourier continua para señales con una secuencia finita de datos. Para una señal de tiempo discreto, $x(n)$, $n=0, \dots, N-1$, N muestras largas, su transformada discreta de Fourier se encuentra definida como se observa en la ecuación 2, donde se tiene que la salida de la transformación es una secuencia de N coeficientes. (Guinon et al., 2010)

$$X(k) = \sum_{n=0}^{N-1} x(n) \exp\left(-j \frac{2\pi}{N} kn\right), \quad k = 0, \dots, N-1,$$

Ecuación 2. Transformada Discreta de Fourier. Fuente: (Guinon et al., 2010)

La DFT provee una forma eficiente para calcular la convolución en el dominio del tiempo de dos señales digitales, esta convolución es una de las aplicaciones con más importancia de la DFT. La convolución puede representar matemáticamente a las señales que pasan a través de un filtro lineal e invariable en el tiempo. La convolución lineal de dos señales se encuentra dada por la expresión de la ecuación 3 donde nos permite analizar la respuesta de un sistema lineal que no varía en el tiempo a una secuencia arbitraria de la entrada. (Arar, 2017)

$$y(n) = \sum_{k=-\infty}^{+\infty} x(k)h(n-k)$$

Ecuación 3. Ecuación de la convolución. Fuente: (Arar, 2017)

Un ejemplo se puede ver en las gráficas de las funciones que se muestran en las figuras 9 y 10, y la convolución de estas se encuentra en la figura 11.

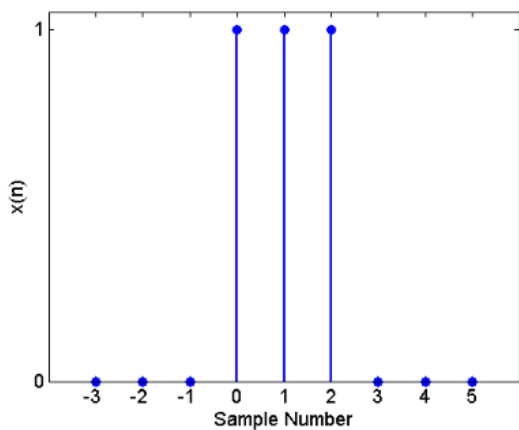


Figura 9. Gráfica de la señal $x(n)$.

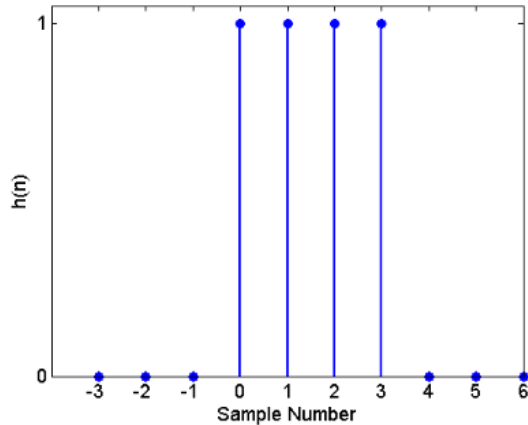


Figura 10. Gráfica de la señal $h(n)$

Fuente: (Arar, 2017)

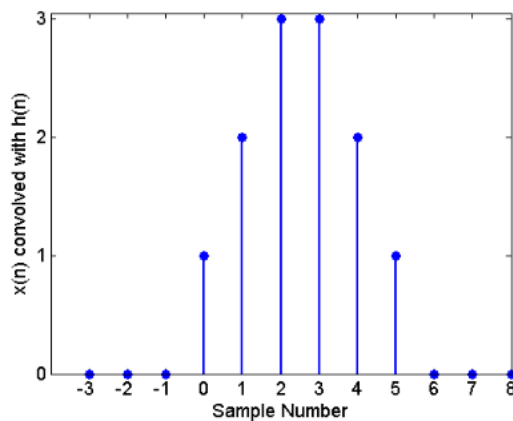


Figura 11. Convolución de las dos señales. Fuente: (Arar, 2017)

Se puede usar el análisis de Fourier para poder calcular la convolución donde se puede expresar matemáticamente como la ecuación 4, en la cual se tiene que $X(\omega)$ y $H(\omega)$ son las transformadas de Fourier en tiempo discreto (DTFT) de $x(n)$ y $h(n)$.

$$F\{x(n) * h(n)\} = X(\omega)H(\omega)$$

Ecuación 4. Ecuación de la convolución utilizando el análisis de Fourier. Fuente:(Kovacs & Kollar, 2017)

Solo se utiliza las muestras de DTFT que son las DFT para poder analizar las señales y realizar la convolución, es necesario tener señales de duración finita por lo que la ecuación se representa como se muestra en la ecuación 5.

$$X(\omega) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n} = 1 + e^{-j\omega} + e^{-i2\omega}$$

Ecuación 5. Convolución con análisis de Fourier. Fuente: (Arar, 2017)

2.2.3.2.3. Criterios de Diseño. Los criterios de diseños se refieren al razonamiento que tienen las personas, el diseño de sistemas tiene como propósito obtener la configuración, especificaciones e identificación de los parámetros de un sistema para satisfacer la necesidad de la realidad de las personas. (Adhikari et al., 2017). En la siguiente imagen se puede observar un diagrama de flujo donde contiene los pasos en el proceso de diseño del sistema integrado.

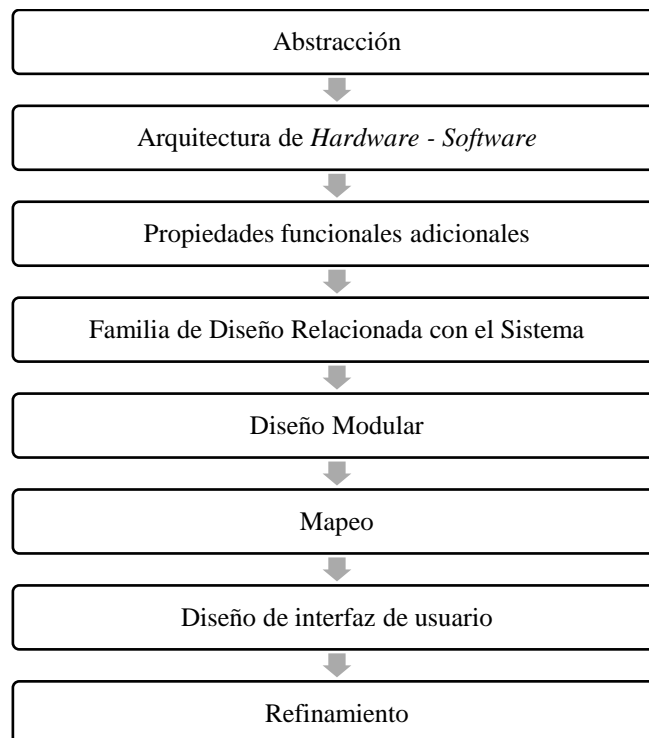


Figura 12. Diagrama de flujo de los pasos para el diseño de sistemas integrados. Adaptado de: (Jansson & Schwabe, 2010)

- La etapa de abstracción se extrae el problema relacionado con el sistema
- En la arquitectura de *hardware* se tiene el conocimiento adecuado de *hardware* y *software* antes de iniciar con el proceso del diseño.
- Las funciones adicionales que se van a implementar se deben entender desde el diseño principal.
- Al momento de diseñar un sistema se debe referir a una familia de diseño previa.
- Es necesario realizar diseños de modulas separados con la finalidad de ser usados más adelante cuando sean necesarios.
- La cartografía se refiere a que debe estar basado en el mapeo de *software*, por ejemplo, el flujo de datos y del programa.
- Se realiza un diseño de interfaz de usuario, este depende de los requisitos que tengan los usuarios, el análisis del entorno y la función del sistema.
- Para que el equipo de *software* pueda ser entendido se debe refinar cada componente y modulo. (Jansson & Schwabe, 2010)

2.2.4. MySignals

Para (Libelium Comunicaciones, 2017), *MySignals* se considera como una plataforma para desarrollar dispositivos médicos y aplicaciones de *eHealth*. Esta plataforma es utilizada para desarrollar la web de *eHealth* propio o insertar propios sensores y así poder construir nuevos dispositivos médicos que ayuden al bienestar de las personas.

En la plataforma de *MySignal* se puede medir más de 20 parámetros biométricos entre los cuales se tiene frecuencia respiratoria, respuesta galvánica de la piel, ritmo cardíaco, entre otras, esta plataforma es una de las más completas que se puede encontrar. (Sarraipa et al., 2019). En la figura 13 se observa la tarjeta de *MySignals* donde se pueden conectar los diferentes sensores diseñados para medir los parámetros que se necesite.

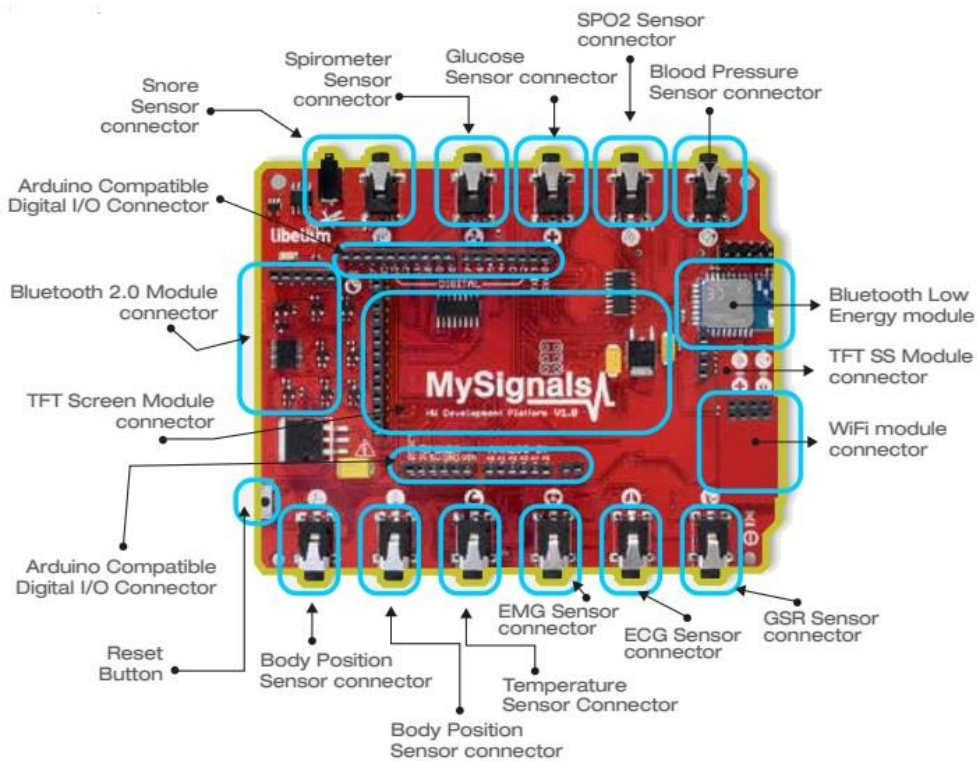


Figura 13. Tarjeta de MySignals. Fuente: (Sarraiya et al., 2019)

Cabe mencionar que los productos de *MySignals* no se consideran como dispositivos médicos o como servicios de atención médica, tampoco se encuentran destinados para dar un diagnóstico médico y tratamientos, puesto que no son productos terminados ya que solo dan a conocer los valores tomados, es por ello que los productos de *MySignals* son destinados para que lo adquieran los desarrolladores o investigadores para que trabajen mediante esta herramienta. (Libelium Comunicaciones, 2017)

2.3. Diseño del Sistema Electrónico

Para realizar el diseño del sistema electrónico se debe tener en cuenta metodologías y normas que se deben seguir, es por ello que se debe conocer el modelo en V, la normativa IEEE 291847 y los diagramas de bloque y flujo.

2.3.1. Modelo en V

Como se menciona en (Tutorialspoint, 2016), el modelo en V se conoce como un modelo de verificación y validación, este modelo es una extensión del modelo de cascada y se encuentra basado en la asociación de una fase de prueba para cada etapa de desarrollo que se encuentra. En este modelo las fases que sigan deben comenzar una vez que se complete la etapa anterior de lo contrario no se puede seguir con el desarrollo del proyecto que se esté realizando.

Las fases del modelo en V se encuentran en la figura 14 y se definen las funciones de cada una de estas.

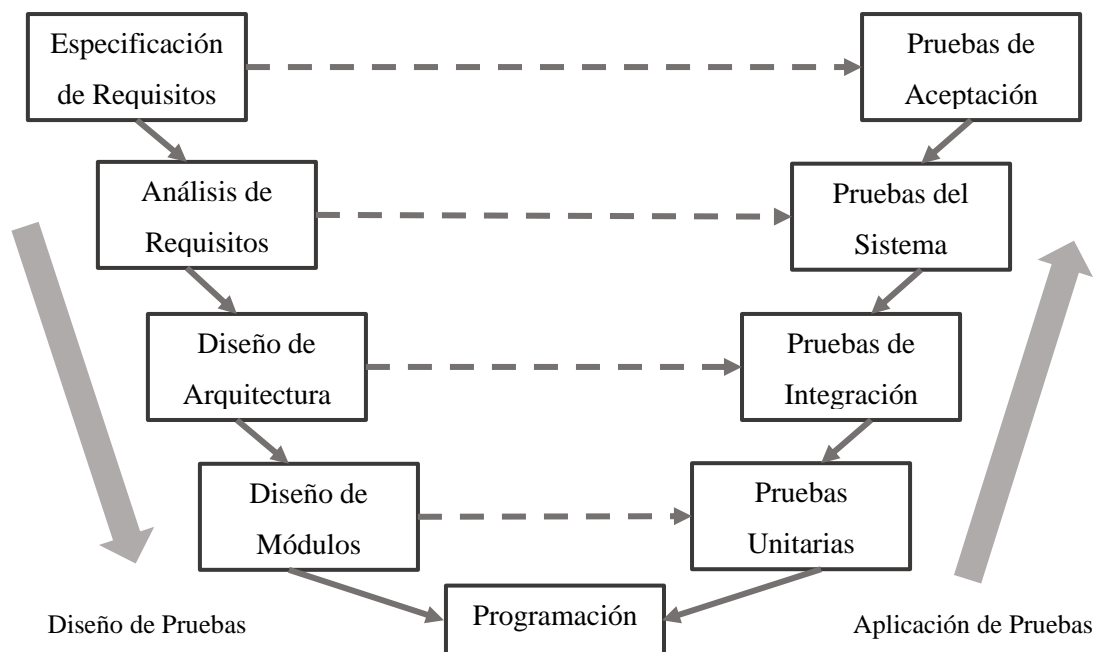


Figura 14. Fases del modelo en V. Adaptado de: (Planning, 2017)

- Fase 1: es la definición de especificaciones donde se detallan los requisitos del sistema que mencionan los *stakeholders*.
- Fase 2: es el diseño de arquitectura donde se obtiene una visión general del sistema que se va a realizar.

- Fase 3: es la fase de programación donde se materializa el diseño del sistema que se está desarrollando.
- Fase 4: esta fase tiene como objetivo la verificación de cada uno de los módulos *Hardware* y *Software* para comprobar que funcionen correctamente.
- Fase 5: es la fase de integración donde se combinan los módulos que hacen parte del sistema y se demuestra el cumplimiento de los requisitos que estaban previamente establecidos. (Mathur & Malik, 2010)

2.3.2. *IEEE 29148*

Para (Jiménez Mediero, 2015), la norma ISO/IEC/IEEE 29148 es un resumen conciso de diferentes normativas de requisitos y procesos para desarrollar productos software donde se hace énfasis sobre el proceso de ingeniería de requisitos. En esta normativa se encuentran secciones principales las cuales pueden ser:

- Conceptos
 - Fundamentos de requisitos: en esta parte se describen los conceptos más importantes utilizados en los requisitos tales como pueden ser la definición de *Stakeholders*, características individuales de cada requisito, criterios de lenguaje, entre otros.
 - Consideraciones prácticas: se detallan los métodos para poder obtener los requisitos.
 - Elementos de información de requisitos: se refiere a la relación de los elementos de información con los requisitos.
- Procesos

- Procesos de requisitos: se debe detallar las guías de creación de los requisitos mediante la normativa ISO/IEC 15288:2008.
- Proceso de definición de requisitos de usuario (*Stakeholders*): se encuentra el proceso para obtener los requisitos.
- Procesos de análisis de requisitos: se hallan las tareas que se lleva a cabo para tener documentados y preparados de la mejor manera.
- Actividades de ingeniería de requisitos en otros procesos técnicos: se encuentran los procesos de requisitos en el diseño de la arquitectura, verificación de requisitos y validación de requisitos.
- Gestión de requisitos: se puede encontrar las tareas de gestión de cambios, gestión de configuración, gestión de información y medición de requisitos.
- Elementos de información
 - Documento de especificación de requisitos de *Stakeholders*.
 - Documento de especificación de requisitos del sistema.
 - Documento de especificación de requisitos de *software*.

2.3.3. Diagramas de Bloque y Flujo

Los diagramas de flujo según (Chart, 2015) se consideran como un esquema de los pasos de un proceso con su respectiva secuencia. Estos diagramas modelan una perspectiva del sistema que es más entendible para los usuarios, ya que el flujo de información se puede representar gráficamente. Los diagramas de flujo se pueden usar cuando se requiera analizar un proceso que ya exista y cuando se necesite estandarizar o rediseñar un proceso. (Batovski, 2014)

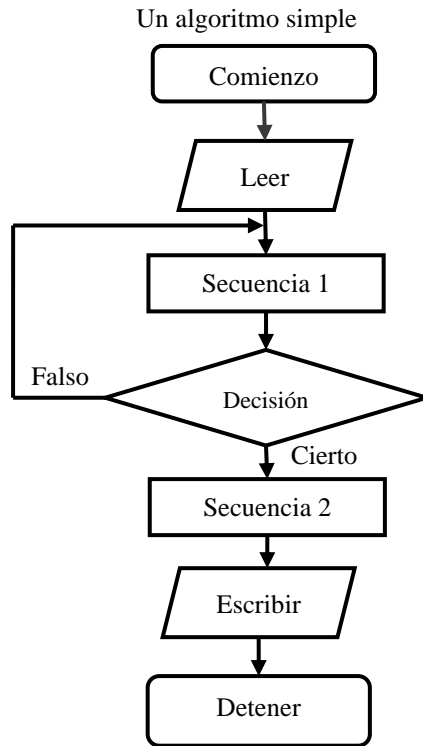


Figura 15. Ejemplo de un diagrama de flujo. Adaptado de: (Batovski, 2014)

Un diagrama de bloque puede describir los bloques de construcción de sistemas complejos donde cada bloque se muestra como un rectángulo con entradas y salidas conectadas a otros bloques del sistema mediante flechas. Los diferentes bloques del sistema se pueden presentar como cajas negras que se interconectan, esto se puede ver en la figura 16. En cada cuadro se debe introducir una etiqueta concisa sobre el fondo blanco que está rodeado de bordes negros. (Flow, 2017)

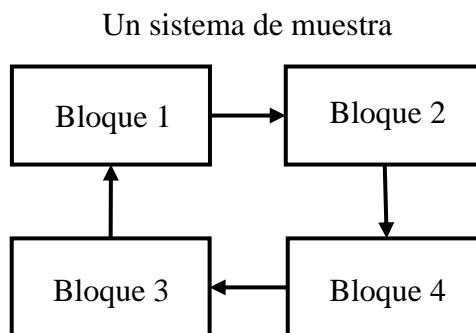


Figura 16. Ejemplo de un diagrama de bloques. Adaptado de: (Batovski, 2014)

2.3.4. Benchmark

Según (Burn, 2017) se define *benchmark* como un análisis comparativo mediante una lista de productos o servicios que permite fijar criterios de evaluación de rendimiento o de eficacia, por lo general esta lista es presentada a través de un cuadro, esto se desarrolla para el diseño de un nuevo producto.

En el caso de este trabajo de titulación se elabora una lista de productos que se encuentran en el mercado y se efectúa un análisis comparativo presentado en tablas, el cual se encuentra regido por requerimientos para seleccionar el que mejor cumpla con dichos requisitos y de esta manera realizar correctamente el *benchmark* para satisfacer los objetivos planteados.

2.4. Biofeedback

2.4.1. Introducción

Biofeedback es el conjunto de procedimientos con el objetivo de posibilitar al sujeto que a partir de la información relativa que es proporcionada de manera puntual, inmediata, constante y precisa, pueda conseguir modificar los valores de forma voluntaria sin necesitar de la intervención de instrumentos químicos. (Conde & Menéndez, 2016)

Se puede decir que *Biofeedback* consiste en brindar información sobre un estado del proceso biológico de una persona, de manera que cualquier técnica que le pueda proporcionar información sobre su actividad fisiológica se pueda controlar voluntariamente, en ese caso se considera *Biofeedback*. (Martínez & Rodríguez, 2019)

2.4.2. Funcionamiento

Dentro del funcionamiento del *Biofeedback* según (Marín Agudelo & Vinaccia Alpi, 2005), el primer paso se basa en la recepción y transmisión al dispositivo de detección mediante cualquier tipo de sensor, después de que la señal es captada se traduce a señales eléctricas, posteriormente el nivel de las señales es potenciado (amplificado)

para que a continuación se dé lugar al filtraje de las interferencias. Estas señales tratadas se pueden traducir a diferentes maneras que son más estimulantes como señales visuales o auditivas que son más fáciles de percibir por las personas. Por último, se tiene que esta información llega de inmediato a las personas. En la figura 17 y 18 se puede ver como con el *feedback* la persona puede controlar y modificar la respuesta en cuestión.

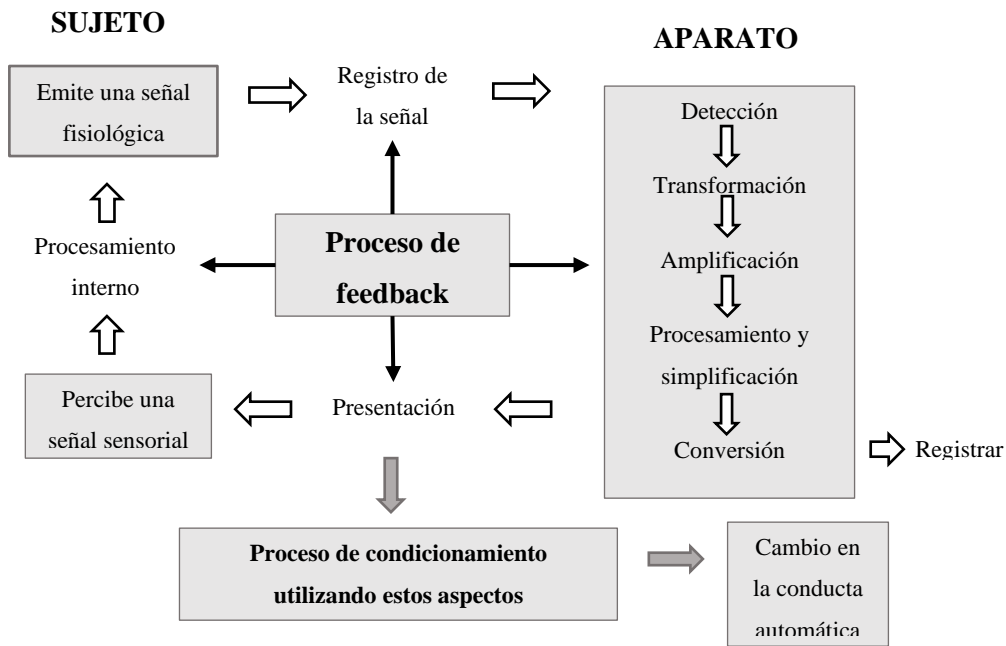


Figura 17. Funcionamiento del Biofeedback. Adaptado de: (Marín Agudelo & Vinaccia Alpi, 2005)

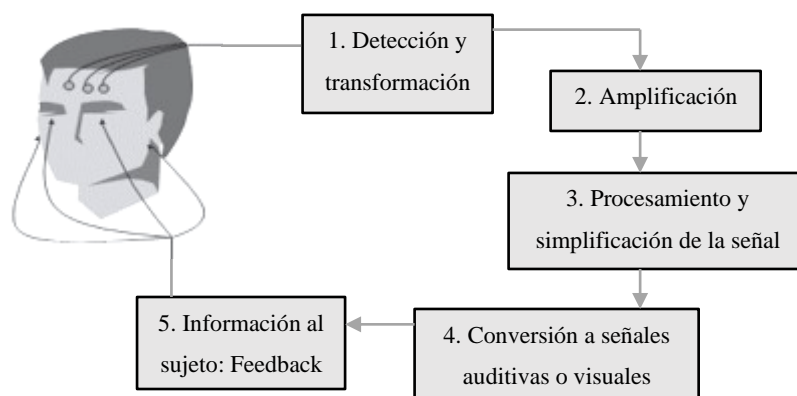


Figura 18. Proceso de Biofeedback. Adaptado de: (Carrobbles, 2016)

2.4.3. Tipos

Debido a los diferentes estudios que se ha venido realizando sobre el Biofeedback se puede conocer los diferentes tipos de Biofeedback que existen los cuales se puede observar en la tabla 1. (Martínez & Rodríguez, 2019)

Tabla 1. Tipos de Biofeedback

TIPOS	SEÑALES
Biofeedback electromiográfico	Actividad muscular de los músculos donde se encuentran los electrodos.
Biofeedback electroencefalográfico	Actividad eléctrica de la zona de la corteza cerebral donde se encuentran los electrodos.
Biofeedback de temperatura periférica	Temperatura superficial de la piel.
Biofeedback de la respuesta psicogalvánica	Nivel de conductancia de la piel.
Biofeedback de la presión sanguínea	Presión arterial.
Biofeedback de la tasa cardíaca	Latidos cardíacos por minuto.
Biofeedback de la erección del pene	Diámetro y longitud del pene.
Biofeedback de los músculos del cérvix	Contracción de las paredes vaginales.
Biofeedback rectal	Sonido del recto.
Biofeedback del esfínter anal	Presión del esfínter anal.
Biofeedback del PH estomacal	Concentración del PH y el ácido hidroclohídrico.
Biofeedback de respiración forzada	Volumen del aire respirado.

Adoptado de: (Conde & Menéndez, 2016). Elaborado por: Tutillo S. 2021

2.5. Machine Learning

Machine Learning o Aprendizaje Automático es una rama de la inteligencia artificial que tiene por objetivo desarrollar técnicas que permitan a los computadores aprender y es visto como un intento de automatizar a través de métodos matemáticos (Gago, 2017). Los algoritmos de *Machine Learning* se pueden encontrar en tres categorías las cuales son aprendizaje supervisado, aprendizaje no supervisado y aprendizaje por refuerzo (Zamorano, 2018).

2.5.1. Aprendizaje Supervisado

Este tipo de algoritmos cuentan con información denominada de entrenamiento para que puedan tomar decisiones o realizar predicciones. A partir del análisis del conjunto de entrenamiento, el algoritmo de aprendizaje ocasiona una función inferida para hacer predicciones sobre los valores de salida, el sistema es capaz de brindar objetivos para las nuevas entradas después de una capacitación suficiente (Páez, 2019). Algunos algoritmos de clasificación pueden ser:

- **Árbol de decisión.** Es una técnica que consiste en analizar las decisiones secuenciales basadas en el uso de resultados y probabilidades asociadas. En este algoritmo se divide la muestra en dos conjuntos homogéneos que se basan en la variable de entrada más significativa y se usa principalmente en problemas de clasificación (Charris et al., 2018).
- **Red Neuronal.** Es un sistema de conexión capaz de simular capacidades cerebrales mediante la interconexión de neuronas artificiales. La neurona artificial consiste en un procesamiento numérico simulando la expansión de un impulso electro-químico, gracias a esto la neurona puede propiciar e inhibir impulsos electro-numéricos según el nivel de estimulación que posea (Sterpin, 2011).

- **Clasificador Naive Bayes.** Es basado en la clasificación denominada teorema de Bayes, en este algoritmo se asume que las variables predictoras son independientes entre sí. Consiste en construir modelos que predicen la probabilidad de posibles resultados (Chaparro et al., 2015).
- **k-NN Vecinos más cercanos.** Este algoritmo se caracteriza por memorizar todos los ejemplos de entrenamiento que se encuentren disponibles durante la fase de prueba, en esta etapa los datos a clasificar se comparan con el conjunto de datos de entrenamiento basándose en una medida de distancia. La clasificación de la información nueva más similar se denomina vecino más cercano, pero es posible basar en los k ejemplos en el cálculo (Lüschoy & Wartena, 2017).

2.5.2. Aprendizaje No Supervisado

En el aprendizaje no supervisado los algoritmos no cuentan con un conocimiento previo es decir no usan etiquetas, tienen la finalidad de la comprensión y abstracción de patrones de información de manera directa, este aprendizaje no es usado para automatizar decisiones y predicciones sino más bien para identificar patrones (Páez, 2019).

2.5.3. Aprendizaje Por Refuerzo

Este tipo de aprendizaje tiene como principal objetivo que el algoritmo aprenda en función de su propia experiencia, es decir que no se entrena con datos de muestra, ya que interactúa con su entorno descubriendo errores y es capaz de tomar la mejor decisión frente a diversas situaciones de acuerdo a un proceso de prueba (Zamorano, 2018).

3. CAPÍTULO III

DESARROLLO EXPERIMENTAL

En el presente capítulo se tratará temas acordes a la elección del *Hardware* y *Software* para la realización del diseño de sistema de detección de estrés, que se basa en una metodología denominada modelo en V y en el estándar IEEE 29148.

3.1. Metodología

Es importante llevar a cabo una secuencia de pasos y procedimientos de investigación, ya que permitirán abrir aún más las perspectivas que se tiene del proyecto. La ejecución clara y objetiva de estos procedimientos de investigación son los que nos ayudarán a obtener un enfoque preciso de lo que se desea obtener y como se desea lograr. El modelo que se utiliza para el desarrollo de este proyecto es el “Modelo en V”, por motivo que cumple con un proceso metodológico adecuado para la investigación realizada.

3.2. Análisis

El proceso permitirá establecer los requerimientos que debe tener el sistema en base a la cantidad de sensores y datos que se recolecta, además esto permitirá determinar los requisitos de *software* y *hardware*. Las técnicas de investigación brindan un planteamiento de la problemática que se quiere solucionar y definir un método para realizar la investigación.

3.2.1. Situación Actual.

Como punto de partida se establece la información necesitada en base al problema del proyecto que se está realizando, en este caso se define las variables que causan el estrés en estudiantes, para lo cual se requiere identificar las necesidades que afecta a los *Stakeholders* del proyecto.

En la actualidad se puede evidenciar en el mundo un gran convencimiento de que las enfermedades cardiovasculares son influenciadas por factores psicosociales, por estilos de vida perjudiciales, factores alimenticios y hasta producir efectos secundarios por estar expuestos al estrés crónico. (Sánchez, 2019) “Uno de cada cuatro personas en las ciudades posee cierto problema peligroso de estrés, aunque también se estima que problemas mentales de salud de este tipo recaen sobre el 50% de las personas” (Campo et al., 2015) En base a esto, el estrés es considerado un factor epidemiológico grave debido a que es el causante de varias patologías. (Sánchez, 2019)

Según encuestas realizadas en diferentes universidades del país, se determinó que existe una relación entre el rendimiento académico y el estrés. Los resultados se evidenciaron de la siguiente manera, en el estrés académico se obtuvo un 42.2%, de los cuales el 57.8% poseían un nivel de estrés no preocupante, el 38.7% tenía un nivel de estrés preocupante y el 3.6% se encontraba con estrés peligroso. (Tomalá, 2018) Esto nos demuestra que existen técnicas no precisas que se utilizan frecuentemente para medir el estrés en las personas.

Hoy en día la tecnología ha ido evolucionando y gracias a ello se puede innovar para mejorar y facilitar nuestra vida. La electromedicina permite introducir al mercado sistemas que ayude con el cuidado de la salud. (Tomalá, 2018) Según una entrevista realizada al Psicólogo Jorge Gordón, docente de la carrera de Psicología General de la Universidad Técnica del Norte, existen dispositivos tecnológicos que monitorean las señales bioeléctricas, pero no específicamente para determinar el estrés, además supo manifestar que estos dispositivos son de gran costo y no son accesibles para todos los ciudadanos más aún para los estudiantes. Es por ello que se realiza el sistema BIOSTRESS el cual se basa en un sistema que monitorea señales bioeléctricas para

determinar los niveles de estrés académico que posee una persona y posteriormente realizar el *Biofeedback* correspondiente para que se pueda controlar su estado.

3.2.2. *Técnicas de investigación*

Se utiliza la técnica de investigación llamada encuesta ya que brinda pautas que sirven para el desarrollo del proyecto.

3.2.2.1. **Encuesta.**

La encuesta se realizó a un cierto número de estudiantes de la carrera de Ingeniería en Electrónica y Redes de comunicación de la ciudad de Ibarra, donde se puede observar que las preguntas son de selección múltiple y corresponden a un tema importante para su bienestar no solo de salud sino también académico. La muestra de la población se tasa según la cantidad de estudiantes que se encuentran siguiendo la carrera antes mencionada.

Para la realización del cálculo de la muestra que se debe tomar se emplea la siguiente ecuación.

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

Ecuación 6. Cálculo de la muestra. (Aguilar Barojas, 2005)

Donde:

n = Tamaño de la muestra.

N = Tamaño de la población.

Z = Nivel de confianza. Valor constante, por lo general se presenta dos valores que dependen del grado de confianza de la investigación, donde el 99% es el valor más alto y equivale a 2,58 y 95% equivale a 1,96 siendo un valor mínimo aceptable. Se debe tomar de acuerdo al criterio del investigador.

σ = Desviación estándar de la población. Se usa la constante 0,5 cuando no se tiene un valor fijo.

e = Límite aceptable de error muestral admitido por el encuestador. Se encuentra desde el 1% al 9% (1% = 0,01 – 9% = 0,09).

Con lo antes mencionado se obtiene los siguientes datos:

$N = 300$, $Z = 1,96$, $\sigma = 0,5$ y $e = 0,08$

$$n = \frac{300 \cdot (1,96)^2 \cdot (0,5)^2}{(300 - 1) \cdot (0,08)^2 + (1,96)^2 \cdot (0,5)^2} = 100,25 \text{ encuestas}$$

De acuerdo con el cálculo realizado en base a la Ecuación 6 se obtiene como resultado que el número de encuestas a realizarse debe ser 100. Las preguntas de la encuesta y tabulación se pueden observar en el Anexo 1. Teniendo en cuenta cuales son las causas que producen estrés en los estudiantes, mediante la encuesta se puede constatar dicha teoría, ya que, según las preguntas planteadas se obtiene los resultados esperados y se llega a las siguientes conclusiones:

- En el Anexo 1 se observa que, en la figura 1 el 91% de los estudiantes encuestados no cuentan con un horario establecido para descansar durante el período académico, esto es alarmante porque como se puede ver en la Figura 2 solo el 59% llega a dormir máximo 5 horas diarias en el transcurso del período académico. Estos datos generan preocupación debido a que pasar muchas horas sin dormir puede producir diferentes enfermedades como el un aumento de peso y a tener problemas cardiovasculares ya que son provocados por el incremento de la hormona del estrés.
- En la Figura 3 se aprecia que el 92% de los encuestados durante el período académico no llevan una dieta equilibrada y en la Figura 4 los encuestados especifican que más de la mayoría en un 86% consumen comida rápida y el 80% snacks, con esto se puede ver que durante el período académico los estudiantes

tienen una mala alimentación que puede ser generada por el estrés académico ya que se agobian con las responsabilidades de la universidad, su organización con el tiempo destinado a la alimentación y sus tareas personales como indica el 93% de los encuestados en la Figura 9, esto puede conducir a una baja inmunidad provocando graves problemas de salud.

- Las Figuras 5, 6 y 7 hacen referencia a la economía de los estudiantes, y su resultado fue que el 18% de los encuestados no reciben una ayuda económica para solventar sus estudios académicos por lo que tienen que trabajar, el 48% si posee un trabajo, lo que quiere decir que algunos estudiantes pese a que reciben ayuda económica cuentan con un empleo. La mayoría de los estudiantes encuestados que son el 66% tiene como presupuesto semanal para sus gastos académicos de 16 a 25 dólares. Algunos de los encuestados los cuales son un 18% se ven afectados por la economía lo que les puede elevar su nivel de estrés y afectar a su rendimiento académico.
- En la Figura 8 se puede concluir que casi todos los estudiantes que son un 94% prefieren cumplir con todas las responsabilidades académicas, aunque no descansen lo necesario, esto es un problema muy grave puesto que como se mencionó en la Figura 1 y 2 los estudiantes no cuentan con hábitos de descanso saludables y les produce altos niveles de estrés afectando a su desarrollo en sus actividades diarias y académicas.
- En la Figura 10 se observa que menos de la mitad de encuestados, un 21% de los estudiantes cuentan con cargas familiares, esto podría ser un factor que se relacione con los niveles de estrés académicos ya que estos estudiantes podrían tener más problemas que el 79% de los que no cuentan con cargas familiares.

- En la Figura 11 se concluye que todas las personas encuestadas consideran que el estrés académico afecta a la salud del estudiante y el 94% de los estudiantes encuestados que se muestra en la Figura 12, consideran útil la creación de un dispositivo electrónico que ayude a conocer los niveles de estrés con la finalidad de prevenir futuras enfermedades.

3.3. Requerimientos

Para comenzar con la primera etapa se define los requerimientos que va a tener el sistema y se procede a la elaboración de estos para cumplir con las necesidades del dispositivo.

3.3.1. Stakeholders.

Según (Ángel, 2010) se define como *Stakeholders* a cualquier individuo o colectividad a la cual puede afectar directa o indirectamente el desarrollo del proyecto. En la Tabla 2 se evidencia la lista de los *Stakeholders* que participan en el diseño del prototipo llamado BIOSTRESS.

Tabla 2. Stakeholders del Sistema

Stakeholders
1. Srta. Samia Tutillo (Desarrollador del proyecto)
2. Ing. Jaime Michilena (Director de Tesis)
3. Estudiantes de la Universidad Técnica del Norte y familiares.
4. Ing. Carlos Vásquez (Asesor de Tesis)
5. Ing. Luis Suárez (Asesor de Tesis)

Fuente: Autoría.

3.3.2. *Requerimientos de los Stakeholders.*

Los requerimientos de *Stakeholders* (STSR) se interpretan como los requisitos que el dispositivo debe tener para que satisfaga las necesidades de los mismos, estos se pueden observar en la Tabla 3 con su respectiva nomenclatura y prioridad.

Tabla 3. *Requerimientos de Stakeholders*

STSR				
#	REQUERIMIENTOS	PRIORIDAD		
		Alta	Media	Baja
Requerimientos de Usuarios				
STSR1	El sistema debe ser liviano.		X	
STSR2	El sistema debe poseer identificaciones para la correcta ubicación de los sensores.	X		
STSR3	La información proporcionada a los usuarios debe ser clara y precisa.		X	
STSR4	El case del dispositivo debe ser de bajo costo económico.			X
Requerimientos Operacionales				
STSR5	El sistema debe tener una fuente de alimentación continua sin ruido.	X		
STSR6	El sistema debe tener la capacidad de procesar las respuestas fisiológicas del paciente para detectar el nivel de estrés.	X		

STSR7	El sistema debe tener la capacidad de funcionamiento durante el desarrollo del test.	X
STSR8	Se necesita una aplicación que evidencie la detección de los niveles de estrés.	X

Fuente: Autoría.

3.3.3. *Requerimientos del sistema.*

En los requerimientos del sistema (SYSR) se encuentran detalladas las funciones específicas que va a realizar el sistema, estos requisitos se encuentran en la Tabla 4.

Tabla 4. Requerimientos del Sistema

SYSR		
#	REQUERIMIENTOS	PRIORIDAD
		Alta Media Baja
Requerimientos de Uso		
SYSR1	El sistema debe poseer sensores que garanticen comodidad en la prueba.	X
SYSR2	Los sensores deben ser localizados correctamente para la toma de datos.	X
SYSR3	El usuario debe poseer una computadora para la observación de los datos.	X
SYSR4	El dispositivo necesita una fuente de alimentación de 5V.	X
SYSR5	Para encender el dispositivo se debe conectar a su fuente de energía.	X

SYSR6	La toma de datos debe ser en tiempo real.	X
--------------	---	----------

Requerimientos de Interfaz

SYSR7	El sistema debe tener un puerto serial.	X
--------------	---	----------

SYSR8	El sistema embebido debe tener pines de entrada y salida	X
--------------	--	----------

SYSR9	Los sensores deben tener cables plug para su conexión a la placa.	X
--------------	---	----------

Requerimientos de Performance

SYSR10	Los datos recolectados de los 3 sensores deben almacenarse en un archivo csv.	X
---------------	---	----------

SYSR11	La prueba debe iniciar cuando el usuario se encuentre con los sensores ubicados correctamente y el test debe iniciarse después de 30 segundos.	X
---------------	--	----------

SYSR12	El sistema debe tener comunicación serial.	X
---------------	--	----------

SYSR13	Visualización gráfica de las señales fisiológicas.	X
---------------	--	----------

Requerimientos de Modo y Estado

SYSR14	La aplicación proporciona mensajes para su correcto funcionamiento.	X
---------------	---	----------

SYSR15	El sistema brinda el respectivo feedback.	X
---------------	---	----------

Requerimientos Físicos

SYSR16	El sistema debe contener un armazón (case) para la protección de las placas.	X
---------------	--	----------

SYSR17	El case debe ser elaborado en acrílico para su conservación.	X
---------------	--	----------

Fuente: Autoría.

3.3.4. *Requerimientos de arquitectura.*

Los requerimientos de arquitectura (SRSH) hacen referencia a los componentes de software y hardware para su respectivo análisis y selección de los mismos con forme a las necesidades. Estos requisitos se observan en la Tabla 5.

Tabla 5. Requerimientos de Arquitectura

SRSH		
#	REQUERIMIENTOS	PRIORIDAD
		Alta Media Baja
Requerimientos de Diseño		
SRSH1	Diseñar un case que permita la protección de las placas.	X
SRSH2	Los sensores deben ir conectados a la placa MySignals.	X
SRSH3	Sistema protegido por un case.	X
SRSH4	Transmisión de la información mediante comunicación serial.	X
SRSH5	Hardware y Software de distribución libre.	X
SRSH6	La aplicación por computador debe ser práctica y entendible.	X
Requerimientos Lógicos		

SRSH7	Las placas que se van a utilizar deben ser compatibles entre sí.	X
SRSH8	La placa debe tener entradas y salidas analógicas y/o digitales para la conectividad con el sensor de conductancia de la piel.	X
SRSH9	El lenguaje de programación debe ser compatible entre las dos placas.	X
Requerimientos de Hardware		
SRSH10	El sistema embebido debe permitir la conexión de los 3 sensores.	X
SRSH11	El sensor debe medir la frecuencia cardíaca.	X
SRSH12	El sensor debe medir la frecuencia respiratoria.	X
SRSH13	El sensor debe medir la conductancia de la piel.	X
SRSH14	Los 3 sensores deben adaptarse a la placa.	X
SRSH15	El sistema no debe perder la conexión a ningún momento.	X
SRSH16	Los datos de los sensores deben ser medidos al menos 30 segundos.	X
Requerimientos de Software		
SRSH17	La placa principal debe ser de código abierto para que sea editable por el desarrollador.	X
SRSH18	Las señales de los sensores de frecuencia cardíaca y frecuencia respiratoria deben ser visualizadas en la aplicación por computador.	X

SRSH19	La aplicación por computador debe ser realizada en código abierto.	X
SRSH20	El software para realizar la aplicación debe tener acceso a la inserción de archivos multimedia como imágenes y videos.	X
SRSH21	El software para realizar la aplicación debe ser compatible con el archivo csv para el almacenamiento de los datos obtenidos por los sensores.	X
Requerimientos Eléctricos		
SRSH22	Las placas deben poseer un voltaje para el funcionamiento desde 5V hasta 12V.	X

Fuente: Autoría.

3.4. Elección del Software y Hardware para el diseño

Para la elección del *Software* y *Hardware* se basa en los requerimientos antes mencionados que se necesitan y se procede a realizar una comparativa con las diferentes herramientas con la finalidad de determinar la que cumpla con el máximo número de requisitos.

3.4.1. Elección del Hardware.

La elección del *Hardware* se elabora con referencia a la Tabla 4 donde se encuentra los requerimientos del sistema y a la Tabla 5 de los requerimientos de arquitectura. Esta elección se realizará para conocer el sistema embebido que se utilizará, para lo cual se asigna el valor “1” cuando cumple con el requerimiento y el valor “0”

cuando no cumple, dando como resultado el que más puntuación obtenga será el seleccionado para trabajar.

Tabla 6. Elección del Hardware

Hardware	Requerimientos							Valor
	SYSR6	SRS7	SRS8	SRS10	SRS17	SRS9	SRS22	Total
Arduino Uno	1	1	1	1	1	1	1	7
Raspberry Pi 2	1	1	0	0	1	1	1	5

1 Cumple

0 No cumple

Elección: Según la tabla de requerimientos, el hardware más apto es el Arduino Uno ya que cumple con el mayor número de requisitos.

Fuente: Autoría.

En la Tabla 7 se puede evidenciar las características que presenta el *hardware* seleccionado que es en este caso el Arduino Uno ya que es el más compatible para la placa *MySignals*.

Tabla 7. Especificaciones Técnicas de Arduino UNO

Arduino UNO	
Microcontrolador	ATmega328
Voltaje Operativo	5v
Voltaje de Entrada	7-12v
Pines de Entrada/Salida Digital	14 (de las cuales 6 son salidas PWM)

Pines de Entradas Análogas	6
Memoria Flash	32KB (ATmega328) de los cuales 0,5KB es usado por Bootloader
SRAM	2KB (ATmega328)
EEPROM	1KB (ATmega328)
Velocidad de Reloj	16 MHZ

Fuente:(Arduino Community, 2018). Elaborado por: Tutillo S. 2021

- Elección de *MySignals HW*.

Las características principales de *MySignals* se puede observar en la Tabla 8 donde se encuentran las especificaciones técnicas de dicha placa.

Tabla 8. Especificaciones Técnicas de MySignals.

MySignals	
Arquitectura	Compatible con Arduino
Microprocesador	ATmega328 (Arduino UNO)
Alimentación	Conexión por puerto USB 5V o adaptador de alimentación
Memoria RAM	2KB
Memoria Flash	32KB
Pantalla	TFT (gráficos básicos)
Pantalla Touch	Si
Almacenamiento en la nube	Si
Aplicación iOS/Android	Si

Número de sensores	16 (se utiliza 3 que son ECG, Airflow y GSR)
Tecnologías inalámbricas	BLE, WiFi

Fuente: (Libelium Comunicaciones, 2017) & (Sarraipa et al., 2019). Elaborado por: Tuttillo S. 2021

En base a las especificaciones técnicas de *MySignals* se obtiene que esta placa es apta para el desarrollo del sistema ya que es compatible con la placa Arduino y posee sensores para su trabajo que en este caso serían los sensores de Frecuencia Respiratoria, Frecuencia Cardíaca (ECG) y Conductancia de la piel (GSR) como se muestra en la figura 19.

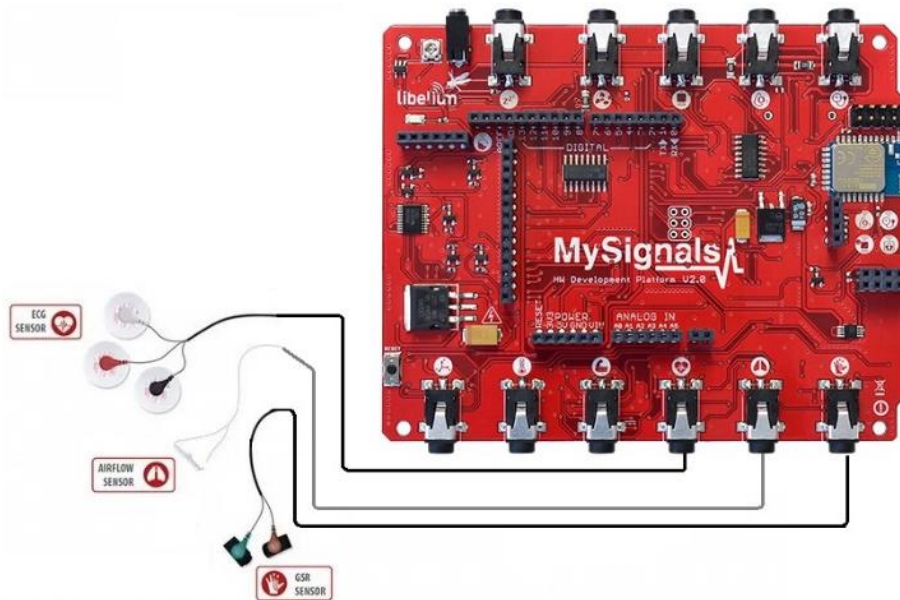


Figura 19. Placa MySignals con sensores seleccionados. Fuente: Autoría basado en (Sarraipa et al., 2019)

3.4.2. Elección del Software.

Esta elección del *Software* se realiza en base a los requerimientos presentados anteriormente en las tablas 4 y 5. En la tabla 9 se presenta la selección del mejor *software* para la realización de la aplicación y el lenguaje de programación que se va a utilizar.

Tabla 9. Elección del Software

Software	Requerimientos							Valor
	SYSR6	SYSR10	SRSR6	SRSR17	SRSR9	SRSR19	SRSR20	Total
IDE de Arduino	1	1	0	1	1	0	1	5
Raspberry (Phyton)	1	1	0	0	1	1	1	5
Processing	0	1	1	0	1	1	1	5
Visual Studio	0	1	1	0	0	1	1	4

1 Cumple

0 No cumple

Fuente: Autoría.

Los tipos de programación IDE de Arduino y Raspberry (Phyton) cumplen con los mismos requerimientos planteados anteriormente ya que se basan en librerías de módulos los cuales son de ayuda para trabajar en el desarrollo del proyecto. Como se va a trabajar con la placa Arduino Uno se considera que el mejor lenguaje de programación a utilizar en este proyecto es el IDE de Arduino. Y también, para el desarrollo de la aplicación se selecciona Processing puesto que cumple con los requisitos que presenta el proyecto y, además, es un *software* de visualización gratuito donde se permite la creación de plataformas graficas interactivas mediante la lectura de diferentes sensores.

3.5. Recursos

En recursos se hace referencia a los elementos que son necesarios e importantes para el desarrollo del sistema denominado BIOSTRESS en base al análisis que se realiza

previamente. Se detalla a continuación los recursos humanos, económicos y tecnológicos que se utilizan en el proyecto.

3.5.1. Recursos Humanos.

Se detalla a las personas que integran de forma directa en la investigación, esto se observa en la Tabla 10.

Tabla 10. Recursos Humanos

Recursos Humanos	
ESTUDIANTES	Estudiantes de la Universidad Técnica del Norte (CIERCOM)
TUTOR	Ing. Jaime Michilena
DESARROLLADOR	Srta. Samia Tutillo

Fuente: Autoría.

3.5.2. Recursos Económicos.

Los recursos económicos se enlistan en la Tabla 11, donde se observa el *hardware* utilizado para el dispositivo y el costo final aproximado. Cabe recalcar que el estudiante financia los gastos.

Tabla 11. Recursos Económicos

RECURSOS ECONÓMICOS			
DESCRIPCIÓN	CANTIDAD	COSTO	COSTO
		UNITARIO	TOTAL
Placa MySignals Basic	1	120	120
Arduino UNO	1	11	11

Sensor de Frecuencia Cardíaca AD8232	1	26	26
Sensor GSR	1	36	36
Electrodos	25	1,50	37,50
		TOTAL	230,50

Fuente: Autoría.

3.5.3. Recursos Tecnológicos.

En este apartado se especifican los diferentes *softwares* que se utilizan en el desarrollo del proyecto, esto se observa en la Tabla 12 a continuación.

Tabla 12. Recursos Tecnológicos

RECURSOS TECNOLÓGICOS	
Software	Costo
IDE Arduino	0
Processing	0
Microsoft Excel	0
TOTAL 0	

Fuente: Autoría.

3.6. Diseño del sistema

De acuerdo con la metodología del Modelo en V, se realiza la siguiente etapa que es el diseño del prototipo donde se especifica el funcionamiento del proyecto de manera ordenada.

En esta fase se encuentran los aspectos fundamentales para el desarrollo de un sistema *Biofeedback* de sensores de detección de estrés, tomando en cuenta la adquisición

de datos mediante bioseñales producidas por el cuerpo en estudiantes universitarios, considerando los criterios propuestos durante los requerimientos del sistema.

3.6.1. Descripción general del funcionamiento del sistema BIOSTRESS.

Este sistema consiente en detectar el nivel de estrés de los estudiantes universitarios a través de algoritmos de aprendizaje automático.

El funcionamiento de BIOSTRESS consiste en iniciar con el test donde el administrador del sistema debe ejecutar la aplicación en computador, seguidamente se presiona el botón continuar y se procede a medir las bioseñales causadas por los estudiantes universitarios, para ello se ubica de manera correcta los sensores al paciente.

Se empieza obteniendo los datos de frecuencia cardíaca, donde se presiona iniciar y se espera un minuto hasta que se obtenga las pulsaciones del paciente, seguidamente se conecta el sensor de frecuencia respiratoria y de igual manera se presiona iniciar para obtener las respiraciones en el lapso de un minuto y por último se obtiene los valores de la conductancia de la piel igualmente en el lapso de un minuto. Una vez obtenido los valores se presiona en continuar y se visualiza el nivel de estrés que posee el estudiante. La información obtenida es almacenada automáticamente en un archivo csv.

Finalmente, para los resultados existen tres niveles de estrés que son normal, algo elevado y alto; de acuerdo con el resultado de cada paciente se inicia con el Biofeedback correspondiente donde el paciente debe realizar un ejercicio de respiración para calmar sus datos y consecuentemente ayudar a que disminuya el nivel de estrés que poseen los estudiantes.

3.6.2. Arquitectura.

Se presenta el funcionamiento del sistema por medio de un diagrama de bloques que se observa en la figura 20, donde constan los bloques del sistema BIOSTRESS.

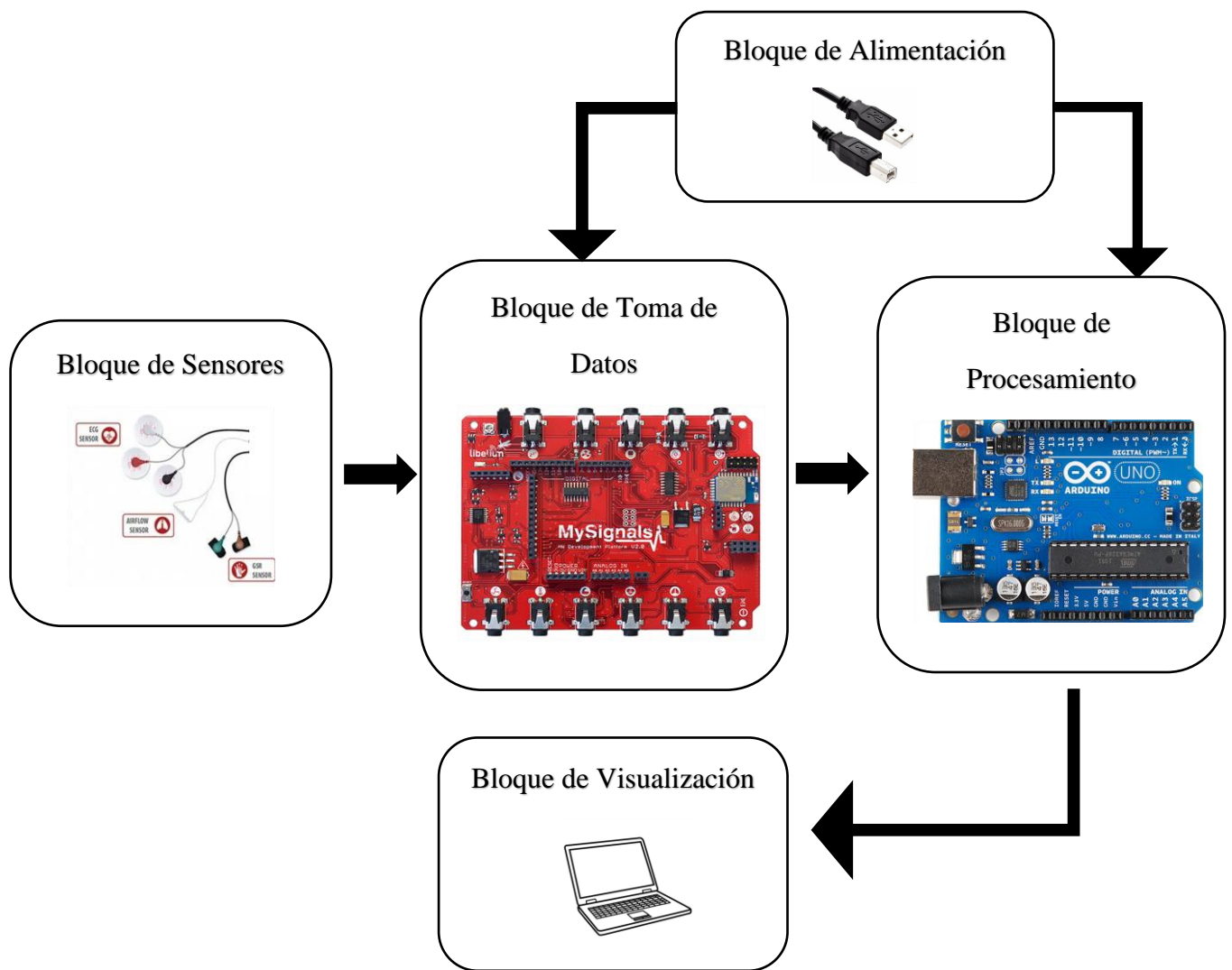


Figura 20. Diagrama de Bloques de BIOSTRESS. Fuente: Autoría.

La arquitectura del sistema BIOSTRESS consta de 5 bloques los cuales cuentan con funciones propias para cada uno, estas se describen a continuación:

- **Bloque de alimentación.** - Es primordial la funcionalidad de este bloque puesto que el sistema trabaja con una alimentación de 5V, los cuales proporciona la entrada USB del ordenador.
- **Bloque de sensores.** - Este bloque cuenta con los sensores del sistema los cuales son los encargados de la recolección de bioseñales de los pacientes tales como la frecuencia cardíaca, frecuencia respiratoria y conductancia de la piel. Esta fase

debe tener un análisis de la ubicación adecuada de los sensores para que los datos sean medidos correctamente.

- **Bloque de toma de datos.** – En esta fase, la placa *MySignals* recibe los datos proporcionados por el bloque de sensores, los cuales generan señales analógicas para su correspondiente análisis.
- **Bloque de procesamiento.** – Este bloque se encuentra conformado por la placa Arduino que trabaja juntamente con la placa *MySignals*, En esta fase se procesa la información anteriormente obtenida para enviarla a la interfaz de la aplicación por computador.
- **Bloque de visualización.** - Por último, se tiene el bloque donde el administrador puede ver los resultados del test y el paciente puede observar el respectivo Biofeedback. Esta sección es trabajada gracias a la ayuda del *software Processing*.

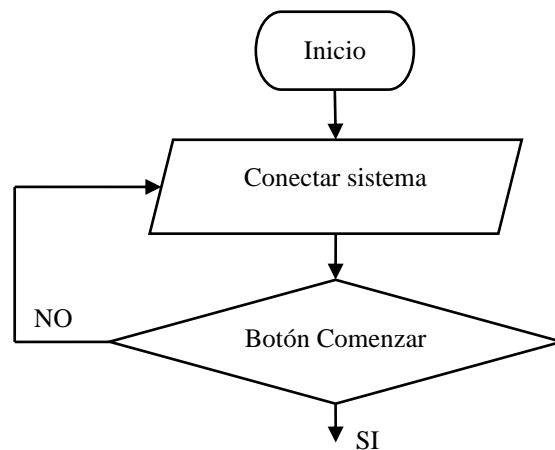
3.6.2.1. Diagrama de Flujo del Proceso.

Posteriormente de la elaboración del diagrama de bloques se realiza el proceso que ejecutará el sistema con la obtención de datos proporcionados por los sensores, esto se representa a través un diagrama de flujo que describe el proceso para detectar mediante algoritmos de aprendizaje de máquinas el nivel de estrés de los estudiantes.

En la figura 21 se puede observar dicho diagrama, para el cual se empieza conectando el sistema y presionando la opción comenzar, si este botón no es accionado se vuelve a conectar el sistema. Continuando con el proceso, se conecta el sensor ECG, se presiona el botón inicio y posteriormente se adquiere sus valores, si esto no se cumple se vuelve a enlazar el sensor y obtener los datos correctamente. Una vez terminado con el sensor ECG se conecta el sensor *Airflow* y se sigue el mismo procedimiento hasta obtener los valores, de igual forma si no sucede de la manera adecuada se vuelve a

conectar el sensor. Para terminar con la etapa de los sensores se obtienen los valores del sensor GSR, y, si toda la información obtenida por los tres sensores se encuentra disponible se continua con la visualización del resultado, si esto no sucede así se regresa a conectar el primer sensor ECG y realizar el procedimiento mostrado hasta obtener todos los datos apropiados.

En el resultado se puede evidenciar tres opciones que son normal, algo elevado y alto, cuando el resultado es normal se llega al final del proceso del sistema, y cuando el resultado varía, es decir, resulta algo elevado y alto se procede con la ejecución del *Biofeedback* para normalizar los datos obtenidos y con eso se concluye todo el proceso del sistema BIOSTRESS.



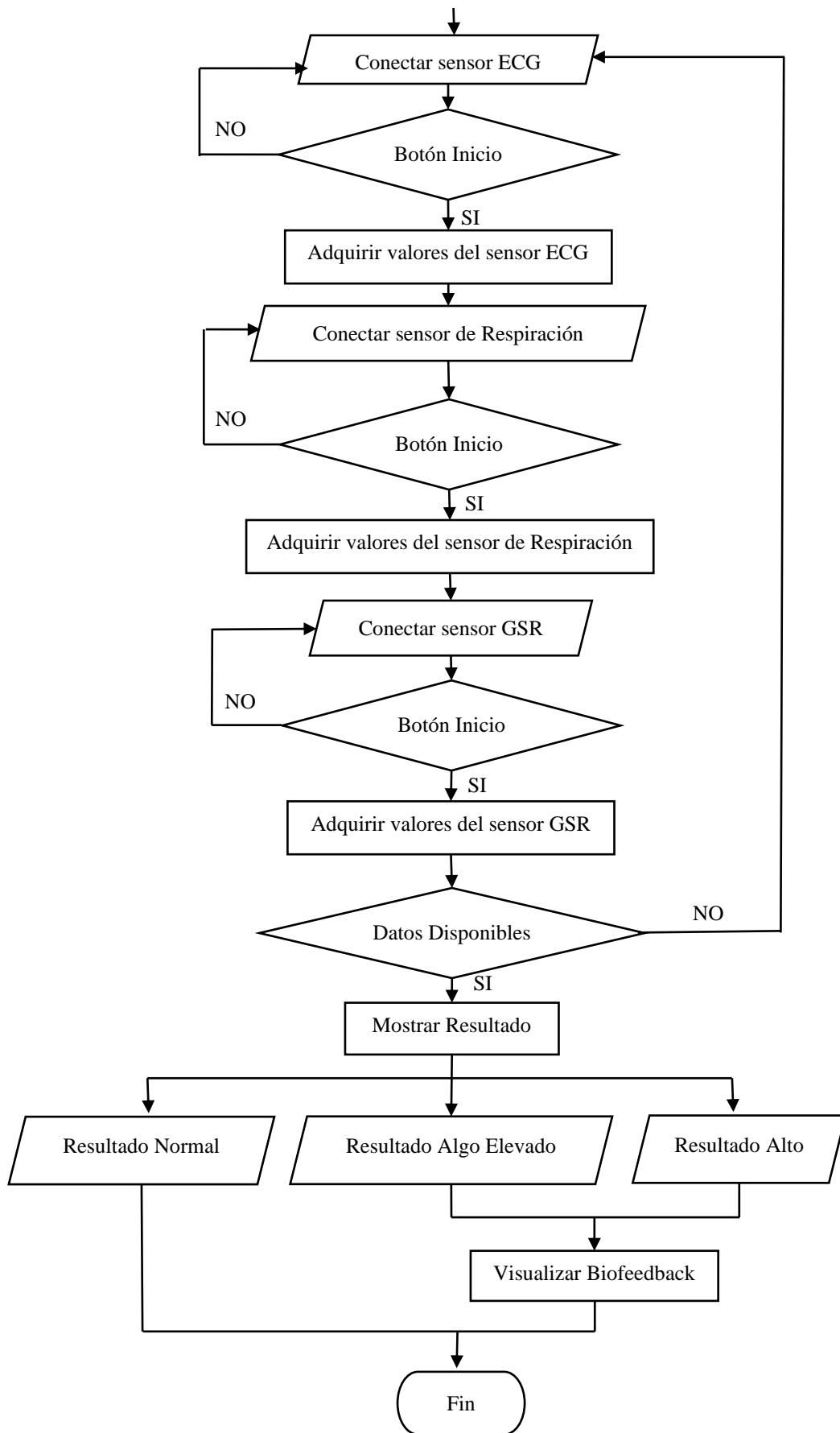


Figura 21. Diagrama de Flujo del proceso de BIOSTRESS. Fuente: Autoría.

3.7. Diseño detallado del Sistema BIOSTRESS.

En esta fase se elabora el diseño en detalle de cada uno de los bloques del sistema BIOSTRESS considerando los requerimientos planteados anteriormente en los apartados 3.3.3 y 3.3.4.

3.7.1. Diseño del bloque de alimentación.

Para el diseño de este bloque se tiene en consideración los requerimientos de los apartados antes mencionados, se necesita mínimo 5V para el funcionamiento del sistema, la alimentación de la entrada USB del computador es la encargada de proporcionar este voltaje, esto se puede observar en la figura 22, donde se evidencia que desde el computador se hace uso de ese voltaje trasladándolo directamente a la placa Arduino y alimentando al mismo tiempo a la placa *MySignals*.

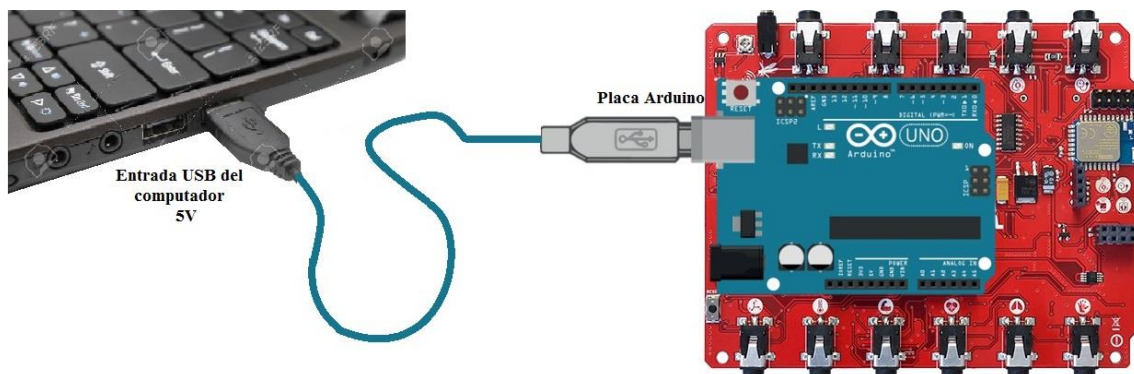


Figura 22. Diseño de conexión del circuito de encendido del computador hacia la placa Arduino. Fuente:

Autoría basado en (Sarraipa et al., 2019)

Para la alimentación de la placa *MySignals* se necesita de la misma manera 5V, para lo cual el voltaje es el mismo suministrado por el ordenador, puesto que, esta placa se encuentra conectado con la placa Arduino y consumen el mismo voltaje. Los pines de la placa *MySignals* son los que se conectan con la placa Arduino, posee 12 entradas análogas y 18 digitales que se acoplan de una forma fácil, en la figura 23 se puede observar la conexión de las dos placas utilizadas en el proyecto.

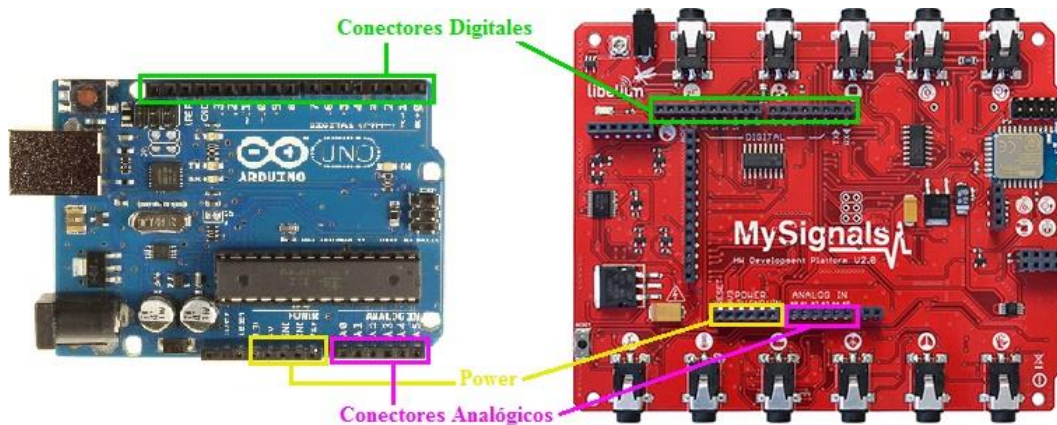


Figura 23. Diagrama de conexión de las placas. Fuente: Autoría basado en (Sarraipa et al., 2019)

3.7.2. *Diseño del bloque de sensores.*

En este bloque se utiliza los sensores idóneos para la recolección de bioseñales de frecuencia cardíaca, frecuencia respiratoria y conductancia de la piel, los cuales son tres de los parámetros médicos que determinan el estrés. En la figura 24 se encuentra el flujograma donde se observa el orden en el cual deben ir ubicados los sensores en el dispositivo para que se obtenga las bioseñales del paciente.

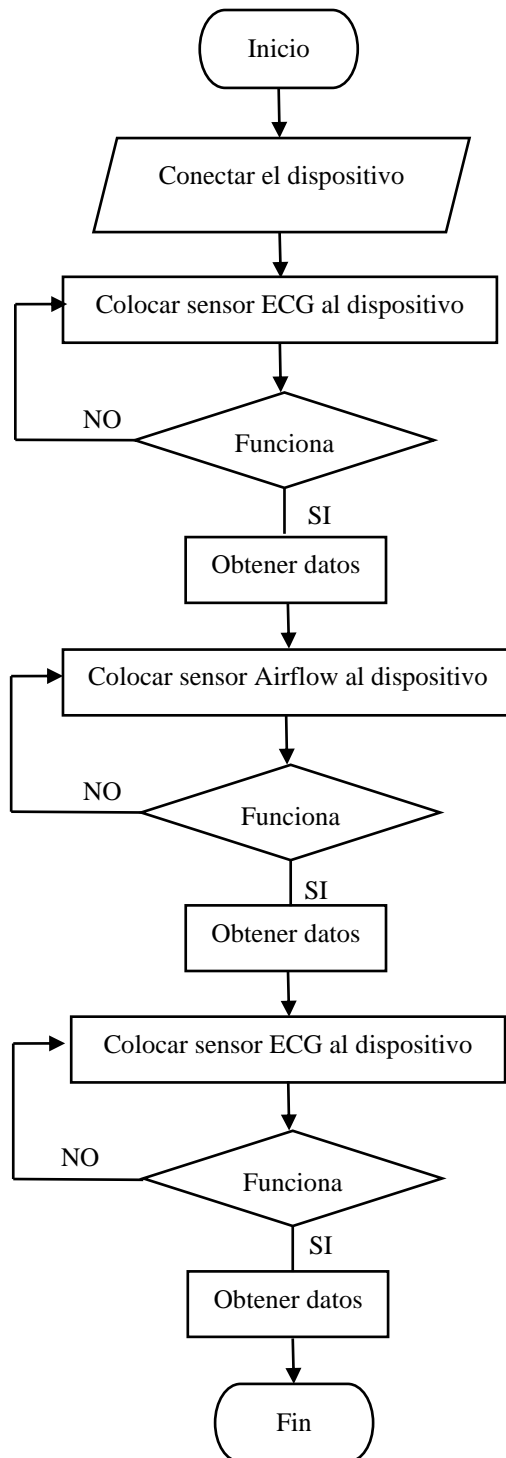


Figura 24. Diagrama de flujo del bloque de sensores. Fuente: Autoría.

- **Sensor de Flujo de aire.**

Este sensor (*Airflow*) se observa en la figura 25 y sirve para calcular la frecuencia respiratoria de las personas, esta se mide en respiraciones por minuto (RPM) cuando el paciente se encuentra en estado de reposo. Como este proyecto se encuentra dirigido a

estudiantes universitarios se les considera personas adultas por lo cual las respiraciones por minuto normales van de 15 a 20. (Tomalá, 2018)



Figura 25. Sensor de Flujo de aire. Fuente: (Sarraipa et al., 2019)

En la tabla 13 se puede encontrar los parámetros que puede llegar a medir el sensor *Airflow*. Para tener la lectura precisa de la frecuencia respiratoria se debe colocar el sensor en el paciente de manera correcta como se puede observar en la figura 26.

Tabla 13. Parámetros del sensor *Airflow*

Parámetro	Unidad	Rango
Frecuencia Respiratoria	RPM	0-200 RPM
Intensidad Respiratoria	Voltios	0 – 3,3 V

Fuente: (Libelium Comunicaciones, 2017)

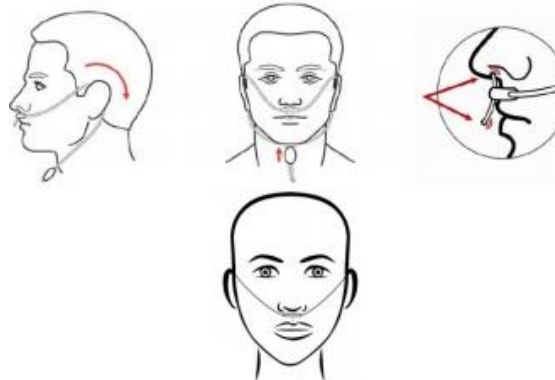


Figura 26. Posición adecuada del sensor *Airflow*. Fuente: (Sarraipa et al., 2019)

El diagrama de conexión del sensor *Airflow* a la placa *MySignals* se representa de la manera como se encuentra en la figura 27, donde se observa que el adaptador plug macho va ubicado en el adaptador plug hembra de la placa simbolizado por el gráfico de pulmones.

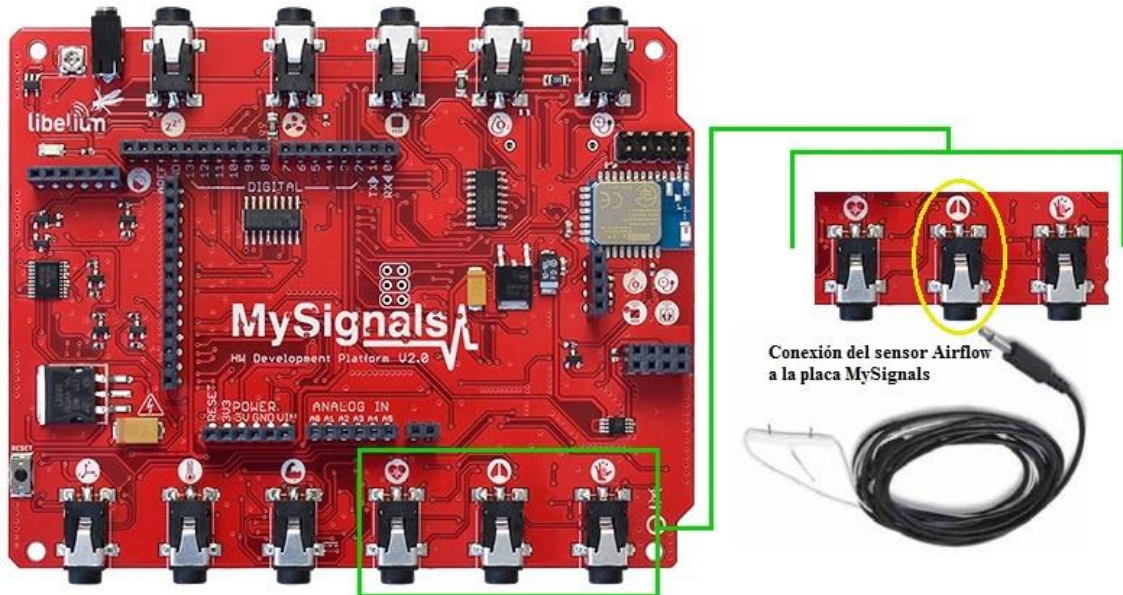


Figura 27. Conexión del sensor Airflow. Fuente: Autoría basado en (Sarraipa et al., 2019)

- **Sensor de Frecuencia Cardíaca (ECG).**

Este sensor se puede ver en la figura 28, sirve para medir la actividad eléctrica del corazón lo que produce una señal de salida confiable que puede ser graficada como electrocardiograma y se emite como una lectura analógica.

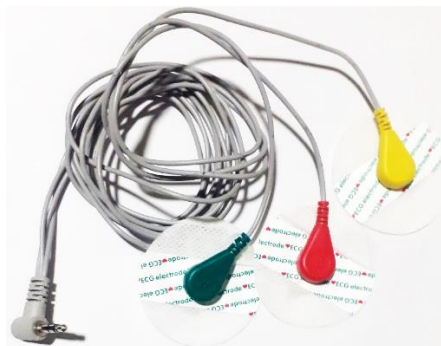


Figura 28. Sensor ECG. Fuente: Autoría.

En la tabla 14 se puede encontrar las mediciones que posee este sensor ECG entre las cuales se encuentra el rango de latidos por minuto (Beats per minute BPM) que puede llegar a medir.

Tabla 14. Parámetros del sensor ECG.

Parámetro	Unidad	Rango
Frecuencia del pulso	BPM	0-200 BPM
Señal de electrocardiograma	Voltios	0 – 5 V

Fuente: (Libelium Comunicaciones, 2017)

Los valores que se encuentran en el rango para personas adultas son entre 60 y 80 latidos por minuto (BPM). (Tomalá, 2018). Este sensor debe ser colocado correctamente para obtener la señal con el menor ruido posible, la manera idónea para colocar al paciente es como se observa en la figura 29.

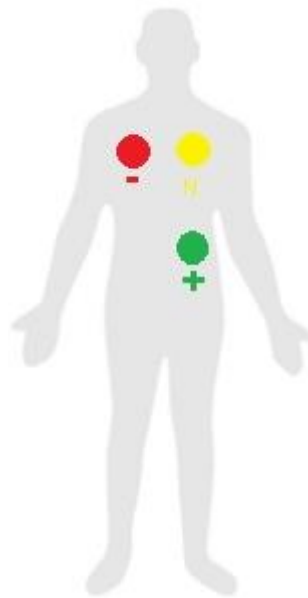


Figura 29. Posición adecuada del sensor ECG. Fuente: (Sarraipa et al., 2019)

El diagrama de conexión del sensor ECG a la placa *MySignals* se observa en la figura 30, donde se encuentra que el adaptador plug macho va ubicado en el adaptador plug hembra de la placa donde se encuentra señalado por el gráfico de un corazón.

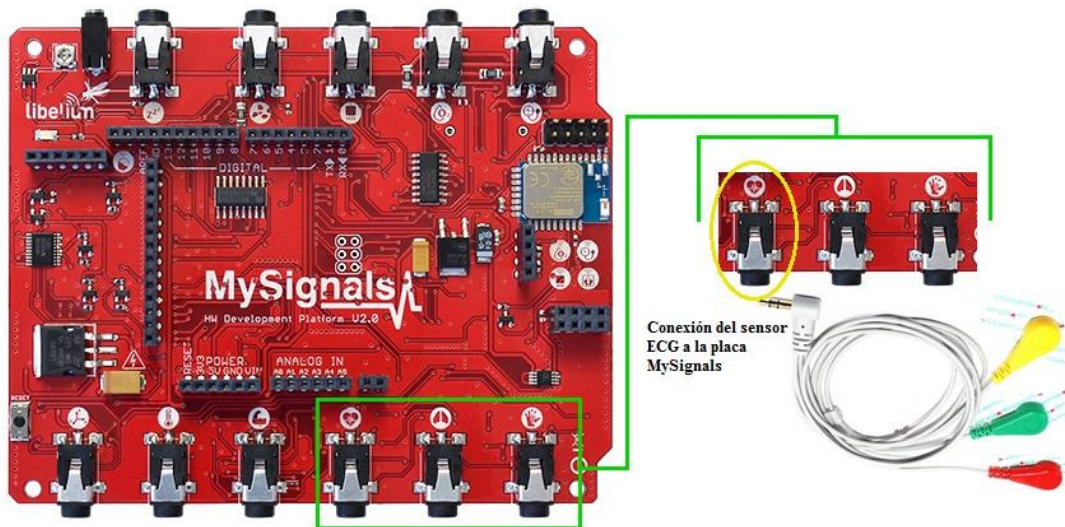


Figura 30. Conexión del sensor ECG. Fuente: Autoría basado en (Sarraipa et al., 2019)

- **Sensor de Conductancia de la piel (GSR)**

Este sensor GSR es utilizado para medir la conductividad eléctrica de la piel la cual varía por el nivel de humedad, esta se mide entre dos puntos y es esencialmente un tipo de ohmímetro. En la figura 31 se puede observar el sensor GSR.



Figura 31. Sensor GSR. Fuente: (Libelium Comunicaciones, 2017)

El funcionamiento del sensor GSR consiste en calcular la resistencia eléctrica de la piel en función del sudor producido por el cuerpo, es decir que mide el reflejo psico galvánico del cuerpo. Los parámetros de este sensor se encuentran en la tabla 15.

Tabla 15. Parámetros del sensor GSR

Parámetro	Unidad	Rango
Conductancia	Siemens	0-20 Siemens

Resistencia	Ohmios	10K – 100KOhms
Voltaje	Voltios	0 – 5 V

Fuente:(Libelium Comunicaciones, 2017)

Los valores del sensor GSR que se encuentran en el rango normal para las personas adultas están entre 350 a 450 y la correcta ubicación de los electodos se observan en la figura 32.

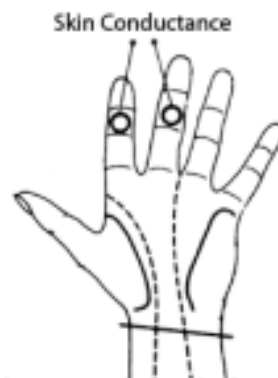


Figura 32. Posición adecuada del sensor GSR. Fuente: (Libelium Comunicaciones, 2017)

El diagrama de conexión del sensor GSR a la placa *MySignals* sigue el mismo procedimiento de los anteriores sensores con la única diferencia que se encuentra conectado al plug de la placa que posee el gráfico de una mano, esto se observa en la figura 33.

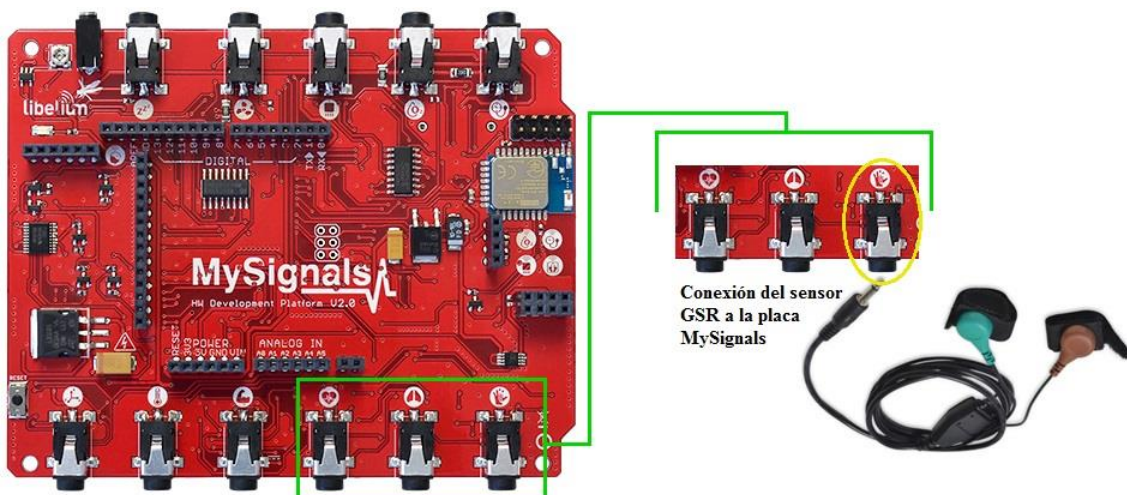


Figura 33. Conexión del sensor GSR. Fuente: Autoría basado en (Libelium Comunicaciones, 2017)

3.7.3. *Diseño del bloque de toma de datos y bloque de procesamiento.*

En estos bloques se explica la conexión del sistema que consiste en indicar todos los elementos que conforman BIOSTRESS para realizar la obtención de datos y procesamiento de la señal. Este sistema como se mencionó en el anterior apartado está compuesto de la placa *MiSignals*, Arduino Uno y los tres sensores mencionados.

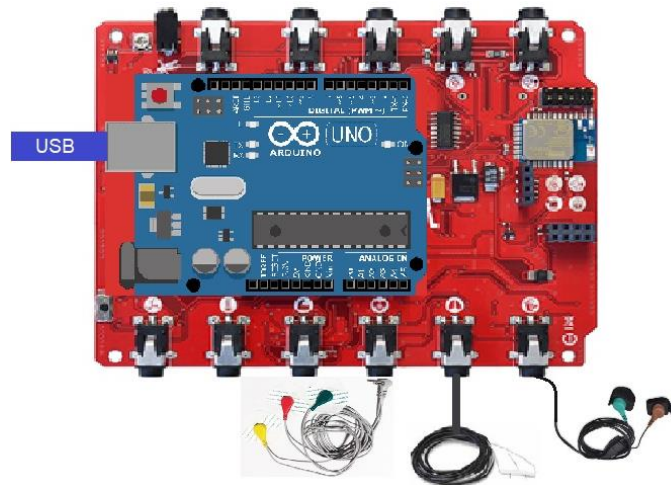


Figura 34. Esquema de conexión del sistema. Fuente: Autoría basado en (Sarraipa et al., 2019).

En la figura 34 se observa el esquema de conexión del sistema BIOSTRESS donde se debe tener en cuenta las siguientes especificaciones:

- La placa *MySignals* es montada sobre la placa Arduino Uno de tal manera que los pines se acoplen adecuadamente entre sí.
- La fuente de alimentación consta del cable USB conectado al Arduino y al ordenador que se utilice.
- Los tres sensores utilizados no requieren de alimentación externa puesto que trabajan adecuadamente con la alimentación proporcionada por el cable USB conectado al ordenador.
- Los sensores *Airflow* y ECG son conectados mediante un cable tipo plug al pin de los iconos de pulmones y corazón respectivamente que se encuentran dibujados en la placa *MySignals*.

3.7.3.1. Bloque de Toma de Datos.

La recolección de la información proveniente del bloque de sensores se analiza en esta etapa donde se crea los códigos de programación respectivos. Se elabora un algoritmo para obtener los datos, en el cual se lee la lectura analógica de los sensores y posteriormente se envía la información al bloque de procesamiento, esto se puede observar en el diagrama de flujo de la figura 35.

En esta figura se observa que para iniciar se definen las librerías del módulo *MySignals*, la librería I2C que sirve para la comunicación de las dos placas y la librería SPI que es requerida por I2C. Como siguiente punto, se declaran las variables que sirven para almacenar los datos de los sensores y se procede a inicializar el puerto serial con la función `Serial.begin()` y la función `MySignals.begin()` para inicializar la comunicación con la placa, esto se realiza con la finalidad de conectar el Arduino con *MySignals*.

Una vez conectado los sensores pasa a la lectura de datos donde cada sensor cuenta con una función diferente, la función que se utiliza para la lectura del sensor *Airflow* es `MySignals.getAirflow ()`, para el sensor ECG se utiliza la función `MySignals.getECG ()` y para la lectura del sensor GSR se utiliza la función `analogRead ()` puesto que este sensor se encuentra conectado directamente a la placa. Para almacenar los datos se utiliza la función `saveTable()` y se encuentra en el archivo csv.

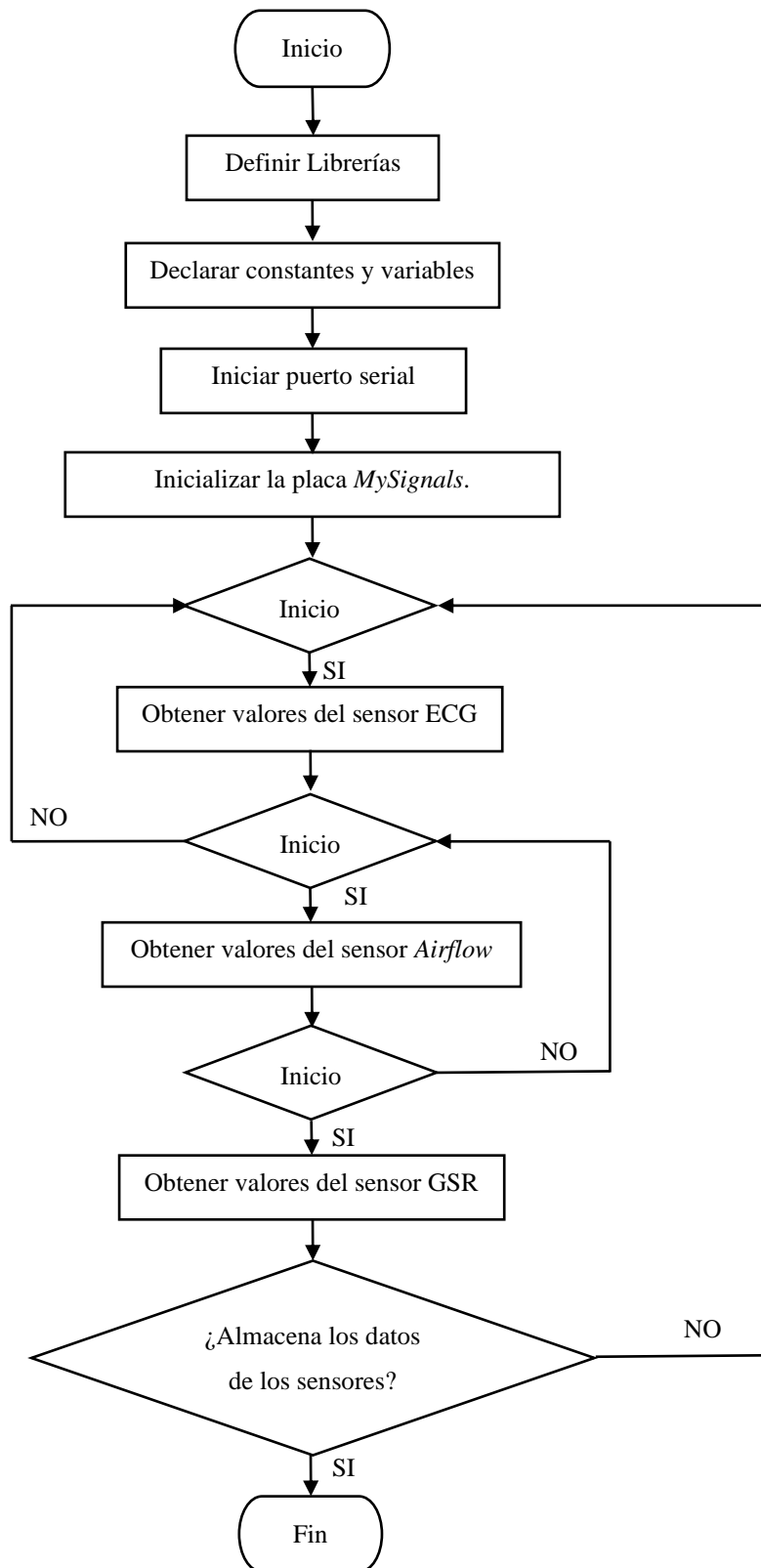


Figura 35. Diagrama de flujo de la toma de datos del sistema BIOSTRESS. Fuente: Autoría.

El conjunto de entrenamiento para el desarrollo del sistema BIOSTRESS se lo realiza mediante el análisis de pruebas realizadas a estudiantes y en base a datos

proporcionados por profesionales en el ámbito de la salud, obteniendo una base de datos apropiada para las pruebas del sistema. Los datos obtenidos por los sensores se almacenan en un archivo csv, puesto que, cuenta con una especie de tabla con filas y columnas donde se encuentra el registro separado y clasificado con los valores de cada sensor, esto ayuda al procesamiento de los mismos para obtener el resultado esperado, en este caso los tres niveles de estrés. La estructura de la base de datos de entrenamiento se muestra en la tabla 16, donde se tiene las columnas de los valores del sensor ECG, *Airflow* y GSR.

Tabla 16. Estructura de la BDD inicial.

BPM	RPM	GSR
X	X	X

Fuente: Autoría.

Donde: **X** es igual al valor obtenido por los sensores.

3.7.3.2. Bloque de Procesamiento.

Una vez que se obtiene los valores de los sensores en una BDD inicial se continua con el procesamiento donde se trata la información obtenida y consecuentemente se muestra una predicción sobre los niveles de estrés de los estudiantes, en esta etapa se debe implementar un algoritmo de aprendizaje de máquinas. En este proyecto se implementará el algoritmo de clasificación KNN puesto que este algoritmo trabaja con dos conjuntos de datos y utiliza la distancia más corta entre el nuevo ejemplo y todos los datos de entrenamiento para determinar el grupo correcto. (Godoy Trujillo, 2017). Como lo menciona (Berástegui Arbeloa & Galar Idoate, 2018), el algoritmo KNN (*k – Nearest Neighbors*) clasifica los datos a partir de otros de los que ya se conoce su clase y a los que se denomina conjunto de entrenamiento.

El proceso que realiza la placa para determinar el nivel de estrés de los estudiantes se observa en el diagrama de flujo de la figura 36.

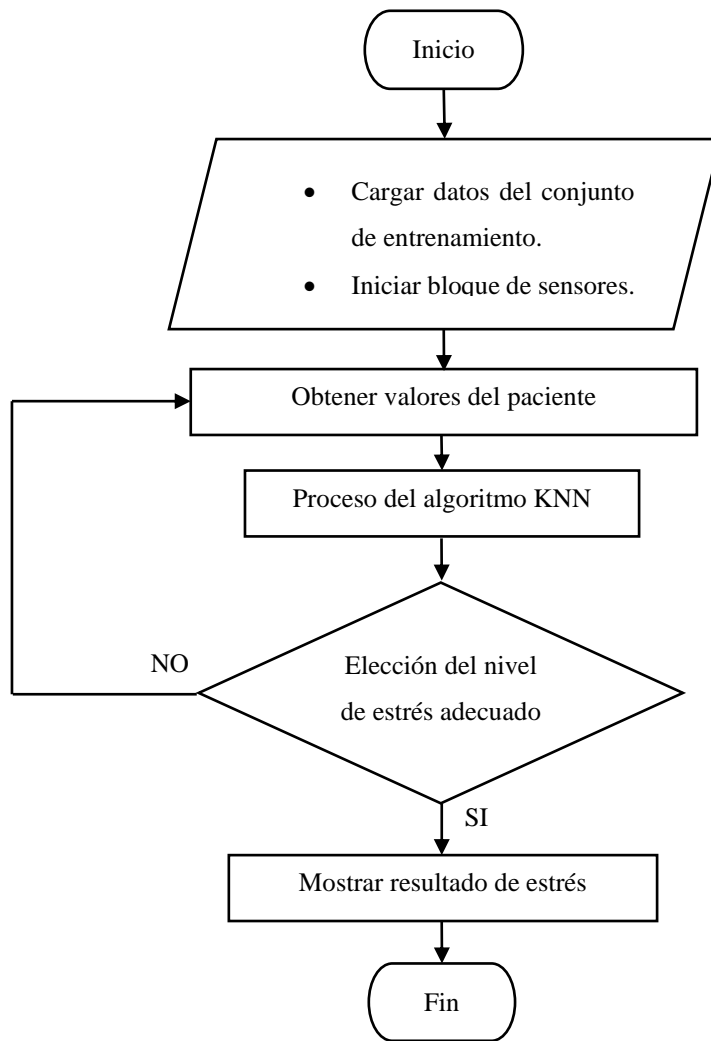


Figura 36. Diagrama de flujo del proceso del algoritmo KNN. Fuente: Autoría.

Con el conjunto de datos de entrenamiento se procede a clasificar en tres estados establecidos como se observa en la tabla 17, donde se añade dos campos a la base de datos anteriormente presentada en el apartado 3.7.3.1.

Tabla 17. Estructura de la BDD inicial más dos parámetros.

BPM	RPM	GSR	Clase	Estado
X	X	X	Y	Z

Fuente: Autoría.

Donde:

X = valor obtenido por los sensores

Y = valor 1,2 o 3 que selecciona el algoritmo

Z = nivel de estrés (normal, algo elevado o alto)

Para agregar los dos campos a la BDD inicial se basa en la tabla 18, donde se tiene el rango de los niveles de estrés, el cual será seleccionado por el trabajo que realiza el algoritmo KNN.

Tabla 18. Rangos de los niveles de estrés.

Rango			Nivel de estrés
BPM	RPM	GSR	
65 o menos	20 o menos	De 350 a 450	Normal
De 66 a 78	De 21 a 27	De 300 a 350	Algo Elevado
Más de 79	Más de 28	Menos de 300	Alto

Fuente: Autoría.

3.7.3.2.1. Algoritmo KNN. El trabajo que realiza el algoritmo KNN se presenta en el diagrama de bloques de la figura 37, donde como primer punto de este diagrama se debe cargar el conjunto de datos de entrenamiento que se realizó anteriormente.

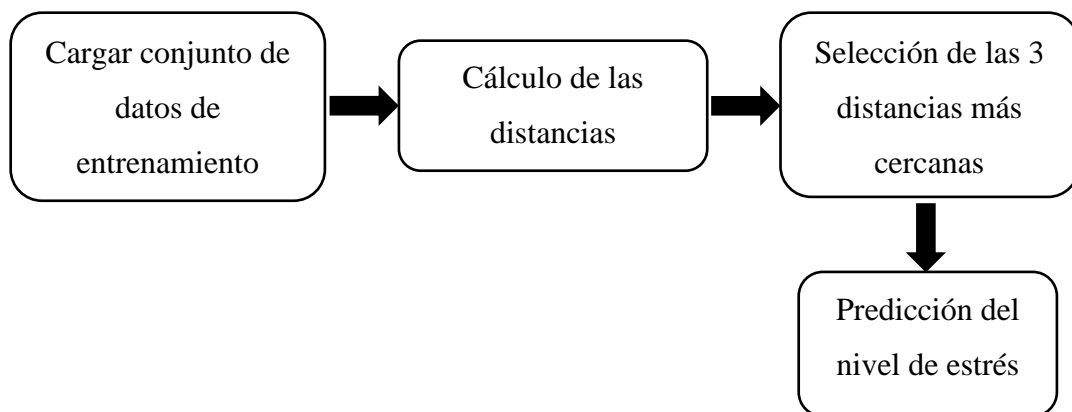


Figura 37. Diagrama de bloques del algoritmo KNN. Fuente: Autoría.

Siguiendo con el diagrama de bloques es necesario calcular las distancias de todos los valores de la BDD de entrenamiento hacia el nuevo dato ingresado al momento de comenzar el test, con esto se puede encontrar el vecino más cercano y obtener las tres menores distancias para así determinar el nivel de estrés al que pertenece el dato nuevo, para que se pueda enviar la información al bloque de visualización

En Arduino se ingresa la clase void denominada Entrenamiento KNN donde se encuentra el conjunto de datos de entrenamiento, como se muestra en la figura 38.

```
void EntrenamientoKNN() {  
  float example1[] = {14, 50, 440};  
  float example2[] = {13, 54, 425};  
  float example3[] = {17, 63, 364};  
  float example4[] = {18, 55, 355};  
  float example5[] = {17, 55, 430};  
  float example6[] = {18, 64, 420};  
  float example7[] = {16, 50, 440};  
  float example8[] = {14, 58, 375};  
  float example9[] = {20, 65, 435};  
}
```

Figura 38. Algoritmo KNN en Arduino. Fuente: Autoría.

3.7.3.2.2. Casos Críticos. Para que el algoritmo KNN cumpla un buen funcionamiento se parte de los casos críticos del proyecto, que son los siguientes:

- **Caso 1:** Cuando un parámetro medido es normal y los otros dos algo elevados.
- **Caso 2:** Cuando un parámetro medido es normal y los otros dos altos.
- **Caso 3:** Cuando se tiene un valor normal, un valor algo elevado y otro valor alto.
- **Caso 4:** Cuando se tiene un parámetro algo elevado y los otros dos normales.
- **Caso 5:** Cuando se tiene un parámetro algo elevado y los otros dos altos.
- **Caso 6:** Cuando se tiene un parámetro alto y los otros dos normales.
- **Caso 7:** Cuando se tiene un parámetro alto y los otros dos algo elevados.

Según el profesional en el tema del estrés se tiene que el valor que debe primar es el de la conductancia de la piel, con esto se debe obtener los niveles de estrés como se muestra en la tabla 19.

Tabla 19. Casos Críticos.

Rango			Nivel de estrés
BPM	RPM	GSR	
Normal	Algo Elevado	Algo Elevado	Algo Elevado
Normal	Alto	Alto	Alto
Normal	Algo Elevado	Alto	Alto
Algo Elevado	Alto	Alto	Alto
Algo Elevado	Normal	Normal	Normal
Alto	Normal	Normal	Normal
Alto	Algo Elevado	Algo Elevado	Algo Elevado

Fuente: Autoría.

Para que el algoritmo KNN brinde los resultados presentados de los casos críticos se debe tener el conjunto de entrenamiento especificado el valor del sensor GSR con rango más altos, es por ello que se tiene rangos de 50, esto se puede evidenciar en la tabla 18 presentada anteriormente.

3.7.3.2.3. Matriz de confusión. Esta matriz permite ver el rendimiento del algoritmo, en este caso del algoritmo KNN, la estructura de la matriz de confusión se puede encontrar en la tabla 20, donde en cada fila se representan las instancias en una clase predicha, mientras que en cada columna se representa las instancias de una clase real.

Tabla 20. Estructura de la matriz de confusión.

	Real Normal	Real Algo Elevado	Real Alto
Predicho Normal	Verdadero Normal	Falso Normal	Falso Normal

Predicho Algo	Falso Algo Elevado	Verdadero Algo	Falso Algo
Elevado		Elevado	Elevado
Predicho Alto	Falso Alto	Falso Alto	Verdadero Alto

Fuente: Autoría.

Con ayuda de la matriz de confusión se puede verificar la proporción de veces que el algoritmo prefijo correctamente el resultado en base a los datos que se van a medir, esto se obtiene gracias a la ecuación 6.

$$Exactitud = \frac{VN + VAE + VA}{VN + VAE + VA + FN + FAE + FA}$$

Ecuación 6. Exactitud del algoritmo. Fuente: (Paredes Inilupu, 2020)

3.7.4. Diseño del bloque de visualización.

Esta etapa se realiza gracias a la ayuda del *software Processing* ya que fue seleccionado en el apartado 3.4.2. El desarrollo de la interfaz de visualización del sistema sigue el proceso que se observa en el diagrama de bloques de la figura 39.

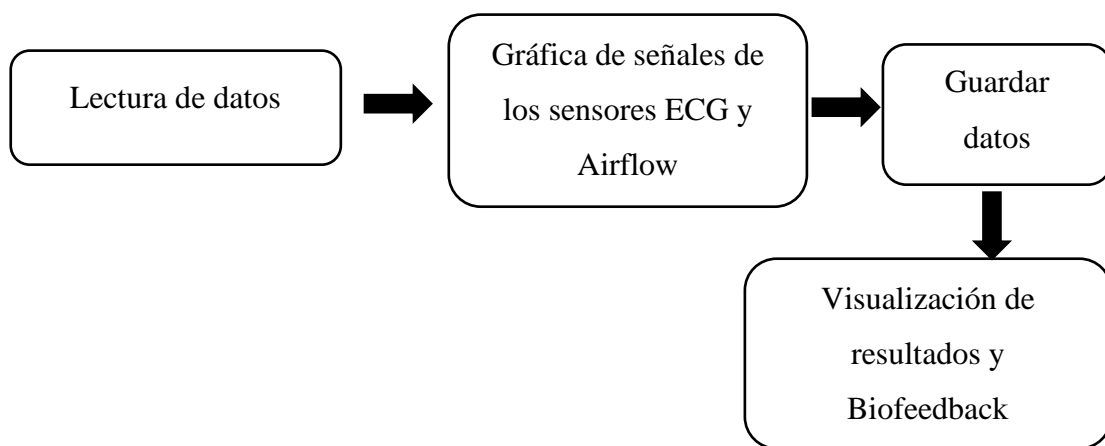


Figura 39. Diagrama de bloques de la fase de visualización. Fuente: Autoría.

- **Lectura de datos:** Los datos obtenidos por medio del bloque de procesamiento son transmitidos a través de comunicación serial debido a que la información tomada es en tiempo real y se necesita el resultado para continuar con el respectivo Biofeedback.

- **Gráfica de señales:** La gráfica de señales se realiza de los sensores ECG y *Airflow* al momento de obtener los datos puesto que, como es en tiempo real, el administrador puede ver como se obtiene la información de los pacientes al realizar el test.
- **Guardar datos:** Una vez que se obtienen los datos se procede a guardar en un archivo csv.
- **Visualización de resultados y Biofeedback:** Se muestra el nivel de estrés que posee la persona en diferentes colores que son verde, amarillo y rojo de acuerdo con el resultado y se procede a realizar el Biofeedback correspondiente siempre y cuando el nivel de estrés sea algo elevado y alto para posteriormente volver a medir los valores y ayudar con la regulación de las bioseñales.

En la figura 40 se puede observar el diagrama de flujo del proceso de la fase de visualización.

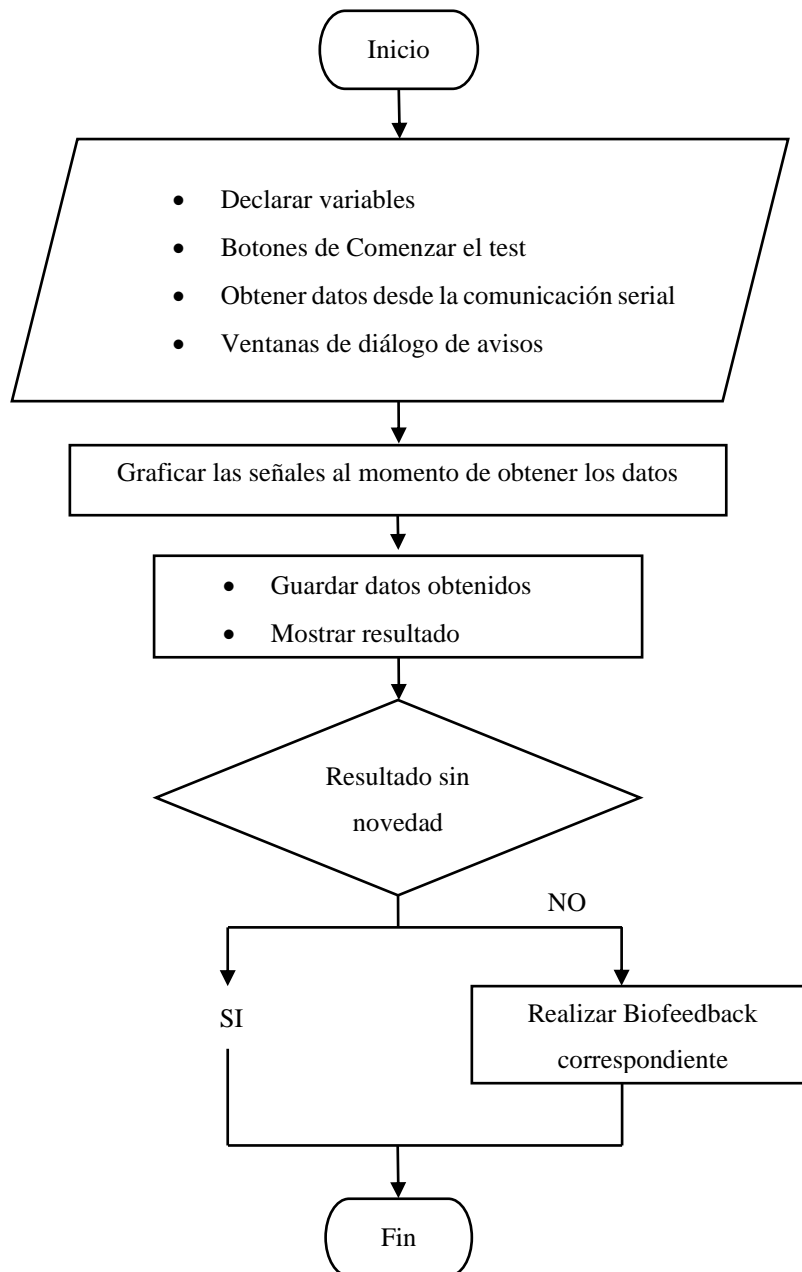


Figura 40. Diagrama de flujo de la fase de visualización. Fuente: Autoría.

3.7.4.1. Diseño del Biofeedback. Para el desarrollo del *Biofeedback* se sigue un procedimiento que consiste en facilitar al paciente de forma inmediata el registro de la actividad biológica que se está monitorizando, transformándola a señales visuales que son percibidas por la persona y mediante un *feedback* regular o controlar estas señales biológicas del paciente, este proceso se observa en el diagrama de bloques de la figura 41.

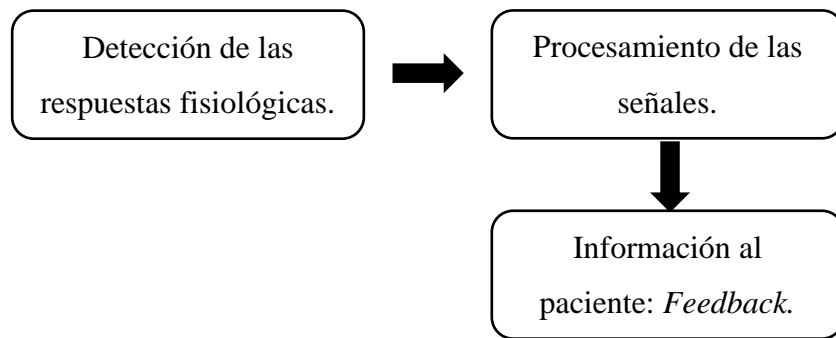


Figura 41. Proceso del *Biofeedback*. Fuente: Autoría.

- Las fases de detección de las respuestas fisiológicas y procesamiento de las señales se encuentran desarrolladas en el diseño de bloque de toma de datos y procesamiento que se mencionaron anteriormente.
- La información al paciente: *Feedback* se debe conseguir de fuentes confiables puesto que cualquier error en el tema de salud puede provocar daños en lugar de ayudar al paciente, es por ello que se solicitó la ayuda de un profesional en el tema y mediante una entrevista con el psicólogo Jorge Gordón docente de la Universidad Técnica del Norte, se determinó que la única función que se puede controlar es la respiración y una vez que esta es controlada las otras funciones de los órganos que en este caso es la frecuencia cardíaca y la conductancia de la piel de igual manera se estabilizan. Con ello se resuelve que para desarrollar el *feedback* es necesario reproducir un video de ejercicios de respiración con una duración de 2 minutos puesto que es un tiempo considerable para normalizar las funciones de los órganos.

Cuando el resultado del sistema es diferente de normal se procede a realizar el respectivo ejercicio de respiración, cabe mencionar que para las dos opciones que son algo elevado y alto el video es el mismo ya que se consideró normalizar la función de respiración. En la figura 42 se puede observar el video utilizado en el sistema.

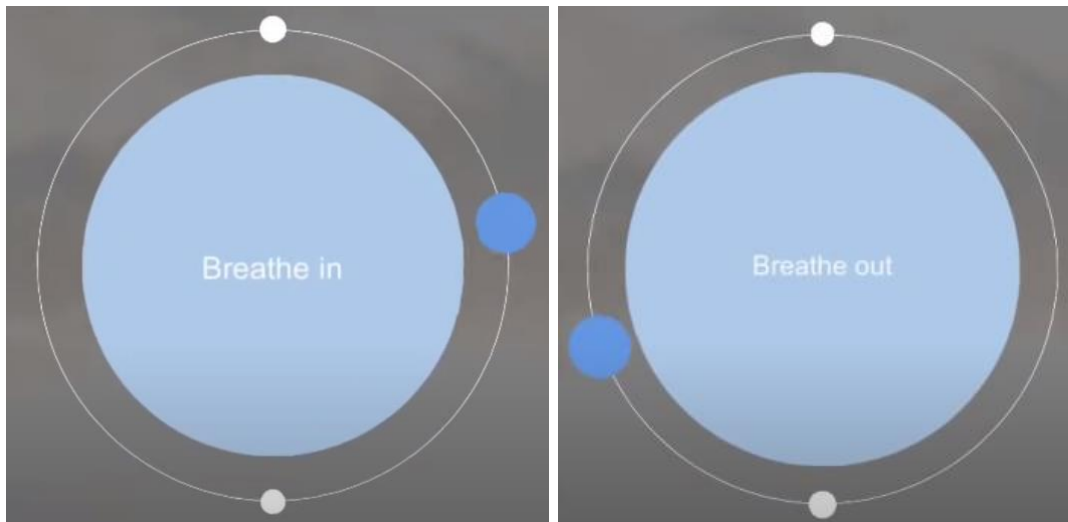


Figura 42. Ejercicio de respiración. Fuente: (Relaxing white noise, 2018)

3.8. Implementación

Para la implementación del diseño antes realizado del sistema BIOSTRESS se elabora en dos partes que son la implementación del *hardware* y *software*.

3.8.1. Implementación del hardware.

En esta etapa se procede a elaborar el armazón (*case*) para lo cual se selecciona el material que es en este caso acrílico ya que entre sus características se puede destacar que es un material liviano, posee resistencia al impacto, tiene estabilidad dimensional, cuenta con alto nivel de duración y además tiene un precio bajo. Para la fabricación del *case* se toma en cuenta el espacio necesario para los elementos electrónicos, posee perforaciones en el lado de arriba para la conexión de los sensores, en el costado derecho para la alimentación del sistema. Sus dimensiones son de 13cm de largo, 10cm de ancho y 3,5cm de alto. Este *case* se puede observar en la figura 43.



Figura 43. Case del sistema BIOSTRESS. Fuente: Autoría.

También se realiza la implementación de los elementos electrónicos que se utiliza en el sistema, como se observa en la figura 44, donde se encuentran conectadas las placas para el correcto funcionamiento de BIOSTRESS.



Figura 44. Implementación del Hardware. Fuente: Autoría.

3.8.2. Implementación del software.

Para la implementación del *software* en el sistema se debe crear el código de programación del bloque de toma de datos, procesamiento, implementación del algoritmo de aprendizaje de máquinas y de la interfaz de la aplicación.

3.8.2.1. Bloque de toma de datos

En este apartado se elabora el código de programación para la toma de datos de los tres sensores utilizados, parte de la programación se puede observar en la figura 45 y el código en su totalidad se encuentra en el Anexo 2.

```
#include <Arduino KNN.h> //Libreria para clasificación KNN
#include <MySignals.h> //Libreria Modulo MySignals
#include "Wire.h" //Libreria I2C para comunicación con mysignals
#include "SPI.h" //Libreria requerida por I2C

const int GSR = A0; //Puerto donde se encuentra conectado el sensor GSR
int sensorValue; //Variable para almacenar valor de conductividad del sensor GSR
int gsr_average; //Variable que almacena el promedio de la conductividad
int BPM; //Variable que almacena la frecuencia cardiaca en pulsos por minuto
int RPM ; //Variable que almacena la frecuencia respiratoria

//*****VARIABLES DE CALCULO DE FRECUENCIA RESPIRATORIA Y CARDIACA*****//

long ultimamedicion;
int ciclo = 0; //1=alto 0=bajo
int cambiodeciclo = 0;
int picodetension;
int valledetension;
long contadorciclo;

uint16_t air = (uint16_t)MySignals.getAirflow(DATA); //Lectura del sensor del flujo de aire
uint16_t ECG = (uint16_t) MySignals.getECG(DATA); //Lectura del sensor ECG
sensorValue = analogRead(GSR); //Lectura del Sensor GSR
```

Figura 45. Código de la toma de datos del sistema BIOSTRESS. Fuente: Autoría.

En la figura 46 se puede observar la toma de datos a estudiantes.



Figura 46. Obtención de datos. Fuente: Autoría.

Para la toma de los datos en estudiantes se procede a ubicar correctamente los tres sensores como se indicó en el apartado 3.7.2, se sigue las siguientes instrucciones:

- Sensor de Frecuencia Cardíaca ECG

1. Este sensor cuenta con electrodos los cuales deben ir colocados al cable ECG.
2. Los electrodos son de color amarillo, rojo y verde. El color amarillo que es neutro debe ir ubicado en el pecho en la parte izquierda donde se encuentra el corazón del paciente, el color rojo que es negativo debe ir ubicado en la parte derecha del pecho del paciente, y el color verde que es positivo debe ir ubicado en la parte alta del abdomen.
3. Se conecta el cable del sensor ECG a la placa *MySignals* en la parte que se encuentra dibujado un corazón y se procede a tomar los datos.

- Sensor de Frecuencia Respiratoria *Airflow*

1. Este sensor integra un cable de extensión con un conector de "ojo de cerradura", tiene una posición específica para ubicar la polaridad correcta, se encuentra unas marcas en el lateral de los dos conectores para ubicar adecuadamente.
2. El sensor *Airflow* debe ir colocado al paciente en la parte de la nariz para obtener el flujo de aire adecuado y sujetado en la parte de las orejas para prevenir algún desajuste del sensor.
3. Se conecta el cable del sensor *Airflow* a la placa *MySignals* en la parte que se encuentra dibujado unos pulmones y se espera un minuto para estabilizar la medida del sensor.

- Sensor de Conductancia de la piel GSR

1. Los electrodos del sensor deben ir conectados al cable GSR.
2. Los contactos de los electrodos deben ir ubicados en la falange de los dedos índice y medio de la mano izquierda del paciente.
3. Se ubica el sensor GSR a la placa y se procede a tomar los valores.



Figura 47. Correcta ubicación de los sensores al paciente. Fuente: Autoría.

3.8.2.2. Bloque de procesamiento

En esta etapa se puede ver el desarrollo del proceso del algoritmo de clasificación KNN con la información proporcionados por en bloque de toma de datos.

Puesto que el algoritmo KNN posee dos conjuntos de datos, se procede a realizar el conjunto de entrenamiento primordial para la posterior clasificación, esta base de datos inicial se elabora con información proporcionada por el psicólogo Jorge Gordón docente de la Universidad Técnica del Norte, además con la ayuda de datos bibliográficos de fuentes confiables, y también, se procede a realizar pruebas a estudiantes de la universidad donde fueron sometidos a varios factores estresantes como por ejemplo elaborar un examen, después se tomó los datos en un ambiente de relajación para obtener valores normales de las personas. Con esto se logra adquirir el conjunto de entrenamiento completo y variado para realizar las pruebas del sistema. En este conjunto se puede encontrar 4 campos con 30 variables que nos da como total 120 datos.

En la tabla 21 a continuación se encuentra un extracto del conjunto de entrenamiento que utiliza el algoritmo KNN.

Tabla 21. Extracto del conjunto de datos de entrenamiento.

BPM	RPM	GSR
64	18	420
50	16	440
58	14	375
65	20	435
60	19	400
68	22	315
72	26	305
71	25	320
73	25	307
69	23	339
86	38	297
80	29	270
80	31	290
85	35	295
90	32	288

Fuente: Autoría.

Con los datos de entrenamiento, se procede a añadir los campos de clase y estado, donde en clase se tiene el número 1,2 o 3 haciendo referencia a los niveles de estrés, y en estado se tiene los niveles: normal, algo elevado y alto que son basados en la tabla 18 presentada anteriormente. Estos estados se clasifican de acuerdo con los parámetros mencionados por el psicólogo y en base a pruebas realizadas con los estudiantes, dando como resultado un conjunto con valores base que determinan el nivel de estrés. En la tabla 22 se observa un extracto de estos datos clasificados correctamente.

Tabla 22. Clasificación de los niveles de estrés.

BPM	RPM	GSR	Clase	Estado
64	18	420	1	Normal
50	16	440	1	Normal
58	14	375	1	Normal
65	20	435	1	Normal
60	19	400	1	Normal
68	22	315	2	Algo Elevado
72	26	305	2	Algo Elevado
71	25	320	2	Algo Elevado
73	25	307	2	Algo Elevado
69	23	339	2	Algo Elevado
86	38	297	3	Alto
80	29	270	3	Alto
80	31	290	3	Alto
85	35	295	3	Alto
90	32	288	3	Alto

Fuente: Autoría.

La nueva tabla de datos se encuentra completa puesto que cuenta con los datos obtenidos de los sensores y con la etiqueta de cada nivel de estrés, la totalidad de la tabla de datos de entrenamiento se encuentra en el Anexo 3.

Para tener una mayor eficacia del sistema se procede a evaluar los casos críticos que se presentan al momento del desarrollo, estos casos se especificaron en la parte de diseño en el apartado 3.7.3.2.2. En la tabla 23 se encuentra los datos obtenidos de cada caso crítico con su estado.

- **Caso 1:** Cuando un parámetro medido es normal y los otros dos algo elevados.
- **Caso 2:** Cuando un parámetro medido es normal y los otros dos altos.
- **Caso 3:** Cuando se tiene un valor normal, un valor algo elevado y otro valor alto.
- **Caso 4:** Cuando se tiene un parámetro algo elevado y los otros dos normales.
- **Caso 5:** Cuando se tiene un parámetro algo elevado y los otros dos altos.
- **Caso 6:** Cuando se tiene un parámetro alto y los otros dos normales.
- **Caso 7:** Cuando se tiene un parámetro alto y los otros dos algo elevados.

Tabla 23. Valores para cada caso crítico.

Caso	Rango			Nivel de estrés
	BPM	RPM	GSR	
1	63	24	342	Algo Elevado
2	58	33	298	Alto
3	61	25	296	Alto
4	72	39	299	Alto
5	77	17	434	Normal
6	89	16	429	Normal
7	92	26	348	Algo Elevado

Fuente: Autoría.

Con la implementación del algoritmo para cada caso crítico se obtiene los resultados previstos anteriormente. La simulación del algoritmo se realiza en base a 100 pruebas y arroja una matriz de confusión la cual se muestra en la tabla 24.

Tabla 24. Matriz de confusión.

	Real Normal	Real Algo Elevado	Real Alto
Predicho Normal	95	2	3

Predicho Algo Elevado	1	97	2
Predicho Alto	1	1	98

Fuente: Autoría.

Para calcular la exactitud del algoritmo utilizado se emplea la ecuación 6 presentada en el apartado 3.7.3.2.3, lo cual da el siguiente resultado.

$$Exactitud = \frac{95+97+98}{95+97+98+2+3+1+2+1+1} = \frac{290}{300} = 0.966 \approx 96,6\%$$

Se obtiene una exactitud del 96,6% al implementar el algoritmo seleccionado, lo que nos da la confiabilidad de que el algoritmo trabaja de la manera correcta al momento de seleccionar los niveles de estrés en los test que se realicen.

3.8.2.3. Bloque de visualización

Este bloque se realizó en el *software Processing* con los datos obtenidos del bloque de procesamiento el cual es enviado por comunicación serial para que pueda ser observado por el administrador y paciente al momento de comenzar el test. El diseño inicial de la interfaz de la aplicación de BIOSTRESS se observa en la figura 48.



Figura 48. Interfaz inicial de BIOSTRESS. Fuente: Autoría.

Después de la interfaz inicial del sistema se observa otra interfaz en la cual se realiza las mediciones correspondientes presionando los botones adecuados y siguiendo las instrucciones, esta interfaz se encuentra en la figura 49.

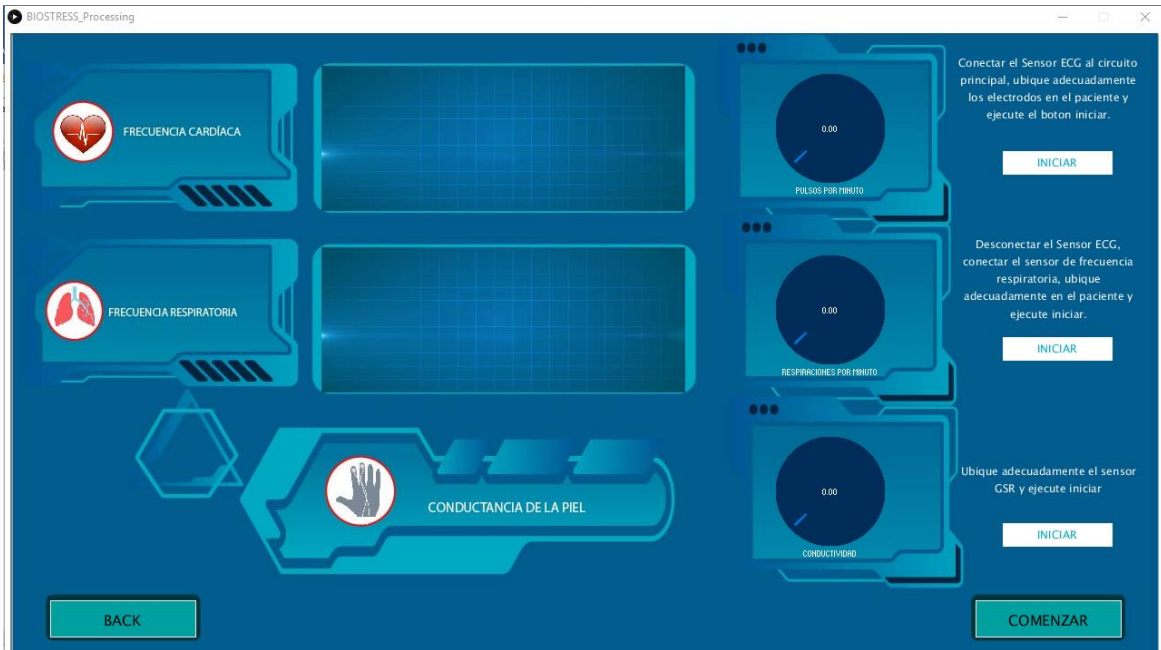


Figura 49. Interfaz de las mediciones de BIOSTRESS. Fuente: Autoría.

La siguiente interfaz se muestra al momento de presionar comenzar con el resultado y se observa el nivel de estrés de los estudiantes, esto se encuentra en la figura 50.



Figura 50. Interfaz del estado del paciente. Fuente: Autoría.

Cuando el estado del paciente presenta novedad, es decir es diferente de normal se procede a realizar el *Biofeedback* para lo cual se presenta un video que indica un ejercicio de respiración y al final se procede a iniciar las mediciones y salir, esto se observa en la figura 51.

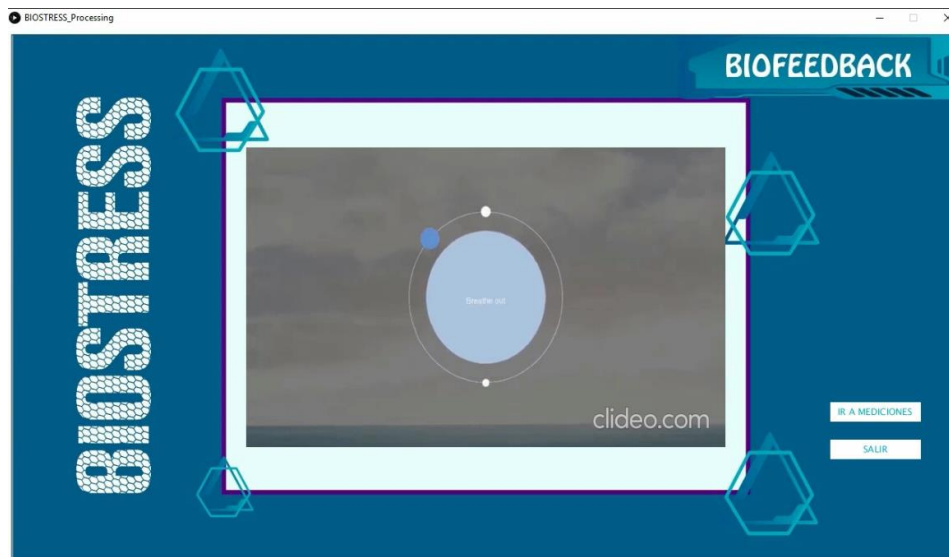


Figura 51. Interfaz del Biofeedback. Fuente: Autoría.

4. CAPÍTULO IV

PRUEBAS DE FUNCIONAMIENTO Y RESULTADOS

En este capítulo se da a conocer las pruebas de funcionamiento del sistema de detección de niveles de estrés en los estudiantes. Con esta etapa se da por concluida la metodología planteada para este proyecto que es el Modelo en V en la fase de validación.

4.1. Test unitarios

Los test unitarios se realizan con la finalidad de comprobar el correcto funcionamiento del sistema BIOSTRESS, para ello se comienza con la verificación del estado del sensor ECG donde se procede a calcular la frecuencia cardíaca con un *Smarthwash* y el sensor ECG ocupado en el sistema, en la figura 52 se puede ver como con los dos elementos utilizados se obtiene el mismo resultado.



Figura 52. Prueba del sensor ECG. Fuente: Autoría.

La calibración de los sensores se desarrolló conectándolos a la placa *MySignals*, la figura 53 muestra la señal de los sensores al momento de conectarlos sin ser calibrados y después con la implementación del algoritmo que se encuentra en el Anexo 2 se observa como los valores tomados son calibrados para su posterior análisis.

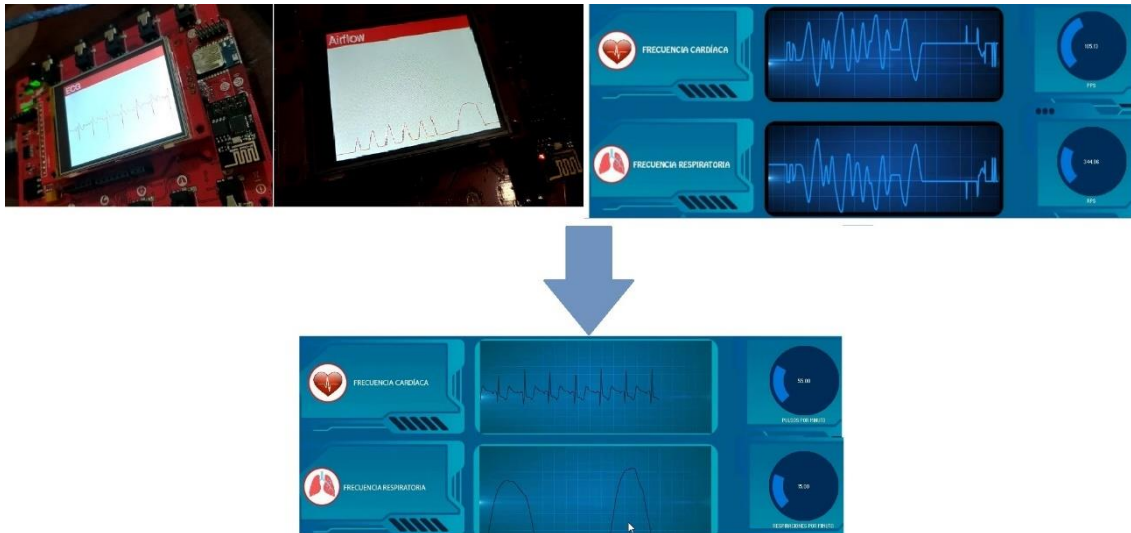


Figura 53. Calibración de los sensores. Fuente: Autoría.

Para concluir con los test unitarios se realiza la prueba de funcionamiento de la interfaz donde se encuentran los resultados obtenidos de los pacientes, es decir muestra el nivel de estrés que posee la persona, esto se comprueba en la figura 54.



Figura 54. Prueba de Funcionamiento de la interfaz. Fuente: Autoría.

4.2. Integración del sistema BIOSTRESS

En este apartado se muestra la incorporación de todos los componentes del sistema tanto exteriormente como interiormente y esto se observa en las figuras 55 y 56.



Figura 55. Vista externa del sistema BIOSTRESS. Fuente: Autoría.

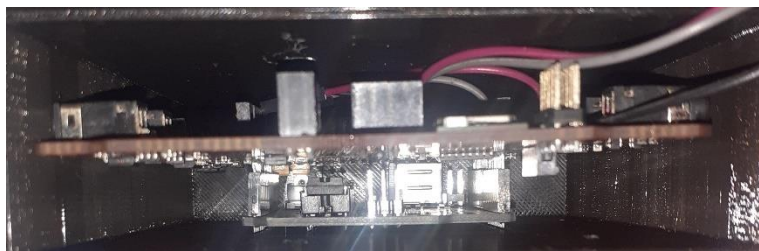


Figura 56. Vista interna del sistema BIOSTRESS. Fuente: Autoría.

4.3. Pruebas

En esta etapa se realizan dos tipos de pruebas que son las pruebas de *hardware* y las pruebas a los usuarios para observar la funcionalidad del sistema.

4.3.1. Pruebas de hardware.

Se realiza una serie de parámetros los cuales se encuentran detallados en la tabla 25, estos comprueban el trabajo que realiza el sistema BIOSTRESS. Los resultados que se observan del *hardware* verifican la correcta conexión de los sensores a la placa *MySignals* puesto que se conectan mediante cables *plug* y en el *case* se encuentran dibujados las señales para su ubicación, cabe mencionar que, si se conecta mal los sensores, los datos no serán tomados.

Tabla 25. Parámetros de las pruebas del Hardware.

Pruebas de Hardware	
Parámetro	Cumplimiento
Funcionan las placas del sistema	Si
Funciona el sensor ECG	Si
Funciona el sensor <i>Airflow</i>	Si, debe calentarse 30 segundos antes del test
Funciona el sensor GSR	Si
El sistema recibe los datos correctamente	Si

Fuente: Autoría.

4.3.2. Pruebas con los usuarios.

En esta prueba se demuestra el funcionamiento en su totalidad del sistema BIOSTRESS, se verifica que el *hardware* y el *software* funcionen correctamente para mostrar el nivel de estrés correspondiente.

En esta etapa se considera datos que son completamente nuevos para el sistema, esta información es obtenida de 10 estudiantes universitarios los cuales se detalla en la tabla 26, con los sujetos presentados se tiene una población variada para realizar las

pruebas necesarias puesto que se encuentran en distintas carreras y cuentan con diferentes edades entre hombres y mujeres.

Tabla 26. Sujetos de prueba.

N°	Edad	Sexo	Carrera
Sujeto 1	19	F	Marketing
Sujeto 2	20	M	Marketing
Sujeto 3	21	M	Ingeniería Forestal
Sujeto 4	22	F	Ingeniería Forestal
Sujeto 5	23	M	Nutrición
Sujeto 6	24	M	Psicología
Sujeto 7	25	F	CIERCOM
Sujeto 8	25	M	CIERCOM
Sujeto 9	25	F	CIERCOM
Sujeto 10	26	F	Turismo

Fuente: Autoría.

Cuando se determina el conjunto de estudiantes con los cuales se va a trabajar se procede a realizar la prueba, para demostrar los tres niveles de estrés se escogió a un sujeto en este caso el número 10 y se realizó el test para lo cual se procede a tomar los datos dando como resultado normal y sometiendo al sujeto a estresores para demostrar los niveles algo elevado y alto, esto se observa a continuación.

4.3.2.1. Análisis del nivel del estrés normal.

Para iniciar con este test se debe colocar al paciente los sensores tal como se indicó en el apartado 3.8.1, consecuentemente el administrador del sistema procede a ubicar el plug del sensor ECG en el sistema BIOSTRESS para comenzar con su toma de datos, el

mismo procedimiento se realiza con los demás sensores para la obtención de información del paciente, esto se observa en la figura 57 y 58.

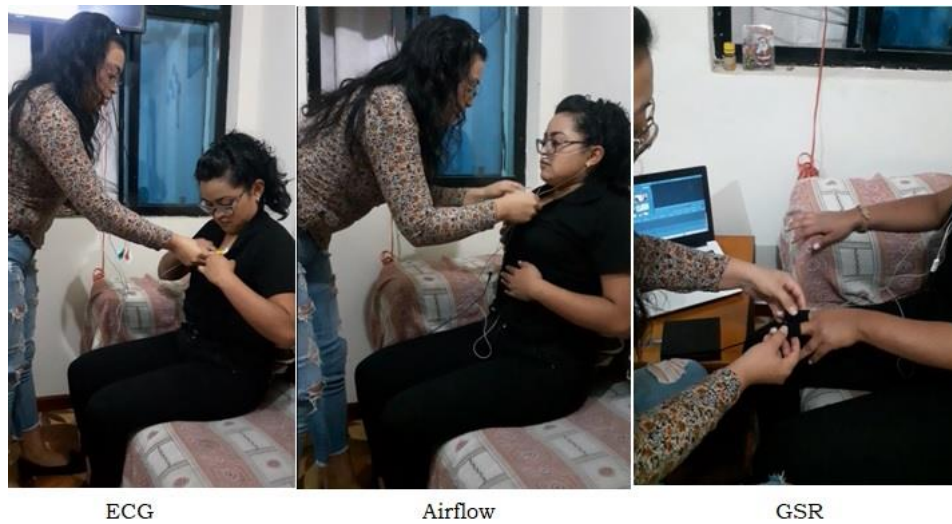


Figura 57. Colocación de los sensores al paciente. Fuente: Autoría.



Figura 58. Ubicación de los sensores al sistema. Fuente: Autoría.

Una vez ubicado los sensores, el administrador ejecuta la interfaz de BIOSTRESS donde se presiona el botón comenzar y se sigue los pasos mostrados en el Anexo 4, seguidamente se procede a tomar los valores de los sensores, cada sensor toma los datos en un tiempo de 1 minuto, esto se muestra en la figura 59.

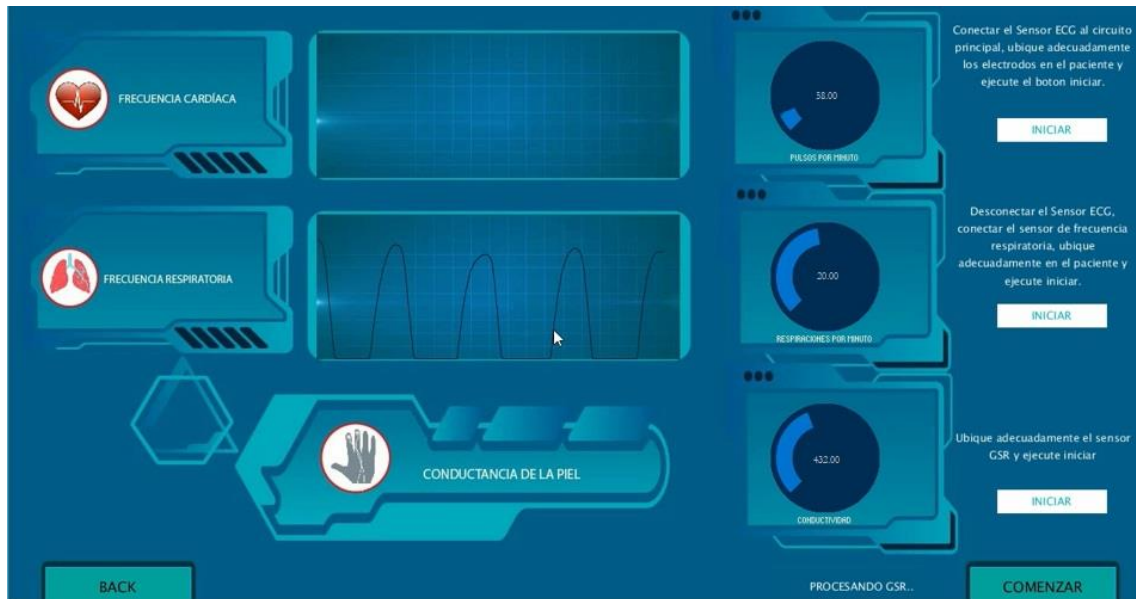


Figura 59. Interfaz de la toma de datos del estado normal. Fuente: Autoría.

Cuando se obtengan los tres valores de los sensores se puede observar el resultado que sería el nivel de estrés del paciente. En la figura 60 se encuentra el estado de la persona, en este caso como los valores están en el rango mencionado anteriormente el nivel de estrés del paciente es normal.

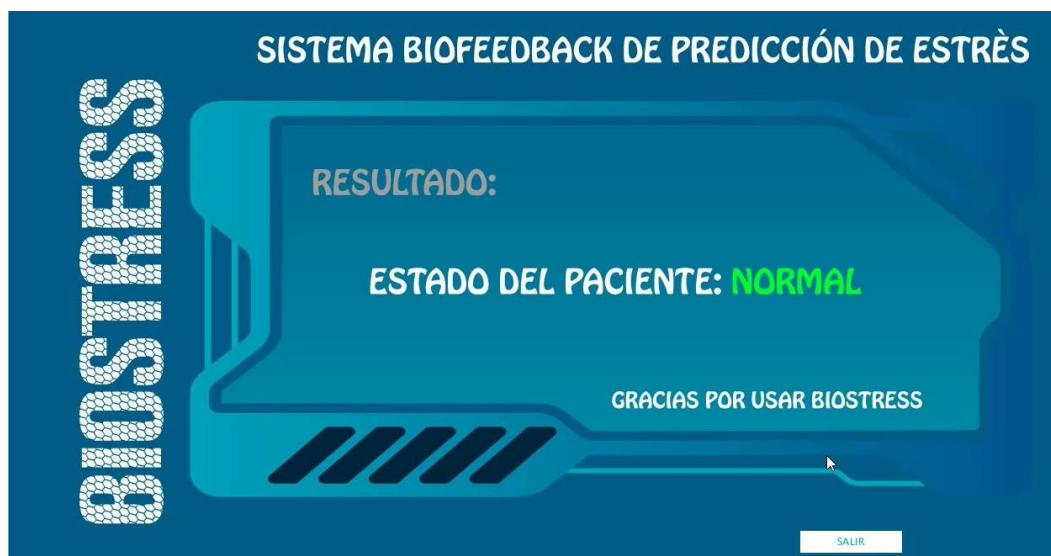


Figura 60. Nivel de estrés normal. Fuente: Autoría.

En la figura 61 se muestra la BDD del archivo csv donde se evidencia los valores obtenidos por el paciente y se le asigna el nivel de acuerdo con el procesamiento del algoritmo KNN.

	A	B	C	D	E
5	3	16	61	400	NORMAL
6	4	17	69	387	NORMAL
7	5	18	59	390	NORMAL
8	6	23	71	316	ALGO ELEVADO
9	7	26	69	343	ALGO ELEVADO
10	8	20	62	380	NORMAL
11	9	14	59	430	NORMAL
12	10	16	80	435	NORMAL
13	11	29	82	298	ALTO
14	12	28	85	293	ALTO
15	13	30	87	291	ALTO
16	14	29	83	299	ALTO
17	15	34	92	289	ALGO ELEVADO
18	16	20	58	432	NORMAL
19	17	27	79	326	ALGO ELEVADO

Figura 61. BDD del archivo csv del estado Normal. Fuente: Autoría.

4.3.2.2. Análisis del nivel del estrés algo elevado.

Para demostrar el nivel de estrés algo elevado se procede a someter al paciente a una serie de estresores, en este caso se le entrega una secuencia de ejercicios y preguntas para que sean respondidos simulando un examen y así poder medir los parámetros para detectar estrés en la persona, esto se observa en la figura 62.



Figura 62. Someter al paciente a estresores. Fuente: Autoría.

Los sensores deben estar ubicados de la misma manera que se mencionó anteriormente, el administrador ejecuta el programa y procede a tomar los datos en la interfaz de la aplicación. La obtención de información se evidencia en la figura 63 y en la figura 64 se observa el archivo csv donde se guarda los datos obtenidos del paciente y el nivel de estrés que posee.

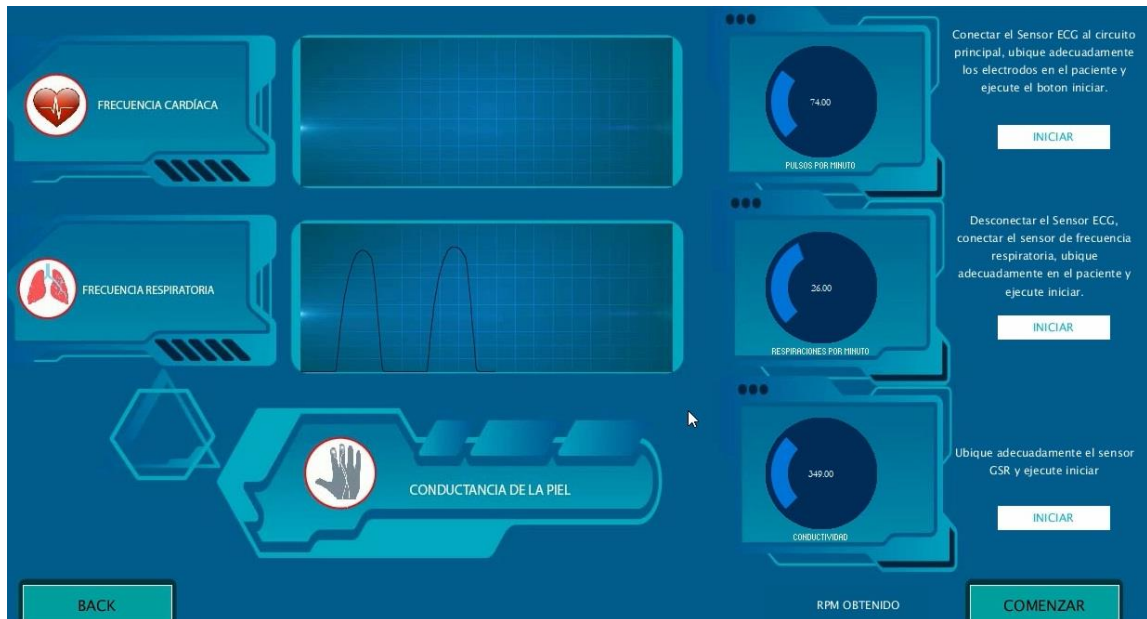


Figura 63. Interfaz de la toma de datos del estado algo elevado. Fuente: Autoría.

Algunas características del libro se pueden perder si lo guarda como CSV (delimitado por comas). Para conservar estas características, guárdelo como archivo de Excel. No most

POSSIBLE PÉRDIDA DE DATOS

	A	B	C	D	E
6	4	17	69	387	NORMAL
7	5	18	59	390	NORMAL
8	6	23	71	316	ALGO ELEVADO
9	7	26	69	343	ALGO ELEVADO
10	8	20	62	380	NORMAL
11	9	14	59	430	NORMAL
12	10	16	80	435	NORMAL
13	11	29	82	298	ALTO
14	12	28	85	293	ALTO
15	13	30	87	291	ALTO
16	14	29	83	299	ALTO
17	15	34	92	289	ALGO ELEVADO
18	16	20	58	432	NORMAL
19	17	26	74	349	ALGO ELEVADO
20	18	24	60	302	ALTO

Figura 64. BDD del archivo csv del estado algo elevado. Fuente: Autoría.

Después de obtener los tres valores de los sensores se comienza con el resultado que en este caso sería nivel de estrés algo elevado puesto que los datos obtenidos han subido gracias al estresor que se utilizó en el paciente. En la figura 65 se muestra la interfaz de este estado.

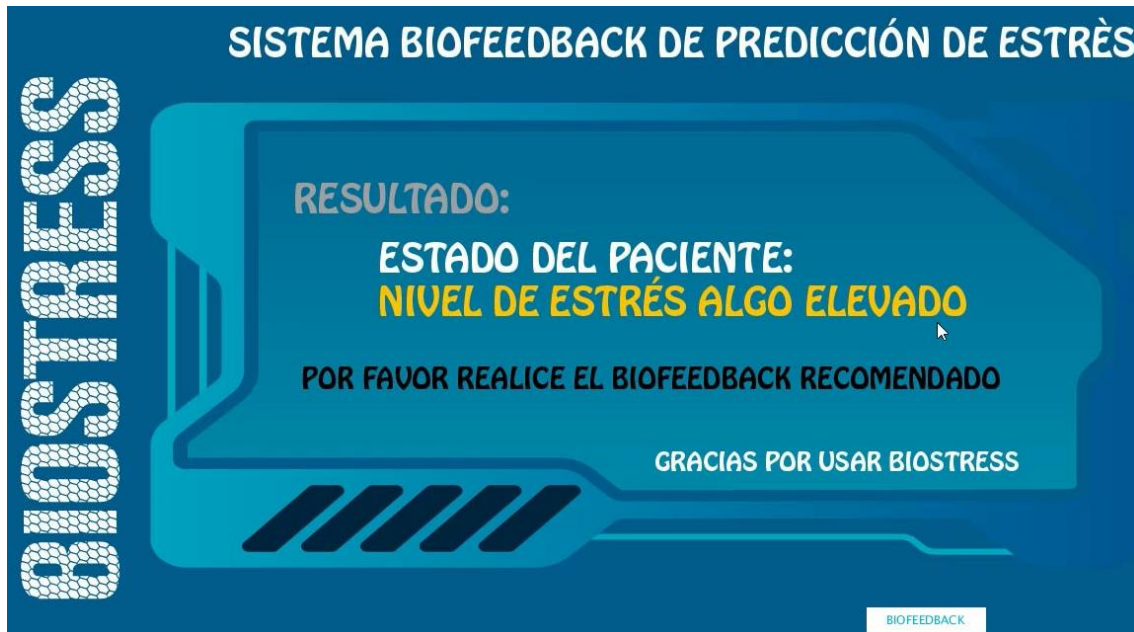


Figura 65. Nivel de estrés algo elevado. Fuente: Autoría.

Cuando el resultado es diferente de normal se debe seguir el *Biofeedback*, el cual consiste en realizar un ejercicio de respiración para controlarla y consecuentemente volver a tomar los datos del paciente y así poder comprobar que este posible tratamiento aplicado resultó favorable. En la figura 66 se muestra la interfaz donde se encuentra el *Biofeedback* y en la figura 67 se observa al paciente realizando dicho ejercicio de respiración.

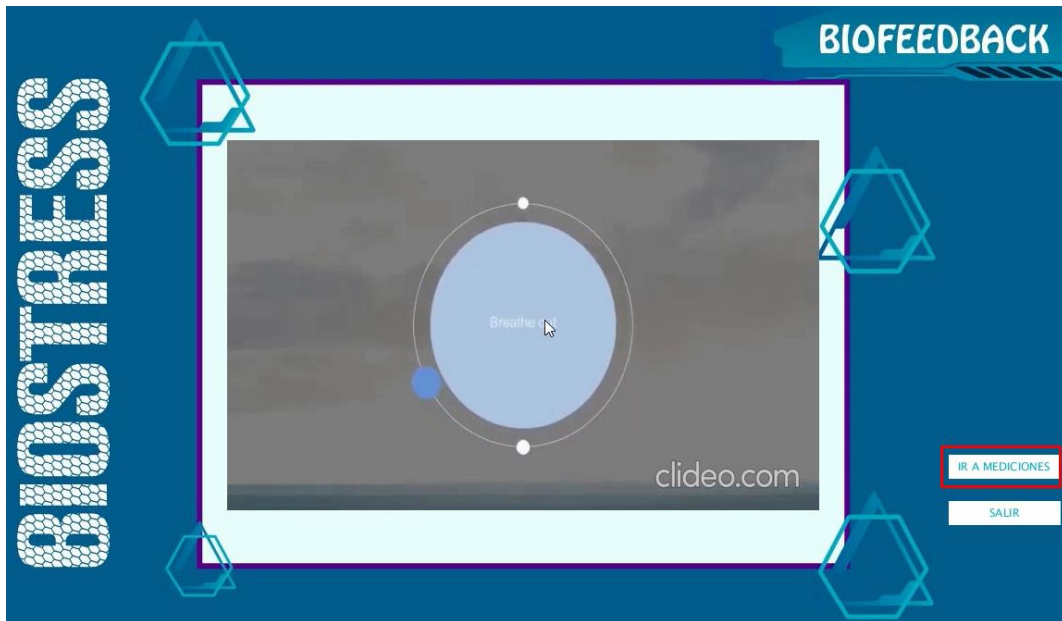


Figura 66. Interfaz del Biofeedback. Fuente: Autoría.

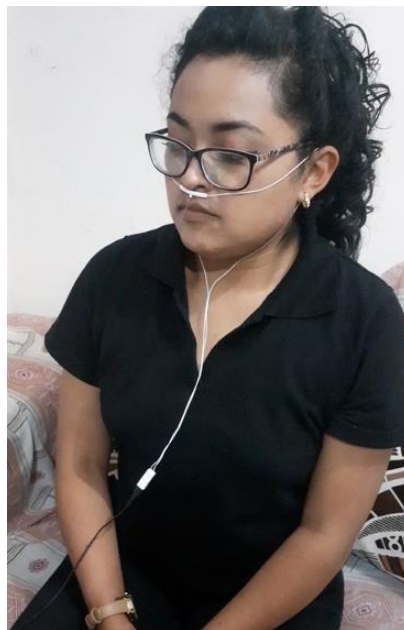


Figura 67. Ejecución del Biofeedback por parte del paciente. Fuente: Autoría.

4.3.2.3. Análisis del nivel del estrés alto.

En este nivel de estrés se somete al paciente a estresores, que en este caso es la simulación de un examen para determinar el nivel de estrés académico del estudiante, como lo evidencia la figura 62 presentada anteriormente. Una vez que el paciente se encuentre resolviendo el examen se procede a tomar los valores, esto se evidencia en la

figura 68 y en la figura 69 se encuentra los datos obtenidos con el resultado en el archivo CSV.

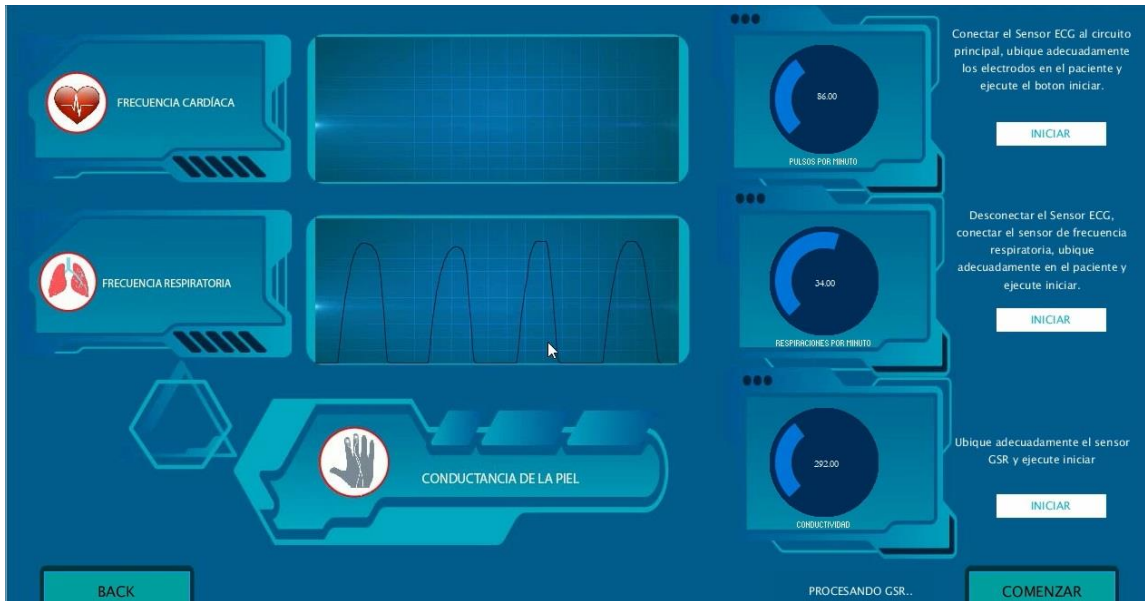


Figura 68. Interfaz de la toma de datos del estado alto. Fuente: Autoría.

Algunas características del libro se pueden perder si lo guarda como CSV (delimitado por comas). Para conservar estas características, guárdelo como archivo de Excel.

POSIBLE PÉRDIDA DE DATOS No muestra

H15

	A	B	C	D	E	F
7	5	18	59	390	NORMAL	
8	6	23	71	316	ALGO ELEVADO	
9	7	26	69	343	ALGO ELEVADO	
10	8	20	62	380	NORMAL	
11	9	14	59	430	NORMAL	
12	10	16	80	435	NORMAL	
13	11	29	82	298	ALTO	
14	12	28	85	293	ALTO	
15	13	30	87	291	ALTO	
16	14	29	83	299	ALTO	
17	15	34	92	289	ALGO ELEVADO	
18	16	20	58	432	NORMAL	
19	17	26	74	349	ALGO ELEVADO	
20	18	34	86	292	ALTO	
21	19	33	83	278	ALTO	

Figura 69. BDD del archivo csv del estado alto. Fuente: Autoría.

Con los datos y el resultado obtenido se procede a visualizar en la interfaz de la aplicación donde para este caso se evidencia que el nivel de estrés es alto y para lo cual

el paciente debe seguir el *Biofeedback* correspondiente como se mencionó anteriormente.

Esto se evidencia en las figuras 70 y 71.



Figura 70. Nivel de estrés alto. Fuente: Autoría.

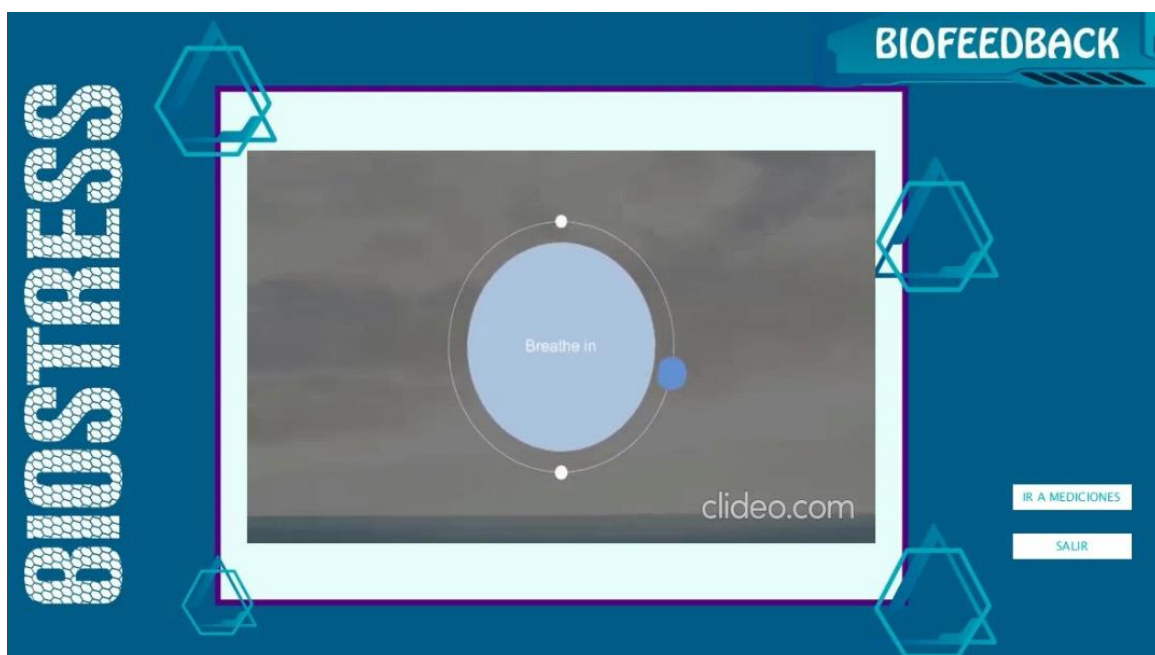


Figura 71. Interfaz del *Biofeedback* de nivel de estrés alto. Fuente: Autoría.

4.3.3. *Discusión de resultados.*

Las pruebas desarrolladas a los sujetos de la tabla 26 dan como resultado los valores que se encuentran en la tabla 27, donde se evidencia que los estudiantes de acuerdo con sus responsabilidades poseen diferentes niveles de estrés, esto depende de los estresores que posean en su vida académica y más aún por las circunstancias que pasan en la pandemia mundial que se vive actualmente.

Tabla 27. Nivel de estrés de los sujetos de prueba.

N°	BPM	RPM	GSR	Nivel de estrés
Sujeto 1	62	18	412	Normal
Sujeto 2	67	22	344	Algo Elevado
Sujeto 3	69	25	331	Algo Elevado
Sujeto 4	81	29	297	Alto
Sujeto 5	73	26	348	Algo Elevado
Sujeto 6	55	16	429	Normal
Sujeto 7	74	23	336	Algo Elevado
Sujeto 8	86	30	294	Alto
Sujeto 9	77	27	343	Algo Elevado
Sujeto 10	58	20	432	Normal

Fuente: Autoría.

En la tabla presentada anteriormente se muestra que algunas personas en este caso el 30% de los sujetos de prueba resultaron con un nivel de estrés normal lo cual no es de alarmarse ya que sus respuestas fisiológicas se encuentran moderadas, el 70% restante presentó un incremento en sus respuestas fisiológicas, esto puede deberse a su ajeteo en la vida diaria provocada por sus diferentes estresores, este 70% de sujetos de prueba

realizaron el respectivo *Biofeedback* lo cual ayuda a que sus respuestas fisiológicas se normalicen lo que conlleva a su relajación y disminución de nivel de estrés.

4.4. Análisis del nivel de estrés en estudiantes

En las pruebas cuyos datos obtenidos dieron como resultado diferente de normal se procedió a realizar el *feedback* correspondiente y se siguió con la medición de los datos de los pacientes para constatar si sus respuestas fisiológicas se normalizaron, esto se puede observar en las figuras a continuación.

En la figura 72 se encuentran la información de la frecuencia respiratoria, donde se evidencia que la línea de color rojo son los datos medidos inicialmente del nivel de estrés algo elevado y alto de los pacientes, y, la línea de color verde son los nuevos valores después de realizar el *Biofeedback*.

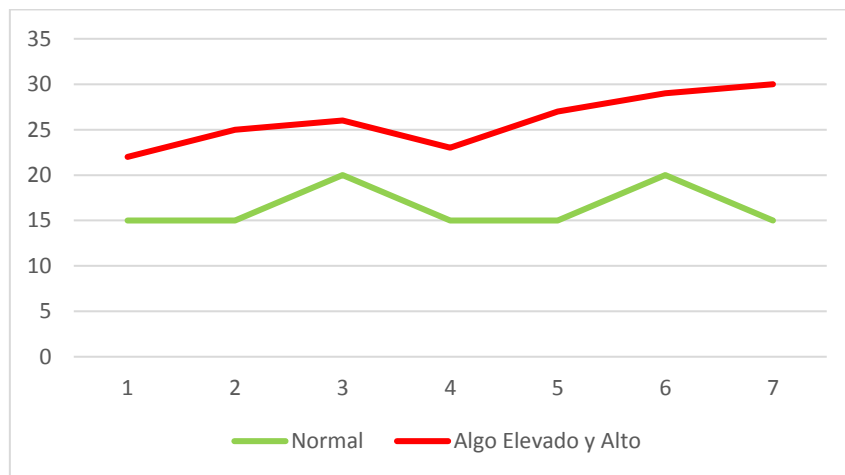


Figura 72. Datos de la Frecuencia Respiratoria. Fuente: Autoría.

De igual manera se observa en la figura 73 los datos de la frecuencia cardíaca, donde la línea de color verde son los datos obtenidos después de realizar el *Biofeedback*, lo cual demuestra que los valores tomados por el sensor ECG se han normalizado.

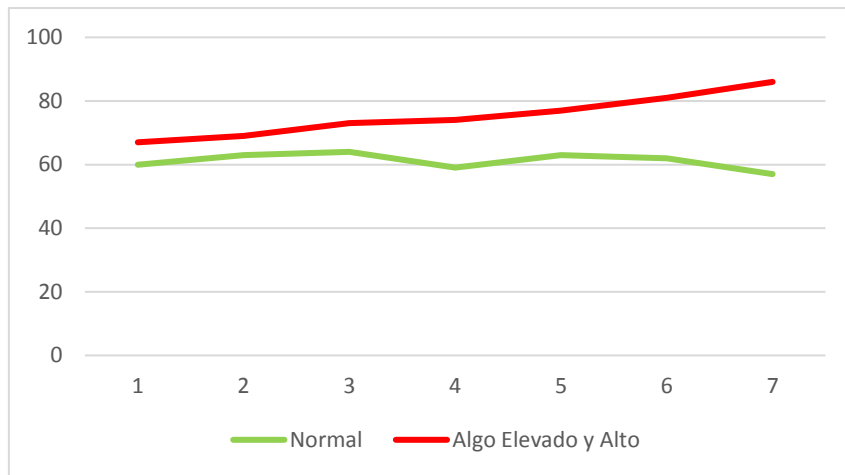


Figura 73. Datos de la Frecuencia Cardíaca. Fuente: Autoría.

Por último, en la figura 74 se muestran los datos de la conductancia de la piel, que igualmente como se mencionó la línea de color verde demuestra los datos normalizados luego de realizar el ejercicio de respiración del *Biofeedback*.

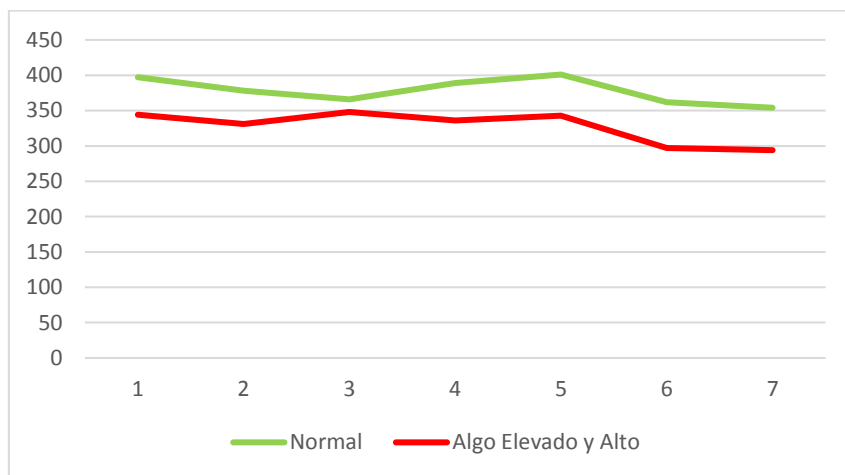


Figura 74. Datos de la Conductancia de la piel. Fuente: Autoría.

Después de realizar las pruebas necesarias con los pacientes en este caso los estudiantes, se puede resumir que debido a diferentes circunstancias que pasan en el período académico como la parte de exámenes va aumentando el estrés y si no es controlado a tiempo puede volverse crónico y causar enfermedades graves para su salud.

5. CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Se elaboró un sistema *Biofeedback* de predicción de estrés en estudiantes universitarios denominado BIOSTRESS que se encuentra conformado por dos sistemas embebidos que son Arduino UNO y *MySignals* los cuales mediante filtros digitales procesan las bioseñales obtenidas por sensores biomédicos y a través de un algoritmo de aprendizaje de máquinas determinan el nivel de estrés de los pacientes.

Se realizó una interfaz en *Processing* que permite tanto al administrador como a los usuarios obtener los niveles de estrés en estudiantes.

La aplicación realizada complementa el funcionamiento del *Biofeedback* en la cual tanto profesionales de psicología como los pacientes pueden conocer los datos obtenidos por el sistema y realizar la retroalimentación.

La base de datos de entrenamiento cuenta con valores para ser comparados con la nueva información y se almacenan en una base de datos con las casillas adecuadas que son utilizadas por el profesional al momento de evaluar y diagnosticar el estrés de los estudiantes.

Se demostró la validez del sistema puesto que funcionó de manera adecuada bajo los parámetros de las pruebas que se realizaron y permite encontrar los niveles de estrés en el rango entre normal a alto en circunstancias académicas de los estudiantes.

5.2. Recomendaciones

Para todo trabajo de investigación se debe usar bibliografías que sean de fuentes confiables como artículos científicos para tener bases teóricas verídicas.

Se recomienda hablar con profesionales en el tema de los cambios fisiológicos provocados por el estrés para así poder comprender que parámetros se deben medir y como deben ser tratados puesto que una mala práctica del *Biofeedback* puede causar daños en la salud de los pacientes.

Es recomendable disminuir el ruido de las señales obtenidas mediante filtros digitales puesto que existen cambios muy repentinos al momento de obtener los datos de los sensores biomédicos.

Para el funcionamiento del sistema BIOSTRESS es recomendable que cuente con una alimentación continua sin interrupciones ya que las señales que se toman y se visualizan son en tiempo real, por ese motivo se utiliza la alimentación por el cable USB de Arduino y la comunicación es serial para que no se pierda ningún dato al momento de la toma de información.

Se debe tomar en cuenta la versión de la placa *MySignals* para descargar las librerías que se utilizan de tal forma que no se altere el rendimiento de las dos placas utilizadas en el proyecto.

Profundizar la investigación en algoritmos supervisados de *Machine Learning* para mejorar la forma automática del algoritmo kNN y así aumentar la eficiencia del proyecto.

Es recomendable colocar al paciente el sensor *Airflow* y dejar un tiempo previo de 30 segundos hasta que se caliente para que pueda realizar la medición correcta.

Para no generar ruido en la toma de datos es recomendable ubicar los electrodos de los sensores al paciente siguiendo con las indicaciones recomendadas y también

verificar que el ordenador utilizado en el test no esté conectado a la corriente eléctrica puesto que esto genera ruido al momento de medir la frecuencia cardíaca y altera valores.

Existen diferentes cambios fisiológicos que pueden ser medidos con la placa *MySignals*, en este proyecto se mide tres de ellas, sin embargo, se recomienda para futuros trabajos tomar las señales al mismo tiempo y aumentar los parámetros a medir para poder detectar más afecciones mentales.

6. BIBLIOGRAFÍA

- Acevedo, M., Acevedo, M., L. A. (2001). Sistema de medición de estrés. *Revista Mexicana de Ingeniería Biomedica*, XXII, 20–25.
- Adhikari, P., Bender, F., Ricco, A. J., & Josse, F. (2017). Design criteria for plasticized polymer coatings for long-term acoustic-wave sensor measurements in aqueous phase. *2017 Joint Conference of the European Frequency and Time Forum and IEEE International Frequency Control Symposium, EFTF/IFC 2017 - Proceedings*, 24–25. <https://doi.org/10.1109/FCS.2017.8088790>
- Aguilar Barojas, S. (2005). Fórmulas para el cálculo de la muestra. *Salud En Tabasco*, 11, 333–338. <http://www.redalyc.org/articulo.oa?id=48711206>
- Alen, N., & Hostinar, C. (2018, November). *Stress Effects: Endocrine*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-endocrine#menu>
- Aloisio, A., Giordano, R., Izzo, V., & Perrella, S. (2015). A frequency agile, self-adaptive serial link on xilinx FPGAs. *IEEE Transactions on Nuclear Science*, 62(3), 955–962. <https://doi.org/10.1109/TNS.2015.2423674>
- Ángel, J. E. (2010). Manejo de stakeholders como estrategia para la administración de proyectos. *Agronomía Colombiana*, 28(3), 491–499.
- Arar, S. (2017, October 19). *Linear Filtering Based on the Discrete Fourier Transform*. Keysight Technologies. <https://www.allaboutcircuits.com/technical-articles/linear-filtering-based-on-the-discrete-fourier-transform/>
- Arduino Community. (2018, February 5). *Arduino UNO*. <https://www.arduino.cc/en/Guide/ArduinoUno>
- Batovski, D. (2014). *How to Prepare Block Diagrams*. January 2008, 2–4.
- Berástegui Arbeloa, G., & Galar Idoate, M. (2018). *Implementación del algoritmo de los k vecinos más cercanos (k-NN) y estimación del mejor valor local de k para su cálculo*. 50. <https://academica-e.unavarra.es/xmlui/bitstream/handle/2454/29112/Memoria.pdf?sequence=2&isAllowed=y>
- Berntson, G. G. (2018, November). *Stress Effects: Nervous System*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-nervous#menu>
- Bogdan, M., Oberla, E., Frisch, H. J., & Wetstein, M. (2016). A modular data

- acquisition system using the 10 GSa/s PSEC4 waveform recording chip. 2016 *IEEE-NPSS Real Time Conference, RT 2016*, 4, 6–8.
<https://doi.org/10.1109/RTC.2016.7543167>
- Bossche, A. (1996). Intelligent sensor systems. *Sensors and Actuators A: Physical*, 55(2–3), 226. [https://doi.org/10.1016/s0924-4247\(97\)80083-0](https://doi.org/10.1016/s0924-4247(97)80083-0)
- Burbano, Y. (2019, May 14). *Sistema Embebido Móvil De Reconocimiento De Patrones De Calidad Del Aire En La Ciudad De Ibarra*. 33–38.
[http://repositorio.utn.edu.ec/bitstream/123456789/9155/1/04 RED 224 TRABAJO DE GRADO.pdf](http://repositorio.utn.edu.ec/bitstream/123456789/9155/1/04%20RED%20224%20TRABAJO%20DE%20GRADO.pdf)
- Burg, M. M. (2018, November). *Stress Effects: Cardiovascular*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-cardiovascular#menu>
- Burn, I. (2017, November 17). *Benchmark*. Test de Rendimiento.
<https://es.ccm.net/faq/12976-evaluar-el-rendimiento-benchmark-de-tu-pc>
- Campo, A., Oviedo, H. C., & Herazo, E. (2015). Escala de Estrés: Desempeño psicométrico en estudiantes. *Revista de La Facultad de Medicina*, 62(3), 407–413.
<https://doi.org/10.15446/revfacmed.v62n3.43735>
- Carrobbles, J. A. (2016). Bio/neurofeedback. *Clinica y Salud*, 27(3), 125–131.
<https://doi.org/10.1016/j.clysa.2016.09.003>
- Carter, B., & Huelsman, L. P. (2001). Handbook Of Operational Amplifier Active RC Networks. *Networks, October*, 1–85.
- Chaniotakis, C. (2006). *Operational Amplifier. 1*, 1–34.
- Chaparro, J., Giraldo, B. G., & Rondón, S. (2015). Evaluación del clasificador Naïve Bayes como herramienta de diagnóstico en Unidades de Cuidado Intensivo. *Revista de Tecnología*, 12(2). <https://doi.org/10.18270/rt.v12i2.774>
- Charris, L., Henriquez, C., Hernandez, S., Jimeno, L., Guillen, O., & Moreno, S. (2018). Análisis comparativo de algoritmos de árboles de decisión en el procesamiento de datos biológicos. *Investigacion y Desarrollo En TIC*, 1, 10.
- Chart, F. (2015). *Flow Diagram*. 1–2.
- Coen, S., & Haelterman, M. (2000). Generation of ultra-high repetition rate pulse trains in a quadruply resonant optical fiber parametric oscillator. 39, 2000.
<https://ieeexplore.ieee.org/document/907459>

- Conde, M., & Menéndez, J. (2016). BIOFEEDBACK TECHNIQUES AND THEIR APPLICATIONS: A REVIEW. *Departamento de Psicología Básica II. Universidad Nacional de Educación a Distancia, February.*
- De Luca, P., Sánchez, a, Pérez, G., & Leija, L. (2004). Medición integral del estrés crónico. *Revista Mexicana de Ingeniería Biomédica*, 25(1), 60–66.
<http://www.medigraphic.com/pdfs/inge/ib-2004/ib041i.pdf>
- Essel, G., & Owusu, P. (2017). *Causes of students' stress, its effects on their academic success*. 2, 15–20.
- Fernández Martínez, R., Ordieres Meré, J., Martínez, F., González Marcos, A., Alba, F., Lostado Lorza, R., & Pernía, V. (2009). *Redes inalámbricas de sensores: teoría y aplicación práctica* (n° 26).
https://www.researchgate.net/publication/260265697_Redex_Inalambricas_de_sensores_teor%C3%ADa_y_aplicaci%C3%B3n_pr%C3%A1ctica
- Flow, D. (2017). *Diagrams Introduction to data-flow diagrams*. 1–37.
https://www.cs.uct.ac.za/mit_notes/software/pdfs/Chp06.pdf
- Gago, R. (2017). Uso de algoritmos automático aplicado a bases de datos genéticos. *Programación Para La Bioinformática*, 10(2), 44.
- Garzón, J. M. (2017). *Enfoque Psicosomático De La Medición De Estrés*.
- Godoy Trujillo, P. E. (2017). *Un Primer enfoque para el reconocimiento de lenguaje de señas basado en un guante inteligente que utiliza técnicas de Machine Learning*. 93. <https://repositorio.espe.edu.ec/bitstream/21000/13743/5/T-ESPE-057806.pdf> <http://repositorio.espe.edu.ec/bitstream/21000/10846/1/T-ESPE-049674.pdf>
- Gorgues, J. (2017, December 15). *Sensors and Transducers and Introduction*.
https://www.electronics-tutorials.ws/io/io_1.html
- Guinon, J. L., Ortega, E., García-Antón, J., & Pérez-Herranz, V. (2010). Digital Filtering by means of the Fourier Transform Using Mathcad. *International Conference on Engineering Education ICEE*, 1–6.
- Hasanbasic, A., Spahic, M., Bosnjic, D., Adzic, H. H., Mesic, V., & Jahic, O. (2019). Recognition of stress levels among students with wearable sensors. *2019 18th International Symposium INFOTEH-JAHORINA, INFOTEH 2019 - Proceedings, March*, 20–22. <https://doi.org/10.1109/INFOTEH.2019.8717754>
- Hernandez, C. A., & Jacinto, E. (2009). A New Methodology in the Design of Digital

- Filters. *Revista Visión Electronica*, 3(2), 40–47.
- Jansson, P. M., & Schwabe, U. K. W. (2010). Photodiode sensor array design for photovoltaic system inter-row spacing optimization: Calculating module performance during in-situ testing/simulated shading. *2010 IEEE Sensors Applications Symposium, SAS 2010 - Proceedings*, 235–240.
<https://doi.org/10.1109/SAS.2010.5439398>
- Jiménez Mediero, D. (2015). Gestión De La Calidad En Los Requisitos. *UNIVERSIDAD CARLOS III DE MADRID ESCUELA POLITÉCNICA SUPERIOR*, 132. https://e-archivo.uc3m.es/bitstream/handle/10016/25794/PFC_David_Jimenez_Mediero.pdf
- Jiménez, R., & Ángeles, J. C. (2015, October 19). *Pulse Train Converter*. *Electronic Desing*.
<https://www.electronicdesign.com/technologies/analog/article/21801030/versatile-voltage-topulse-train-converter-supports-sensor-data-io>
- Kovacs, M., & Kollar, Z. (2017). Software implementation of the Recursive Discrete Fourier Transform. *2017 27th International Conference Radioelektronika, RADIOELEKTRONIKA 2017, September*.
<https://doi.org/10.1109/RADIOELEK.2017.7936647>
- Kumar, C. S., & Julian, A. (2015). Wearable messaging device for visually impaired person. *Proceedings of 2014 IEEE International Conference on Advanced Communication, Control and Computing Technologies, ICACCCT 2014*, 978, 994–998. <https://doi.org/10.1109/ICACCCT.2014.7019243>
- Labott-Stewart, S. (2018, November). *Stress Effects: Respiratory*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-respiratory#menu>
- Lee, S. P., Lee, J. G., Chang, J. W., Kim, J. N., Lee, S. H., & Chowdhury, S. (2007). Integrated sensor system for humidity sensing of robots using analog mixed CMOS technology. *ROSE 2007 - International Workshop on Robotic and Sensor Environments, Proceedings, October*, 25–28.
<https://doi.org/10.1109/ROSE.2007.4373962>
- Libelium Comunicaciones, D. (2017). *MySignals SW eHealth and Medical IoT Development Platform Technical Guide*.
<https://doi.org/10.5382/econgeo.2018.4561>

- Lin, J. D., Cheng, A. M. K., & Gercek, G. (2016). Partitioning Real-Time Tasks with Replications on Multiprocessor Embedded Systems. *IEEE Embedded Systems Letters*, 8(4), 89–92. <https://doi.org/10.1109/LES.2016.2620473>
- Luque Ordóñez, J. (2016). Dispositivos y tecnologías wearables. *Javier Luque Ordóñez*, 1, 18. https://www.acta.es/medios/articulos/ciencias_y_tecnologia/041001.pdf
- Lüscho, A., & Wartena, C. (2017). Classifying medical literature using k-nearest-neighbours algorithm. *CEUR Workshop Proceedings, 1937(Ddc)*, 26–38.
- Marín Agudelo, H., & Vinaccia Alpi, S. (2005). BIOFEEDBACK. *Informes Psicológicos*, 7(7), 109–121.
- Marshall, D., & Sidorov, K. (2018). Filters and the Fourier Transform Digital Signal Processing. *School of Computer Science & Informatics Cardiff University*, 2, 1–108.
http://users.cs.cf.ac.uk/Dave.Marshall/Multimedia/PDF/02_Filters_and_Fourier_Transforms.pdf
- Martínez, A., & Rodríguez, L. (2019). *BIOFEEDBACK DE LA VARIABILIDAD DE LA FRECUENCIA CARDIACA*. 95–119.
- Martínez Díaz, E. S. (2016, July 25). *¿Qué es el estrés? AEPSAL*.
<https://www.aepsal.com/estres-causas-y-sintomas/>
- Mathur, S., & Malik, S. (2010). Advancements in the V-Model. *International Journal of Computer Applications*, 1(12), 30–35. <https://doi.org/10.5120/266-425>
- Michel, D., Xiao, F., Skillman, L., & Alameh, K. (2016). Surface Plasmon Resonance Sensor for In Situ Detection of Xanthan Gum. *IEEE Journal on Selected Topics in Quantum Electronics*, 22(3), 45–48. <https://doi.org/10.1109/JSTQE.2015.2477054>
- Miyara, F. (2004). *CONVERSORES D/A Y A/D Segunda Edición -Año 2004*.
Electrónica III, 1, 43. <http://www.fceia.unr.edu.ar/enica3/da-ad.pdf>
- Páez, S. (2019). ANÁLISIS COMPARATIVO DE HERRAMIENTAS OPEN SOURCE PARA DATA MINING SOBRE DATOS PÚBLICOS DEL MINISTERIO DE EDUCACIÓN DE LA REPÚBLICA DEL ECUADOR. *Rabit : Jurnal Teknologi Dan Sistem Informasi Univrab*, 1(1), 219.
- Paredes Inilupu, D. (2020, June 26). *Data Science con R*. Análisis de Datos y Algoritmos de Predicción. <https://bookdown.org/dparedesi/data-science-con-r/>
- Planning, T. (2017). *V_Model*. 1–3. <http://www.softwaretestingclass.com/v-model/>

- Qing, L., & Yao, C. (2013). *Real-Time Concepts for Embedded Systems* (R. Ward & M. Briand (eds.)). CMP Books. <http://www.kelm.ftn.uns.ac.rs/literatura/mrv/Real-Time.Concepts.For.Embedded.Systems.eBook-LiB.pdf>
- Quespaz, C. (2019, June 11). *Reconocimiento de patrones de datos de WSN enfocado a regadíos por ambientes controlados de rosas a escala de laboratorio*. <http://repositorio.utn.edu.ec/handle/123456789/9197>
- Relaxing white noise. (2018, July 9). *2 min Breathe Bubble | Breathe Exercises - Sea - Think Nothing Exercise I Breathe In Calm App*. https://www.youtube.com/watch?v=9tOJZQhO_Uw
- Robinson, J. (2017, December). *The Effects of Stress*. <https://www.webmd.com/balance/stress-management/effects-of-stress-on-your-body>
- Sánchez, J. (2019). *ESTRÉS ACADÉMICO, INTERNOS ROTATIVOS DE ENFERMERÍA DE LA UNIVERSIDAD TÉCNICA DEL NORTE EN LA PROVINCIA DE IMBABURA*.
- Sarraipa, J., Páez, J., López, J., & Fino, R. (2019). Manual de MySignals. *ACACIA*, 1–27.
- Sathish Kannan, K., & Thilagavathi, G. (2013). Online farming based on embedded systems and wireless sensor networks. *Proceedings of International Conference on Computation of Power, Energy, Information and Communication, ICCPEIC 2013*, 71–74. <https://doi.org/10.1109/ICCPEIC.2013.6778501>
- Shaw, W. (2018, November). *Stress Effects: Musculoskeletal*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-musculoskeletal#menu>
- Shrivastava, P., Bhatnagar, A., Desai, J., & Gajjar, S. (2019). DDAS: Distributed data acquisition system using wireless sensor networks. *2019 IEEE 16th India Council International Conference, INDICON 2019 - Symposium Proceedings, i*, 19–22. <https://doi.org/10.1109/INDICON47234.2019.9029077>
- Singh, S., & Ranjan, P. (2011). Towards a new low cost, simple implementation using embedded system wireless networking for UAVs. *International Symposium on Advanced Networks and Telecommunication Systems, ANTS*, 1–3. <https://doi.org/10.1109/ANTS.2011.6163667>
- Sterpin, D. (2011). *Perceptrón Auto-Supervisado De Una Red Neuronal Artificial*. 90–101.

- Tan, L., & Jiang, J. (2019). *Digital Signal Processor*. Digital Signal Processing (Third Edition). <https://www.sciencedirect.com/topics/computer-science/digital-signal-processor>
- Tilburg, M. (2018, November). *Stress Effects: Gastrointestinal*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-gastrointestinal#menu>
- Timothy J., L. (2017, June 5). *The Effects of Stress*. Healthline. <https://www.healthline.com/health/stress/effects-on-body#1>
- Tomalá, V. (2018). *SISTEMA ELECTRÓNICO DE MONITOREO DE NIVELES DE ESTRÉS ACADÉMICO*. 2, 227–249.
- Torgaev, S. N., Di, W., Chertikhina, D. S., Kozhemyak, O. A., & Ponomarev, S. V. (2015). Hardware-software complex for studying of digital filters. *2015 International Siberian Conference on Control and Communications, SIBCON 2015 - Proceedings*, 9–11. <https://doi.org/10.1109/SIBCON.2015.7147303>
- Tovian, S. M. (2018, November). *Stress Effects: Male Reproductive System*. American Psychological Association. <https://www.apa.org/helpcenter/stress/effects-male-reproductive#menu>
- Tutorialspoint. (2016). SDLC - V-Model. *Tutorialspoint*, http://www.tutorialspoint.com/sdlc/sdlc_v_model.ht.
http://www.tutorialspoint.com/sdlc/sdlc_v_model.ht
- Wei, H., & Wang, T. (2007). Curriculum of embedded system for software colleges. *Proceedings of the 2nd IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA 2006*. <https://doi.org/10.1109/MESA.2006.296948>
- Widanti, N., Sumanto, B., Rosa, P., & Fathur Miftahudin, M. (2015). Stress level detection using heart rate, blood pressure, and GSR and stress therapy by utilizing infrared. *2015 International Conference on Industrial Instrumentation and Control, ICIC 2015, Iic*, 275–279. <https://doi.org/10.1109/IIC.2015.7150752>
- Wilson, J. S. (2008). *Intro to Sensors Overview*. https://books.google.de/books?hl=en&lr=&id=5UE6YCjDG-MC&oi=fnd&pg=PP1&dq=introduction+to+sensor+technology&ots=jfGSg8gGLV&sig=bVPdfgAc04QyWenTwCtQyy8588o&redir_esc=y#v=onepage&q=introduction+to+sensor+technology&f=false

- Zahradnik, P., & Šimák, B. (2012). Education in real-time digital signal processing using digital signal processors. *2012 35th International Conference on Telecommunications and Signal Processing, TSP 2012 - Proceedings*, 625–628.
<https://doi.org/10.1109/TSP.2012.6256372>
- Zamorano, J. (2018). *Comparativa Y Análisis De Algoritmos de Aprendizaje Automáticos*. 116.
- Zumbahlen, H. (2008). The Op Amp. *Linear Circuit Design Handbook*, 1–82.
<https://doi.org/10.1016/b978-0-7506-8703-4.00001-8>

7. GLOSARIO

AIRFLOW: El sensor de flujo de aire nasal es un dispositivo que se utiliza para medir la frecuencia respiratoria en un paciente.

BIOFEEDBACK: Retroalimentación biológica, se define como un procedimiento que permite al paciente aprender a controlar respuestas fisiológicas de forma voluntaria.

BIOSTRESS: Sistema *Biofeedback* de sensores de predicción de estrés mediante algoritmos de aprendizaje.

CASE: Armazón del sistema BIOSTRESS.

CSV: Valores separados por comas, es un tipo especial de archivo que puede crear o editar en Excel.

ECG: El electrocardiograma es un sensor de diagnóstico que se utiliza de forma rutinaria para evaluar las funciones eléctricas y musculares del corazón.

E-HEALTH: Conocida como ciber salud consiste en el apoyo que la utilización costo eficaz y segura de las tecnologías de la información y las comunicaciones ofrece a la salud y a los ámbitos relacionados con ella.

GSR: Este sensor mide la conductancia eléctrica de la piel, que varía con su nivel de humedad.

HARDWARE: Conjunto de elementos físicos o materiales que constituyen un sistema.

KNN: Es un algoritmo de aprendizaje supervisado de *Machine Learning*. Puede usarse para clasificar nuevas muestras o para predecir, sus siglas en inglés significan K vecinos más cercanos.

MYSIGNALS: Es una plataforma de desarrollo para dispositivos médicos y aplicaciones de *e-Health*.

OPEN SOURCE: Es un código diseñado de manera que sea accesible al público.

PROCESSING: Es un software libre diseñado para el desarrollo de interfaces gráficas.

SOFTWARE: Conjunto de programas y rutinas que permiten al sistema realizar determinadas tareas.

STAKEHOLDERS: Es cualquier individuo que, de alguna manera, es impactado por las acciones de determinada empresa, significa partes interesadas.

TEST: Es una prueba para comprobar el funcionamiento de un sistema.

8. ANEXOS

ANEXO 1. Formato de Encuestas con Tabulación de los Resultados y Entrevista

ENCUESTA DESTINADA A ESTUDIANTES UNIVERSITARIOS

Esta encuesta está dirigida a los estudiantes de la Universidad Técnica del Norte con la finalidad de obtener información sobre los requerimientos de usuarios para la elaboración del sistema BIOSTRESS.

1. Cuándo se encuentra en el ciclo académico, ¿Usted cuenta con un horario adecuado para descansar por las noches?
 - a) Si
 - b) No

2. ¿Cuántas hora duerme diariamente cuando se encuentra en el período académico?
 - a) 8 o más horas
 - b) 7 horas
 - c) 6 horas
 - d) 5 o menos horas

3. ¿Durante el período académico usted lleva una dieta equilibrada?
 - a) Si
 - b) No

4. ¿Qué alimentos consume frecuentemente cuando se encuentra en la universidad?
 - a) Yogurt, batidos, jugos, frutas.
 - b) Comida rápida
 - c) Snacks
 - d) Ensaladas
 - e) Almuerzos

f) Desayunos

g) Meriendas

5. ¿Usted cuenta con ayuda económica para solventar sus estudios académicos?

a) Si

b) No

6. ¿Usted trabaja para poder solventar sus gastos académicos?

a) Si

b) No

7. ¿Cuánto de presupuesto semanal dispone para sus gastos académicos?

a) De 5 a 15 dólares

b) De 16 a 25 dólares

c) De 26 a 35 dólares

d) Más de 35 dólares

8. ¿Usted prefiere cumplir con todas sus responsabilidades académicas a pesar de no dormir las horas establecidas saludablemente?

a) Si

b) No

9. ¿Al terminar la jornada diaria, usted se siente agobiado por las tareas de la universidad y sus tareas personales?

a) Si

b) No

10. ¿Usted tiene cargas familiares?

a) Si

b) No

11. ¿Cree usted que el estrés académico afecta a la salud del estudiante?

a) Si

b) No

12. ¿Usted considera útil un dispositivo electrónico que ayude a conocer los niveles de estrés con el fin de prevenir posibles enfermedades?

a) Si

b) No

- **Tabulación de resultados.**

1. Cuando se encuentra en el ciclo académico, ¿Usted cuenta con un horario adecuado para descansar por las noches?

100 respuestas

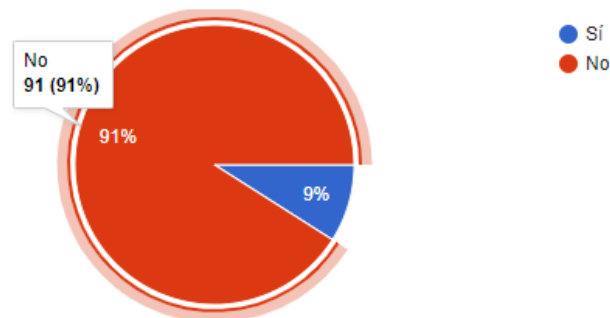


Figura. 1. Gráfica del Análisis - Pregunta 1

Fuente: autoría.

Como resultado de la pregunta 1 se puede observar que el 91% de la población encuestada no cuenta con un horario adecuado para descansar por las noches cuando se encuentran cursando un período académico. Esto quiere decir que los estudiantes no establecen un tiempo adecuado para su descanso.

2. ¿Cuántas horas duerme diariamente cuando se encuentra en el periodo académico?

100 respuestas



Figura. 2. Gráfica del Análisis - Pregunta 2

Fuente: autoría.

El resultado de la pregunta 2 indica que durante el período académico el 59% de los encuestados duermen 5 o menos horas, seguido del 35% de los encuestados que duermen 6 horas diarias y el resto que sería el 6% están entre 7 y 8 horas de descanso, lo cual queda demostrado que los estudiantes encuestados no cuentan con un horario fijo de descanso.

3. ¿Durante el periodo académico usted lleva una dieta equilibrada?

100 respuestas

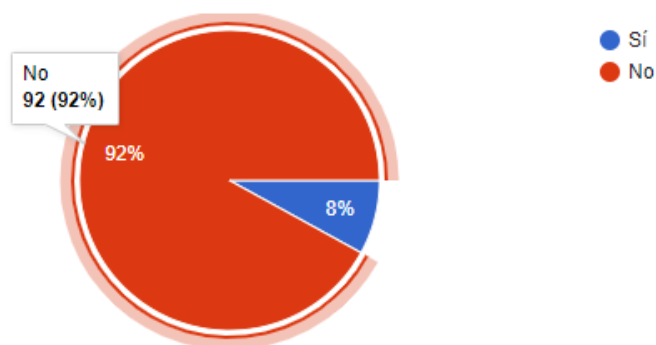


Figura. 3. Gráfica del Análisis - Pregunta 3

Fuente: autoría.

Como resultado de la pregunta 3 se obtiene que durante el período académico los estudiantes encuestados en un 92% no llevan una dieta equilibrada y sólo el 8% de ellos cuentan con una dieta equilibrada, lo que es preocupante puesto que se puede producir diferentes enfermedades a lo largo del período académico.

4. ¿Qué alimentos consume frecuentemente cuando se encuentra en la universidad?

100 respuestas

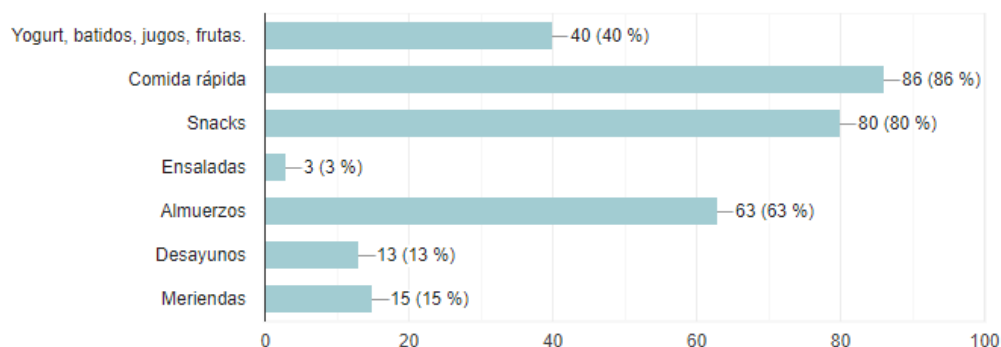


Figura. 4. Gráfica del Análisis - Pregunta 4

Fuente: autoría.

El resultado a la pregunta 4 indica los alimentos que consumen frecuentemente los estudiantes a la hora que se encuentran en la universidad, el 86% de los estudiantes encuestados afirman que consumen comida rápida seguidamente con el 80% que consumen snacks, por lo que estas dos opciones serían las más elegidas por los encuestados. El resto de las comidas que se podría considerar un poco más saludables cuentan con los siguientes porcentajes: el 63% consume almuerzos, el 40% está entre yogurt, batidos, jugos y frutas, el 15% consume meriendas, el 13% desayunos y el 3% consume ensaladas.

5. ¿Usted cuenta con ayuda económica para solventar sus estudios académicos?

100 respuestas

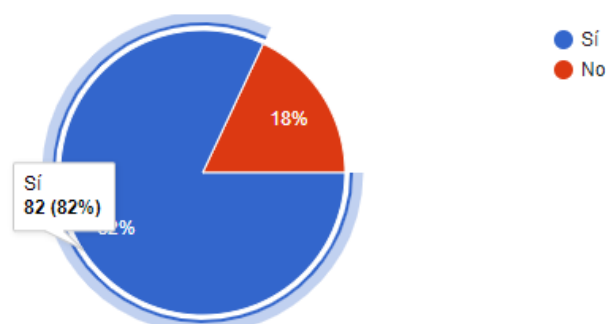


Figura. 5. Gráfica del Análisis - Pregunta 5

Fuente: autoría.

Como resultado de la pregunta 5 se obtiene que el 82% de los estudiantes encuestados poseen una ayuda económica para solventar sus estudios, al contrario que el 18% de los encuestados no poseen esta ayuda económica.

6. ¿Usted trabaja para poder solventar sus gastos académicos?

100 respuestas

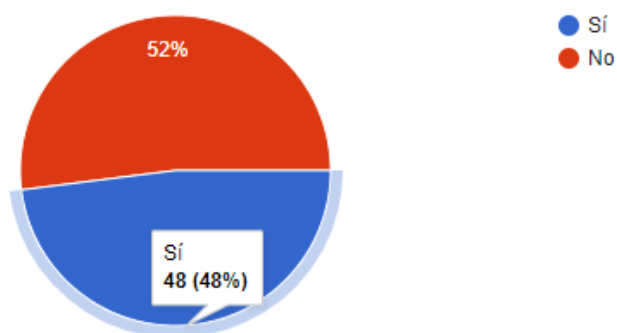


Figura. 6. Gráfica del Análisis - Pregunta 6

Fuente: autoría.

El resultado de la pregunta 6 indica que el 52% de los estudiantes encuestados no trabajan para solventar sus gastos académicos al contrario del 48% de los encuestados que si trabajan para solventar sus gastos.

7. ¿Cuánto de presupuesto semanal dispone para sus gastos académicos?

100 respuestas

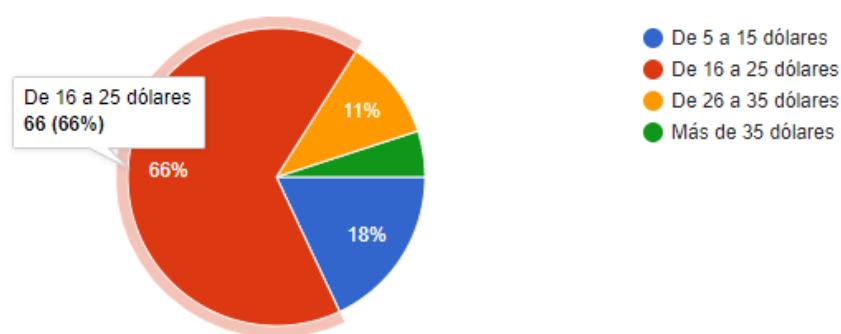


Figura. 7. Gráfica del Análisis - Pregunta 7

Fuente: autoría.

Como resultado de la pregunta 7 se obtiene que el 66% de los encuestados cuentan con un presupuesto semanal entre 16 a 25 dólares disponibles para sus gastos académicos,

el 18% de los estudiantes encuestados cuentan con 5 a 15 dólares para dichos gastos y el restante que es el 16% están entre 26 hasta más de 35 dólares.

8. ¿Usted prefiere cumplir con todas sus responsabilidades académicas a pesar de no dormir las horas establecidas saludablemente?

100 respuestas

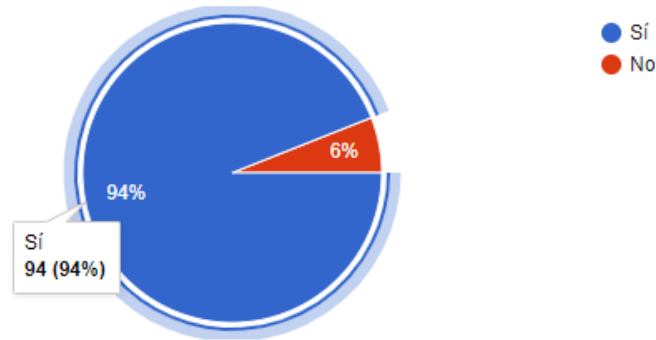


Figura. 8. Gráfica del Análisis - Pregunta 8

Fuente: autoría.

El resultado de la pregunta 8 indica que el 94% de los estudiantes encuestados si prefieren cumplir con todas las obligaciones académicas a pesar de no dormir las horas establecidas saludablemente, y el restante 6% no prefiere cumplir con sus responsabilidades académicas y dormir las horas establecidas saludablemente.

9. ¿Al terminar la jornada diaria, usted se siente agobiado por las tareas de la universidad y sus tareas personales?

100 respuestas

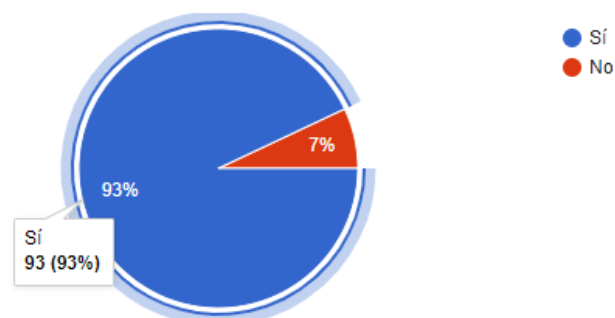


Figura. 9. Gráfica del Análisis - Pregunta 9

Fuente: autoría.

Como resultado a la pregunta 9 se obtiene que, al terminar la jornada diaria el 93% de los estudiantes encuestados se sienten agobiados por las tareas que tienen de la

universidad y sus tareas personales, solo el 7% de los encuestados no se sienten agobiados.

10. ¿Usted tiene cargas familiares?

100 respuestas

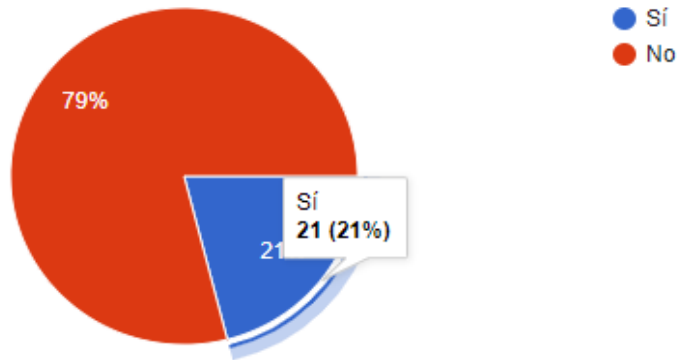


Figura. 10. Gráfica del Análisis - Pregunta 10

Fuente: autoría.

El resultado de la pregunta 10 indica que en la mayoría de encuestados con un 79% no cuentan con cargas familiares y el 21% de los estudiantes encuestados si tienen cargas familiares.

11. ¿Cree usted que el estrés académico afecta a la salud del estudiante?

100 respuestas

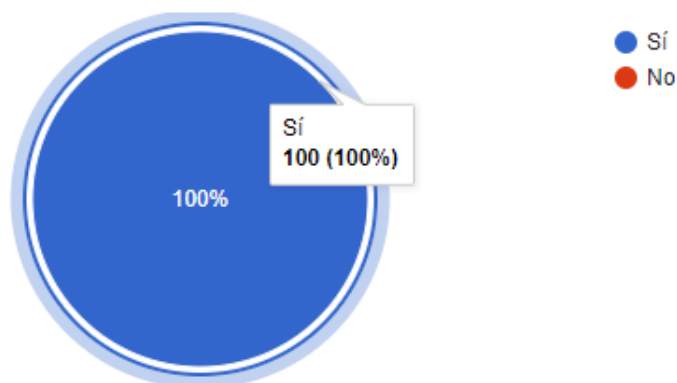


Figura. 11. Gráfica del Análisis - Pregunta 11

Fuente: autoría.

Como resultado de la respuesta 11 se obtiene que todos los estudiantes encuestados creen que el estrés académico afecta a la salud de los estudiantes.

12. ¿Usted considera útil un dispositivo electrónico que ayude a conocer los niveles de estrés con el fin de prevenir posibles enfermedades?

100 respuestas

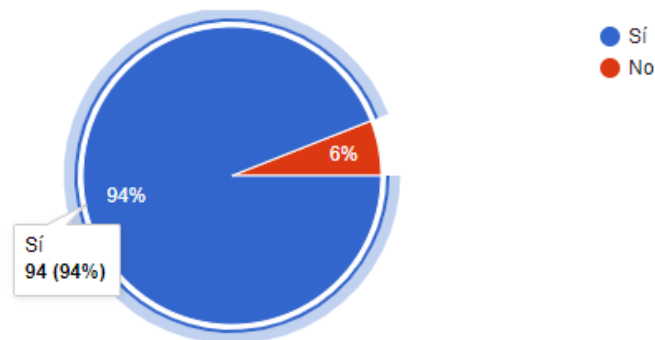


Figura. 12. Gráfica del Análisis - Pregunta 12

Fuente: autoría.

El resultado de la pregunta 12 indica que más de la mayoría de los encuestados con un 94% consideran útil la creación de un dispositivo electrónico que ayude a conocer los niveles de estrés con el fin de prevenir posibles enfermedades.

ENCUESTA UTILIZADA PARA LA OBTENCIÓN DE REQUERIMIENTOS DE USUARIO

Esta encuesta está dirigida a los estudiantes de la Universidad Técnica del Norte con la finalidad de obtener información sobre los requerimientos de usuarios para la elaboración del sistema BIOSTRESS.

1. ¿Considera usted que, el dispositivo BIOSTRESS debe ser liviano?
 - a) Si
 - b) No
2. ¿Usted cree que, para facilidad de uso del dispositivo, este debe tener identificadores de uso?
 - a) Si
 - b) No
3. ¿Cree usted que el dispositivo debe tener información clara y precisa?

a) Si

b) No

4. ¿Usted considera que el armazón del dispositivo debe ser de bajo costo económico?

a) Si

b) No

- **Tabulación de resultados.**

1. ¿Considera usted que, el dispositivo BIOSTRESS debe ser liviano?

100 respuestas

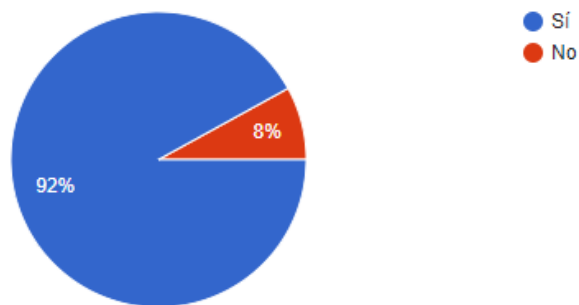


Figura. 13. Gráfica del Análisis - Pregunta 1-Encuesta 2

Fuente: autoría

El 92% de los estudiantes encuestados consideran que el dispositivo debe ser liviano, esta información se utiliza para realizar el requerimiento de usuario número 1.

2. ¿Usted cree que, para facilidad de uso del dispositivo, este debe tener identificadores de uso?

100 respuestas

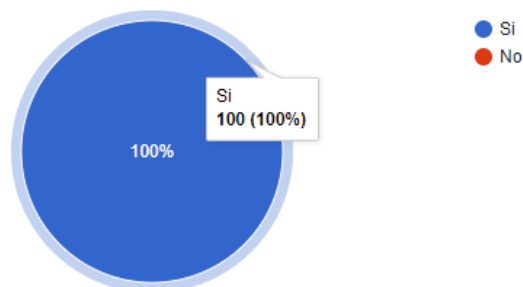


Figura. 14. Gráfica del Análisis - Pregunta 2-Encuesta 2

Fuente: autoría

Todos los estudiantes encuestados creen que el dispositivo debe contar con identificadores para la facilidad de uso del dispositivo.

3. ¿Cree usted que el dispositivo debe tener información clara y precisa?

100 respuestas

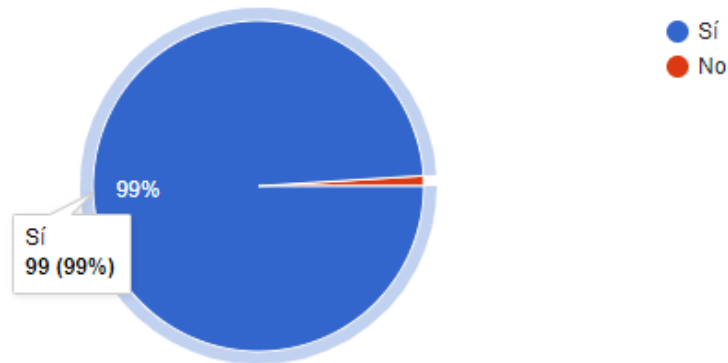


Figura. 15. Gráfica del Análisis - Pregunta 3-Encuesta 2

Fuente: autoría

El 99% de los encuestados están de acuerdo en que el dispositivo debe tener información clara y precisa.

4. ¿Usted considera que el armazón del dispositivo debe ser de bajo costo económico?

100 respuestas

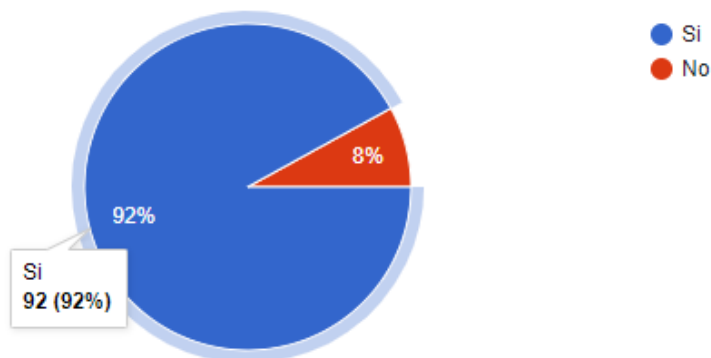


Figura. 16. Gráfica del Análisis - Pregunta 4-Encuesta 2

Fuente: autoría

El armazón del dispositivo debe ser de bajo costo económico según el 92% de los estudiantes encuestados.

ENTREVISTA UTILIZADA PARA LA OBTENCIÓN DE REQUERIMIENTOS DE USUARIO

La siguiente entrevista tiene como finalidad recaudar información para obtener requerimientos operacionales de los usuarios, por ese motivo se realizó la entrevista a un profesional en el ámbito (psicólogo). La entrevista cuenta de las siguientes preguntas:

1. ¿Considera usted que es de gran importancia utilizar la tecnología para ayudar a solucionar enfermedades que afecten a los estudiantes?

Sí, pienso que la tecnología juega un papel muy importante en muchos aspectos en la actualidad y en el ámbito de la salud se está viendo cada vez que tiene aportes muy significativos, y un dispositivo elaborado por estudiantes de la universidad es un aporte muy grande puesto que ayuda a involucrarse más en los problemas de la actualidad.

2. ¿Usted conoce dispositivos tecnológicos que ayuden a detectar estrés en las personas?

Existen dispositivos que detectan cambios fisiológicos que produce el estrés, conozco dispositivos que son costosos y no están especializados solamente para detectar estrés.

3. ¿Qué parámetros se podría medir para detectar estrés en las personas?

Se debe comprender como actúa el sistema nervioso del ser humano ante el estrés, para ello voy a dar a conocer que el sistema nervioso parasimpático es el sistema en el que debemos vivir, y el sistema nervioso simpático es el que se activa ante una situación estresante y como respuesta a esta condición subjetiva se tiene respuestas objetivas como el aumento de las pulsaciones, concentrar la sangre en los músculos, la respiración se acelera, la presión arterial aumenta, el estado de insulina comienza a elevarse. En el estrés crónico se mantiene activo el sistema nervioso simpático y la manera de controlar las respuestas fisiológicas es mediante la respiración ya que es la única forma de que se diga

al cerebro que se encuentra en sistema parasimpático entonces se debe regular los otros órganos.

4. ¿De los dispositivos que conoce o a utilizado para detectar estrés, necesitan estar conectados a la energía siempre?

Claro, los dispositivos que se utilizan siempre deben estar conectados a la energía para que funcione y se pueda obtener los cambios fisiológicos de humano.

5. ¿Cómo consideraría a un dispositivo electrónico que tenga inconvenientes al momento que se encuentre realizando un test?

Sería muy factible que el dispositivo funcione durante el tiempo que se demore en realizar el test puesto que se toma los cambios fisiológicos del cuerpo al instante, y si se tuviera inconvenientes perdería la credibilidad en ese dispositivo.

6. ¿Usted cree que sería bueno evidenciar los niveles de estrés en una aplicación?

Sí, eso sería lo ideal porque se podría ver los valores obtenidos de los pacientes y así poder regular estos valores siempre y cuando se encuentren fuera de lo normal.



Figura. 17. Entrevista virtual con el psicólogo.

Fuente: autoría

ANEXO 2. Programación de BIOSTRESS

```
#include <Arduino_KNN.h>           //Libreria para clasificación KNN

#include <MySignals.h>             //Libreria Modulo MySignals

#include "Wire.h"                  //Libreria I2C para comunicación con mysignals

#include "SPI.h"                   //Libreria requerida por I2C

const int GSR = A0;               //Puerto donde se encuentra conetado el sensor GSR

int sensorValue;                 //Variable para almacenar valor de conductivida del sensor GSR

int gsr_average;                 //Variable que alacena el promedio de la conductividad

int BPM;                          //Variable que almacena la frecuencia cardíaca en pulsos por minuto BPM

int RPM ;                         //Variable que almacena la frecuencia respiratoria

//VARIABLES DE CALCULO DE FRECUENCIA RESPIRATORIA Y CARDÍACA//

long ultimamedicion;

int ciclo = 0; //1=alto 0=bajo

int cambiodeciclo = 0;

int picodetension;

int valledetension;

long contadorciclo;

///Variables requeridas para el cálculo de tiempos de toma de datos usando el función
millis///

boolean estadoBPM = true;

boolean estadoLed = true;

int IntervaloEncendido = 200;

int IntervaloApagado = 50;
```

```

unsigned long IntervaloBPM = 60000;    ///Tiempo de toma de datos

int IntervaloBPM2 = 1;

extern volatile unsigned long timer0_millis;

unsigned long nuevo_valor = 0;

unsigned long tiempoAnteriorEncendido = 0; //guarda el tiempo de la referencia para
comparar

unsigned long tiempoAnteriorApagado = 0;

unsigned long tiempoAnteriorBPM = 0;

unsigned long tiempoAnteriorBPM2 = 0;

int pulsos = 0; //Almacena los pulsos para el cálculo de la frecuencia

int pulsos2 = 0;

String dato; // Almacena la cadena de envío de datos

int codigo = 0; //Variable que permite identificar el sensor que se está ejecutando

///Variables KNN///

KNNClassifier myKNN(3);

void setup()

{Serial.begin(9600); //Inicializa comunicación serial a 9600 baudios

  MySignals.begin(); //Inicializa la Cx con la placa MySignals

  EntrenamientoKNN(); //Llama la función de entrenamiento del sistema la misma
que contiene la base de datos para la clasificación KNN

}

void loop()

{ if (Serial.available() > 0) { // Condición que espera la recepción de datos

  char letra = Serial.read();

```

```

    if (letra == '1') {          //Si el dato recibido es igual a 1 inicializa la lectura del sensor
de pulso cardíaco

    delay(1000);

    pulsos2 = 0;

    codigo = 1;

    setMillis(nuevo_valor);    //Reinicia el contador Millis para un adecuado uso del
temporizador

    tiempoAnteriorBPM = 0;

    }

    if (letra == '2') {          //Si el dato recibido es igual a 2 inicializa la lectura del sensor
de frecuencia respiratoria

    delay(1000);

    pulsos2 = 0;

    codigo = 2;

    setMillis(nuevo_valor);

    tiempoAnteriorBPM = 0;

    }

    if (letra == '3') {          //Si el dato recibido es igual a 3 inicializa la lectura del
sensor GSR

    delay(1000);

    codigo = 3;

    pulsos2 = 0;

    setMillis(nuevo_valor);

    tiempoAnteriorBPM = 0;

```

```

}

if (letra == '4') {          //Si el dato recibido es igual a 4 cancela el envío de datos

delay(1000);

codigo = 4;

}

if (letra == '5') {        ///Si el dato recibido es igual a 5 realiza el proceso de KNN
para la clasificación

codigo = 5;

}

}

if (codigo == 1) {

uint16_t air = (uint16_t)MySignals.getAirflow(DATA);          //Lectura del sensor del
flujo de aire para el cálculo de la frecuencia respiratoria

CalculoRPM(air);          // Llama la función que calcula la
frecuencia respiratoria en respiraciones por minuto RPM

RPM = pulsos2;

dato = "";

dato += air;

dato += ",";

dato += RPM;

Serial.println(dato);

if (RPM > 0) {

codigo = 4;

}
}

```

```

}

if (codigo == 2) {

    uint16_t ECG = (uint16_t) MySignals.getECG(DATA);           //Lectura del sensor
EGC para el cálculo de frecuencia cardíaca

    CalculoRPM(ECG);

    BPM = pulsos2 ;

    dato = "";

    dato += ECG;

    dato += ",";

    dato += BPM;

    Serial.println(dato);

    delay(5);

    if (BPM > 0) {

        codigo = 4;

    }

}

if (codigo == 3) {

    long sum = 0;

    for (int i = 0; i < 30; i++)

    {

        sensorValue = analogRead(GSR);                       //Lectura del Sensor GSR para
obtener la conductividad de la piel

        sensorValue = map(sensorValue, 1023, 0, 450, 0);

        dato = "";

```



```

    dato += sensorValue;

    dato += ",";

    dato += i;

    Serial.println(dato);

    delay(1000);

}

codigo = 4;

}

if (codigo == 5) {                                     //Calcula KNN

    float input[] = { RPM, BPM, sensorValue};

    Serial.println(sensorValue);

    int classification = myKNN.classify(input, 5); // clasifica en base a K=3

    int confidence = myKNN.confidence();

    dato = "";

    dato += classification;

    dato += ",";

    dato += confidence;

    Serial.println(dato);

    codigo = 4;

}

}

void CalculoRPM(int Datos) {                             //Función para el cálculo de frecuencia

    int senal = Datos;

```

```
if (senal >= ( ultimamedicion + 200))
```

```
{  ultimamedicion = senal;
```

```
  ciclo = 1;
```

```
  if (senal > picodetension)
```

```
  {  picodetension = senal;
```

```
  }
```

```
}
```

```
if (senal <= (ultimamedicion - 200))
```

```
{
```

```
  ultimamedicion = senal;
```

```
  ciclo = 0;
```

```
  if (senal < valledetension)
```

```
  {
```

```
    valledetension = senal;
```

```
  }
```

```
}
```

// Función que permite calcular la frecuencia cardíaca y respiratoria en 1 minuto

```
if (millis() - tiempoAnteriorBPM >= IntervaloBPM) {
```

```
  estadoBPM = false;
```

```
  pulsos2 = pulsos;
```

```
  tiempoAnteriorBPM = millis();
```

```
  pulsos = 0;
```

```
}
```

```
if (millis() - tiempoAnteriorBPM2 >= IntervaloBPM2) {
```

```

    estadoBPM = true;

    pulsos2 = (pulsos2 + pulsos2) / 2;

    tiempoAnteriorBPM2 = millis();
}

if ((millis() - tiempoAnteriorEncendido >= IntervaloEncendido) && estadoLed ==
true && ciclo == 0)
{
    estadoLed = false;

    picodetension = senal;

    valledetension = senal;

    tiempoAnteriorApagado = millis();
}

if ((millis() - tiempoAnteriorApagado >= IntervaloApagado) && estadoLed == false
&& ciclo == 1) //pulso
{
    pulsos++;

    picodetension = senal;

    valledetension = senal;

    estadoLed = true;

    tiempoAnteriorEncendido = millis();
}
}

void EntrenamientoKNN() {

    float example1[] = { 14, 50, 440};

    float example2[] = { 13, 54, 425};

```

```
float example3[] = {17, 63, 364};  
float example4[] = {18, 55, 355};  
float example5[] = {17, 55, 430};  
float example6[] = {18, 64, 420};  
float example7[] = {16, 50, 440};  
float example8[] = {14, 58, 375};  
float example9[] = {20, 65, 435};  
float example10[] = {19, 60, 400};  
float example11[] = {24, 72, 310};  
float example12[] = {27, 78, 350};  
float example13[] = {22, 78, 340};  
float example14[] = {25, 69, 310};  
float example15[] = {26, 69, 335};  
float example16[] = {22, 68, 315};  
float example17[] = {26, 72, 305};  
float example18[] = {25, 71, 320};  
float example19[] = {25, 73, 307};  
float example20[] = {23, 69, 339};  
float example21[] = {30, 82, 299};  
float example22[] = {29, 85, 280};  
float example23[] = {33, 81, 240};  
float example24[] = {34, 87, 285};  
float example25[] = {38, 86, 297};  
float example26[] = {29, 80, 270};
```

```
float example27[] = {31, 80, 290};  
float example28[] = {35, 85, 295};  
float example29[] = {32, 90, 288};  
float example30[] = {28, 79, 295};  
myKNN.addExample(example13, 2); // 1=ESTADO NORMAL  
myKNN.addExample(example6, 1); // 2=ALGO ELEVADO  
myKNN.addExample(example14, 2); // 3=ALTO  
myKNN.addExample(example1, 1);  
myKNN.addExample(example27, 3);  
myKNN.addExample(example18, 2);  
myKNN.addExample(example28, 3);  
myKNN.addExample(example5, 1);  
myKNN.addExample(example7, 1);  
myKNN.addExample(example10, 1);  
myKNN.addExample(example21, 3);  
myKNN.addExample(example22, 3);  
myKNN.addExample(example24, 3);  
myKNN.addExample(example8, 1);  
myKNN.addExample(example25, 3);  
myKNN.addExample(example9, 1);  
myKNN.addExample(example26, 3);  
myKNN.addExample(example11, 2);  
myKNN.addExample(example15, 2);  
myKNN.addExample(example23, 3);
```

```

myKNN.addExample(example16, 2);

myKNN.addExample(example2, 1);

myKNN.addExample(example3, 1);

myKNN.addExample(example4, 1);

myKNN.addExample(example17, 2);

myKNN.addExample(example29, 3);

myKNN.addExample(example12, 2);

myKNN.addExample(example30, 3);

myKNN.addExample(example19, 2);

myKNN.addExample(example20, 2);

}

void setMillis(unsigned long nuevo_valor) {

    uint8_t oldSREG = SREG;

    cli();

    timer0_millis = nuevo_valor;

    SREG = oldSREG;

}

```

Programación de visualización de datos

```

import javax.swing.JOptionPane; //Libreria para ejecutar ventanas de dialogo y avisos

import controlP5.*; //Libreria para el control de Botones y Medidores digitales de
la Interfaz

import processing.serial.*; //Libreria para envio y recepción de datos desde arduino

import processing.video.*;

Movie myMovie;

```

```

ControlP5 cp5;          //Declaración de Variable para el uso de control P5 que
permite controlar los botones y medidores digitales

Serial arduino;        //Declaración de variable para el uso de la comunicación
serial que permite el envío y recepción de datos de //arduino.

boolean newData = false;    //Declaración de variable requerida para identificar
si se reciben nuevos datos a travez de comunicación serial

boolean reset=false;      //Variable utilizada para cancelar la graficación de
datos en pantalla

boolean obtenerResultado;

///Variables requeridas para el control y graficación de botones de cambio de pantalla///

static final int MAX = 2, GAP = 50, DIM = 120, RAD = DIM >> 1;

int page, cx, cy;

Button back, next;

///Variables requeridas para graficación de medidores digitales///

Knob myKnobA; //Medidor digital de frecuencia cardíaca

Knob myKnobB; //Medidor digital de frecuencia respiratoria

Knob myKnobC; //Medidor digital de conductividad de la piel

///Variables de ubicación de la gráfica Electrocardiograma///

int xPos = 350 ;

int lastxPos=350;

int lastheight=400;

///Variables de ubicación de la gráfica de respiraciones///

int xPos2 = 350 ;

int lastxPos2=350;

```

```

int lastheight2=200;

//Variables para graficación de botones//

int x=1100, y=135, w=120, h=25;      //Boton de inicio de toma de frecuencia
cardíaca x:ubicación en el eje X, y: ubicación en el eje Y de la ventana, w: ancho de
boton, h: altura del boton

int x2=1100, y2=340, w2=120, h2=25;  //Boton de inicio de toma de frecuencia
respiratoria x:ubicación en el eje X, y: ubicación en el eje Y de la ventana, w: ancho de
boton, h: altura del boton

int x4=1100, y4=545, w4=120, h4=25;  //Boton de inicio de toma de conductividad
x:ubicación en el eje X, y: ubicación en el eje Y de la ventana, w: ancho de boton, h:
altura del boton

int x5=950, y5=630, w5=120, h5=25;   //Boton de inicio de toma de conductividad
x:ubicación en el eje X, y: ubicación en el eje Y de la ventana, w: ancho de boton, h:
altura del boton

int x6=1100, y6=500, w6=120, h6=25;  //Boton de inicio de toma de conductividad
x:ubicación en el eje X, y: ubicación en el eje Y de la ventana, w: ancho de boton, h:
altura del boton

int x7=1100, y7=550, w7=120, h7=25;  //Boton de inicio de toma de conductividad
x:ubicación en el eje X, y: ubicación en el eje Y de la ventana, w: ancho de boton, h:
altura del boton

//Variables para la recepción de datos desde Arduino//

int valores[] = {0, 0};

float signal;

int frecuencia;

int sensor;

```



```

int estado;

int confiabilidad;

///Variables Tipo PImage para cargar imagenes a la interfaz///  

PImage imagen;

PImage imagen2;

PImage imagen3;

PImage imagen4;

PImage imagen5;

PImage imagen6;

PImage imagen7;

///Variables que almacenan los resultados obtenidos///  

int BPM;

int RPM;

float GSR;

Table table;

int estado2;

boolean play;

void setup() {

    size(1280, 720);           ///Tamaño de la ventana de la interfaz

    imagen=loadImage("1.jpg");  ///Cargar imagen de portada

    imagen2=loadImage("4.jpg");  ///cargar imagen de pantalla de toma de datos

    imagen4=loadImage("NORMAL.jpg");  ///Cargar imagen de portada

    imagen5=loadImage("ELEVADO.jpg");  ///cargar imagen de pantalla de toma de  

datos

```

```

imagen6=loadImage("ALTO.jpg"); //Cargar imagen de portada

imagen7=loadImage("video.jpg"); //Cargar imagen de portada

table = loadTable("datos.csv", "header");

textAlign(CENTER, CENTER); //Función que permite tomar como referencia el
centro de la pantalla para la ubicación de textos

cx = width >> 1; //Variables requeridas para la función de cambio de pantalla

cy = height >> 1;

back = new Button("BACK", GAP, height - Button.H - GAP);

next = new Button("COMENZAR", width - Button.W - GAP, height - Button.H -
GAP);

/// Parametros para graficación de los medidores digitales tipo knob///

cp5 = new ControlP5(this);

myKnobA = cp5.addKnob("Pulsos por minuto") //Titulo de knob

.setRange(0, 200) //Rango de 0 - 200 para le medidor de frecuencia cardíaca

.setPosition(850, 50) //Posición dentro de la ventana de la interfaz

.setRadius(60); //Tamaño

myKnobB = cp5.addKnob("Respiraciones por minuto")

.setRange(0, 60)

.setPosition(850, 250)

.setRadius(60);

myKnobC = cp5.addKnob("Conductividad")

.setRange(0, 1023)

.setPosition(850, 450)

.setRadius(60);

```

```

///Inicialización de la Cx Serial en el Puerto 8///

arduino = new Serial(this, "COM8", 9600);

arduino.bufferUntil('\n');          //Establece un byte específico para
almacenar en búfer hasta antes de llamar a serialEvent ().

}

void draw() {

  pageSelector();    // Función para el cambio de pantalla

}

////////////////////Función que permite identificar las pulsaciones del raton para realizar las
acciones de cambio de pantalla////////////////////

void mousePressed() {

  if (page > 0 && back.isHovering) --page;

  else if (page < MAX && next.isHovering) ++page;

  redraw();

}

///Función que permite identificar si el selector del raton se encuentra dentro del boton
para ejecutar la acción al ser pulsado///

void mouseMoved() {

  back.isInside();

  next.isInside();

  redraw();

}

///Pantalla de Inicio///

void page0() {

```

```

imageMode(CENTER); //Toma como referencia el centro de la ventana para la
ubicación de imagen de fondo de la pantalla de inicio

image(imagen, width/2, height/2); //Carga la imagen en el centro de la pantalla

textSize(Button.TXTSZ); //Tamaño del texto de los botones de cambio de
pantalla

if (page > 0) back.display(); //Condición para mostrar el boton parair a la pantalla
siguiente

if (page < MAX) next.display(); //Condición para mostrar el boton de retroceder a la
pantalla anterior

myKnobA.hide(); //Oculta de la pantalla de inicio los medidores

myKnobC.hide();

myKnobB.hide();

reset=false; //Variable para activar/desactivar el proceso de medición o
toma de datos

BPM=0;

RPM=0;

GSR=0;

}

///Pantalla 2 : Toma de Mediciones///  

void page1() {

if (reset==false) {

background(#00353b);

imageMode(CENTER);

```

```

image(imagen2, width/2, height/2); //drawing meter image

textAlign(CENTER);

String s = "Conectar el Sensor ECG al circuito principal, ubique adecuadamente los
electrodos en el paciente y ejecute el boton iniciar.";

fill(255);

textSize(12);

text(s, 1050, 30, 200, 100); // Text wraps within text box

s = "Desconectar el Sensor ECG, conectar el sensor de frecuencia respiratoria, ubique
adecuadamente en el paciente y ejecute iniciar.";

fill(255);

textSize(12);

text(s, 1050, 230, 200, 100); // Text wraps within text box

s = "Ubique adecuadamente el sensor GSR y ejecute iniciar";

fill(255);

textSize(12);

text(s, 1050, 480, 200, 100); // Text wraps within text box

textAlign(CENTER, CENTER);

newData =false;

reset=true;

}

//Gráfica boton de inicio de medición de frecuencia cardíaca ///

fill(255);          //Relleno del boton 255: color blanco

stroke(255);        //Color del borde 255: color blanco

rect(x, y, w, h);   //grafica un rectangulo en la posición x,y de tamaño w,h

```

```

fill(#00a9c2);          //Color del texto

textSize(12);          //Tamaño del texto

text("INICIAR", x+60, y+h-14);          //Texto y ubicación del texto

// verificar si se dio click en el boton

if (mouseX > x && mouseX < x+w && mouseY > y && mouseY < y+h &&
mousePressed==true) {

    delay(100);

    int input = JOptionPane.showConfirmDialog(null, "Se encuentra el Sensor ECG
conectado?", "Aviso", JOptionPane.YES_NO_OPTION);

    if (input==0) {

        arduino.write('2');

        sensor=2;

        delay(1000);

    }

    else {

        mousePressed=false;

        JOptionPane.showMessageDialog(null, "Solicitud cancelada");

        arduino.write('4');

        noStroke();

        fill(#005c8d);

        rect(850, 630, 180, 50);

        String s = "PROCESO CANCELADO";

        fill(255);

        textSize(12);

```

```

    text(s, 850, 600, 200, 100); // Text wraps within text box
}
}

//Grafica boton de inicio de medición de frecuencia respiratoria //
fill(255);

stroke(255);

rect(x2, y2, w2, h2);

fill(#00a9c2);

textSize(12);

text("INICIAR", x2+60, y2+h2-14);

// verificar si se dio click en el boton

if (mouseX > x2 && mouseX < x2+w2 && mouseY > y2 && mouseY < y2+h2 &&
mousePressed==true) {

    delay(100);

    int input = JOptionPane.showConfirmDialog(null, "Se encuentra el Sensor de
frecuencia respiratoria conectado?", "Aviso", JOptionPane.YES_NO_OPTION);

    if (input==0) {

        arduino.write('1');

        sensor=1;

        delay(1000);

    } else {

        mousePressed=false;

        JOptionPane.showMessageDialog(null, "Solicitud cancelada");

        arduino.write('4');

```

```

noStroke();

fill(#005c8d);

rect(850, 630, 180, 50);

String s = "PROCESO CANCELADO";

fill(255);

textSize(12);

text(s, 850, 600, 200, 100); // Text wraps within text box
}

}

//Gráfica boton de inicio de medición de conductividad de la piel GSR //

fill(255);

stroke(255);

rect(x4, y4, w4, h4);

// 2. Dibujar el texto del boton color negro

fill(#00a9c2);

textSize(12);

text("INICIAR", x4+60, y4+h4-14);

// verificar si se dio click en el boton

if (mouseX > x4 && mouseX < x4+w4 && mouseY > y4 && mouseY < y4+h4 &&
mousePressed==true) {

    // si hubo click en el botón borrar la pantalla con color blanco

    int input = JOptionPane.showConfirmDialog(null, "Se encuentra el Sensor GSR
conectado?", "Aviso", JOptionPane.YES_NO_OPTION);

    if (input==0) {

```



```

    arduino.write('3');

    sensor=3;

    delay(1000);

    mousePressed=false;

}

else {

    mousePressed=false;

    JOptionPane.showMessageDialog(null, "Solicitud cancelada");

    arduino.write('4');

    noStroke();

    fill(#005c8d);

    rect(850, 630, 180, 50);

    String s = "PROCESO CANCELADO";

    fill(255);

    textSize(12);

    text(s, 850, 600, 200, 100); // Text wraps within text box

}

}

/// Si se inicio la medición del sensor de frecuencia cardíaca y se reciben datos se activa
la graficación/////

if (newData && sensor==1) {

    frecuencia = int(valores[1]);           //Almacena e valor recibido de la
frecuencia

    if (frecuencia==0) {

```

```

noStroke();

fill(#005c8d);

rect(850, 630, 180, 50);

String s = "PROCESANDO RPM..";

fill(255);

textSize(12);

text(s, 850, 600, 200, 100); // Text wraps within text box

} else {

noStroke();

fill(#005c8d);

rect(850, 630, 180, 50);

String s = "RPM OBTENIDO";

fill(255);

textSize(12);

text(s, 850, 600, 200, 100); // Text wraps within text box

RPM=frecuencia;

}

myKnobB.setValue( frecuencia ); //Muestra la frecuencia total
obtenida en el medidor digital

signal = map(valores[0], 0, 1023, 400, 200); //Mapea los datos recibidos
para una adecuada graficación

stroke(#131E36); //color de la linea de graficación "azul"

strokeWeight(1); //Espesor de la linea de graficación

line(lastxPos, lastheight, xPos, signal); // graficación de la señal

```

```

lastxPos= xPos;

lastheight= int(signal);

if (xPos >= 750) {

    xPos = 350;

    lastxPos= 350;

    reset = false;

} else {

    xPos++;

}

newData =false;

}

// Si se inicio la medición del sensor de frecuencia respiratoria y se reciben datos se
activa la graficación//

if (newData && sensor==2) {

    signal = map(valores[0], 0, 700, 50, 200); //map to the screen height.

    frecuencia = int(valores[1]);

    myKnobA.setValue( frecuencia );

    if (frecuencia==0) {

        noStroke();

        fill(#005c8d);

        rect(850, 630, 180, 50);

        String s = "PROCESANDO BPM..";

        fill(255);

        textSize(12);

```

```

text(s, 850, 600, 200, 100); // Text wraps within text box

} else {

noStroke();

fill(#005c8d);

rect(850, 630, 180, 50);

String s = "BPM OBTENIDO";

fill(255);

textSize(12);

text(s, 850, 600, 200, 100); // Text wraps within text box

BPM=frecuencia;

}

//Drawing a line from Last inByte to the new one.

stroke(#131E36); //stroke color

strokeWeight(1); //stroke wider

line(lastxPos2, lastheight2, xPos2, signal);

lastxPos2= xPos2;

lastheight2= int(signal);

if (xPos2 >= 750) {

xPos2 = 350;

lastxPos2= 350;

reset = false;

} else {

xPos2++;

```

```
}  
  
newData =false;  
  
}
```

/// Si se inicio la medición del sensor GSR y se reciben datos se activa la graficación///

```
if (newData && sensor==3) {  
  
    signal = int(valores[0]);  
  
    frecuencia = int(valores[1]);  
  
    if (frecuencia==29) {  
  
        noStroke();  
  
        fill(#005c8d);  
  
        rect(850, 630, 180, 50);  
  
        String s = "GSR OBTENIDO";  
  
        fill(255);  
  
        textSize(12);  
  
        text(s, 850, 600, 200, 100); // Text wraps within text box  
  
        GSR=(signal);  
  
        frecuencia=100;  
  
        newData =false;  
  
    } else {  
  
        noStroke();  
  
        fill(#005c8d);  
  
        rect(850, 630, 180, 50);  
  
        String s = "PROCESANDO GSR..";  
  
        fill(255);
```

```

    textSize(12);

    text(s, 850, 600, 200, 100); // Text wraps within text box

    myKnobC.setValue( signal );

    stroke(#1A99FF);

    strokeWeight(1);

}

}

//Botones de cambio de pantalla//

textSize(Button.TXTSZ);

if (page > 0)  back.display();

if (page < MAX)  next.display();

//Activación de medidoresdigitales //

myKnobA.show();

myKnobC.show();

myKnobB.show();

myKnobB.hideTickMarks();

obtenerResultado=true;

}

//Se programa muestra de resultados en la tercera pantalla//

void page2() {

    if(obtenerResultado){

        myKnobA.hide();

        myKnobC.hide();

        myKnobB.hide();

```

```

obtenerResultado=false;

reset=false;

if(GSR==0 || BPM ==0 || RPM==0){

int input = JOptionPane.showConfirmDialog(null,"Realice todas las mediciones para
obtener el pronóstico del nivel de estrés", "Error!", JOptionPane.DEFAULT_OPTION);

    if(input == 0){
page=1;
} }

else{

arduino.write('5');

}}

if(estado2==1){

//Gráfica boton de inicio de medición de frecuencia cardíaca //

fill(255);          //Relleno del boton 255: color blanco

stroke(255);        //Color del borde 255: color blanco

rect(x5, y5, w5, h5);    //grafica un rectangulo en la posición x,y de tamaño w,h

fill(#00a9c2);        //Color del texto

textSize(12);        //Tamaño del texto

text("SALIR", x5+60, y5+h-14);    //Texto y ubicación del texto

// verificar si se dio click en el boton

if (mouseX > x5 && mouseX < x5+w5 && mouseY > y5 && mouseY < y5+h5 &&
mousePressed==true) {

page=0;

} }

```

```

else{

play = true;

///Grafica boton de inicio de medición de frecuencia cardíaca ///

fill(255);           ///Relleno del boton 255: color blanco

stroke(255);         ///Color del borde 255: color blanco

rect(x5, y5, w5, h5);    ///grafica un rectangulo en la posición x,y de tamaño w,h

fill(#00a9c2);        ///Color del texto

textSize(12);         ///Tamaño del texto

text("BIOFEEDBACK", x5+60, y5+h-14);    ///Texto y ubicación del texto

// verificar si se dio click en el boton

if (mouseX > x5 && mouseX < x5+w5 && mouseY > y5 && mouseY < y5+h5 &&
mousePressed==true) {

page=3;

} }

if (newData) {

estado = int(valores[0]);

confiabilidad = int(valores[1]);

println(estado);

println(confiabilidad);

if(estado==1){

estado2=1;

imageMode(CENTER);

image(imagen4, width/2, height/2); ///drawing meter image

TableRow newRow = table.addRow();

```



```

newRow.setInt("id", table.getRowCount() - 1);

newRow.setInt("RPM", RPM);

newRow.setInt("BPM", BPM);

newRow.setInt("GSR", int(GSR));

newRow.setString("ESTADO", "NORMAL" );

saveTable(table, "data/datos.csv");

} if(estado==2){

estado2=2;

        imageMode(CENTER);

image(imagen5, width/2, height/2); //drawing meter image

TableRow newRow = table.addRow();

newRow.setInt("id", table.getRowCount() - 1);

newRow.setInt("RPM", RPM);

newRow.setInt("BPM", BPM);

newRow.setInt("GSR", int(GSR));

newRow.setString("ESTADO", "ALGO ELEVADO" );

saveTable(table, "data/datos.csv");

} if(estado==3){

estado2=2;

        imageMode(CENTER);

image(imagen6, width/2, height/2); //drawing meter image

TableRow newRow = table.addRow();

newRow.setInt("id", table.getRowCount() - 1);

newRow.setInt("RPM", RPM);

```

```

newRow.setInt("BPM", BPM);

newRow.setInt("GSR", int(GSR));

newRow.setString("ESTADO", "ALTO" );

saveTable(table, "data/datos.csv");

} newData =false;

} }

void page3() {

if(play){

imageMode(CENTER); //Toma como referencia el centro de la ventana para la
ubicación de imagen de fondo de la pantalla de inicio

image(imagen7, width/2, height/2); //Carga la imagen en el centro de la pantalla

int input = JOptionPane.showConfirmDialog(null, "En el siguiente ejercicio se procede
a inhalar (tomar aire) cuando la pelota baje en dirección de las manecillas del reloj y a
exhalar (soltar aire) cuando la pelota suba.", "Biofeedback",

JOptionPane.DEFAULT_OPTION);

if(input == 0){

myMovie = new Movie(this, "video.mp4");

myMovie.play();

}play=false;

} image(myMovie, width/2, height/2, 640,400);

/////Gráfica boton de inicio de medición de frecuencia cardíaca ////

fill(255); //Relleno del boton 255: color blanco

stroke(255); //Color del borde 255: color blanco

rect(x6, y6, w6, h6); //grafica un rectangulo en la posición x,y de tamaño w,h

```

```

fill(#00a9c2);          //Color del texto

textSize(12);          //Tamaño del texto

text("IR A MEDICIONES", x6+60, y6+h6-14);          //Texto y ubicación del texto

// verificar si se dio click en el boton

if (mouseX > x6 && mouseX < x6+w6 && mouseY > y6 && mouseY < y6+h6 &&
mousePressed==true) {

page=1;

}

////////Gráfica boton de inicio de medición de frecuencia cardíaca //////

fill(255);          //Relleno del boton 255: color blanco

stroke(255);          //Color del borde 255: color blanco

rect(x7, y7, w7, h7);          //grafica un rectangulo en la posición x,y de tamaño w,h

fill(#00a9c2);          //Color del texto

textSize(12);          //Tamaño del texto

text("SALIR", x7+60, y7+h7-14);          //Texto y ubicación del texto

// verificar si se dio click en el boton

if (mouseX > x7 && mouseX < x7+w7 && mouseY > y7 && mouseY < y7+h7 &&
mousePressed==true) {

    JOptionPane.showMessageDialog(null,

    "Gracias por usar BIOSTRESS");

    page=0;

} }

////////Función para cambio de pantalla////////

void pageSelector() { // R

```

```

switch(page) {
case 0:
    page0();
    break;
case 1:
    page1();
    break;
case 2:
    page2();
    break;
case 3:
    page3();
    break;
}}

```

///*Función botones de cambio de pantalla*///

```

class Button {
    static final int W = 160, H = 40, TXTSZ = 020;
    static final color BTNC = #00A0A0, HOVC = #00FFFF, TXTC = 0;
    final String label;
    final int x, y, xW, yH;
    boolean isHovering;
    Button(String txt, int xx, int yy) {
        label = txt;
        x = xx;

```

```

    y = yy;

    xW = xx + W;

    yH = yy + H;

}

void display() {

    fill(isHovering? HOVC : BTNC);

    rect(x, y, W, H);

    fill(TXTC);

    text(label, x + W/2, y + H/2);

}

boolean isInside() {

    return isHovering = mouseX > x & mouseX < xW & mouseY > y & mouseY < yH;

} }

//////////Interrupción Seria cuando recibe datos//////////

void serialEvent (Serial arduino) {

    String dato = arduino.readStringUntil('\n'); // leemos el buffer del puerto serie

    if (dato != null) {

        dato = trim(dato);

        valores = int(split(dato, ','));

        newData = true;

    } }

void movieEvent(Movie m) {

    m.read();

}

```

ANEXO 3. Tabla del conjunto de datos de entrenamiento.

BPM	RPM	GSR	Clase	Estado
50	14	440	1	Normal
54	13	425	1	Normal
63	17	364	1	Normal
55	18	355	1	Normal
55	17	430	1	Normal
64	18	420	1	Normal
50	16	440	1	Normal
58	14	375	1	Normal
65	20	435	1	Normal
60	19	400	1	Normal
72	24	310	2	Algo Elevado
78	27	350	2	Algo Elevado
78	22	340	2	Algo Elevado
69	25	310	2	Algo Elevado
69	26	335	2	Algo Elevado
68	22	315	2	Algo Elevado
72	26	305	2	Algo Elevado
71	25	320	2	Algo Elevado
73	25	307	2	Algo Elevado
69	23	339	2	Algo Elevado
82	30	299	3	Alto
85	29	280	3	Alto
81	33	240	3	Alto
87	34	285	3	Alto
86	38	297	3	Alto
80	29	270	3	Alto
80	31	290	3	Alto
85	35	295	3	Alto
90	32	288	3	Alto
79	28	295	3	Alto

ANEXO 4. Fotografías del trabajo.

En la figura 18 se muestra el desarrollo de la encuesta online realizada para la recolección de información y en la figura 19 se muestra las respuestas de dicha encuesta.



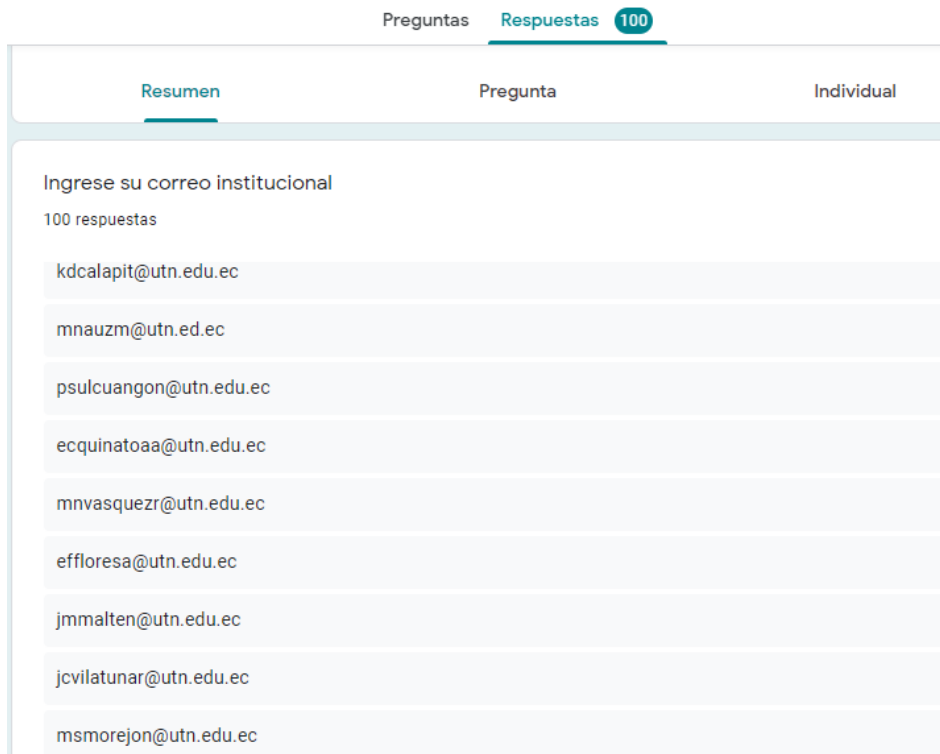
The screenshot shows the BIOSTRESS online survey interface. At the top, there is a navigation bar with 'Preguntas' and 'Respuestas 100'. The main header features the text 'BIOSTRESS SISTEMA BIOFEEDBACK DE DETECCIÓN DE ESTRÉS'. Below this, there is a form with the following elements:

- A text input field labeled 'BIOSTRESS' with the instruction: 'Lea detenidamente cada pregunta planteada y responda con lo más adecuado.'
- A text input field labeled 'Ingrese su correo institucional' with a red asterisk indicating a required field.
- A text input field labeled 'Texto de respuesta corta'.

On the right side of the form, there is a vertical toolbar with icons for adding, deleting, and other actions.

Figura. 18. Desarrollo de la Encuesta online.

Fuente: Autoría.



The screenshot shows the BIOSTRESS online survey results page. At the top, there is a navigation bar with 'Preguntas' and 'Respuestas 100'. Below this, there is a table with the following columns: 'Resumen', 'Pregunta', and 'Individual'. The table contains the following data:

Resumen	Pregunta	Individual
Ingrese su correo institucional		
100 respuestas		
kdcalapit@utn.edu.ec		
mnauzm@utn.ed.ec		
psulcuangon@utn.edu.ec		
ecquinatoaa@utn.edu.ec		
mvasquezr@utn.edu.ec		
effloresa@utn.edu.ec		
jmmalten@utn.edu.ec		
jcvilatunar@utn.edu.ec		
msmorejon@utn.edu.ec		

Figura. 19. Respuesta de la Encuesta.

Fuente: Autoría.

Implementación de BIOSTRESS

La figura 20 muestra el proceso de implementación del *hardware* del sistema BIOSTRESS.

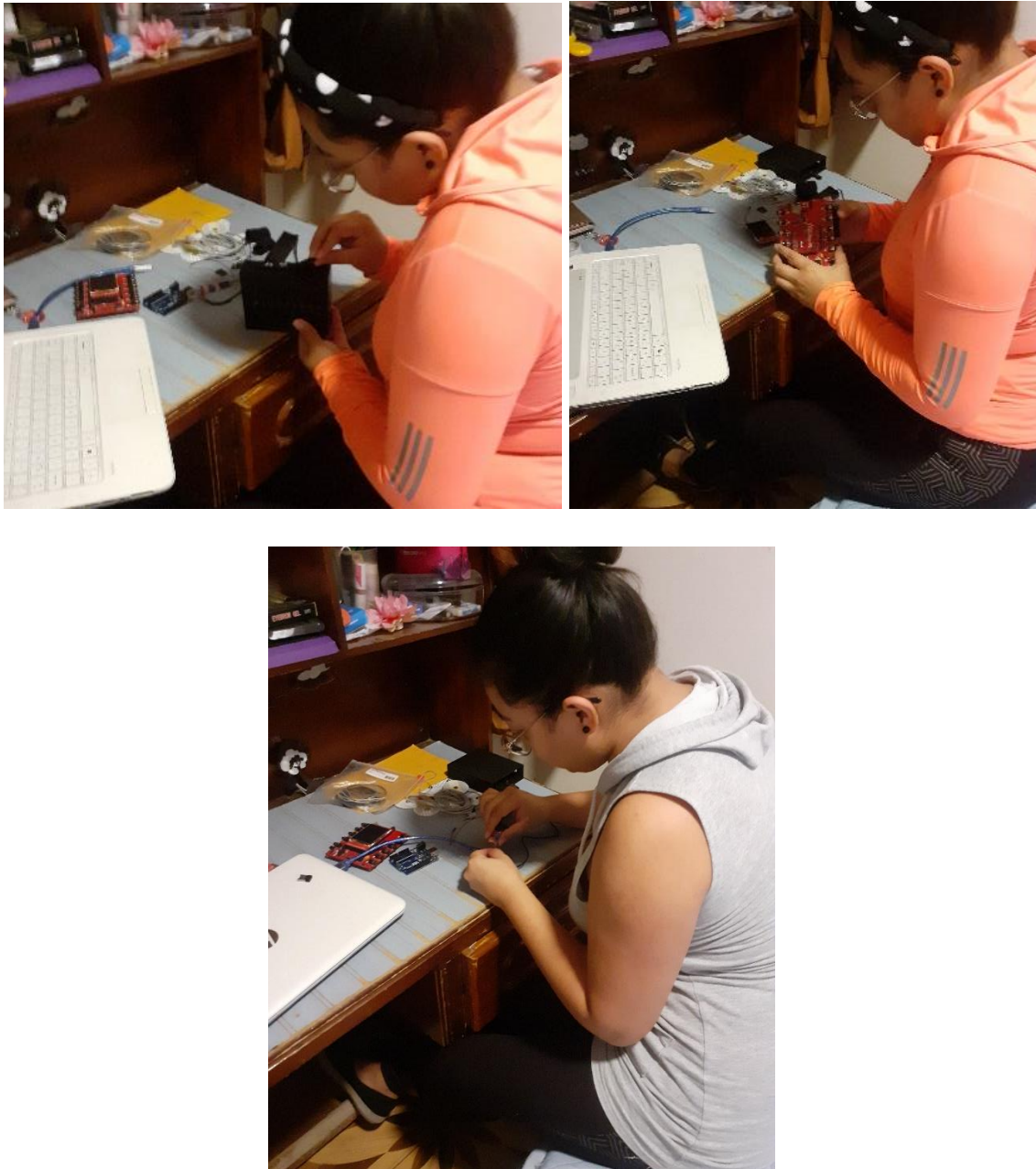
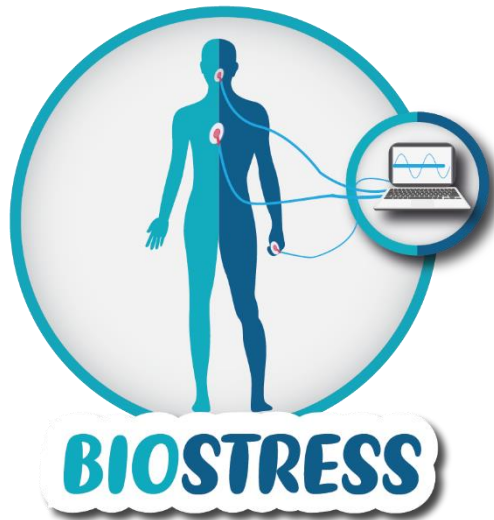


Figura. 20. Construcción de BIOSTRESS.

Fuente: Autoría.



MANUAL DE USUARIO DE BIOSTRESS



BIOSTRESS es un sistema de detección de niveles de estrés mediante algoritmos de aprendizajes de máquinas aplicado a estudiantes universitarios.

El nombre del sistema viene dado por la unión de dos palabras, BIO que hace referencia a Biofeedback y STRESS que es el campo de estudio en inglés. BIOSTRESS cuenta con la placa MySignals y tres de sus sensores que son imprescindibles para el funcionamiento de cada test. Este sistema brinda información tanto al administrador como al usuario sobre el nivel de estrés que poseen los pacientes siendo normal, algo elevado y alto, y, gracias al feedback incorporado regula las reacciones fisiológicas provocadas por el estrés. Con esto se ayuda a los especialistas a brindar un diagnóstico más preciso a los pacientes.

Cómo realizar un test

El sistema BIOSTRESS puede estar a cargo de una persona que realizará la prueba, en este caso el administrador un especialista en el área de estrés (psicólogo). BIOSTRESS cuenta con sensores que van conectados al paciente y el sistema que va conectado al ordenador del administrador, como se muestra a continuación:



Los sensores que se observan van ubicados en el paciente de manera adecuada para la correcta lectura de los datos. BIOSTRESS está destinado a estudiantes universitarios. El ejemplo se realizará con los siguientes pasos:

1. Se conecta el sistema al ordenador mediante el cable USB de Arduino como se puede observar en la ilustración 1.

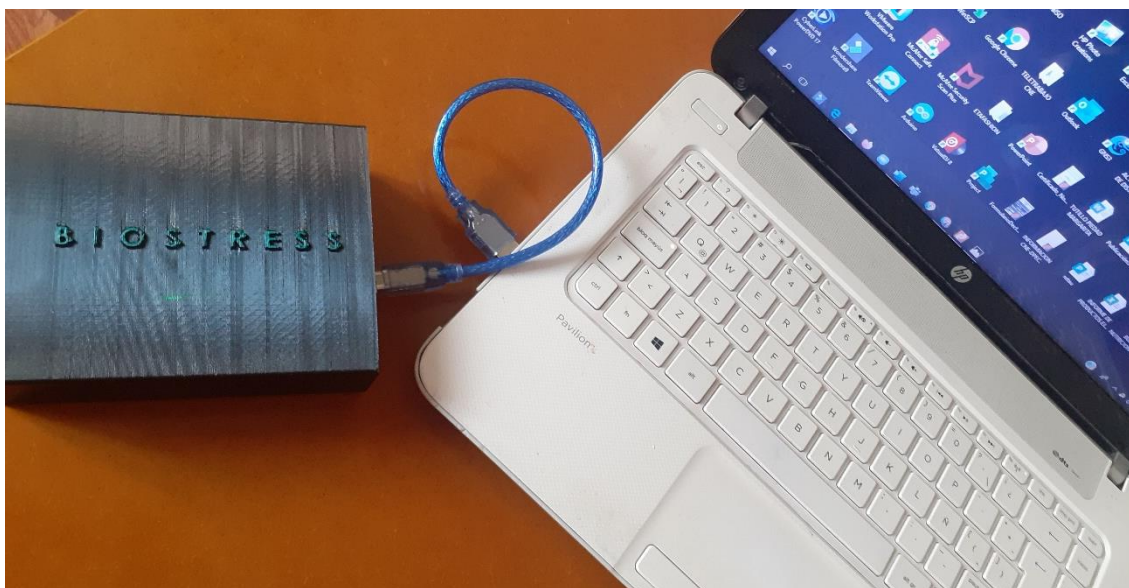


Ilustración 1. Conexión de BIOSTRESS.

- Se debe comprobar que se ha conectado adecuadamente al puerto COM8 del ordenador, para ello vaya a búsqueda en Windows e ingrese Administrador de Dispositivos.

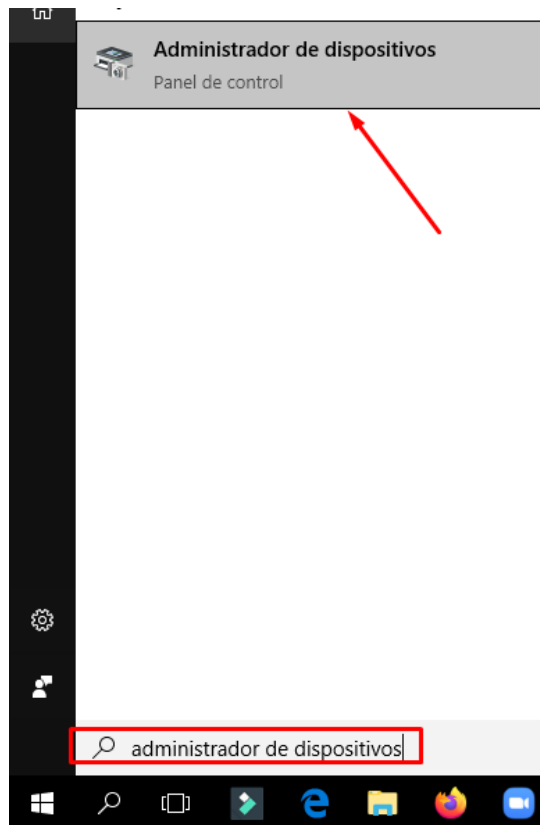


Ilustración 2. Administrador de dispositivos.

- Una vez seleccionado Administrador de dispositivos se despliega la opción de Puertos (COM y LPT), y se verifica que reconozca el puerto COM8 como se observa en la ilustración 3.

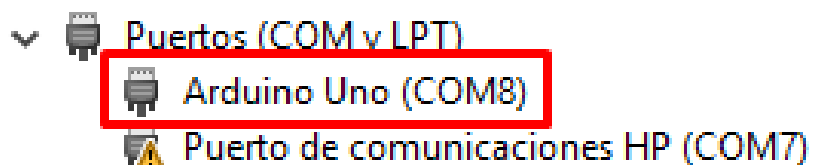


Ilustración 3. Puertos COM y LPT

2. Cuando ya se encuentre correctamente conectado se procede a realizar doble clic al icono de la aplicación y se abrirá la interfaz como se muestra en la ilustración 4.

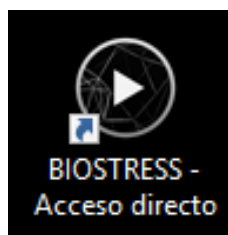




Ilustración 4. Interfaz de BIOSTRESS.

3. Se presiona en la opción COMENZAR y se observa la interfaz para medir los datos del paciente, esto se muestra en la ilustración 5.

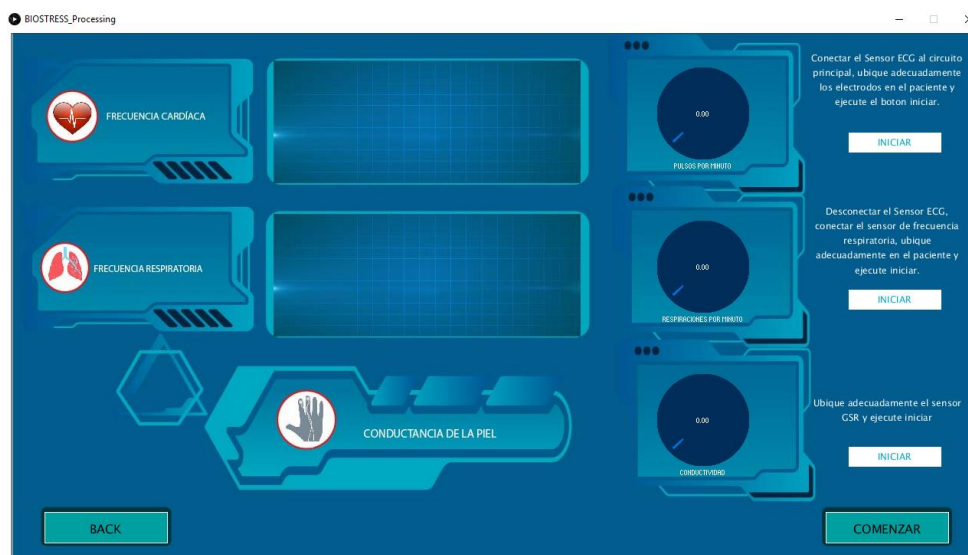


Ilustración 5. Interfaz de toma de datos.

4. Ubique los sensores correctamente al paciente, como se observa en la ilustración 6.



ECG

Airflow

GSR

Ilustración 6. Ubicación de los sensores.

5. En la aplicación, comience con la toma de datos para ello presione la opción de inicio del sensor ECG de frecuencia cardíaca como se observa en la ilustración 7.



Ilustración 7. Inicio del sensor ECG.

6. Se procede a ubicar el cable plug del sensor ECG al sistema como se observa en la ilustración 8.



Ilustración 8. Conexión del sensor ECG.

7. Una vez que se encuentre conectado el sensor, en la aplicación se muestra un mensaje al cual se presiona si y se espera 1 minuto hasta que el sensor tome los datos necesarios de la frecuencia cardíaca del paciente.

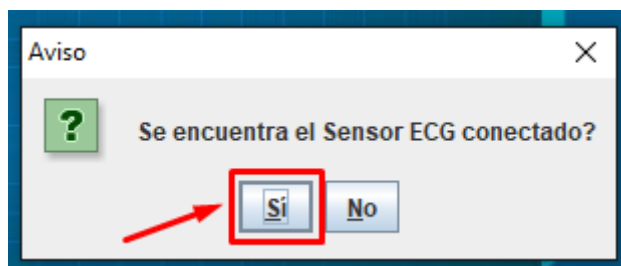


Ilustración 9. Conectar el sensor ECG.

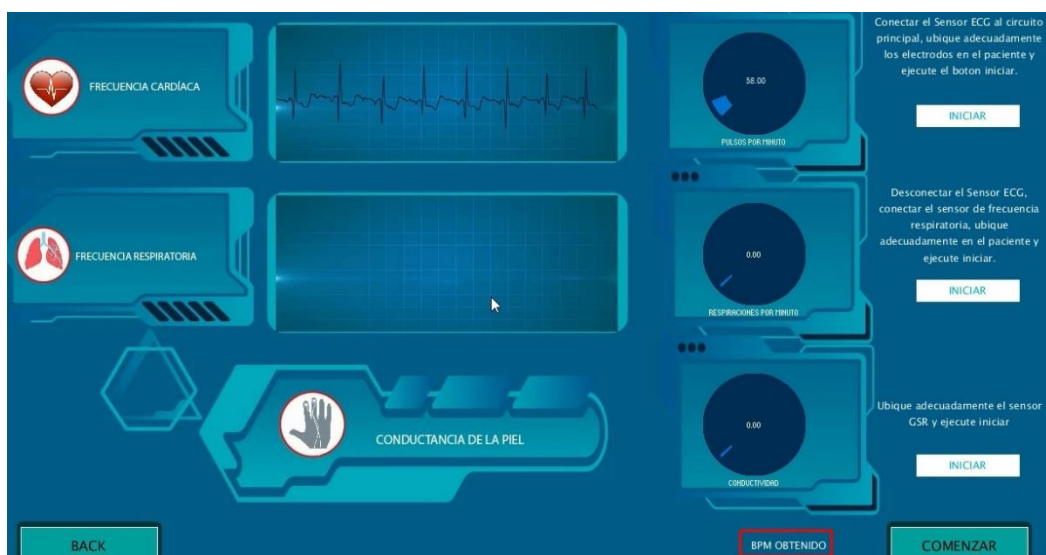


Ilustración 10. BPM obtenidos.

8. Se prosigue con el sensor *Airflow* de frecuencia respiratoria, para ello se desconecta el cable plug del sensor ECG y se conecta el cable plug del sensor *Airflow* en el lugar adecuado como se observa en la ilustración 11.



Ilustración 11. Conexión del sensor Airflow.

9. En la aplicación se presiona INICIAR y en el mensaje de aviso SI para proceder a tomar los datos de la frecuencia respiratoria, como se observa en la ilustración 12.

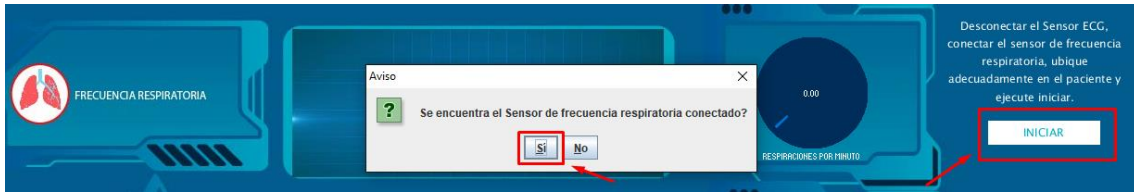


Ilustración 12. Iniciar el sensor Airflow.

10. De igual manera se espera 1 minuto hasta obtener los valores del sensor *Airflow* como se muestra en la ilustración 13.

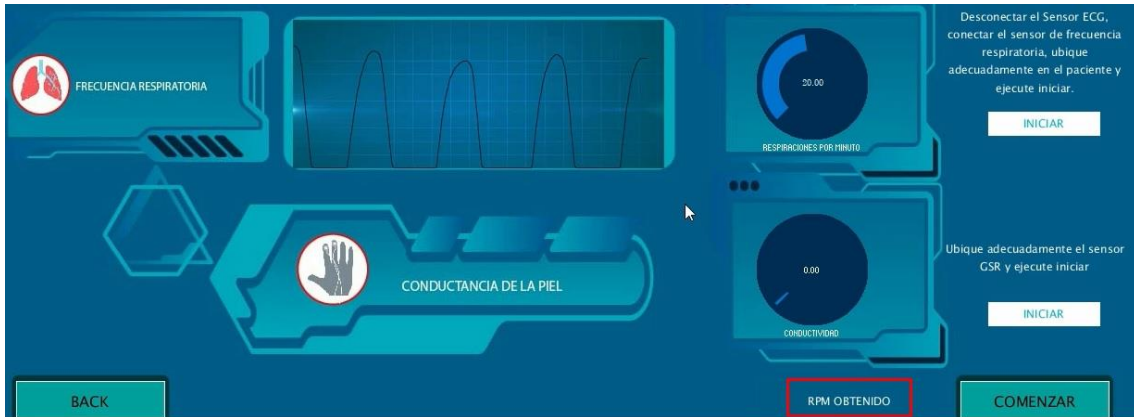


Ilustración 13. RPM obtenidos.

11. A continuación, se debe iniciar con la lectura de datos del sensor GSR prosiguiendo los mismos pasos antes mencionados, esto se evidencia en la ilustración 14.



Ilustración 14. GSR obtenido.

12. Cuando los datos del paciente son obtenidos correctamente se procede a dar clic en la opción comenzar para iniciar el resultado, como se muestra en la ilustración 15.

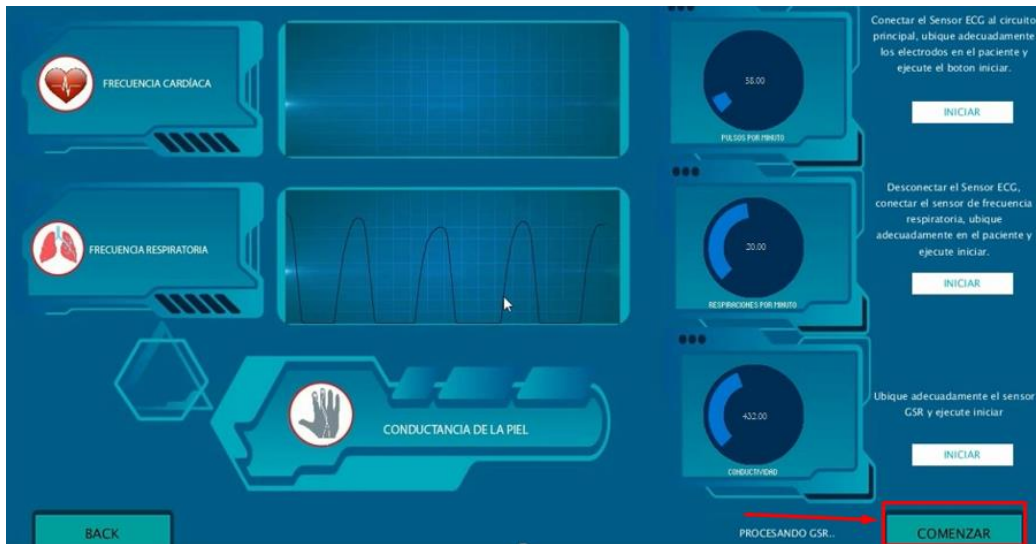


Ilustración 15. Comenzar resultado.

13. Se observa el resultado del nivel de estrés del paciente, esto se evidencia en la ilustración 16.



Ilustración 16. Resultados del test.

14. Si el resultado del test es normal hasta ahí llega la prueba caso contrario se procede a realizar el Biofeedback para los niveles algo elevado y alto. Para ello se presiona la opción BIOFEEDBACK y se observa las indicaciones del ejercicio a realizar, esto se observa en las ilustraciones 17 y 18.



Ilustración 17. Iniciar Biofeedback.

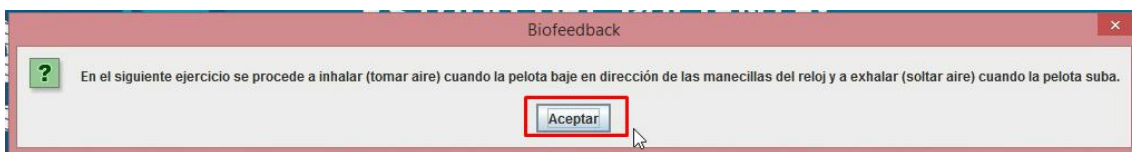


Ilustración 18. Mensaje del ejercicio del Biofeedback.

15. El paciente procede a realizar el Biofeedback y posteriormente se vuelve a tomar los valores para comprobar la normalización de las respuestas fisiológicas. En la ilustración 19 se observa el ejercicio a realizar.

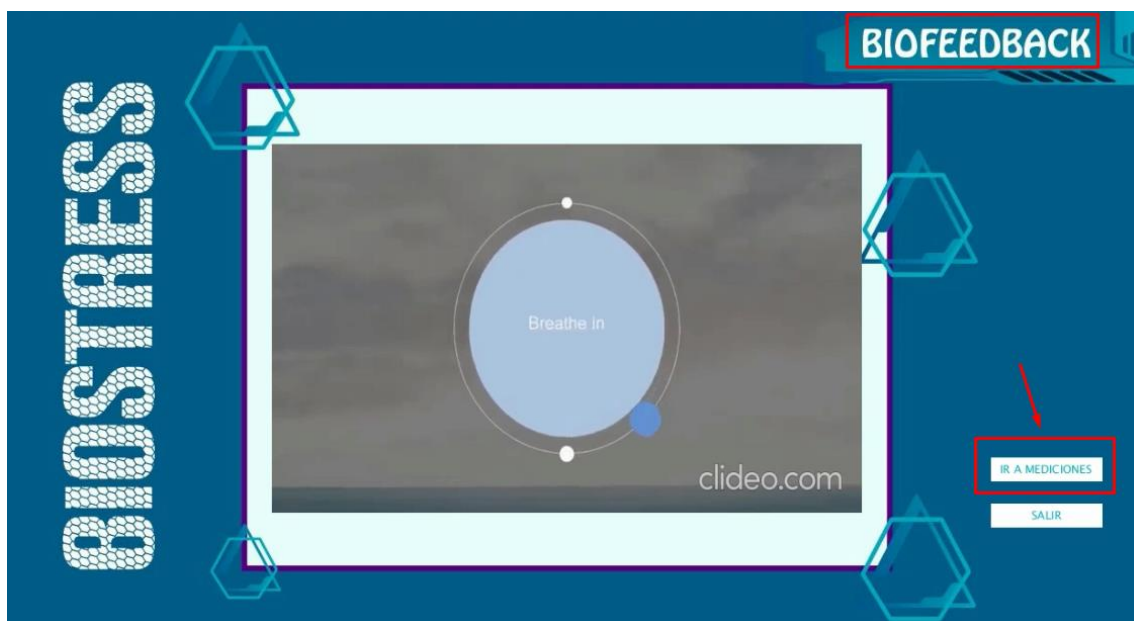


Ilustración 19. Biofeedback.

16. Los datos guardados se pueden observar en el archivo datos.csv, este archivo se encuentra en la carpeta data de Processing como se observa en la ilustración 20.

o > Nuevo vol (D:) > Samy > Trabajo de grado II > BIOSTRESS FINAL > BIOSTRESS_Processing > data

Nombre	Fecha	Tipo	Tamaño
1	10/12/2020 15:21	Archivo JPG	112 KB
4	12/3/2021 9:48	Archivo JPG	85 KB
ALTO	10/3/2021 19:04	Archivo JPG	131 KB
ARADGothicJP-Me...	17/3/2021 14:11	Archivo VLW	74 KB
Century-48.vlw	24/11/2020 19:05	Archivo VLW	230 KB
CenturyGothic-Bol...	24/11/2020 19:11	Archivo VLW	225 KB
datos	17/3/2021 12:46	Archivo de valores separados por comas de Microsoft Excel	3 KB
ELEVADO	10/3/2021 16:48	Archivo JPG	136 KB
Interfaz2.0	7/1/2021 21:19	Archivo JPG	72 KB
NORMAL	10/3/2021 16:21	Archivo JPG	114 KB
Placa1-01	7/1/2021 21:15	Archivo PNG	1.288 KB
Placa2-01-01	7/1/2021 21:15	Archivo PNG	1.288 KB
sensor1	7/1/2021 21:18	Archivo PNG	63 KB
sensor2	7/1/2021 21:18	Archivo PNG	65 KB
video	11/3/2021 8:56	Archivo JPG	90 KB
video	10/3/2021 19:50	Archivo MP4	8.167 KB

Ilustración 20. Archivo csv.

17. Se presiona doble clic al archivo y se hace clic en DATOS en la barra de tareas y texto en columnas para ver de una manera más clara la información guardada. Esto se evidencia en la ilustración 21.

Archivo Inicio Insertar Disposición de página Fórmulas **Datos**

Desde el texto/CSV Desde la web From Table/Range Fuentes recientes Conexiones existentes Obtener y transformar datos Actualizar todo Consultas y conexiones Propiedades Editar vínculos

POSIBLE PÉRDIDA DE DATOS Algunas características del libro se pueden perder si se abre un archivo de Excel.

	A	B	C	D	E	F
101	99	0	0	350	ALGO ELEVADO	
102	100	21	49	416	NORMAL	
103	101	0	0	123	ALTO	
104	102	29	202	361	ALGO ELEVADO	
105	103	7	150	427	NORMAL	
106	104	32	222	450	NORMAL	
107	105	27	226	395	NORMAL	
108	106	11	212	280	ALTO	
109	107	27	220	398	NORMAL	
110	108	23	4	326	ALGO ELEVADO	
111	109	34	40	389	NORMAL	

Ilustración 21. Datos guardados.

18. Para salir de la aplicación se presiona la opción SALIR, se desconecta del ordenador y los sensores del paciente.

Instalación

BIOSTREES puede ser instalado en dos versiones del sistema operativo de Windows que son de 32 y 64 bits, estas versiones se observan en la ilustración 22.

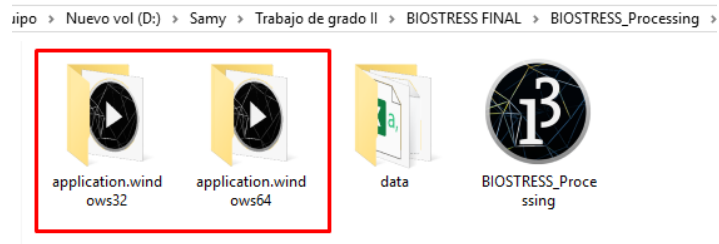


Ilustración 22. Versiones de BIOSTRESS.

En este caso se va a instalar el sistema de 64 bits, para ello, en el ordenador del administrador se debe crear una carpeta denominada BIOSTRESS en la ruta que se prefiera y dentro de esta carpeta se copia la carpeta “application.windows64”.

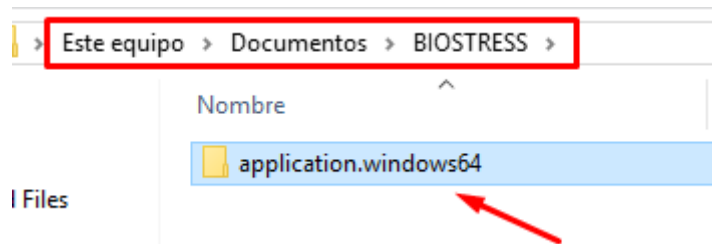


Ilustración 23. Carpeta de la versión de 64 bits.

Se procede a abrir la carpeta dando doble clic y se crea el acceso directo en escritorio de la aplicación para ello se da clic derecho a la aplicación y se elige la opción Enviar a y se selecciona Escritorio (crear acceso directo), esto se indica en la ilustración 24.

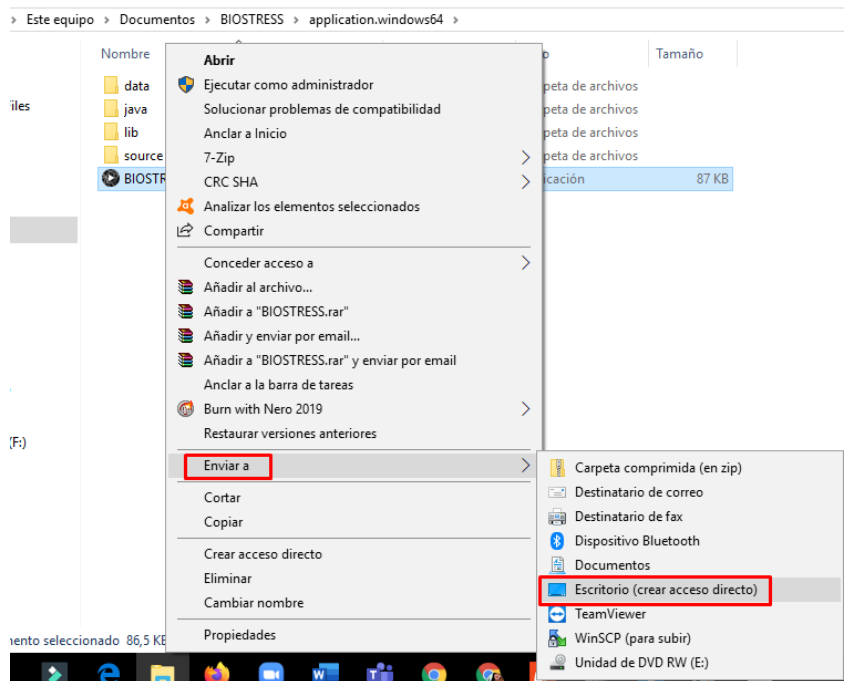


Ilustración 24. Creación de acceso directo de BIOSTRESS.

Se visualiza el icono de la aplicación en Escritorio y se procede a ejecutar el programa BIOSTRESS.



Ilustración 25. Icono de BIOSTRESS en el escritorio.