



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE TELECOMUNICACIONES

**INFORME FINAL DEL TRABAJO DE INTEGRACIÓN CURRICULAR,
PROYECTO DE INVESTIGACIÓN**

TEMA:

**“DISPOSITIVO ELECTRÓNICO COMO APOYO A LA PREVENCIÓN
DEL STC (SÍNDROME DEL TUNEL CARPIANO) EN ENTORNOS
LABORALES USANDO INTELIGENCIA ARTIFICIAL”**

Trabajo de titulación previo a la obtención del título de Ingeniero en Telecomunicaciones

Línea de investigación: Desarrollo, aplicación de software y cibersecurity (seguridad cibernética)

AUTOR:

Samantha Mishell Pérez Cuatin

DIRECTOR:

Ing. Suárez Zambrano Luis Edilberto, Msc

Ibarra – Ecuador 2025

UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA
IDENTIFICACIÓN DE LA OBRA

La Universidad Técnica del Norte dentro del proyecto Repositorio Digital Institucional, determinó la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la Universidad. Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	1050449261		
APELLIDOS Y NOMBRES:	SAMANTHA MISHHELL PÉREZ CUATIN		
DIRECCIÓN:	IBARRA, AV. ATAHUALPA 49-19		
EMAIL:	smperezo@utn.edu.ec /samthamishell@gmail.com		
TELÉFONO FIJO:	2652-127	TELF. MOVIL	0994085934

DATOS DE LA OBRA	
TÍTULO:	DISPOSITIVO ELECTRÓNICO COMO APOYO A LA PREVENCIÓN DEL STC (SÍNDROME DEL TÚNEL CARPIANO) EN ENTORNOS LABORABLES USANDO INTELIGENCIA ARTIFICIAL
AUTOR (ES):	SAMANTHA MISHHELL PÉREZ CUATIN
FECHA:	2025/7/16
SOLO PARA TRABAJOS DE INTEGRACIÓN CURRICULAR	
CARRERA/PROGRAMA:	<input checked="" type="checkbox"/> GRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	INGENIERO EN TELECOMUNICACIONES
DIRECTOR:	MSC. SUÁREZ ZAMBRANO LUIS EDILBERTO
ASESOR:	MSC. JAIME ROBERTO MICHILENA CALDERÓN

AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Samantha Mishell Pérez Cuatin, con cédula de identidad Nro. 1050449261, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de integración curricular descrito anteriormente, hago entrega del ejemplar respectivo en formato digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión; en concordancia con la Ley de Educación Superior Artículo 144.

Ibarra, a los 16 días del mes de julio de 2025

EL AUTOR:



.....
Samantha Mishell Pérez Cuatin

CONSTANCIAS

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

Ibarra, a los 16 días del mes de julio de 2025

EL AUTOR:



.....
Samantha Mishell Pérez Cuatin

CERTIFICACIÓN DEL DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Ibarra, 16 de julio de 2025

MSC. SUÁREZ ZAMBRANO LUIS EDILBERTO

DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

CERTIFICA:

Haber revisado el presente informe final del trabajo de Integración Curricular, el mismo que se ajusta a las normas vigentes de la Universidad Técnica del Norte; en consecuencia, autorizo su presentación para los fines legales pertinentes.



.....
Msc. Suárez Zambrano Luis Edilberto
C.C. 1002304291

APROBACIÓN DEL COMITÉ CALIFICADOR

El Comité Calificado del trabajo de Integración Curricular "Dispositivo electrónico como apoyo a la prevención del STC (síndrome del túnel carpiano) en entornos laborables usando inteligencia artificial" elaborado por Samantha Mishell Pérez Cuatin, previo a la obtención del título de Ingeniero en Telecomunicaciones, aprueba el presente informe de investigación en nombre de la Universidad Técnica del Norte:



.....
Ing. Suárez Zambrano Luis Edilberto, Msc
C.C.: 1002304291



.....
Ing. Jaime Roberto Michilena Calderón, Msc
C.C.: 1002198438

DEDICATORIA

A mi madre, Maribel Cuatin, mi guía, mi ejemplo y mi mayor inspiración. Desde siempre me enseñaste a creer en mí, a confiar en que podía lograr todo lo que me propusiera y a no rendirme jamás. Todo lo que soy es por ti y para ti. Espero que te sientas tan orgullosa de ser mi madre como yo lo estoy de ser tu hija. Tu amor y apoyo incondicional me dieron la fuerza para superar cada obstáculo y alcanzar esta meta. Sin tus sacrificios, dedicación y esfuerzo, nada de esto habría sido posible. Esta tesis es un regalo para ti, un símbolo de mi gratitud y admiración. Que este logro te haga sentir profundamente orgullosa de mí y de todo lo que hemos logrado juntas.

A mis hermanos, Anderson y Estefy, quienes siempre fueron mi motor. Ustedes han sido mi inspiración, la razón por la que luché con fuerza y perseveré cuando las cosas se pusieron difíciles. Gracias por estar ahí, por motivarme sin saberlo y por llenar mi vida de amor.

A Michel Negrete, por tu infinita paciencia, cariño y por creer en mí incluso en los momentos más complicados. Tal vez no estuvimos juntos desde el inicio de este camino, pero sí hasta el final. Tu amistad ha sido uno de los regalos más valiosos que la carrera me ha dejado.

A mis amigos Yuleissy Cuamacas, Santiago Puetate, Milton Garzón y Johan Cuaspa, gracias por acompañarme en los buenos y malos momentos, por hacerme reír cuando más lo necesitaba, y por demostrarme que la verdadera amistad no conoce distancias. Este logro también les pertenece, por estar presentes durante todos estos años y llenar mi vida de alegría.

Samantha Mishell Pérez Cuatin

AGRADECIMIENTO

A Dios, mi guía, refugio y fortaleza. Esta tesis es fruto de tu infinita gracia. En los momentos de cansancio, dudas o dificultad, siempre estuviste presente dándome luz, fe y esperanza. Que esta obra sea para tu gloria, y que tu nombre sea exaltado por siempre.

A toda mi familia, gracias por su amor incondicional, por sus consejos sinceros y por enseñarme el valor de la perseverancia. Gracias por impulsarme a ser una mejor persona y a nunca renunciar a mis sueños. Este logro también es de ustedes, porque han sido mi base y mi fuerza.

Manifiesto mi más profundo agradecimiento al ingeniero Luis Edilberto Suárez Zambrano, quien fue mucho más que un docente, convirtiéndose en un guía, un confidente y un verdadero ejemplo a seguir. Le estaré eternamente agradecida por creer en mí, por cada palabra de aliento, por la confianza depositada y por brindarme siempre su apoyo incondicional. Su acompañamiento durante este tiempo fue invaluable, y su ejemplo quedará para siempre como una inspiración en mi vida personal y profesional.

Al Ing. Jaime Roberto Michilena Calderón, por su constante apoyo a los estudiantes, por motivarnos a seguir adelante, por su amabilidad, su disposición para colaborarnos y por impulsarnos a crecer no solo como profesionales, sino también como seres humanos.

A la Universidad Técnica del Norte, por abrirme sus puertas y permitirme formarme en sus aulas. Gracias por contar con un cuerpo docente de excelencia y por rodearme de compañeros que marcaron positivamente mi vida académica y personal.

Este logro no es solo mío, sino el resultado del amor, la confianza y el apoyo incondicional de quienes han estado a mi lado en cada paso. A todos ustedes, gracias por creer en mí, por alentarme en los momentos difíciles y por ayudarme a convertirme en la persona que soy hoy. Desde lo más profundo de mi corazón, les dedico este triunfo. ¡Mil gracias!

Samantha Mishell Pérez Cuatin

RESUMEN

El incremento de trastornos musculoesqueléticos como el síndrome del túnel carpiano (STC), especialmente en actividades laborales que implican movimientos repetitivos de las extremidades superiores, ha impulsado la búsqueda de soluciones tecnológicas para su detección oportuna. En este estudio se presenta el desarrollo de un dispositivo electrónico portátil, diseñado para monitorear parámetros fisiológicos de forma no invasiva en trabajadores del área administrativa, con el propósito de prevenir la aparición del STC. El sistema incorpora sensores de electromiografía (EMG) y respuesta galvánica de la piel (GSR), que permiten registrar indicadores de fatiga muscular. Dichos datos son procesados por una red neuronal artificial entrenada para clasificar distintos niveles de esfuerzo físico. La arquitectura del dispositivo incluye una etapa de adquisición con Seeeduino V4.3, un módulo de procesamiento con ESP32 para comunicación inalámbrica, y una aplicación móvil enlazada con Firebase Realtime Database para la visualización remota de datos. Las pruebas iniciales realizadas con personal administrativo permitieron validar la precisión del modelo y la usabilidad del sistema. Los resultados evidencian que esta solución es técnicamente viable como herramienta preventiva en contextos laborales, y establece una base sólida para su futura integración en programas de ergonomía y salud ocupacional.

Palabras clave: Síndrome del túnel carpiano, fatiga muscular, monitoreo no invasivo, sensores EMG, red neuronal artificial, salud ocupacional.

ABSTRACT

The increasing incidence of musculoskeletal disorders, such as Carpal Tunnel Syndrome (CTS), in work environments with repetitive hand movements has driven the development of technological solutions for early symptom detection. This study presents the design and implementation of a portable electronic device for non-invasive physiological monitoring in administrative personnel, aiming to prevent the onset of CTS. The system integrates biomedical sensors, including electromyography (EMG) and galvanic skin response (GSR), to capture signals associated with muscle fatigue. These signals are processed using an artificial neural network (ANN) trained to classify physical effort levels. The system architecture features a Seeeduino V4.3 for data acquisition, an ESP32 module for wireless processing, and a mobile application connected to Firebase Realtime Database for real-time visualization. Preliminary testing with administrative workers validated both the accuracy of the model and the usability of the device. Results demonstrate the feasibility of the proposed approach as a preventive tool in occupational settings, establishing a foundation for future integration into workplace health and ergonomics systems.

Keywords: Carpal tunnel syndrome, muscle fatigue, non-invasive monitoring, EMG sensors, artificial neural network, occupational health.

LISTA DE CONTENIDOS

1.	Capítulo I: Antecedentes.....	22
1.1	Tema.....	22
1.2	Problema.....	22
1.3	Objetivos.....	25
1.3.1.	Objetivo General.....	25
1.3.2.	Objetivos Específicos.....	25
1.4	Alcance.....	25
1.5	Justificación.....	28
2.	Capítulo II: Estado del Arte.....	31
2.1	Síndrome del Túnel del Carpo.....	31
2.1.1.	Epidemiología.....	32
2.1.2.	Fisiopatología.....	32
2.1.3.	Manifestaciones Clínicas.....	33
2.1.4.	Maniobras diagnósticas.....	34
2.1.5.	Tratamiento.....	36
2.1.6.	Prevención.....	36
2.1.7.	Usuarios de computadoras y el Síndrome del Túnel Carpiano.....	37

2.2	Ergonomía.....	37
2.2.1.	Ergonomía en los puestos de trabajo	38
2.2.2.	Mobiliario.....	38
2.2.3.	Apoya muñecas para el teclado y el mouse.....	39
2.2.4.	Teclado.....	40
2.2.5.	Mouse	42
2.2.6.	La postura.....	42
2.3	Inteligencia Artificial	43
2.3.1.	Ámbito de aplicación de los sistemas IA	44
2.3.2.	Progreso de la Inteligencia Artificial.....	45
2.3.3.	Redes Neuronales Artificiales.....	46
2.3.4.	Arquitecturas de Redes Neuronales	47
2.4	Software Libre.....	48
2.4.1.	Lenguajes de Programación	48
2.5	Sensores Biomédicos	50
2.6	Sensores de eSalud	51
2.7	Marco Legal	52
2.7.1.	Constitución Ecuatoriana	52
2.7.2.	Normativa Aplicable a la Seguridad y Salud en el Trabajo	53
2.7.3.	Código del Trabajo	53

3.	Capítulo III: Diseño e implementación.....	55
3.1	Metodología	55
3.2	Situación Actual	56
3.3	Requerimientos.....	57
3.3.1.	Nomenclatura de requerimientos	57
3.3.2.	Identificación de stakeholders	58
3.3.3.	Requerimientos de stakeholders	58
3.3.4.	Requerimientos del sistema.....	60
3.3.5.	Requerimientos de arquitectura	61
3.4	Diseño del sistema.....	63
3.4.1.	Selección de Hardware y Software del sistema.....	63
3.4.2.	Selección de hardware.....	63
3.4.3.	Selección de software	70
3.4.4.	Arquitectura del sistema.....	73
3.4.5.	Diagrama de flujo del funcionamiento del sistema	75
3.5	Desarrollo del dispositivo.....	76
3.5.1.	Diagrama del sistema de alimentación.....	77
3.5.2	Diagrama de conexión de los módulos de adquisición de datos y procesamiento.	78
3.6	Desarrollo de case del dispositivo	79
3.7	Modelo de entrenamiento para el dispositivo.....	79

3.7.1.	Creación del Dataset	81
3.7.2.	Preprocesamiento.....	82
3.7.3.	División de datos de entrenamiento y prueba.....	84
3.7.4.	Escalado	85
3.7.5.	Creación del modelo	86
3.7.6.	Entrenamiento	87
3.7.7.	Evaluación y predicción.....	88
3.8	Conversión del modelo a una matriz C.....	89
3.9	Programación de módulo de adquisición de datos.....	90
3.10	Configuración de Firebase Realtime Database.....	93
3.11	Programación de módulo de procesamiento	94
3.12	Desarrollo de la aplicación	101
3.12.1.	Estructura de la aplicación	102
3.12.2.	Funcionalidades Implementadas	103
3.13	Backend	109
4.	Capítulo IV: Pruebas de Funcionamiento	110
4.1	Evaluación del modelo de red neuronal.....	110
4.1.1.	Configuración y parámetros de entrenamiento del modelo.....	112
4.2	Prueba de funcionamiento del sistema.....	118
4.3	Pruebas de campo del sistema	129

4.3.1. Participantes y ejecución de pruebas	129
4.4 Comparación entre el dispositivo y el Cuestionario de Boston.....	139
4.5 Cálculo del índice de concordancia	141
4.6 Viabilidad Técnica Económica.....	141
4.6.1. Costo General de dispositivo.....	142
4. Conclusiones y Recomendaciones	144
4.4.1 Conclusiones	144
4.4.2 Recomendaciones.....	146
5. Bibliografía.....	147

LISTA DE TABLAS

Tabla 1 <i>Síntomas más comunes</i>	33
Tabla 2 <i>Examen físico de detección del STC</i>	34
Tabla 3 <i>Ámbito de aplicación de la IA</i>	44
Tabla 4 <i>Áreas de progreso de la IA</i>	45
Tabla 5 <i>Aplicación biomédica de los sensores físicos</i>	50
Tabla 6 <i>Sensores del kit e-Health Sensor Platform V1.0</i>	51
Tabla 7 <i>Nomenclatura de Requerimientos</i>	57
Tabla 8 <i>Lista de participantes en el proyecto</i>	58
Tabla 9 <i>Requerimientos de Stakeholders</i>	59
Tabla 10 <i>Requerimientos del sistema</i>	60
Tabla 11 <i>Requerimientos de Arquitectura</i>	61
Tabla 12 <i>Selección del microcontrolador</i>	63
Tabla 13 <i>Selección de placa de conectividad inalámbrica</i>	65
Tabla 14 <i>Selección de sensor EMG</i>	67
Tabla 15 <i>Consumo eléctrico del dispositivo</i>	69
Tabla 16 <i>Selección de framework</i>	73
Tabla 17 <i>Librerías utilizadas</i>	82
Tabla 18 <i>Configuración de nodos</i>	94
Tabla 19 <i>Especificaciones del hardware utilizado durante el entrenamiento del modelo</i>	111
Tabla 20 <i>Especificaciones del dispositivo móvil</i>	111
Tabla 21 <i>Parametros para la red neuronal</i>	113

Tabla 22 <i>Valores de entrenamiento RNA</i>	117
Tabla 23 <i>Pruebas de operatividad el hardware</i>	129
Tabla 24 <i>Pruebas de funcionamiento en los usuarios</i>	131
Tabla 25 <i>Resultados de la evaluación fisiológica: estado normal, tensión y fatiga</i>	134
Tabla 26 <i>Resultados de la evaluación según la influencia de los días</i>	135
Tabla 27 <i>Cuestionario realizado a voluntarios</i>	138
Tabla 28 <i>Resultados obtenidos del test de Boston</i>	139
Tabla 29 <i>Tabla comparativa entre el dispositivo y el cuestionario de Boston</i>	140
Tabla 30 <i>Costo General del dispositivo</i>	142
Tabla 31 <i>Resumen del costo general del dispositivo</i>	144

LISTA DE FIGURAS

Figura 1 <i>Árbol de problemas sobre el síndrome del túnel carpiano</i>	24
Figura 2 <i>Arquitectura genérica del dispositivo electrónico</i>	27
Figura 3 <i>Ejercicios simples</i>	37
Figura 4 <i>Silla de trabajo para oficina</i>	39
Figura 5 <i>Apoya muñecas para el teclado y el mouse</i>	40
Figura 6 <i>Tipos de teclados</i>	41
Figura 7 <i>Postura correcta frente al computador</i>	43
Figura 8 <i>Estructura de una red neuronal monocapa</i>	47
Figura 9 <i>Estructura de una red neuronal artificial multicapa</i>	47
Figura 10 <i>Fases de la metodología Bottom-Up</i>	55
Figura 11 <i>Seeeduino V4.3</i>	64
Figura 12 <i>ESP32</i>	66

Figura 13 <i>Grove – EMG sensor</i>	67
Figura 14 <i>Grove – sensor GSR</i>	68
Figura 15 <i>IDE Arduino</i>	71
Figura 16 <i>Anaconda</i>	72
Figura 17 <i>Arquitectura del sistema</i>	74
Figura 18 <i>Diagrama de flujo del funcionamiento del sistema</i>	76
Figura 19 <i>Diagrama de alimentación del dispositivo</i>	77
Figura 20 <i>Diagrama de conexión de los sensores</i>	78
Figura 21 <i>Case del dispositivo</i>	79
Figura 22 <i>Proceso de generación del modelo</i>	80
Figura 23 <i>Proporción equilibrada de los valores de la variable objetivo dolor</i>	83
Figura 24 <i>Nivel de correlación de las variables</i>	84
Figura 25 <i>Division de datos entrenamiento y prueba</i>	85
Figura 26 <i>Normalizacion de variables</i>	85
Figura 27 <i>Capas de la red neuronal</i>	86
Figura 28 <i>Ajustes para evitar colapsos de epocas</i>	87
Figura 29 <i>Métricas de entrenamiento y validación</i>	88
Figura 30 <i>Prediccion final del modelo</i>	89
Figura 31 <i>Modelo transformado a una matriz C</i>	90
Figura 32 <i>Definicion de pines y variables</i>	91
Figura 33 <i>Funciones para lectura de sensor GSR</i>	92
Figura 34 <i>Funciones para lectura de sensor EMG</i>	92
Figura 35 <i>Monitoreo del puerto serial</i>	93

Figura 36 <i>Librerías de comunicación con Firebase</i>	95
Figura 37 <i>Credenciales de la Base de Datos y Wifi</i>	96
Figura 38 <i>Componentes globales Tensorflow</i>	97
Figura 39 <i>Configuración de la comunicación serial</i>	98
Figura 40 <i>Función para la comunicación serial entre el ESP32 y Seeeduno V4.3</i>	99
Figura 41 <i>Normalización de los datos</i>	100
Figura 42 <i>Predicción de datos</i>	101
Figura 43 <i>Esquema de la estructura de carpetas de la aplicación</i>	102
Figura 44 <i>Pantalla principal</i>	104
Figura 45 <i>Pantalla de registro</i>	106
Figura 46 <i>Guía para el usuario</i>	107
Figura 47 <i>Pantalla de resultados</i>	108
Figura 48 <i>Historial del paciente</i>	109
Figura 49 <i>Gráfico de curvas de entrenamiento y pérdida</i>	115
Figura 50 <i>Matrices de confusión de red neuronal</i>	116
Figura 51 <i>APK de la aplicación móvil</i>	119
Figura 52 <i>Interfaz de configuración de red</i>	120
Figura 53 <i>Colocación de sensor de electromiografía</i>	121
Figura 54 <i>Colocación de sensor de respuesta galvanica</i>	122
Figura 55 <i>Pantalla de validación de sensores</i>	123
Figura 56 <i>Interfaz de registro</i>	124
Figura 57 <i>Interfaz de resultados</i>	125

Figura 58 <i>Mensaje informativo después de 3 lecturas consecutivas de estado de Fatiga</i>	126
Figura 59 <i>Interfaz del administrador</i>	127
Figura 60 <i>Resultados del usuario en formato pdf</i>	128
Figura 61 <i>Cuestionario de Boston</i>	137

Capítulo I: Antecedentes

1.1 Tema

Dispositivo electrónico como apoyo a la prevención del STC en entornos laborables usando inteligencia artificial.

1.2 Problema

El Síndrome del Túnel Carpiano (STC) surge cuando el nervio mediano se comprime. Este síndrome es muy común en individuos que realizan movimientos repetitivos con las manos. Algunos de los síntomas que suelen presentar las personas afectadas por el STC son debilidad, hormigueo y sensación de quemazón tanto en los dedos como en las manos. Además, si no llega a tratarse los primeros indicios que impliquen síntomas de STC es probable que los mismos evolucionen de un dolor leve a uno agudo e incapacitante. Esto último afectaría la capacidad para realizar actividades que se llevan a cabo en la cotidianidad, como los quehaceres domésticos y el trabajo. Es importante mencionar que esta condición impacta física y de forma emocional, social y laboral, ya que implica la pérdida de días laborales y afecta las actividades diarias (Bolaños, 2015). El Síndrome del Túnel Carpiano puede presentarse en personas de diferentes edades y se origina por movimientos realizados de modo reiterativo con las manos. “Ocurre casi siempre en personas de 30 a 60 años y es más común en las mujeres que en los hombres” (MedlinePlus, 2023, párr. 7). Los casos de STC se presentan más en personas mayores de 30 años, habitualmente en mujeres.

Según datos de la Organización Internacional del Trabajo (OIT), alrededor del 5% de la población adulta en su totalidad sufre de este síndrome, con una incidencia anual de 1 caso por cada 1.000 personas cada año (Organización Internacional del Trabajo, 2009). Además, un estudio de Ayala, publicado en 2018, señala los posibles efectos negativos del uso excesivo de la

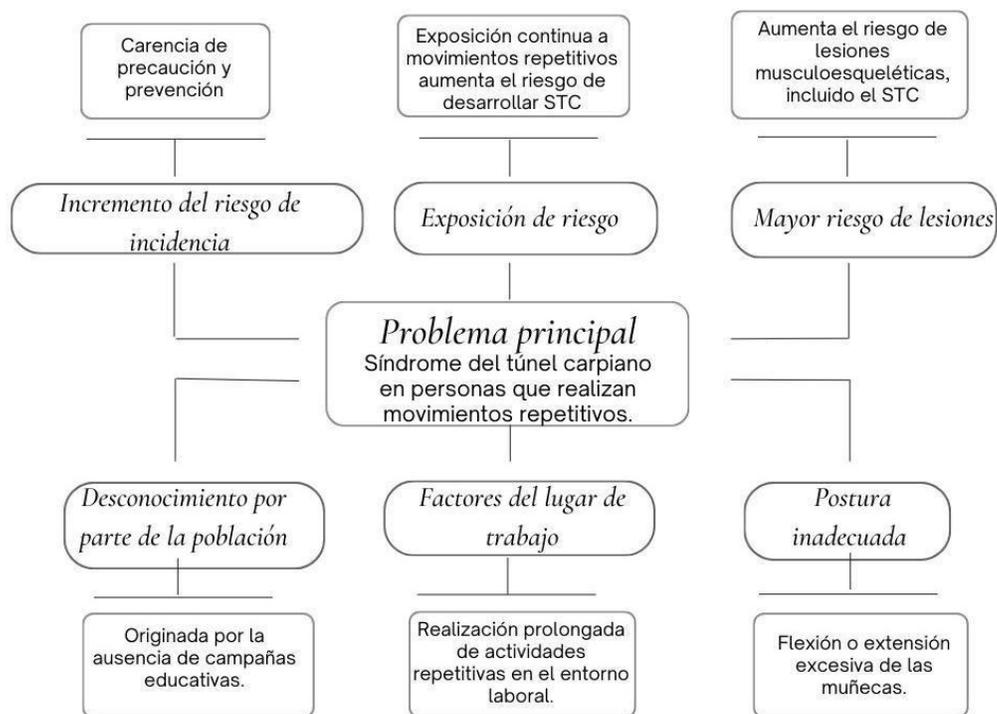
computadora en el nervio medio. Estos efectos incluyen un aumento de la presión en el túnel carpiano cuando se utiliza el mouse, aquello puede afectar la fuerza de agarre de los dedos, la extensión de la muñeca y las pinzas (Ayala, 2018). A lo señalado anteriormente se puede acotar que la prolongada presión en las articulaciones carpometacarpianas puede resultar en un derrame intraneuronal y, como consecuencia, en la desmielinización del nervio mediano (Balbastre et al., 2016; Kafuru et al., 2016).

En Ecuador, de acuerdo con el Seguro General de Riesgos del Trabajo del Instituto Ecuatoriano del Seguro Social, indicaron que las enfermedades presuntivas de origen laboral durante el lapso de 2015 hasta el año 2017 alcanzaron una cifra de 820 casos donde casi la mitad de estos fueron reconocidas. En 2015-2017, las enfermedades profesionales de mayor incidencia en Ecuador son las osteomusculares, la más frecuente en 2015 la lumbalgia crónica más hernia de disco con 103 casos, en 2016 el STC con 70 casos y en 2017 la hernia de disco con 29 casos (Sarango, 2019).

Los trabajadores del área administrativa trabajan un promedio de 8 horas diarias, la mayor parte del tiempo sentados usando el teclado y el mouse, exponiéndose a adoptar posiciones corporales inadecuadas y realizando movimientos repetitivos de los miembros superiores (Ayala, 2018). Como se puede apreciar en la Figura 1, el árbol de causas-efectos sobre el Síndrome del Túnel Carpiano en personas que realizan movimientos repetitivos, resulta evidente la importancia de comprender la sintomatología y la situación actual de esta condición en los trabajadores del área administrativa de la Universidad Técnica del Norte. Se plantea desarrollar un dispositivo que permita recopilar datos sobre las molestias físicas de los trabajadores y facilite la creación de un modelo predictivo para diagnosticar de manera temprana los niveles de estas molestias. El objetivo es reducir el riesgo de lesiones asociadas al Síndrome del Túnel Carpiano.

Figura 1

Árbol de problemas sobre el síndrome del túnel carpiano



Nota. Autoría propia.

1.3 Objetivos

1.3.1. *Objetivo General*

Desarrollar un dispositivo electrónico mediante la utilización de redes neuronales que apoye a la prevención del STC causado por movimientos repetitivos en entornos laborales.

1.3.2. *Objetivos Específicos*

- Recopilar información bibliográfica relevante sobre la realización de movimientos repetitivos, molestias físicas y sintomatología asociados al síndrome del túnel carpiano y redes neuronales.
- Establecer requerimientos para el diseño del sistema utilizando la metodología bottom-up.
- Diseñar e implementar un dispositivo para identificar tempranamente molestias físicas asociadas al síndrome del túnel carpiano mediante redes neuronales.
- Realizar pruebas y validar el funcionamiento del dispositivo en los trabajadores que realicen actividades que impliquen movimientos repetitivos en la UTN (Universidad Técnica del Norte, Departamento Seguridad, Gestión y Riesgos).

1.4 Alcance

El proyecto se enfoca en el diseño, desarrollo e implementación de un dispositivo que permita identificar molestias físicas asociadas al síndrome del túnel carpiano, basado en sensores y redes neuronales, para prevenir lesiones por esfuerzo repetitivo en entornos laborales. Se sigue la metodología Bottom-up, que permite unir los diferentes componentes del sistema desde las especificaciones del cliente. Las fases que se desarrollarán son: especificaciones, ciclo de verificación, integración de bloques, verificación del sistema y producto final (Muñoz & Pauline, 2010).

En la primera fase de especificaciones, la investigación se llevará a cabo mediante un enfoque de investigación aplicada, utilizando teorías y principios de la ergonomía, movimientos repetitivos, respuesta galvánica de la piel, electromiografía, dolores musculares asociados al síndrome del túnel carpiano y redes neuronales. Se examinarán artículos de investigación recientes sobre la fatiga del ratón para comprender mejor el problema y las soluciones existentes. Se aplicarán métodos de recopilación de información bibliográfica, análisis de datos y diseño de sistemas basados en sensores y redes neuronales. En esta fase, se definirán detalladamente los requisitos y se identificarán los componentes necesarios para el diseño del dispositivo.

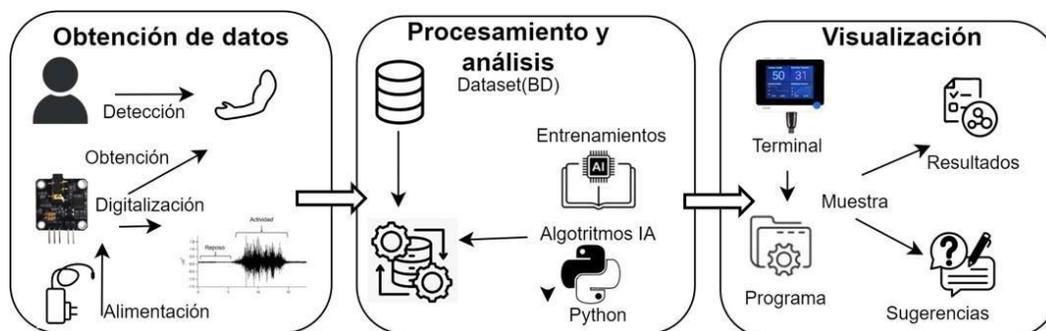
En la fase de ciclo de verificación y simulación en relación con lo obtenido en la fase anterior se seleccionarán los elementos adecuados para la recopilación de datos y almacenamiento de datos, se llevarán a cabo pruebas preliminares para validar la viabilidad técnica del proyecto. Se realizarán pruebas de concepto con los sensores para asegurar su funcionamiento adecuado y su capacidad para capturar datos de manera precisa y confiable. Además, se evaluarán diferentes modelos de redes neuronales para determinar cuál es el más adecuado, considerando factores como la precisión y la eficiencia computacional.

En la Figura 2, se puede visualizar un esquema que representa la arquitectura genérica propuesta para la adquisición, procesamiento y visualización de datos del dispositivo electrónico, en esta fase se procederá a integrar los componentes individuales en un sistema funcional y cohesivo. Se emplea un enfoque basado en mediciones de respuesta galvánica de la piel y electromiografía, recopiladas a través de un dispositivo basado en un microcontrolador conectado a los sensores específicos. Estos datos se utilizan para entrenar un modelo de red neuronal artificial (ANN) que pueda predecir los niveles de dolor muscular y para ello se utiliza un software de código abierto como herramienta principal para el desarrollo y entrenamiento del modelo. Una vez

entrenado y probado, el modelo se convierte para ser ejecutado de manera autónoma, permitiendo la detección precisa de los niveles de dolor muscular y proporcionando advertencias proactivas para prevenir posibles lesiones.

Figura 2

Arquitectura genérica del dispositivo electrónico



Nota: Se muestra la topología del funcionamiento del dispositivo electrónico, está dividido en tres etapas para comprender de mejor manera las actividades que se realiza en cada una de ellas.

En la etapa de verificación del sistema y fase final, se llevarán a cabo pruebas exhaustivas del sistema completo para asegurar su correcto funcionamiento y su conformidad con los requisitos establecidos actuando como un prediagnóstico en la detención de dolores musculares. Se realizarán pruebas de funcionamiento con usuarios reales que desempeñen actividades laborales que involucren el uso prolongado del ratón. Se evaluará los niveles de dolor muscular y la facilidad de uso del sistema, recopilando datos y retroalimentación de los usuarios. Una vez completadas las pruebas y realizados los ajustes necesarios se documentará todo el proceso de desarrollo, incluyendo los detalles de diseño, implementación y pruebas del sistema.

1.5 Justificación

Se debe ver la relación entre salud y trabajo de forma integral, considerando al ser humano desde los ámbitos físico, cognitivo, psíquico y social. Desde este punto de vista integral, la relación salud-trabajo es compleja, pues existe la percepción de que el trabajo afecta la salud, como lo señala la Organización Mundial de la Salud: “el estado de adaptación al medio (biológico y sociocultural), el estado fisiológico de equilibrio, el equilibrio entre la forma y la función del organismo (alimentación), y la perspectiva biológica y social (relaciones familiares, hábitos)” (Possidoni, 2015, p. 32)

Los Objetivos de Desarrollo Sostenible (ODS), adoptados por la Asamblea General de las Naciones Unidas en 2015, representan una agenda ambiciosa para abordar los desafíos globales más apremiantes del siglo XXI. La Organización Panamericana de la Salud [OPS] (2020) señala que, si bien los ODS no mencionan explícitamente la salud en el trabajo, “todos los 17 ODS abordan temas que tienen impactos importantes en la salud” (Organización Panamericana de la Salud, s.f.). Por ejemplo, promover entornos de trabajo seguros puede ayudar a reducir las muertes y enfermedades relacionadas con el trabajo, lo que contribuye a la meta 3.8 del ODS 3.

Del mismo modo, crear lugares de trabajo que promuevan el bienestar mental y físico de los trabajadores puede contribuir a la meta 3.4 del ODS 3. Además, proteger los derechos laborales y promover entornos de trabajo decentes, como lo exige la meta 8.8 del ODS 8, es fundamental para garantizar que todos los trabajadores tengan la oportunidad de disfrutar de buena salud y bienestar (Naciones Unidas, 2018).

En Ecuador entre las normas de mayor relevancia encontramos la normativa del Instituto Ecuatoriano de Seguridad Social (IESS), en su resolución C. D. 513 del Reglamento de Seguridad General de Riesgos del Trabajo en el artículo 55: “Mecanismos de la Prevención de Riesgos del

Trabajo: Las empresas deberán implementar mecanismos de Prevención de Riesgos del Trabajo, como medio de cumplimiento obligatorio de las normas legales o reglamentarias, haciendo énfasis en lo referente a la acción técnica que incluye: identificación de peligros y factores de riesgo, medición de factores de riesgo, evaluación de factores de riesgo, control operativo integral, vigilancia ambiental laboral y de la salud y evaluaciones periódicas” (Resolución No. C.D. 513. Reglamento del Seguro General de Riesgos del Trabajo, 2016, p. 24). Esta norma identifica los derechos que tienen los trabajadores de llevar a cabo sus funciones en un ambiente laboral adecuado que asegure el cuidado de su salud, seguridad y bienestar, con lo que compromete y obliga a las empresas a velar por el cuidado de sus colaboradores desde la identificación de riesgos hasta las evaluaciones.

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. Los avances tecnológicos suceden a un ritmo acelerado es un hecho. Esto influye en nuestra profesión, por un lado, porque debemos mantenernos informados sobre todas las novedades en cuanto a técnicas y equipos, y por otro lado porque somos agentes de educación para la salud con capacidad para favorecer cambios de conductas saludables en la población (Cepeda et al., 2012).

Existen ciertas actividades que incrementan el riesgo de aparición del STC. Los trabajadores del área administrativa, donde casi todas sus actividades las realizan sentados frente a un computador y usando constantemente el teclado y el mouse, presentan sintomatología o lesión osteomuscular que permita la instauración o aparición del STC (Ayala, 2018). Por lo tanto, surge la necesidad de conocer la sintomatología presuntiva que puedan presentar en los trabajadores del

área administrativa de la Universidad Técnica del Norte (Departamento Seguridad, Gestión y Riesgos).

También es importante como futuros profesionales, considerar las diferentes enfermedades ergonómicas que pueden presentar para promover y prevenir estas patologías. Los trabajadores administrativos de la Universidad Técnica del Norte (Departamento Seguridad, Gestión y Riesgos) se han incluido en el proyecto, ya que ellos ejecutan actividades que requieren de esfuerzo y movimientos repetitivos sobre mano y muñeca, por lo tanto, es indispensable incluirlos y determinar sus factores de riesgo promoviendo el derecho a la salud con equidad.

Capítulo II: Estado del Arte

Este capítulo contiene una revisión de literatura detallada sobre los principales conceptos teóricos que permiten abordar el problema de investigación. Estos sirven como base para el análisis y la comprensión del tema que se va a investigar, debido a que la información recopilada otorga un sustento fundamental a la línea de investigación. Para ello se aborda la definición del síndrome del túnel del carpo, luego se describirá la frecuencia y los determinantes que presenta, el origen, los síntomas, las formas en las que se puede diagnosticar, el tratamiento y la prevención. Dentro de la misma línea se abarca la relación existente entre el uso de la computadora y el síndrome del túnel carpiano. Para complementar, se explica sobre la ergonomía y cómo es importante en los puestos de trabajos. También se describe las características que deben cumplir ciertos elementos como, por ejemplo, el mobiliario, el apoyo muñecas, el teclado, el mouse; para así evitar tensión en la zona del carpo. Asimismo, se indica la importancia de mantener una buena postura frente a la computadora. Posteriormente se profundiza en la inteligencia artificial (IA), en este aspecto se consideran los ámbitos de aplicación, los progresos, las redes neuronales que permiten su funcionamiento y cómo es la arquitectura de estas. Una vez concluida esto, se precisa qué es el software libre y cuáles son los lenguajes de programación. Del mismo modo, se detalla sobre los microcontroladores y su estructura. Por último, se determina la aplicación que tienen los sensores biomédicos y cuáles son los tipos de sensores de eSalud que existen.

2.1 Síndrome del Túnel del Carpo

El Síndrome del Túnel Carpiano (STC) es una afección que surge por el movimiento repetitivo de las manos, debido a estos el nervio mediano sufre una presión excesiva. Gorostiza (2015) menciona que:

el Síndrome del Túnel de Carpo es una condición clínica que aparece cuando el nervio mediano es comprimido a su paso por el túnel del carpo. Los síntomas de este síndrome aparecen en la zona de inervación del nervio mediano y las principales son dolor, a menudo nocturno; parestesias o disestesias; hormigueos, sobre todo por las yemas de los dedos; adormecimiento de la mano y pérdida de sensibilidad y fuerza (p. 4)

Es decir, es una patología que afecta al túnel del carpo a través de la compresión del nervio mediano.

2.1.1. Epidemiología

En estudios realizados sobre el STC, acerca de la frecuencia y los determinantes que este presenta, se encontraron los siguientes datos:

Su incidencia es de 99 por 100 000 personas por año. Prevalece en hombres en un 0,6%, mientras que en mujeres en un 3,4%. Es una de las afecciones más frecuentes, (...), afecta al 3% de los adultos, siendo más frecuente en las mujeres debido a 7:1 (Fuel, 2017; Salas y Díaz, 2016). Usualmente aparece entre la cuarta y sexta década de vida. Sin embargo, hay que considerar que los casos en los que se presenta antes de los 40 años se debe a la actividad ocupacional (Fuel, 2017; Pozo, 2018; Salas y Díaz, 2016). Además, es común que el STC se presente en trabajadores que durante sus labores realicen movimientos de gran fuerza y muy repetitivos con las manos (Fuel, 2017).

Por tanto, los autores anteriormente mencionados coinciden en que esta patología se encuentra presente mayoritariamente en las mujeres antes que en los hombres.

2.1.2. Fisiopatología

El movimiento repetitivo y constante que, relacionado con los factores de riesgos o la presencia de dos o más enfermedades simultáneas en una persona, trae como consecuencia la fatiga

crónica, la tensión permanente en músculos, tendones o músculos, el dolor y las lesiones. A nivel motriz se puede presentar deficiencia de movimiento y sensibilidad. Asimismo, de acuerdo con el nivel de severidad puede aparecer una desmielinización y en algunos casos degeneración axonal (Naranjo, 2016).

Este síndrome también se produce cuando la presión del túnel del carpo supera los 30 mm Hg, puesto que se produce una alteración en el nervio mediano. Incluso puede aumentar hasta los 90 mm Hg o más. La excesiva compresión provoca una disfunción en el correcto movimiento de la mano y la muñeca (Pozo, 2018). Por lo tanto, el STC se presentará a través de una serie de signos y síntomas relacionados con lo sensomotriz.

2.1.3. Manifestaciones Clínicas

Los síntomas presentes en el STC dependerán de la fase en la que se encuentre la enfermedad. Al inicio se presentan síntomas asociados con la alteración de las funciones sensitivas. En esta fase los signos del síndrome pueden aparecer y desaparecer por momentos, es decir, su presencia no es constante. Mientras que, en una fase avanzada los pacientes presentan afecciones motoras persistentes durante periodos continuos y prolongados. Esto imposibilita la realización de actividades que pueden resultar sencillas en la cotidianidad. Algunos de los síntomas más comunes se encuentran en la Tabla 1.

Tabla 1

Síntomas más comunes

Síntomas	Lugar de aparición
Entumecimiento, cosquilleo, ardor y dolor	Dedos: pulgar, índice, medio y anular.

Espasmo	Irradiación a los dedos: pulgar, índice, medio y anular.
Dolor o cosquilleo	Se extiende del antebrazo al hombro.

Nota. (Gualotuña, 2022, p. 11)

Los pacientes suelen manifestar que en ocasiones tienen una sensación de dolor quemante que llega a despertarlos durante la noche y mejora al mover de un lado al otro la mano. También indican sobre la presencia de debilidad y torpeza al ejecutar movimientos sencillos con la mano.

2.1.4. *Maniobras diagnósticas*

Existen diferentes signos que permiten llevar a un buen diagnóstico del Síndrome del Túnel Carpiano. Estos se realizan mediante exámenes físicos, algunos de ellos son los que se pueden evidenciar en la Tabla 2.

Tabla 2

Examen físico de detección del STC

Examen físico	Evaluación	Referencia
Signo de Phalen	Positivo el 80%, aparece el dolor al momento de flexionar la muñeca en 90° al minuto de prueba. Al hacer esto se reduce el espacio del túnel lo que presenta parestesias.	

Signo de Tinel

Positivo el 60-70%, consiste en percutir con los dedos o martillo reflejo el ligamento anular, el paciente presenta calambres en los dedos 1, 2 y 3.



Signo de Flick

Positivo el 93% signo mayor nivel predictivo, el paciente indica que síntomas se intensifican en las noches y disminuyen cuando agita la mano (movimientos similares al sacudir un termómetro).



Signo del círculo

El médico le indica al paciente oponer el pulgar con el índice y no realiza una flexión adecuada de falanges.



Nota. (Rosado, 2023, pp. 9–10)

Los signos que se mencionaron en la Tabla 2 permiten detectar la severidad del Síndrome del Túnel Carpiano. La característica común que comparten todos estos signos recae en la atrofia muscular de la eminencia tenar y en la ausencia de sensibilidad del primer al tercer dedo. Cabe resaltar que es de vital importancia que se tomen en cuenta los síntomas que permiten reconocer en que estadio de la patología se encuentra el STC.

Además, a través de una adecuada historia clínica el médico podrá implementar el proceso más apropiado para diagnosticar este síndrome. Por lo tanto, el médico tiene la potestad de indagar a los pacientes sobre el uso continuo de objetos manipulables que aumenten el riesgo de desarrollar esta sintomatología. Hay que considerar que el uso de estos objetos puede presentarse tanto en el hogar como en el área de trabajo.

2.1.5. Tratamiento

Existen dos procedimientos para tratar el Síndrome del Túnel Carpiano, estos se dividen en quirúrgico y no quirúrgico. La primera consiste en la liberación del túnel carpiano mediante métodos o técnicas que realizan incisiones en la zona del ligamento transversal del carpo con el fin de disminuir la presión presente sobre el nervio mediano. El segundo, también conocido como tratamiento conservador, consiste en que se debe inmovilizar durante las noches la muñeca con una férula, emplear esteroides corticales, fisioterapia, la terapia de ultrasonido, entre otras más (Ghasemi-rad et al., 2014).

El tipo de tratamiento a aplicar dependerá del estado en el que se encuentre el STC. “Los tratamientos no quirúrgicos se recomiendan para pacientes con síntomas leves de STC. Se recomienda una evaluación quirúrgica para los pacientes con síntomas moderados a graves” (Ghasemi-rad et al., 2014, p. 295). Por lo tanto, el procedimiento a seguir recae sobre la levedad o gravedad de los síntomas que posea el paciente afectado.

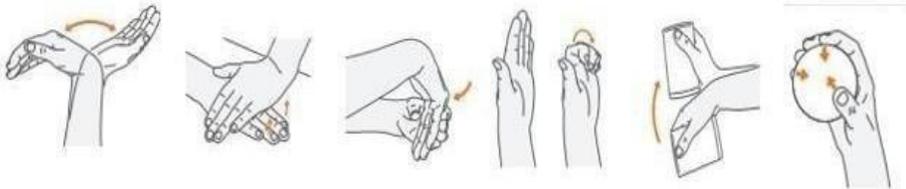
2.1.6. Prevención

Hay diferentes tipos de ejercicios simples que se pueden usar para prevenir el Síndrome del Túnel Carpiano. Estos se pueden llevar a cabo al culminar la jornada laboral o en el transcurso del día. Resulta importante que cada persona propensa a sufrir STC ponga en práctica estos

ejercicios. Algunos de los ejercicios que ayudan a prevenir el STC se encuentran ejemplificados gráficamente en la Figura 3.

Figura 3

Ejercicios simples



Nota. (Arthritis Research UK, s.f.)

2.1.7. Usuarios de computadoras y el Síndrome del Túnel Carpiano

En una investigación titulada *Relationship Between Carpal Tunnel Syndrome and Wrist Angle in Computer Workers* se encontró que las personas que usan prolongadamente la computadora llegan a presentar síntomas relacionados al STC. Esto se debe a que al momento de escribir en el teclado las personas colocan la muñeca en un ángulo de extensión que causa incomodidad. Además, en este estudio se evidenció que uno de los factores predisponente para el desarrollo de STC es la extensión de la muñeca en un ángulo mayor a 20° (Liu et al., 2003).

2.2 Ergonomía

La Asociación Internacional de Ergonomía (2000) define a la ergonomía como:

la disciplina científica que se ocupa de la comprensión de las interacciones entre los seres humanos y otros elementos de un sistema, y la profesión que aplica teoría, principios, datos y métodos al diseño con el fin de optimizar el bienestar humano y el rendimiento general del sistema (párr. 1).

Por tanto, la ergonomía realiza ajustes a los aspectos de un sistema con el objetivo de brindar un espacio que garantice el confort necesario para cada personal que usará los espacios.

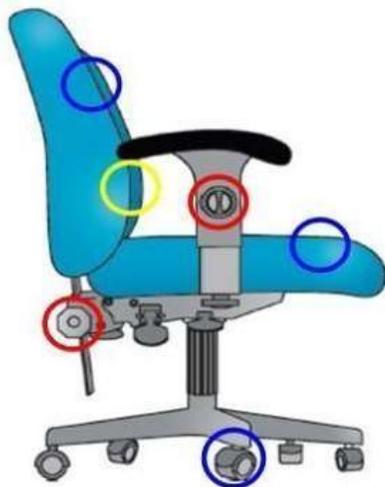
2.2.1. Ergonomía en los puestos de trabajo

El puesto de trabajo debe tener un buen diseño, ya que esto otorga grandes beneficios en la salud del trabajador y le permite una mejor productividad. Por lo tanto, en los puestos de oficina se debe tomar en cuenta elementos como el diseño de instalaciones de trabajo, el equipamiento de mobiliario con calidad ergonómica. Asimismo, hay que considerar que la organización del trabajo es vital para evitar el estrés y la desmotivación de los trabajadores (Confederación Regional de Organizaciones Empresariales de Murcia, s.f.).

2.2.2. Mobiliario

La silla de trabajo debe contar con un asiento y respaldo de altura que permita ser ajustado. Debe tener una base estable de 5 ruedas para lograr una libertad en el movimiento. Los reposabrazos también deben poder ajustarse en altura y ancho. Cabe resaltar que todos los elementos ajustables deben ser de uso fácil. Además, el asiento debe tener una anchura entre 40 y 45 cm, debe tener un acolchado de 20 mm con una cubierta de tela que sea flexible y transpirable, con una inclinación en el borde anterior. El respaldo del asiento puede ser alto o bajo. Cuando es bajo debe conseguir un correcto apoyo de las vértebras lumbares. También debe ser regulable en altura e inclinación. Cuando es alto debe permitir el apoyo lumbar y ser regulable en inclinación hasta 15° hacia atrás (Chavarría, 1989; Instituto de Salud Pública Chile, 2016).

En la Figura 4 se puede observar las características que debe cumplir la silla ergonómica.

Figura 4*Silla de trabajo para oficina*

Nota. (Instituto de Salud Pública Chile, 2016)

El escritorio debe ser lo suficientemente espacioso para colocar el monitor, el teclado, el mouse, el apoya muñecas, el teléfono y los documentos. La altura adecuada de este dependerá de las características del usuario y el espacio que queda por debajo debe permitir el estiramiento de las piernas. Además, se recomienda que quede una altura de 5 cm entre los muslos y la parte inferior de la mesa de trabajo. La altura de la parte superior del escritorio debe ser de 75 cm, con el uso de un apoya pies en los casos que lo requieran. Básicamente, la altura de la mesa debe permitir que el trabajador apoye sus codos y antebrazos cómodamente sobre la superficie, sin exigir un esfuerzo. La profundidad del escritorio debe permitir que la pantalla se desplace de unos 50 cm a 70 cm de distancia entre los ojos y la pantalla. Por tanto, la profundidad recomendable es de 80 cm o más (Instituto de Salud Pública de Chile, 2016).

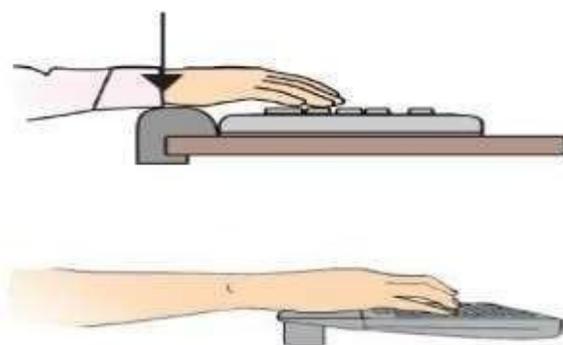
2.2.3. Apoya muñecas para el teclado y el mouse

El apoya muñecas para teclado permite mantener un apoyo de las muñecas sobre una superficie suave. Igualmente mantiene a la muñeca en una posición que evita un excesivo esfuerzo.

Estos accesorios deben ser uniformes y con una base antideslizante. El apoya muñecas para el mouse es similar al del teclado. La única diferencia es que permite alinear la muñeca con respecto al brazo mientras se utiliza el mouse (Instituto de Salud Pública Chile, 2016). En la Figura 5 se puede observar una ilustración que contiene el apoya muñecas para mouse y teclado.

Figura 5

Apoya muñecas para el teclado y el mouse



Nota. (Instituto de Salud Pública Chile, 2016).

2.2.4. Teclado

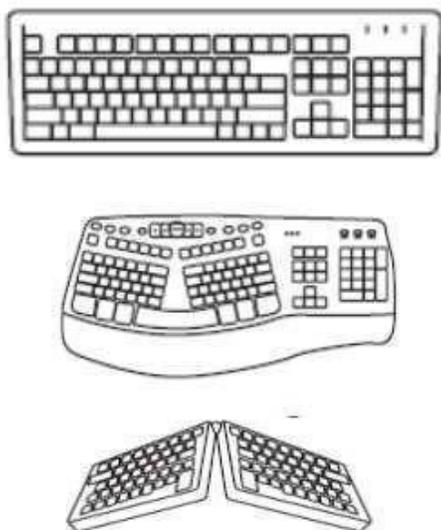
En los trabajos de oficina el teclado resulta indispensable, ya que las personas deben realizar un sinnúmero de documentos. Es por esta razón que resulta recomendable el uso de una técnica y estrategia adecuada que permita una correcta digitalización. Una técnica correcta de tipeo consta al presionar cada tecla de manera suave y haciendo uso de las dos manos. Además, las muñecas y los antebrazos deben permanecer apoyados en el escritorio. Es indispensable que en los espacios de trabajo se haga uso de un teclado ergonómico, pues reducen la desviación cubital de las muñecas (Instituto de Salud Pública Chile, 2016).

Los teclados ergonómicos se presentan como opciones que permiten reducir la aparición de alteraciones musculares. No obstante, hay falta de evidencia que confirmen este beneficio.

Además, algunos teclados pueden favorecer el desarrollo del trabajo de las personas, pero estos beneficios no son aplicables de la misma manera para todas las personas. Existe una gran diversidad en el diseño de los teclados. Un ejemplo de ellos es el teclado partido, que es uno de los más cómodos y una vez las personas aprenden a utilizarlo se reduce las molestias al escribir. Es importante resaltar que el diseño correcto, la colocación del teclado, la realización de pausas y la reducción del ritmo de trabajo ayuda a disminuir las alteraciones producidas por el uso prolongado de estos (Fidalgo y Nogareda, 2001). En la Figura 6 se encuentran diferentes tipos de teclados ergonómicos.

Figura 6

Tipos de teclados



Nota. (Instituto de Salud Pública Chile, 2016).

2.2.5. Mouse

De acuerdo con Gualotuña (2022) el mouse debe adaptarse al tamaño y a las curvas de mano de la persona que lo use. El movimiento que este realice debe ser fácil y sin restricciones. Es por esta razón que los cables no deben cruzarse e interrumpir su uso. De igual manera, al momento de hacer clic no se debe alterar la posición de la mano y la muñeca. Por tanto, el tipo de mouse que se elija dependerá del tiempo de uso en la jornada laboral.

2.2.6. La postura

La postura inadecuada posibilita el desarrollo de alteraciones musculares. Uno de los principales riesgos en el trabajo de oficina es la postura. El permanecer sentado durante tiempos prolongados afecta la circulación sanguínea, produce fatiga y lesiones musculoesqueléticas. Además, las malas posiciones adquiridas al trabajar provocan que los músculos del cuellos, brazos y espalda se pongan tensos (McMutual, 2017).

Al trabajar frente a un ordenador la parte superior e inferior del cuerpo deben formar un ángulo de 90° o 100° grados. Esta postura se logra con la espalda apoyada en el respaldo de la silla, la regulación de la altura del asiento para que los brazos sobre la mesa y pegados al cuerpo formen un ángulo recto con el antebrazo. Además, mientras se trabaja es recomendable mover los pies y las piernas. También se debe variar la posición de los pies entre el suelo y el reposapiés, no cruzar las rodillas o flexionar las piernas debajo de 90° grados. Esto se debe a que dificultan la circulación de la sangre. Finalmente, es importante revisar la postura cada cierto tiempo, para hacer conciencia de si se encuentra sentado correctamente o no (Gualotuña, 2022). En la Figura 7 se encuentra una imagen de referencia que indica cómo debe ser la postura correcta que deben tener las personas frente al computador.

Figura 7

Postura correcta frente al computador



Nota. (FisioPro, 2018).

2.3 Inteligencia Artificial

La inteligencia artificial, cuyas siglas son IA, es una herramienta tecnológica que permite la construcción de sistemas que dan solución a problemas concretos. De acuerdo con Banda (2014) la IA abarca investigaciones tanto científicas como tecnológicas sobre los sistemas inteligentes. Estos últimos son entidades, como máquinas o humanos, con capacidades cognitivas que permiten el cumplimiento de metas en determinadas situaciones. Por lo tanto, el principal desafío de la inteligencia artificial recae en la producción de sistemas que puedan percibir, aprender, almacenar o razonar sobre los contenidos o la información que se conoce. Además, los seres artificiales desarrollados serían capaces de realizar actividades inteligentes complejas mediante el uso del lenguaje humano y la interacción con el entorno físico. Esto se debe a que simularían el comportamiento de otros seres vivos.

2.3.1. *Ámbito de aplicación de los sistemas IA*

En los últimos 60 años la IA ha sido un aspecto central para el avance en campos científicos e intelectuales. Es por esta razón que los sistemas de inteligencia artificial pueden aplicarse en diferentes ámbitos. Algunos de ellos se muestran en la Tabla 3.

Tabla 3

Ámbito de aplicación de la IA

Ámbito	Aplicación
Medicina	Ayuda al diagnóstico
	Análisis de imágenes biomédicas
	Procesado de señales fisiológicas
Ingeniería	Organización de la producción
	Optimización de procesos
	Cálculos de estructuras
	Planificación y logística
	Diagnóstico de fallos
	Toma de decisiones
Economía	Análisis financiero bursátil
	Análisis de riesgo
	Estimación de precios en productos derivados
	Minería de datos
Marketing y fidelización de clientes	Marketing y fidelización de clientes
	Análisis de estructuras biológicas
	Genética médica y molecular
Informática	Procesado de lenguaje natural

Criptográfica

Teoría de juegos

Lingüística computacional

Nota. (Benítez et al., 2018, p. 18).

2.3.2. Progreso de la Inteligencia Artificial

La IA ha marcado un progreso significativo en diversas áreas, debido al gran potencial de transformación que presenta. Este progreso se encuentra en los juegos, el tratamiento de imágenes, la automatización de vehículos, la traducción del lenguaje, entre otros. Además, su evolución ha permitido el desarrollo de tecnologías cognitivas en distintas industrias. Algunos de los avances más importantes de la IA, mencionados por Lepage (2016), se encuentran sintetizados en la Tabla 4.

Tabla 4

Áreas de progreso de la IA

Área	Descripción	Uso
Ingeniería del conocimiento	Proceso de comprender y luego representar el conocimiento humano en estructuras de datos, modelos semánticos y reglas heurísticas.	Aplicaciones para resolver complejos problemas asociados a la experticia de los humanos.
Robótica	Dispositivo mecánico autónomo que puede realizar tareas e interactuar con el mundo físico o real.	Automatizar tareas repetitivas en áreas que manipulan materiales, ensamblan procesos y realizan actividades de control.

Aprendizaje automático	Está compuesto de herramientas, técnicas y algoritmos para analizar datos.	Crear modelos predictivos o identificar patrones dentro de los datos.
------------------------	--	---

Nota. (Lepage, 2016, pp. 6, 9).

2.3.3. Redes Neuronales Artificiales

Las redes neuronales artificiales, cuyas siglas en inglés son ANNs, simulan a las redes neuronales biológicas. Según Ilunga y Stephenson (2005) “ANNs are networks of interconnected simple units (nodes) that are based on a greatly simplified model of the brain” [Las ANNs son redes de unidades simples interconectadas (nodos) que se basan en un modelo muy simplificado del cerebro] (p. 172). Por tanto, son un modelo tecnológico que imitan el funcionamiento del cerebro al procesar la información.

El funcionamiento de las ANNs se desarrolla a través de los datos de entrada y salida de un proceso de formación. Fernández (2011) menciona que:

Una red neuronal recibe un conjunto de variables de entrada y la respuesta correspondiente o el objetivo fijado en la salida (cuando esto es así se llama el entrenamiento supervisado). Se define como error la diferencia entre la respuesta deseada y la salida proporcionada por la red neuronal; dicho error se vuelve a administrar al sistema (feedback) para reajustar los parámetros de entrenamiento de forma sistemática (regla de aprendizaje). El proceso se repite hasta que el resultado final es aceptable (p. 29).

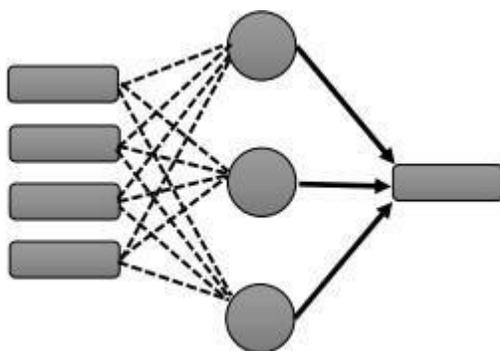
Es decir, la red neuronal se configura mediante una serie de variables que permiten llegar al resultado deseado. Esto se realiza a través de la inserción y ajustes de parámetros que establecen conexiones con los conocimientos previos de entrenamiento. Por esta razón, se debe disponer de un gran número de datos que cubra significativamente las condiciones de funcionamiento.

2.3.4. *Arquitecturas de Redes Neuronales*

La arquitectura de una red neuronal hace referencia a la estructura física y de organización que permitirá que las neuronas artificiales se conecten entre sí. La arquitectura básica de una red neuronal artificial se compone por tres tipos de capas: entrada, oculta y de salida. Además, se pueden encontrar redes artificiales monocapa o multicapa. Díaz (2023) señala que “las RNA[redes neuronales artificiales] monocapa o perceptrón son una arquitectura simple que se compone de una capa de entrada y una de salida” (p.10). Es decir, las redes neuronales artificiales monocapa tienen una arquitectura simple por la forma en la que se encuentran constituidas. En la Figura 8 se presenta un ejemplo de esta.

Figura 8

Estructura de una red neuronal monocapa

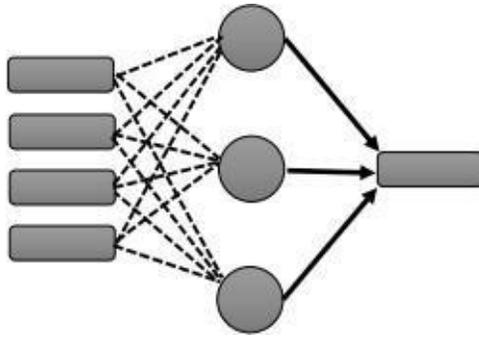


Nota. (Díaz, 2023, p. 11)

Las redes neuronales artificiales multicapas están formadas por varias capas. “A diferencia de la red monocapa tiene capas ocultas para identificar patrones exhaustivamente y resolver problemas más complejos” (Díaz, 2023, p. 11). En la Figura 9 se indica un ejemplo de esta.

Figura 9

Estructura de una red neuronal artificial multicapa



Nota. (Díaz, 2023, p. 11)

2.4 Software Libre

Es un software cuyo código puede experimentar alteraciones o modificaciones por parte del usuario. De acuerdo con Stallman (2004) “con software libre nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software” (p. 45). Es decir, la persona que haga uso del software tendrá total libertad para controlar el programa y sus funciones.

2.4.1. *Lenguajes de Programación*

El lenguaje de programación hace referencia a aquellos programas que permiten la construcción de una aplicación. Guevara (2008) menciona que este lenguaje es “un sistema de comunicación entre el programador y la máquina (computador), creado con el único fin de programar computadoras, esto se hace partiendo del uso de instrucciones preestablecidas” (p. 15). Por consiguiente, las instrucciones son los pasos que seguirá la máquina para realizar una actividad. Dentro de los softwares de programación se encuentran Python, Visual Studio, Open CV.

2.4.1.1 Python

Es un lenguaje de programación que se usa en el desarrollo de aplicaciones web y softwares. Pajankar (2017) menciona que es un lenguaje de alto nivel, el cual tiene como características principales la sencillez y la legibilidad. Además, algunos de sus aspectos más relevantes son la robustez, el soporte a múltiples plataformas, la facilidad de aprendizaje y la constante actualización. Es importante destacar que este lenguaje cuenta con varias bibliotecas que permiten la adición de diversas herramientas al desarrollo de proyectos o prototipos.

2.4.1.2 Visual Studio

Es una plataforma que permite la edición, depuración y traducción del código fuente. Microsoft (2022) señala que es el entorno de desarrollo integrado (IDE) más completo para quienes compilan aplicaciones de .NET y C++ en el entorno de Windows. Asimismo, presenta una amplia gama de herramientas que posibilita la potencia y la perfección de cada etapa del proceso de desarrollo de software. Algunas de sus funciones son el autocompletado de código mediante tecnología de IA, trabajo de codificaciones compartidas en tiempo real, producción de clones de los distintos repositorios y la configuración automática de flujos de trabajo de CI/CD.

2.4.1.3 Open Computer Vision

Open Computer Vision, también conocido como OpenCV, es una biblioteca libre que permite el procesamiento de imágenes en Python. Según Jarrín (2020) OpenCV es una biblioteca que contiene softwares de dominio público, es decir, son de código abierto. Los softwares que contiene son para visión artificial y aprendizaje automático. Además, esta biblioteca ofrece más de 2500 algoritmos que permiten el reconocimiento de objetos, rostros, acciones humanas. También pueden corregir los ojos rojos, hacer un seguimiento de movimiento ocular, identificar escenarios, extraer modelos 3D, entre otras cosas.

2.5 Sensores Biomédicos

Los sensores biomédicos son dispositivos que permiten medir parámetros relevantes acerca de la condición de salud de una persona.

Los sensores biomédicos toman señales que representan variables biomédicas y las convierten en lo que usualmente son señales eléctricas. Los sensores biomédicos sirven como una interfaz entre los sistemas biológico y electrónico y deben funcionar de manera tal que no afecten adversamente a ninguno de estos sistemas (Chamorro J, s.f., p. 2)

En otras palabras, los sensores biomédicos son una interfaz que convierten las variables biomédicas en señales eléctricas. En la Tabla 5 se pueden evidenciar algunos ejemplos de aplicación biomédica de los sensores físicos.

Tabla 5

Aplicación biomédica de los sensores físicos

Sensor	Aplicación	Rango de la señal
Indicador de estiramiento de metal	Movimiento respiratorio	0 – 0.05
Sensor magnético de desplazamiento	Movimiento respiratorio	0 – 10 mm
LVDT	Contracción muscular	0 – 20
	Sensor de contracción	0.5 mm
Calla de carga	Escala electrónica	0 - 440 (0 – 200 kg)
Acelerómetro	Actividad del sujeto	0 – 20 mts ²
Sensor de presión miniatura	Presión intra-arterial	0 – 50 Pa
	Presión de la vejiga	0 – 10 Pa
	Presión infrauterina	0 – 15 Pa

	Gasto cardíaco	0 – 500 ml/min
Sensor electromagnético de flujo	Flujo sanguíneo de un órgano.	0 – 100 ml/min
Flujo sanguíneo de un órgano	Pletismógrafo	0 – 0.02

Nota. (Gavilema y Mullo, 2014).

2.6 Sensores de eSalud

La plataforma Arduino es una de las plataformas de prototipado rápido y desarrollo de hardware que más apogeo ha tenido en los últimos años. Buechley et al. (2008) indica que Arduino brinda un hardware de uso fácil y efectivo, cuyo fin es conectar y emplear varios componentes electrónicos con una gama de aplicaciones. Además, en la página web oficial de Arduino se puede encontrar una colección completa de tutoriales, formas de descarga e información de las diferentes versiones de la plataforma (Arduino, 2017).

Otro sensor de eSalud es el kit eHealth SensorPlatform V1.0, el cual fue creado por Libelium. Este kit permite a sus usuarios la adquisición de un cúmulo de señales fisiológicas ECG, EMG, frecuencia respiratoria, temperatura superficial y GSR. En la Tabla 6 se pueden observar estos aspectos.

Tabla 6

Sensores del kit e-Health Sensor Platform V1.0.

Nombre del sensor	Descripción	Tipo de datos
Sensor de temperatura	La temperatura corporal depende del lugar del cuerpo en el que se realiza la medición, la hora del día y el nivel de	Discretos

	actividad del sujeto. Diferentes partes del cuerpo tienen diferentes temperaturas.	
Sensor de flujo de aire	La frecuencia respiratoria es un indicador amplio de inestabilidad fisiológica importante. El sensor mide el flujo respiratorio de una persona en las vías respiratorias superiores (nariz). El sensor de flujo de aire también puede proporcionar una advertencia de hipoxemia y apnea.	Continuos
Sensor Galvánico de Respuesta de la Piel (GSR)	Se puede utilizar para medir la conductancia eléctrica de la piel, que varía con su nivel de humedad. La conductancia de la piel se utiliza como indicación de excitación psicológica o fisiológica. GSR mide la conductancia eléctrica entre 2 puntos y es esencialmente un tipo de medidor de la resistencia eléctrica u ohmímetro.	Continuos

Nota. (Bayo, 2023, pp. 79–80).

2.7 Marco Legal

2.7.1. *Constitución Ecuatoriana*

La Constitución de la República del Ecuador, vigente desde el 2008 y modificada en el 2021, constituye una base legal que sustenta la presente investigación. El Art. 33 referente al Trabajo y Seguridad Social puntualiza que:

el trabajo es un derecho y un deber social, y un derecho económico, fuente de realización personal y base de la economía. El Estado garantizará a las personas trabajadoras el pleno respeto a su dignidad, una vida decorosa, remuneraciones y retribuciones justas y el desempeño de un trabajo saludable y libremente escogido o aceptado (p. 19).

Del mismo modo, el Art. 326 referente a las Formas de Trabajo y su Retribución, literal (5) estipula que “toda persona tendrá derecho a desarrollar sus labores en un ambiente adecuado y propicio, que garantice su salud, integridad, seguridad, higiene y bienestar” (p. 162).

2.7.2. Normativa Aplicable a la Seguridad y Salud en el Trabajo

Asimismo, se debe señalar que la Normativa Aplicable a la Seguridad y Salud en el Trabajo, vigente desde 2016, también es un referente legal del presente objeto de estudio. Dentro de esta se toma en cuenta la Resolución No. C.D. 513 en la que en el Art. 6, ubicado en Capítulo II, alude a las Enfermedades Profesionales u Ocupacionales y se puede destacar que:

son afecciones crónicas, causadas de una manera directa por el ejercicio de la profesión u ocupación que realiza el trabajador y como resultado de la exposición a factores de riesgo, que producen o no incapacidad laboral. Se considerarán enfermedades profesionales u ocupacionales las publicadas en la lista de la Organización Internacional del Trabajo OIT, así como las que determinare la CVIRP para lo cual se deberá comprobar la relación causa - efecto entre el trabajo desempeñado y la enfermedad crónica resultante en el asegurado, a base del informe técnico del SGRT (p. 10).

2.7.3. Código del Trabajo

De igual forma, se considera al Código del Trabajo, vigente desde el 2005 y actualizado en el 2020, como otra base legal que respalda a esta investigación. El Art. 38 referente a los Riesgos Provenientes del Trabajo indica que:

Los riesgos provenientes del trabajo son de cargo del empleador y cuando, a consecuencia de ellos, el trabajador sufre daño personal, estará en la obligación de indemnizarle de acuerdo con las disposiciones de este Código, siempre que tal beneficio no le sea concedido por el Instituto Ecuatoriano de Seguridad Social (p. 19).

Por su parte el Art. 347 sobre los Riesgos del Trabajo señala que estos son “las eventualidades dañosas a que está sujeto el trabajador, con ocasión o por consecuencia de su actividad. Para los efectos de la responsabilidad del empleador se consideran riesgos del trabajo las enfermedades profesionales y los accidentes” (p. 102). Por otra parte, el Art. 349 relacionado con las Enfermedades Profesionales menciona que “enfermedades profesionales son las afecciones agudas o crónicas causadas de una manera directa por el ejercicio de la profesión o labor que realiza el trabajador y que producen incapacidad” (p. 103).

Capítulo III: Diseño e implementación

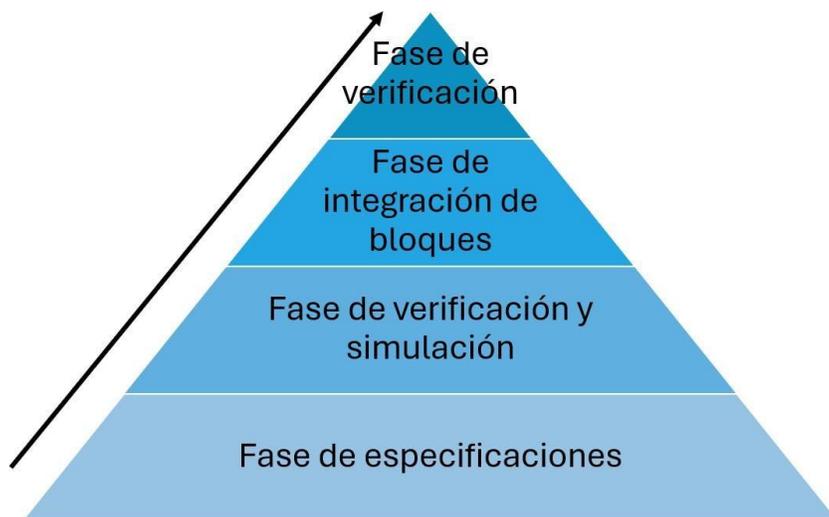
Este capítulo aborda en detalle el proceso de desarrollo del dispositivo, comenzando con la descripción de la metodología establecida y el análisis de la situación actual relacionada con los problemas del túnel carpiano en entornos laborales. A continuación, se presentan los requisitos del sistema, definidos a partir de la identificación de los stakeholders, con el objetivo de cumplir con los criterios establecidos para el diseño, implementación y verificación del dispositivo.

3.1 Metodología

La metodología utilizada en el desarrollo del proyecto es Bottom-Up, la cual facilita la integración gradual de los distintos componentes del sistema hasta conformar un sistema completo y complejo. En la Figura 10 se ilustran las fases que componen esta metodología, las cuales se detallan a continuación.

Figura 10

Fases de la metodología Bottom-Up



Nota. Adaptado de:(Muñoz & Pauline, 2010)

En la fase de especificaciones, se recopila información bibliográfica mediante un enfoque de investigación aplicada, proporcionando una base sólida para el dispositivo electrónico. Esta fase se evidencia en los capítulos I y II.

Durante la fase de verificación y simulación, se definen los requerimientos del sistema y se desarrollan esquemas iniciales para validar la viabilidad del diseño. La fase de integración por bloques se centra en el desarrollo e integración de los componentes individuales para conformar un sistema funcional. Ambas fases se reflejan en el capítulo III.

Finalmente, en la fase de verificación del sistema, se procede a la instalación y a la realización de pruebas para asegurar su correcto funcionamiento y el cumplimiento de los objetivos establecidos. Esta fase se desarrolla en el capítulo IV.

3.2 Situación Actual

En la actualidad, la Universidad Técnica del Norte (UTN) cuenta con una nómina de personal administrativo distribuido en distintas unidades organizacionales. Según datos recientes de la Dirección de Gestión de Talento Humano (DGTH), la universidad dispone de un equipo administrativo compuesto por diversas denominaciones de puestos, encargados de tareas operativas y de gestión dentro de la institución.

El presente estudio se llevará a cabo en las instalaciones de la UTN, ubicadas en la ciudad de Ibarra, provincia de Imbabura, específicamente en las dependencias del Departamento de Seguridad, Gestión y Riesgos, y la Facultad de Ingeniería en Ciencias aplicadas situadas en la Av. 17 de Julio 5-21 y General José María Córdova.

Las pruebas del sistema serán aplicadas a un segmento del personal administrativo, sin distinción de género, que desempeña actividades de oficina en su entorno laboral. Estas tareas, en

su mayoría, consisten en actividades repetitivas y rutinarias, fundamentales para el funcionamiento diario de la institución.

En el marco de este estudio, se desarrollará y evaluará un dispositivo basado en redes neuronales, cuyo objetivo es la identificación temprana de molestias físicas asociadas al síndrome del túnel carpiano. Las pruebas se realizarán con el personal administrativo que labora presencialmente en las instalaciones mencionadas, permitiendo analizar la efectividad del dispositivo en la detección de posibles afecciones.

3.3 Requerimientos

A continuación, basándose en la fase de verificación y simulación de la metodología previamente explicada, se detallan los requerimientos necesarios para la creación del dispositivo. Este análisis abarca desde los requerimientos del sistema y los stakeholders hasta los requerimientos de arquitectura, con el objetivo de facilitar el posterior diseño del sistema.

3.3.1. *Nomenclatura de requerimientos*

Con el objetivo de facilitar la comprensión de los términos utilizados durante el desarrollo, en la tabla 7 se presentan las abreviaturas junto con el significado de cada criterio.

Tabla 7

Nomenclatura de Requerimientos

Abreviatura	Significado
StSR	Requerimientos de Stakeholders
SySR	Requerimientos del Sistema
SrSH	Requerimientos de Arquitectura

Se priorizaron los requerimientos clasificándolos en niveles de alta, media y baja prioridad, para determinar cuáles funciones deben cumplir sin errores y cuáles se pueden omitir sin comprometer el funcionamiento general del sistema.

3.3.2. Identificación de stakeholders

Los stakeholders se integran en todas las etapas del proyecto, asumiendo responsabilidades en distintas áreas del sistema, con el propósito de asegurar que la solución final sea la óptima. La tabla 8 detalla la participación de quienes contribuirán al desarrollo de este proyecto.

Tabla 8

Lista de participantes en el proyecto

Nro.	Lista de Stakeholders	
1	Ing. Luis Suárez	Tutor de tesis
2	Ing. Jaime Michilena	Asesor de tesis
3	Srta. Samanta Pérez	Tesista
4	Lic. Emily Carlosama	Fisioterapeuta
5	Personal UTN	Usuarios

3.3.3. Requerimientos de stakeholders

Los requisitos establecidos por los stakeholders (StSR) se fundamentan en las necesidades operativas y de usuario especificadas en la Tabla 9. Es esencial que el sistema cumpla con cada uno de estos requisitos para asegurar su funcionamiento.

Tabla 9*Requerimientos de Stakeholders*

Requerimientos de Stakeholders				
No.	Requerimientos	Prioridad		
		Alta	Media	Baja
Requerimientos operacionales				
StSR1	El sistema debe reconocer señales de dolor muscular en el antebrazo.	x		
StSR2	El sistema debe monitorizar en tiempo real niveles de dolor muscular en los movimientos de la muñeca.	x		
StSR3	El sistema debe proporcionar el estado del dolor muscular para cada registro de datos nuevo.	x		
StSR4	El dispositivo debe ser eficiente energéticamente.		x	
StSR5	El dispositivo debe apoyar en el diagnóstico del STC.	x		
StSR6	Para el correcto funcionamiento del dispositivo es necesario disponer de un método para recopilar datos que permita implementar los algoritmos de inteligencia artificial.	x		
Requerimientos de Usuario				
StSR7	La interfaz de usuario debe ser intuitiva y fácil de usar.	x		
StSR8	La interfaz de usuario debe mostrar el estado del monitoreo.	x		

StSR9	El dispositivo debe ser fácil de colocar y retirar sin ayuda externa.	x
-------	---	---

3.3.4. *Requerimientos del sistema*

La especificación de requerimientos del sistema (SySR) define las características funcionales que el dispositivo debe tener, las cuales se describen en detalle en la Tabla 10.

Tabla 10

Requerimientos del sistema.

Requerimientos del sistema SySR				
No.	Requerimientos	Prioridad		
		Alta	Media	Baja
Requerimientos Funcionales				
SySR1	El sistema debe tener detección adecuada de los músculos del antebrazo para identificar signos de fatiga o dolor.	x		
SySR2	El sistema debe mostrar en tiempo real el estado en el que se encuentra al momento de realizar el análisis.	x		
SySR3	El sistema debe registrar y almacenar datos para análisis posteriores.		x	
Requerimientos de Interfaz				
SySR4	El sistema debe presentar pines de transmisión y recepción para datos.	x		

SySR5	La interfaz debe mostrar datos de monitoreo de forma clara y accesible.	x
SySR6	El sistema debe emitir reportes de resultados de forma ágil y oportuna.	
Requerimientos de uso		
SySR7	El dispositivo debe ser fácil de configurar sin necesidad de asistencia técnica	x
SySR8	Disponer de un instructivo.	x
SySR9	El dispositivo debe contar con una batería que garantice su funcionamiento.	x

3.3.5. *Requerimientos de arquitectura*

Los requerimientos de arquitectura (SrSH) definen los elementos que compondrán el dispositivo, lo que facilita la selección eficiente y adecuada de los componentes de hardware y software, tal como se detalla en la Tabla 11.

Tabla 11

Requerimientos de Arquitectura

Requerimientos de arquitectura SrSH				
No.	Requerimientos	Prioridad		
		Alta	Media	Baja
Requerimientos de Diseño				
SrSH1	El dispositivo debe ser compacto para garantizar la portabilidad y comodidad del usuario.	x		

SrSH2	El diseño debe permitir la fácil instalación de los sensores en el usuario, de manera que se mantengan en contacto constante con la piel durante el monitoreo.	x
-------	--	---

Requerimientos de Hardware

SrSH4	El microcontrolador debe proporcionar suficiente potencia de procesamiento para analizar las señales de los sensores.	X
SrSH5	El microcontrolador debe disponer de entradas y salidas analógicas y digitales.	x
SrSH6	El microcontrolador debe ser compatible con los sensores y viceversa.	x
SrSH7	El dispositivo debe incluir conectividad inalámbrica para la transmisión de datos.	x
SrSH8	Los sensores deben trabajar con voltajes de hasta 5v para eficiencia en el consumo energético.	x
SrSH9	El sistema debe tener una batería integrada con suficiente capacidad.	x

Requerimientos de Software

SrSH11	El IDE y lenguaje de programación deben ser de código abierto.	x
SrSH12	Compatibilidad de librerías para el uso algoritmos de inteligencia artificial.	x

3.4 Diseño del sistema

En esta sección se desarrolla el diseño del sistema, tomando como base el análisis realizado en los capítulos 2 y 3, donde se identificaron los requerimientos necesarios para el correcto funcionamiento del dispositivo. A continuación, se presenta la fase de diseño, la selección de los dispositivos de hardware y software, junto con la arquitectura del sistema en general que se implementara.

3.4.1. Selección de Hardware y Software del sistema.

Con base en los stakeholders definidos anteriormente, se cumplió con la elección del hardware y software adecuados para asegurar la óptima funcionalidad del dispositivo. Se empleó para la elección tablas comparativas relacionando requisitos anteriormente mencionados con valores de puntuación 1 y 0 (1: Cumple y 0: No cumple), seleccionado así el equipo o software de mayor puntuación.

3.4.2. Selección de hardware

La selección del hardware es una de las más cruciales a la hora de diseñar un dispositivo, ya que este define las capacidades tanto de procesamiento como de comunicación necesarias para los componentes.

3.4.2.1 Selección de microcontrolador

Para la selección del microcontrolador, se consideraron los requisitos establecidos por los stakeholders y se evaluaron diversas alternativas: Arduino UNO, Arduino Mega y Seeeduino V4.3. La tabla 12 presenta el análisis detallado realizado para elegir la opción más adecuada.

Tabla 12

Selección del microcontrolador

Hardware	Requerimientos							Valoración
	StSR2	StSR4	SySR4	SySR5	SrSH5	SrSH6	SrSH7	Total
Arduino UNO	1	0	1	1	0	1	1	5
Arduino Mega	1	0	1	1	1	1	1	6
Seeeduino V4.3	1	1	1	1	1	1	1	7

Según la evaluación presentada en la tabla 12, el microcontrolador Seeeduino V4.3 (ver Figura 11) es la opción ideal para la construcción del dispositivo debido a su compatibilidad con el entorno de programación Arduino, lo cual facilita su implementación. Además, su sistema ajustable de voltaje entre 3.3V y 5V brinda flexibilidad para integrarse con distintos sensores y módulos, mientras que su puerto micro-USB y regulador de bajo consumo permiten una alimentación estable, tanto desde USB como desde fuentes externas.

Figura 11

Seeeduino V4.3



Nota.(Seeed Studio, 2024b)

3.4.2.2 Selección de placa para conectividad inalámbrica

El dispositivo necesita conexión inalámbrica WiFi para comunicarse con la base de datos. Por ello, se seleccionaron las placas Raspberry Pi Pico W, ESP32 y Arduino Nano RP2040 Connect, evaluadas en la tabla 13 de acuerdo con los requisitos del proyecto.

Tabla 13

Selección de placa de conectividad inalámbrica

Hardware	Requerimientos						Valoración
	StSR4	StSR6	SySR3	SySR5	SrSH5	SrSH8	Total
Raspberry Pi pico W	1	0	1	1	1	1	5
Esp32	1	1	1	1	1	1	6
ArduinoNano RP2040 Connect	1	0	1	1	0	1	4

El microcontrolador seleccionado para la conectividad inalámbrica basándonos en la tabla anterior es ESP32 (ver Figura 12). Este dispositivo destaca por su versatilidad y aplicabilidad debido a su capacidad más amplia de SRAM y conexión inalámbrica, al contar con conexiones UART permite las conexiones necesarias para el funcionamiento adecuado del dispositivo. Los entornos de programación que utiliza son de código abierto como MicroPython y C/C++.

Figura 12*ESP32*

Nota.(Espressif Systems, 2025)

3.4.2.3 Selección de sensores.

El dispositivo incorporará sensores de respuesta galvánica de la piel (GSR) y de electromiografía (EMG) para medir la actividad eléctrica generada por los músculos, seleccionados mediante un análisis comparativo basado en la disponibilidad, accesibilidad y fiabilidad de las distintas opciones disponibles en el mercado.

3.4.2.3.1 Sensor de electromiografía (EMG).

El sensor EMG mide la actividad eléctrica generada por los músculos, siendo relevante para el desarrollo del dispositivo debido a que permite monitorear la sobrecarga muscular del antebrazo y muñeca. En la tabla 14, se comparan y evalúan los sensores EMG disponibles para su implementación, tomando en cuenta los requerimientos establecidos para el dispositivo.

Tabla 14*Selección de sensor EMG.*

Hardware	Requerimientos						Valoración
	StR1	StR10	SySR1	SrSH2	SrSH7	SrSH9	
EMG Taidecent	1	1	1	1	1	0	5
EMG MYOWARE	1	0	1	0	1	0	3
Grove - EMG Detector	1	1	1	1	1	1	6

Con los resultados de la tabla 14 se concluye que el sensor EMG ideal para el dispositivo será Grove – EMG Detector (Ver Figura 13). Este sensor es capaz de captar señales musculares mínimas y, posteriormente, las procesa utilizando un segundo amplificador y filtro para mejorar la precisión y claridad de los datos obtenidos.

Figura 13*Grove – EMG sensor*

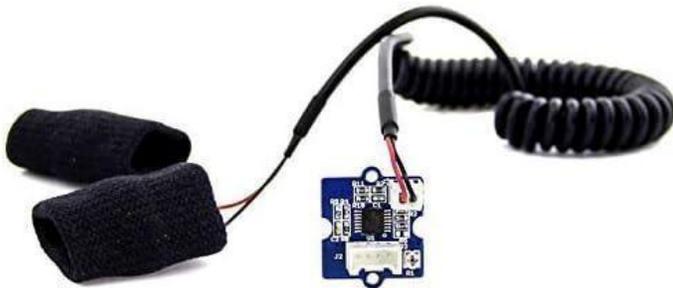
Nota. (Seed Studio, 2025)

3.4.2.3.2 Sensor de respuesta galvánica de la piel (GSR).

El sensor GSR mide la conductancia eléctrica de la piel, este parámetro biométrico resulta de particular interés en el ámbito de la ergonomía y la salud ocupacional, ya que altos niveles de estrés reflejados en un aumento de GSR se asocian con posturas inadecuadas y acciones musculares repetitivas, aumentando el riesgo de STC. Sin embargo, la disponibilidad de sensores GSR en el mercado es limitada, ya que suelen estar integrados en dispositivos como relojes inteligentes y pulseras de monitoreo. El sensor Grove - GSR (ver Figura 14) es una de las pocas opciones independientes en el mercado, elegido por su excelente relación calidad-precio. Además, su diseño es cómodo y permite una fácil colocación y retirada.

Figura 14

Grove – sensor GSR



Nota. (Seeed Studio, 2024a)

3.4.2.4 Selección fuente de alimentación

Dado que todos los sensores integrarán un único dispositivo que será portátil y puede estar en constante movimiento, es imprescindible utilizar una batería que proporcione la energía necesaria para su funcionamiento. Para determinar la batería más adecuada, en la Tabla 15 se

presentan los amperajes de los sensores y microcontroladores empleados. Esta tabla permite evaluar las demandas energéticas y, en consecuencia, seleccionar una batería que garantice un rendimiento óptimo y una autonomía adecuada para el dispositivo.

Tabla 15

Consumo eléctrico del dispositivo

Elemento	En Funcionamiento	Modo Sleep
Seeeduino V4.3	50 mA	50 mA
ESP32-WROOM-32	80 mA	20 mA
Sensor EMG	20 mA	-
Sensor GSR	15 mA	-
Total	165 mA	70 mA

El amperaje necesario para alimentar el dispositivo es de 165 mA, la ecuación (1) permite obtener el consumo de la batería.

$$Consumo = \frac{(T_{cn} * I_{cn}) + (T_{cd} + I_{cd})}{T_{cn} + T_{cd}} \quad (1)$$

Donde:

T_{cn} : Tiempo de consumo en funcionamiento

I_{cn} : Intensidad de consumo de corriente en funcionamiento

T_{cd} : Tiempo de consumo modo sleep

I_{cd} : Intensidad de consumo de corriente en modo sleep

Reemplazando los datos con los siguientes parámetros:

T_{cn} : 10 minutos = 600 segundos

I_{cn} : 165mA

T_{cd} : 10 minutos = 600 segundos

I_{cd} : 70mA

Reemplazando en la ecuación (1) se obtiene el consumo del dispositivo

$$Consumo = \frac{(600 * 165) + (600 * 70)}{600 + 600} = 83.05mA \quad (2)$$

Con base en el consumo energético del dispositivo obtenido, se establece el voltaje y amperaje de la batería a utilizar. Para un consumo de aproximadamente 83.05 mA, se utilizará una batería de **3.7V** con una capacidad de **1000 mAh** o superior. Esto proporcionará una duración adecuada para el funcionamiento continuo del dispositivo, asegurando que todos los componentes reciban la energía necesaria para operar de manera eficiente.

Mediante la siguiente formula (3) se estima el tiempo de duración dividiendo el consumo de batería sobre el consumo del dispositivo.

$$Tiempo\ de\ bateria = \frac{1000mA}{83.05mA} = 12.04\ horas \quad (3)$$

3.4.3. Selección de software

Una vez seleccionado el hardware, y en función de los requerimientos SrSH11 y SrSH12, es imprescindible utilizar entornos de desarrollo integrados (IDE) y lenguajes de programación de

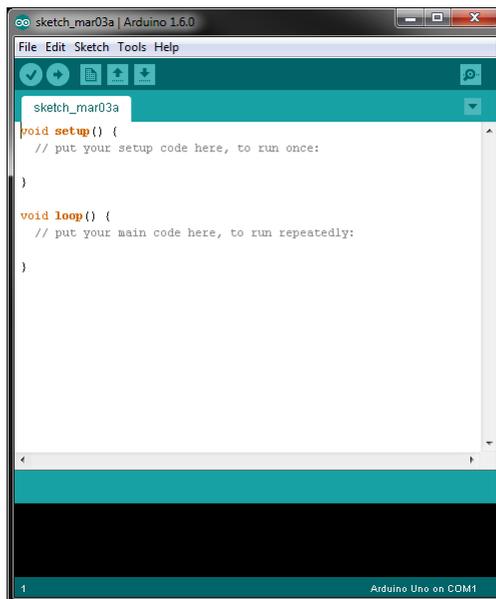
código abierto. Esta elección garantizará la compatibilidad y flexibilidad necesarias para el sistema. De esta manera, se puede asegurar que se están utilizando las mejores opciones disponibles para cumplir con los objetivos del proyecto de manera eficiente.

3.4.3.1 Software para microcontrolador.

El microcontrolador Seeeduion V4.3 y ESP32 utiliza el mismo IDE de Arduino para la programación (Véase la figura 15) basada en lenguaje de programación C++ que cuenta con las librerías necesarias para la configuración de los sensores y facilita la implementación de las comunicaciones necesarias.

Figura 15

IDE Arduino



Nota. (Arduino, 2025)

3.4.3.2 Software de procesamiento.

Para la creación y el entrenamiento de la red neuronal, se empleará el lenguaje de programación Python, acompañado del entorno de distribución de software libre Anaconda. Esta elección es especialmente adecuada debido a que Anaconda es ampliamente utilizado en los campos de las ciencias de datos y el aprendizaje automático. Anaconda facilita la gestión de paquetes y dependencias, permitiendo un entorno de desarrollo integrado y robusto (Anaconda, 2024). Además, proporciona acceso a una amplia gama de bibliotecas y herramientas específicas para la inteligencia artificial y el análisis de datos, tales como TensorFlow, Keras y Scikit-Learn, lo que agiliza el proceso de desarrollo y experimentación.

Figura 16

Anaconda



Nota. (Anaconda, 2024)

3.4.3.3 Software para visualización.

Para la visualización de los datos se desarrollará una interfaz móvil mediante la programación en un entorno de desarrollo de aplicaciones móviles, para esto se realiza una comparación exhaustiva entre dos frameworks de código abierto, React Native y Flutter, con el objetivo de seleccionar el más adecuado para presentar los datos de manera eficiente.

Tabla 16*Selección de framework*

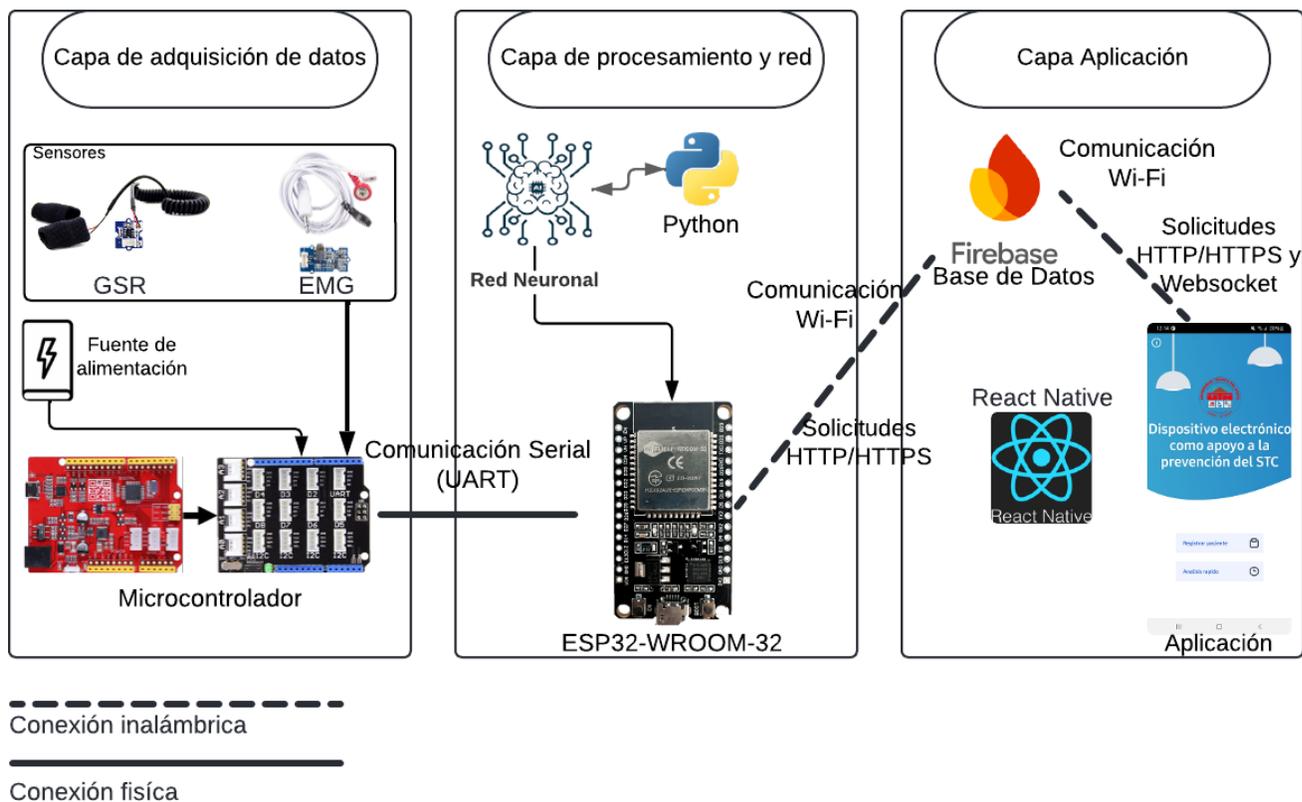
Framework	Requerimientos						Valoración
	StR7	StR8	SySR3	SySR6	SySR7	SySR8	
React Native	1	1	1	1	1	1	6
Flutter	1	1	1	0	1	1	5

Con los resultados de la Tabla 16 se determina que el framework que cumple los requisitos es React Native. Este framework al ser más antiguo relativamente cuenta con una mayor documentación y ofrece la flexibilidad de personalización de usuario necesaria para el dispositivo.

3.4.4. *Arquitectura del sistema*

La Figura 17 presenta la arquitectura en base a los análisis de los stakeholders descritos en detalle en la sección anterior, la misma que se organiza en tres bloques principales: capa de adquisición de datos, capa de procesamiento y red, y capa aplicación.

Figura 17

Arquitectura del sistema

Nota. Autoría propia.

3.4.4.1 Bloque de Adquisición de datos

En el bloque de adquisición de datos se dispone de una fuente de alimentación, sensores, microcontrolador Seeeduino V4.3 junto con su base para los sensores EMG y GSR. Estos componentes son fundamentales para el funcionamiento del dispositivo, los sensores tomarán datos del antebrazo y se enviarán para su procesamiento mediante el protocolo de comunicación serie asíncrono UART además que permitirá al dispositivo ser móvil.

3.4.4.2 Bloque de procesamiento y red

El procesamiento de los datos y red se realizará a través de un microcontrolador ESP32 que cuenta con conexión inalámbrica Wi-Fi, y con la capacidad de integrar una inteligencia artificial de redes neuronales programada en Python para mejorar la predicción de los niveles de dolor. Los resultados obtenidos se enviarán a una base de datos en la nube Firebase para su almacenamiento y publicación utilizando una comunicación inalámbrica Wi-Fi y solicitudes HTTP/HTTPS a la REST API de este mismo.

3.4.4.3 Bloque de Visualización

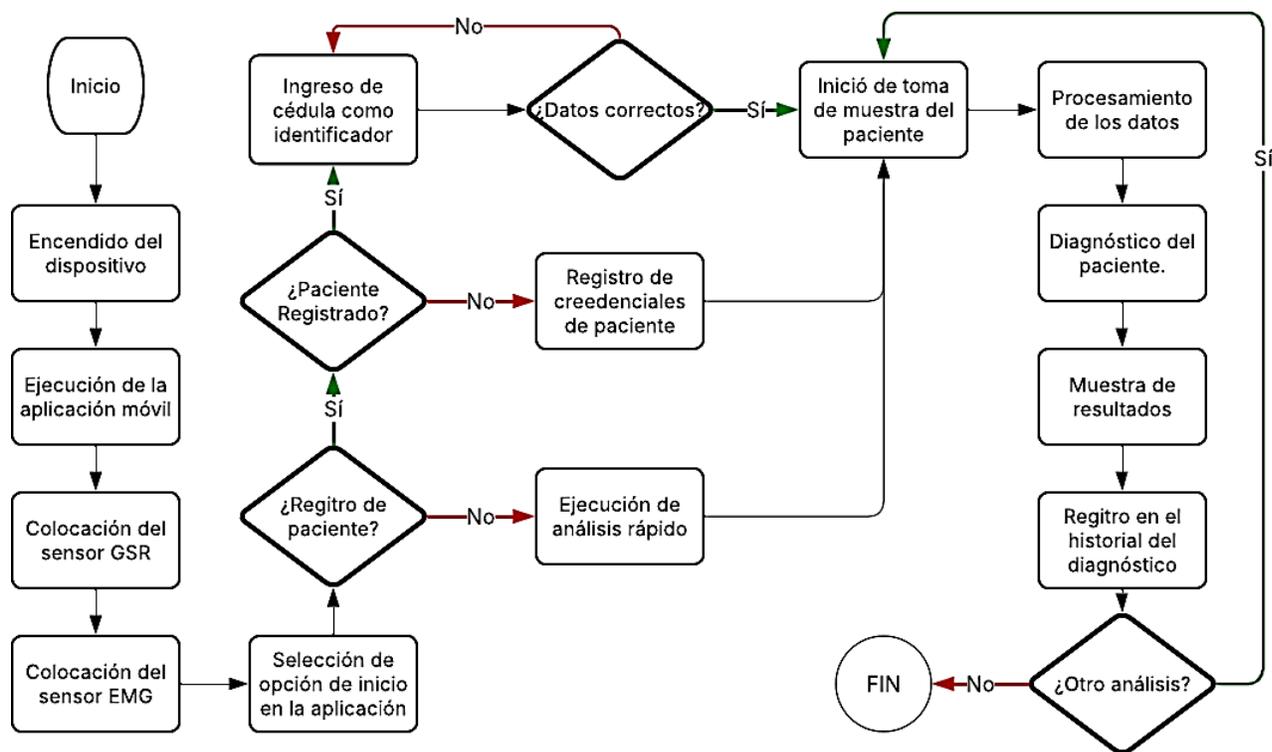
La visualización de los datos se realiza a través de una aplicación móvil desarrollada con el framework de código abierto React Native, este framework permite crear aplicaciones multiplataforma, compatibles tanto con dispositivos Android como iOS, lo que facilita el acceso y la interacción del usuario con los resultados obtenidos por el dispositivo. La aplicación ofrece una interfaz intuitiva y amigable que muestra en tiempo real los niveles de dolor predichos por el sistema, así como gráficos para un mejor seguimiento. La aplicación al contar con Wi-Fi del dispositivo móvil realiza las peticiones HTTP/HTTPS al REST API de Firebase y abre una conexión en tiempo real (WebSocket) para la sincronización con la Base de datos.

3.4.5. Diagrama de flujo del funcionamiento del sistema.

En la Figura 18 se expone de manera detallada la secuencia operativa del sistema, desde la colocación de los sensores en el usuario hasta la obtención de los resultados mediante la aplicación, tal como se ilustra en el siguiente diagrama de flujo.

Figura 18

Diagrama de flujo del funcionamiento del sistema



Nota. Autoría propia.

3.5 Desarrollo del dispositivo

Posteriormente, y en concordancia con la metodología propuesta durante la fase de integración de bloques, se lleva a cabo el desarrollo e integración de los componentes individuales. El dispositivo se compone de dos elementos principales: el módulo de adquisición de datos y el sistema de procesamiento, el cual será responsable de gestionar los datos capturados por los sensores EMG Y GSR y enviarlos a través de una conexión a internet a la base de datos Firebase.

En esta sección, se describe detalladamente la infraestructura física involucrada en la construcción del dispositivo, abordando la conexión de los componentes y su integración para garantizar el correcto funcionamiento del sistema

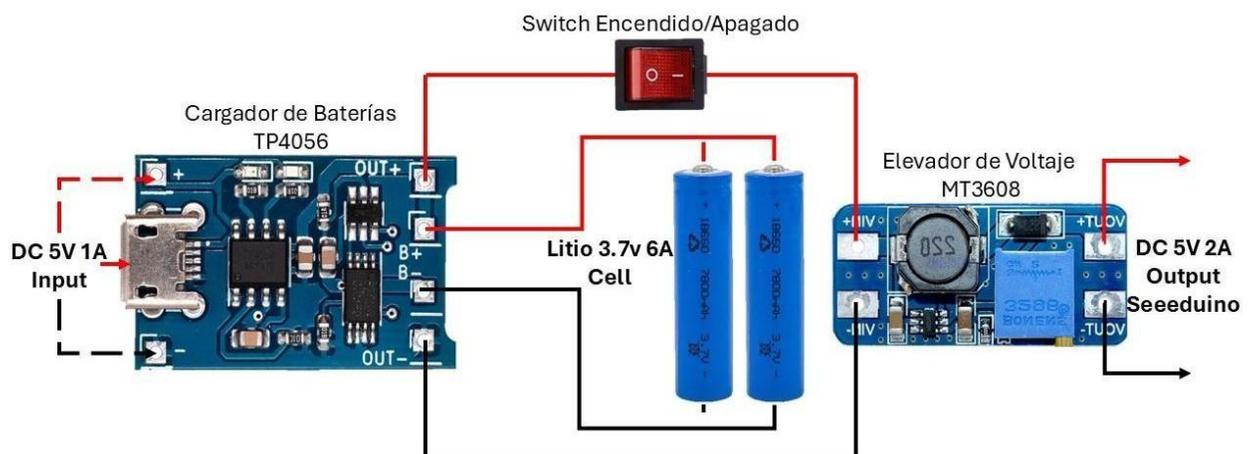
3.5.1. Diagrama del sistema de alimentación.

Para la alimentación del dispositivo (véase figura 19), se utiliza un módulo de carga de baterías TP4056, el cual permite conectar las baterías directamente, este módulo gestiona la carga y detiene la alimentación cuando las baterías alcanzan un nivel de energía bajo, reanudándola una vez que las baterías recuperan un nivel adecuado y se controla el encendido del dispositivo mediante un switch que controla la alimentación general.

Debido a que las baterías solo proporcionan 3.7 V, es necesario elevar el voltaje a 5 V para alimentar correctamente la placa Seeduino V4.3. Para ello, se emplea un elevador de voltaje MT3608, que garantiza el suministro del voltaje y la corriente requeridos para el correcto funcionamiento del dispositivo.

Figura 19

Diagrama de alimentación del dispositivo



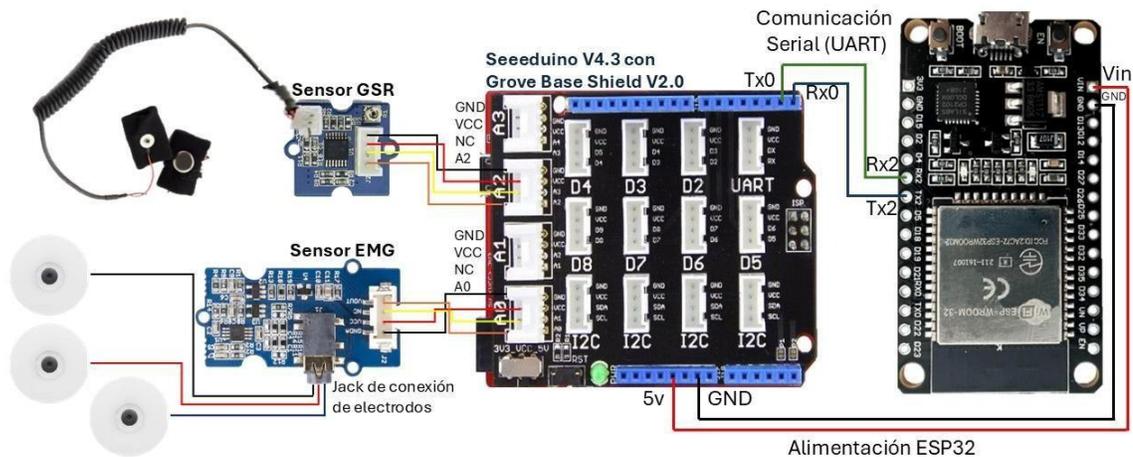
Nota. Autoría propia.

3.5.2 Diagrama de conexión de los módulos de adquisición de datos y procesamiento.

En la figura 20 se presenta el diagrama de conexión de los sensores EMG (electromiografía) y GSR (respuesta galvánica de la piel) hacia el Grove Base Shield la cual facilita la organización y conexión de los cables de los sensores, permitiendo la recolección eficiente de los datos. La placa Seeeduno V4.3 actúa como el controlador principal para la recopilación inicial de los datos que son transmitidos mediante el protocolo I2C hacia la ESP32 misma que realiza el procesamiento de los datos y los envía a la base de datos Firebase en la nube a través de una conexión WiFi.

Figura 20

Diagrama de conexión de los sensores



Nota. Autoría propia.

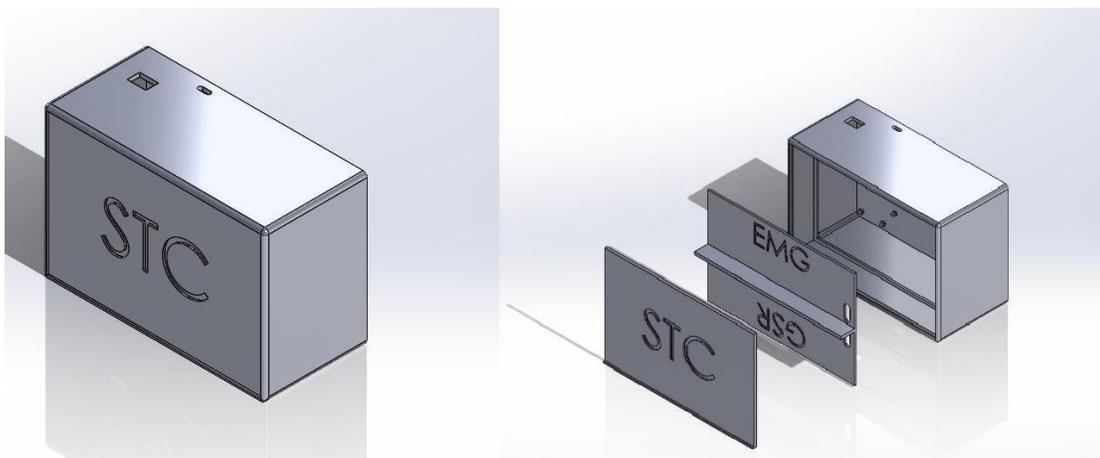
3.6 Desarrollo de case del dispositivo

Para la construcción del case se eligió el material PLA, debido a su resistencia y facilidad de impresión, que permite proteger adecuadamente los componentes del sistema. El diseño interno se dividió para asegurar una correcta distribución y ubicación de los elementos.

Considerando las dimensiones de los componentes, las medidas del case fueron definidas en 14 cm de largo, 6 cm de alto y 6.5 cm de ancho. El diseño del case se desarrolló utilizando el software Fusion360, como se muestra en la figura 21, una vez ensambladas las partes del case, se realizó la incorporación y posicionamiento de cada componente que conforma el sistema.

Figura 21

Case del dispositivo



Nota. Autoría propia.

3.7 Modelo de entrenamiento para el dispositivo

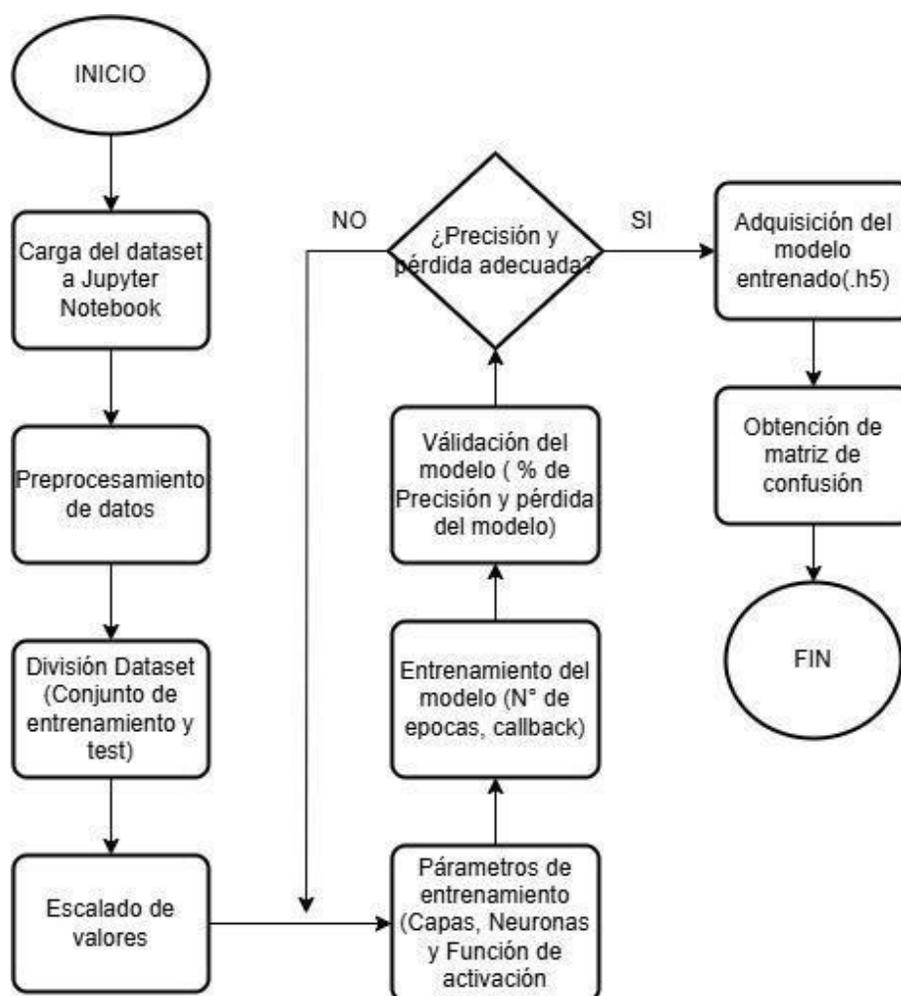
El modelo de entrenamiento para la detección de fatiga muscular asociada al síndrome del túnel carpiano (STC) fue desarrollado utilizando el lenguaje de programación Python, en su versión 3.12.4. El proceso siguió las etapas descritas en la Figura 24, comenzando con la recopilación de datos (dataset), diseñados para clasificar tres niveles de dolor o fatiga.

Una vez recolectados, pasan por un preprocesamiento de visualización y limpieza. A continuación, se realizó la división del conjunto de datos en particiones de entrenamiento y prueba. Posteriormente, los datos fueron escalados para mejorar el rendimiento del modelo.

En la fase de entrenamiento, se utilizaron métricas y funciones específicas para garantizar la fiabilidad del modelo. Finalmente, el modelo entrenado permitió realizar predicciones precisas y visualizar los resultados en tiempo real, facilitando su aplicación práctica en la prevención del STC en entornos laborales.

Figura 22

Proceso de generación del modelo.



Nota. Autoría propia.

3.7.1. Creación del Dataset

Para la creación del dataset, se llevó a cabo el registro de datos relacionados con el dolor muscular utilizando los sensores especializados. La recopilación de información se realizó con usuarios que, de manera habitual, ejecutan movimientos repetitivos o presentan molestias asociadas al síndrome del túnel carpiano (STC). Los movimientos y posturas adecuados fueron definidos y supervisados conforme al criterio técnico-médico de una experta en fisioterapia, garantizando la correcta ejecución durante el proceso de captura.

Con base en esta supervisión técnica, los datos fueron clasificados en tres categorías: relajado o normal, tenso y astenia o agotado. Toda la información recopilada fue almacenada en una tarjeta microSD, y el proceso de registro tuvo una duración de cinco semanas, durante las cuales se recolectaron un total de 300 muestras, representando diferentes estados musculares.

Con el dataset preparado, se utilizó Anaconda Jupyter para llevar a cabo el procesamiento y análisis de los datos de forma local, aprovechando las capacidades del equipo, en una experiencia similar a la que ofrece Google Colab.

El primer paso consistió en la importación de librerías esenciales presentadas en la tabla 17, para facilitar el manejo, visualización y análisis de los datos. Posteriormente, se cargó el dataset obtenido, estableciendo la base para las siguientes etapas del desarrollo del modelo.

Tabla 17*Librerías utilizadas*

Nombre de la Librería	Función
NumPy	Cálculos numéricos
Pandas	Manipulación y estructuras de datos
Seaborn	Visualización conjunta con Matplotlib
Matplotlib	Visualización
Sklearn	División y evaluación del modelo
Tensorflow	Modelado de la red neuronal

3.7.2. Preprocesamiento

La parte del preprocesamiento es una de las más cruciales al momento de la creación del modelo de red neuronal porque permite conocer los datos tratados. De ese modo los datos recolectados por los sensores no requieren de una limpieza o interpolación debido a que no se tienen una presencia de datos nulos por lo que se realizara una visualización de gráficos asociados a las variables tratadas.

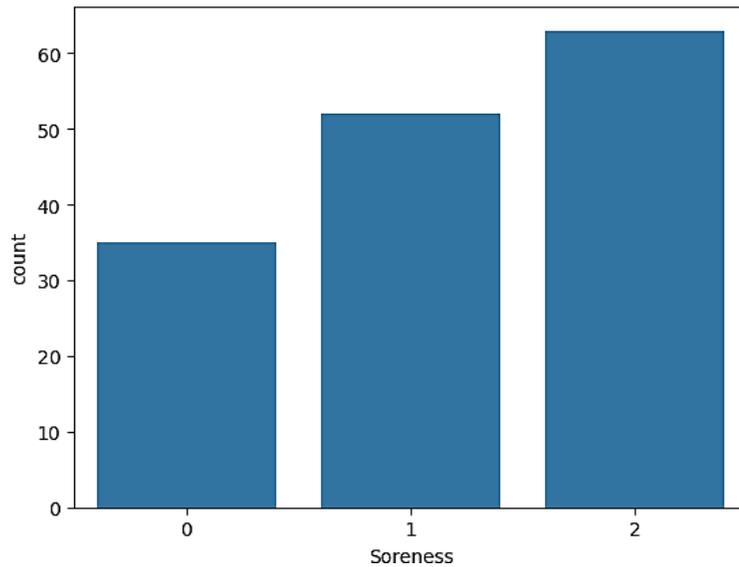
Lo primero a analizar es verificar si existe una proporción equilibrada de los valores de la variable objetivo dolor para obtener un buen modelado posterior. En este caso nuestra variable objetivo es la de dolor, teniendo así una mayor presencia de valores de fatiga, pero sin un gran desbalanceo en general como se muestra en la figura 25.

Figura 23

Proporción equilibrada de los valores de la variable objetivo dolor

```
#Verificación de proporción equilibrada entre los diferentes valores de la variable objetivo dolor
sns.countplot(x='Soreness', data=df)
```

<Axes: xlabel='Soreness', ylabel='count'>

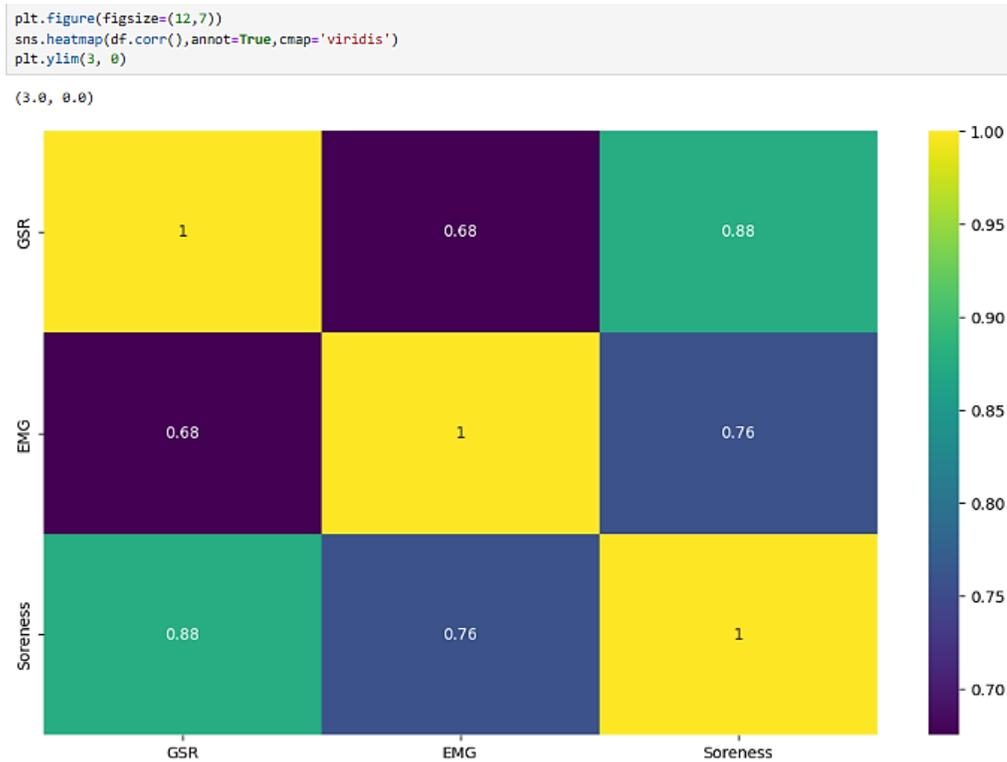


Nota. Autoría propia.

En la figura 26 se presenta un gráfico de verificación del nivel de correlación de las variables con la variable objetivo, en caso una de las variables no tendría un nivel de correlación adecuado deberíamos proceder con la eliminación de dicha variable que no aportara significativamente al modelo, en este caso se tiene un buen nivel de correlación de las variables destacando un mayor nivel el del sensor GSR.

Figura 24

Nivel de correlación de las variables



Nota. Autoría propia.

3.7.3. División de datos de entrenamiento y prueba.

A partir de este punto, se utiliza la biblioteca Scikit-learn para dividir los datos en un conjunto de entrenamiento y un conjunto de prueba.

Como se muestra en la figura 27, el primer paso consiste en separar las variables predictoras (X) de la variable objetivo (Y). Dado que se trata de una tarea de predicción multiclase, es necesario transformar la variable objetivo a formato One-Hot Encoding, una técnica que convierte las categorías en vectores binarios, donde cada categoría se representa como un vector con un único valor "1" y el resto en "0" para facilitar el procesamiento por el modelo.

La división de los datos se realiza asignando el 75% al conjunto de entrenamiento y el 25% al conjunto de prueba, asegurando así un balance adecuado para el entrenamiento y evaluación del modelo.

Figura 25

Division de datos entrenamiento y prueba

```
# Separar variables predictoras (X) y variable objetivo (y)
X = df.drop('Soreness', axis=1).values
y = pd.get_dummies(df['Soreness']).values # Convertir a formato One-Hot

# Dividir datos en entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=101)
```

Nota. Autoría propia.

3.7.4. Escalado

Una vez realizada la división de los datos, se procede a escalarlos para garantizar que no existan diferencias significativas en los órdenes de magnitud entre las variables. Esto es crucial, ya que valores con rangos muy distintos podrían influir de manera desproporcionada en el entrenamiento de la red neuronal, asignando mayor peso a ciertas variables.

Para evitar este problema, se aplica un proceso de escalado que normaliza las variables, ajustándolas a un rango entre 0 y 1.

Figura 26

Normalizacion de variables

```
# Escalar datos (normalización)
scaler = MinMaxScaler() #Se inicializa el escalador
X_train = scaler.fit_transform(X_train) #Ajuste y transformación del conjunto de entrenamiento
X_test = scaler.transform(X_test) #Ajuste del conjunto de prueba con los mismos parámetros
```

Nota. Autoría propia.

3.7.5. Creación del modelo.

La creación del modelo se realiza utilizando una arquitectura de red neuronal secuencial proporcionada por la biblioteca Keras. Esta estructura define las capas de manera ordenada, permitiendo diseñar y ajustar fácilmente el flujo de datos a través de la red.

El modelo comienza con una capa de entrada, que incluye una densamente conectada (*Dense*) con 64 unidades y la función de activación ReLU, ideal para capturar patrones no lineales en los datos. Además, se aplica una capa de Dropout con una tasa del 30% para reducir el sobreajuste, apagando aleatoriamente un porcentaje de neuronas durante el entrenamiento.

A continuación, se incluye una capa oculta con 32 unidades, también con la activación ReLU y un Dropout adicional del 30%, reforzando la capacidad de generalización del modelo.

Finalmente, se configura una capa de salida con tantas unidades como clases tiene la variable objetivo (Y), utilizando la función de activación Softmax. Esta función convierte los resultados en probabilidades, lo que es esencial para tareas como es el caso de clasificación multiclase.

Figura 27

Capas de la red neuronal

```
# Configuración del modelo
model = Sequential()

# Capa de entrada
model.add(Dense(units=64, activation='relu', input_shape=(X_train.shape[1],)))
model.add(Dropout(0.3)) # Añadir regularización con Dropout

# Capa oculta
model.add(Dense(units=32, activation='relu'))
model.add(Dropout(0.3))

# Capa de salida
model.add(Dense(units=y_train.shape[1], activation='softmax'))
```

Nota. Autoría propia.

3.7.6. Entrenamiento.

El entrenamiento del modelo implica ajustar los pesos de la red neuronal para optimizar su capacidad de predicción. Previamente, se realiza la compilación mediante la función `compile`, especificando la función de pérdida `categorical_crossentropy`, adecuada para clasificación multiclase, y el optimizador Adam con una tasa de aprendizaje de 0.001. Asimismo, se define la métrica `categorical_accuracy` para evaluar la precisión en cada iteración.

Para prevenir sobreajustes y garantizar un entrenamiento eficiente, se emplea el callback `EarlyStopping`, que detiene el proceso si la pérdida en el conjunto de validación (`val_loss`) no mejora tras 20 épocas consecutivas, restaurando los mejores pesos alcanzados. Finalmente, el método `fit` entrena el modelo durante un máximo de 300 épocas, con un tamaño de lote de 32, validando el rendimiento en cada iteración con los datos de prueba. Este enfoque equilibra la eficiencia computacional con la robustez del modelo.

Figura 28

Ajustes para evitar colapsos de épocas

```
# Compilación del modelo
model.compile(loss='categorical_crossentropy',
              optimizer=Adam(learning_rate=0.001), # Ajustar learning rate si es necesario
              metrics=['categorical_accuracy'])

# Callbacks (early stopping)
early_stop = EarlyStopping(monitor='val_loss', mode='min', patience=20, verbose=1, restore_best_weights=True)

# Entrenamiento del modelo
history = model.fit(X_train,
                  y_train,
                  validation_data=(X_test, y_test),
                  epochs=300,
                  batch_size=32, # Reducir tamaño del batch para datasets pequeños
                  verbose=1,
                  callbacks=[early_stop])
```

Nota. Autoría propia.

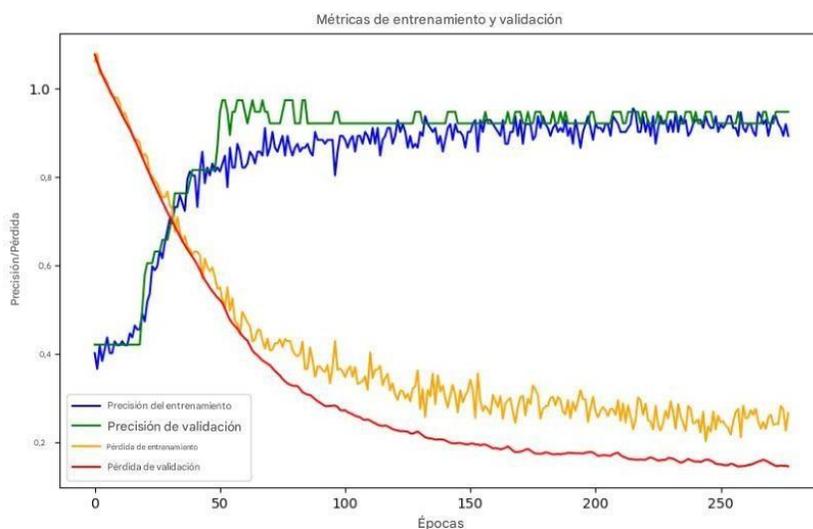
3.7.7. Evaluación y predicción.

Se evalúa el modelo utilizando el historial de entrenamiento almacenado, generamos unos gráficos que comparan las métricas de precisión (categorical_accuracy) y pérdida (loss) entre los conjuntos de entrenamiento y validación a lo largo de las épocas.

En la figura 31, se muestra las métricas de precisión y pérdida tanto para el conjunto de entrenamiento como para el conjunto de validación a lo largo de las épocas del entrenamiento del modelo. El modelo presenta un rendimiento robusto y consistente tanto en los datos de entrenamiento como de validación, lo que sugiere que está bien ajustado para la predicción multiclase. Esto indica que el proceso de preprocesamiento, diseño del modelo y configuración del entrenamiento fueron efectivos.

Figura 29

Métricas de entrenamiento y validación



Nota. Autoría propia.

Finalmente, se realiza una evaluación final del modelo para medir su rendimiento en el conjunto de prueba. En esta etapa, se calcula el nivel de pérdida y precisión, obteniendo un valor

de pérdida de 0.145 y una precisión del 94.73%, como se observa en la figura 32. Estos resultados reflejan un desempeño robusto del modelo, con una pérdida mínima y un alto porcentaje de precisión, lo que demuestra su capacidad para generalizar adecuadamente en datos no vistos.

Figura 30

Prediccion final del modelo

```
# Evaluación final
test_loss, test_accuracy = model.evaluate(X_test, y_test, verbose=0)
print(f"Test Loss: {test_loss}")
print(f"Test Accuracy: {test_accuracy}")

Test Loss: 0.14535948634147644
Test Accuracy: 0.9473684430122375
```

Nota. Autoría propia.

3.8 Conversión del modelo a una matriz C.

Una vez evaluado el modelo y guardado como formato TensorFlow Keras H5 se lo puede volver a utilizar para realizar predicciones en otros análisis, por el contrario, este formato no es legible para el ESP32 y consume gran cantidad de recursos como el tamaño y consumo de energía.

En consecuencia, se requiere utilizar una versión experimental conocida como TensorFlow Lite que ejecuta modelos de aprendizaje automático en microcontroladores y otros dispositivos usando solo algunos kilobytes de memoria (*TensorFlow Lite Para Microcontroladores*, n.d.). El ESP32 tiene compatibilidad con las arquitecturas probadas, por lo que existe la disponibilidad de las bibliotecas de Arduino de código abierto.

En este sentido, resulta imprescindible realizar la conversión del modelo almacenado en formato H5 a un modelo en formato TensorFlow Lite (.tflite). Posteriormente, dicho modelo debe ser transformado a un arreglo de bytes C para ser almacenado en la memoria de los programas en

un archivo .h, siguiendo el procedimiento de conversión establecido como óptimo para garantizar su correcta implementación y ejecución.

Como se muestra en la figura 33, la biblioteca TensorFlow incorpora una función integrada para la conversión a su versión Lite. No obstante, para la conversión a una matriz en lenguaje C, existen diversas herramientas estándar proporcionadas en la documentación oficial de TensorFlow Lite. En este proyecto, se optó por utilizar la conversión básica implementando una función, reconocida por su estabilidad, aunque no alcanza el mismo nivel de optimización en términos de ahorro de memoria en comparación con otros métodos disponibles.

Figura 31

Modelo transformado a una matriz C

```

from hex_to_c_array import hex_to_c_array
# Ruta del archivo del modelo guardado
path = "model/modelo_pred_mousef"
# Cargar el modelo de Keras del modelo guardado en formato h5
model = tf.keras.models.load_model(path + ".h5")
# Conversión del modelo a TensorFlow Lite
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
# Guardado del modelo convertido en un archivo .tflite
with open(path + '.tflite', 'wb') as f:
    f.write(tflite_model)
print("\r\nTensorFlow Keras H5 convertido en un modelo TensorFlow Lite!\r\n")
# Guardado del array de bytes en un archivo .h usando la función importada
model_name = "model_dispositivo"
with open("model/{}.h".format(model_name), 'w') as file:
    file.write(hex_to_c_array(tflite_model, model_name))
print("\r\n Modelo TensorFlow Lite convertido a un archivo de cabecera C (.h)!\r\n")

```

Nota. Autoría propia.

3.9 Programación de módulo de adquisición de datos.

Los sensores implementados al formar parte del ecosistema Grove de Seed Studio cuentan con un sistema de conectores estandarizado que reduce la complejidad en el cableado y

configuración. Estos sensores han sido desarrollados para garantizar su compatibilidad con la plataforma Arduino, eliminando la necesidad de utilizar bibliotecas externas para su correcto funcionamiento.

La programación del Seeeduno V4.3 es equivalente a la del Arduino UNO, ya que ambos se basan en el mismo microcontrolador ATmega328P. Este microcontrolador, en conjunto con el Grove Base Shield, se encarga de recolectar los datos analógicos previamente mencionados. En este proceso, se definen inicialmente los pines y las variables destinadas al almacenamiento de datos como se muestra en la figura 34, además de configurarse dentro del void setup() la velocidad de comunicación serial que se establecerá para la comunicación posterior.

Figura 32

Definición de pines y variables

```
#define GSR A0
#define EMG A2
int gsr_value, emg_value;

void setup() {
  // Inicialización de comunicación serial
  Serial.begin(9600);
  delay(2000);
}
```

Nota. Autoría propia.

Antes de proceder a definir el núcleo del programa, que se ejecutará en un bucle continuo, se crean funciones separadas para la lectura de los sensores. Este enfoque permite obtener una lectura más precisa, facilita la comprensión del código y mejora su mantenimiento a largo plazo.

Para el sensor GSR se calcula una media de las 10 últimas mediciones que permite minimizar los errores o fluctuaciones en las lecturas como lo muestra la figura 35, mejorando la estabilidad de los datos obtenidos y establecemos que se pueda colocar un valor de calibración proporcionado al llamar a la función.

Figura 33

Funciones para lectura de sensor GSR

```
void get_GSR_data(int calibration) {
    long sum = 0;
    // Calcula la media de las diez últimas mediciones
    // del sensor GSR para eliminar el fallo.
    for (int i = 0; i < 10; i++) {
        sum += analogRead(GSR);
        delay(5);
    }
    gsr_value = (sum / 10) - calibration;
}
```

Nota. Autoría propia.

En la figura 36 para el sensor EMG se realiza un proceso similar, pero con las últimas 32 mediciones y con una división por equivalente con el operador de desplazamiento a la derecha. Se agrega a las funciones pequeños retardos para garantizar frecuencias de muestreo adecuadas.

Figura 34

Funciones para lectura de sensor EMG

```
void get_EMG_data() {
    long sum = 0;
    // Evalúa la suma de las 32 últimas mediciones del sensor EMG.
    for (int i = 0; i < 32; i++) {
        sum += analogRead(EMG);
    }
    // Desplazamiento de la suma por cinco
    // operador de desplazamiento a la derecha (>>) para obtener el valor EMG.
    emg_value = sum >> 5;
    delay(10);
}
```

Nota. Autoría propia.

A continuación se muestra en la figura 37 una función principal de bucle se lleva a cabo un monitoreo constante del puerto serial para verificar la disponibilidad de nuevos datos. Si se detecta la presencia de un dato, este se compara con el mensaje de activación predefinido. En caso de coincidencia, se ejecutan las funciones correspondientes para actualizar las variables asociadas a los datos de los sensores. Finalmente, los valores procesados se envían a través del puerto serial para su comunicación.

Figura 35

Monitoreo del puerto serial

```
void loop() {
  if (Serial.available()) {
    String mensaje = Serial.readStringUntil('\n');
    if (mensaje == "1") {
      get_EMG_data();
      get_GSR_data(3);
      Serial.print("EMG: ");
      Serial.print(emg_value);
      Serial.print("\tGSR: ");
      Serial.println(gsr_value);
      delay(2000);
    }
  }
  delay(10);
}
```

Nota. Autoría propia.

3.10 Configuración de Firebase Realtime Database.

Para el funcionamiento del procesamiento de los datos que se detalla posteriormente, se configura los nodos y campos necesarios para el trabajo como puntos de interacción entre el

módulo ESP32 y la aplicación móvil. Al ser una base de datos NoSQL no se usa tablas ni filas siguiendo el modelo basado en JSON como se detalla en la tabla 18.

Tabla 18

Configuración de nodos

Nodo	Campo	Tipo de Dato	Descripción
Análisis	EMG	Entero (int)	Actividad electromiografía.
Análisis	GSR	Entero (int)	Respuesta galvánica de la piel.
Análisis	Inicio	Booleano (0/1)	Indica si el análisis está en curso.
Análisis	Resultado	Entero (int)	Resultado del análisis.
user	email	String	Dirección de correo electrónico del usuario.
user	nombreApellido	String	Nombre completo del usuario.
user	teléfono	String	Número de teléfono del usuario.

3.11 Programación de módulo de procesamiento.

Las bibliotecas de TensorFlow Lite para Arduino IDE no están disponibles de forma predeterminada, por lo que es necesario proceder con su instalación manual. Este proceso debe

realizarse siguiendo las directrices proporcionadas en la documentación oficial disponible en el repositorio de GitHub correspondiente.

Durante las pruebas realizadas con la biblioteca, se identificaron errores de compilación en Arduino IDE al utilizar el sistema operativo Windows. Ante esta situación, se decidió emplear Ubuntu 18.04, en el cual se verificó la compatibilidad con la biblioteca requerida, permitiendo continuar con el desarrollo sin inconvenientes.

En la figura 38, se implementa las librerías de comunicación con Firebase y la integración de TensorFlow Lite, donde se incluye los módulos que proporcionan las herramientas requeridas para la carga, interpretación y ejecución del modelo de aprendizaje realizado.

Figura 36

Librerías de comunicación con Firebase

```
// Biblioteca de Comunicacion con Firebase
#include <Firebase.h>
// Importacion de módulos TensorFlow necesarios.
#include <TensorFlowLite.h>
#include "tensorflow/lite/experimental/micro/kernels/all_ops_resolver.h"
#include "tensorflow/lite/experimental/micro/micro_error_reporter.h"
#include "tensorflow/lite/experimental/micro/micro_interpreter.h"
#include "tensorflow/lite/schema/schema_generated.h"
#include "tensorflow/lite/version.h"
```

Nota. Autoría propia.

En un archivo externo adicional se almacenan las credenciales necesarias para establecer la conexión con la red Wi-Fi, así como la URL correspondiente a la base de datos alojada en Firebase. Es fundamental destacar que, en función de las configuraciones de seguridad implementadas en la plataforma, se deben especificar cuidadosamente los parámetros dentro del

archivo, siguiendo las directrices establecidas en la documentación de la librería. Se añade la matriz C generada de la red neuronal entrenada y las variables donde se guardarán los datos, todo esto se evidencia dentro de la figura 39.

Figura 37

Credenciales de la Base de Datos y Wifi

```
//Importacion de Archivos de Credenciales de la BD y Wifi
#include "secrets.h"
Firebase fb(REFERENCE_URL);
// Importacion de libreria tensorflowLite matrix C (.h)
#include "model_dispositivo.h"
//Variables para guardar los datos
float gsr;
float emg;
```

Nota. Autoría propia.

Se establece los componentes globales necesarios de ejecución de TensorFlow Lite en el programa. Se define un espacio de nombres anónimo para encapsular las variables y evitar conflictos. Dentro de este espacio como se muestra en la figura 40, se declaran los siguientes objetos: un reportador de errores (`error_reporter`), el modelo de TensorFlow Lite (`model`), el intérprete para la ejecución del modelo (`interpreter`), y los tensores de entrada y salida (`model_input` y `model_output`). Además, se reserva un bloque de memoria (`tensor_arena`) de tamaño fijo, definido por `kTensorArenaSize` (15 KB), destinado a almacenar los tensores y cálculos intermedios requeridos del modelo.

Figura 38

Componentes globales Tensorflow

```
// TensorFlow Lite globals
namespace {
tflite::ErrorReporter* error_reporter = nullptr;
const tflite::Model* model = nullptr;
tflite::MicroInterpreter* interpreter = nullptr;
TfLiteTensor* model_input = nullptr;
TfLiteTensor* model_output = nullptr;
// Memoria para el modelo
constexpr int kTensorArenaSize = 15 * 1024;
uint8_t tensor_arena[kTensorArenaSize];
} // namespace
```

Nota. Autoría propia.

En el contexto de la función de inicialización (*setup*), se establece la configuración de la comunicación serial, cuyo propósito principal es permitir la transmisión de datos hacia el dispositivo Seeeduno V4.3. Para garantizar la integridad del proceso y mitigar posibles errores de solapamiento durante la inicialización del módulo ESP32, se implementa una comunicación serial secundaria. Asimismo, se lleva a cabo la inicialización de la conexión Wi-Fi y la configuración de los componentes necesarios para el uso de TensorFlow Lite.

En este proceso en la figura 41, se crea un objeto estático para la gestión de reportes de error, se carga el modelo correspondiente y se verifica su compatibilidad con el esquema actual. Posteriormente, se inicializan las operaciones compatibles mediante un objeto estático de tipo *AllOpsResolver*. A continuación, se instancia un intérprete estático que utiliza el modelo cargado y un área de memoria previamente definida. Este intérprete se encarga de realizar la asignación de memoria para los tensores, así como de configurar los tensores de entrada y salida necesarios para el procesamiento, asegurando así la operatividad del sistema.

Figura 39

Configuración de la comunicación serial

```

void setup() {
  Serial2.begin(9600, SERIAL_8N1, 16, 17);
  delay(500);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
  // Configuración de TensorFlow Lite
  static tflite::MicroErrorReporter micro_error_reporter;
  error_reporter = &micro_error_reporter;
  model = tflite::GetModel(model_dispositivo);
  if (model->version() != TFLITE_SCHEMA_VERSION) {
    error_reporter->Report("La versión del modelo no coincide con el esquema");
    while (1);
  }
  static tflite::ops::micro::AllOpsResolver resolver;
  static tflite::MicroInterpreter static_interpreter(
    model, resolver, tensor_arena, kTensorArenaSize, error_reporter);
  interpreter = &static_interpreter;
  TfLiteStatus allocate_status = interpreter->AllocateTensors();
  if (allocate_status != kTfLiteOk) {
    error_reporter->Report("AllocateTensors() failed");
    while (1);
  }
  model_input = interpreter->input(0); model_output = interpreter->output(0);
}

```

Nota. Autoría propia.

Se crea una función donde se realizará toda la lógica de lectura de los datos de los sensores enviados por la comunicación serial entre el ESP32 y Seeeduno V4.3. Se tiene una sentencia de control que se mantiene monitorizando la comunicación y en caso de que exista datos comienza a parsear los datos para que sean legibles por modelo de red neuronal como se muestra en la figura 42.

Figura 40

Función para la comunicación serial entre el ESP32 y Seeeduno V4.3

```
void run_test_case() {
  if (Serial2.available()) {
    // Lectura de datos del Seeeduno hasta encontrar un salto de línea
    String mensaje = Serial2.readStringUntil('\n');
    // Parsear datos
    if (mensaje.startsWith("EMG:")) {
      // Buscar el tabulador (\t) que separa EMG y GSR
      int separador = mensaje.indexOf('\t');
      if (separador > 0) {
        // Datos GSR y EMG
        // Extracción valor de EMG
        String EMG_str = mensaje.substring(4, separador);
        // Extraer el valor de GSR
        String GSR_str = mensaje.substring(separador + 5);
        // Conversión de cadenas a enteros
        EMG = EMG_str.toInt();
        GSR = GSR_str.toInt();
        // Actualización de datos a la BD
        fb.setInt("Análisis/GSR", gsr);
        fb.setInt("Análisis/EMG", emg);
      }
    }
  }
}
```

Nota. Autoría propia.

La función continúa con el proceso de normalización de los datos, escalándolos mediante la división entre 1000 para ajustarlos al rango esperado por el modelo. Estos valores normalizados se asignan a las posiciones correspondientes en la estructura de datos del modelo, asegurando su compatibilidad con los requisitos de las entradas. Posteriormente, se ejecuta la inferencia del modelo, el cual evalúa los datos procesados y genera una predicción. Dicha predicción es interpretada en una sentencia de control con un umbral del 75% como muestra la figura 43, y el resultado correspondiente se registra en la base de datos. Finalmente, para indicar la conclusión del proceso, el valor asociado al inicio de la función se restablece a 0 en la base de datos, garantizando la preparación del sistema para su próxima ejecución.

Figura 41

Normalizacion de los datos

```

// Normalizacion de datos y pasarlos al modelo
model_input->data.f[0] = gsr / 1000.0;
model_input->data.f[1] = emg / 1000.0;
// Ejecutar inferencia
TfLiteStatus invoke_status = interpreter->Invoke();
if (invoke_status != kTfLiteOk) {
    error_reporter->Report("Invoke falló en la entrada dada.");
    return;
}
// Lectura de predicción del modelo
for(int i = 0; i<3; i++){
    if(model_output->data.f[i] >= 0.75){
        fb.setInt("Análisis/Resultado", i);
    }
    delay(2000);
    fb.setInt("Análisis/Inicio", 0);
}

```

Nota. Autoría propia.

En la figura 44 en la función de bucle principal (void loop()), se supervisa continuamente el estado del sistema mediante la verificación del valor de la clave "Análisis/Inicio" en la base de datos. Si este valor es igual a 1, se envía una señal al dispositivo Seeeduino a través del puerto serial secundario, lo que activa la transferencia de datos para su procesamiento. Posteriormente, se invoca la función run_test_case(), encargada de realizar la inferencia sobre los datos recibidos. Finalmente, se introduce un retraso de 50 milisegundos para garantizar la estabilidad del sistema durante cada iteración del bucle.

Figura 42

Predicción de datos

```
void loop() {  
  int Inicio = fb.getInt("Análisis/Inicio");  
  if (Inicio == 1) {  
    Serial2.print(1);  
    // Realización de una predicción basada  
    //en los datos llegados por comunicacion Serial  
    run_test_case();  
  }  
  delay(50);  
}
```

Nota. Autoría propia.

3.12 Desarrollo de la aplicación

En esta sección se presenta el proceso de diseño y desarrollo de la aplicación móvil, detallando las herramientas utilizadas. Se describe la arquitectura del sistema, la estructura del código y los principales componentes implementados, con el objetivo de garantizar una aplicación eficiente, escalable y mantenible.

El desarrollo se llevó a cabo utilizando React Native, un framework basado en JavaScript que permite la creación de aplicaciones móviles multiplataforma con un único código base. Se seleccionó esta tecnología debido a su flexibilidad, eficiencia en el rendimiento y la posibilidad de reutilización de componentes.

A continuación, se describen las fases clave del desarrollo, incluyendo el diseño de la interfaz de usuario, la implementación de la lógica y la integración con servicios externos como la base de datos en la nube Firebase.

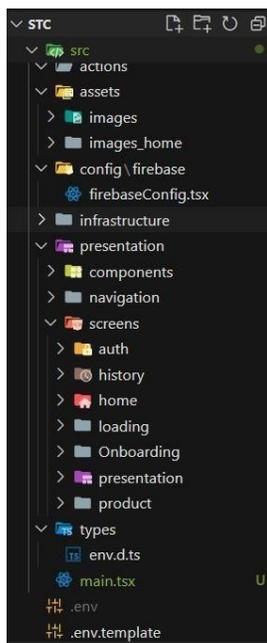
3.12.1. Estructura de la aplicación.

La organización del código y la estructura de carpetas, así como la selección de dependencias y librerías, se fundamentaron en los lineamientos arquitectónicos propuestos por los principios SOLID, complementados con el enfoque de Arquitectura Limpia. Este último, según Martin (2017), promueve la separación de responsabilidades y la independencia entre capas, lo cual permite desarrollar sistemas robustos, escalables y de fácil mantenimiento. La Arquitectura Limpia enfatiza la idea de que la lógica debe permanecer aislada de los detalles de implementación, como frameworks y tecnologías específicas, garantizando así la adaptabilidad y testeo del sistema.

Como se ilustra en la Figura 45, la disposición de las carpetas refleja la segmentación de las distintas capas funcionales y de presentación, asegurando una separación clara de responsabilidades y facilitando la integración de nuevos módulos en futuras iteraciones.

Figura 43

Esquema de la estructura de carpetas de la aplicación



Nota. Autoría propia.

3.12.2. Funcionalidades Implementadas

3.12.2.1 Pantalla principal

En la figura 46, se presenta la interfaz principal de la aplicación móvil diseñada con un enfoque intuitivo y funcional, facilitando el acceso a las principales opciones del sistema. En la parte superior, se encuentra información principal del dispositivo con título central, "*Dispositivo electrónico como apoyo a la prevención del STC*",

En la parte inferior, se presentan dos botones principales con un diseño minimalista y accesible:

1. **Registrar paciente:** Permite ingresar la información de nuevos usuarios en el sistema, asegurando un almacenamiento estructurado de los datos clínicos relevantes.
2. **Análisis rápido:** Brinda la posibilidad de realizar un diagnóstico preliminar o rápido, proporcionando una evaluación inicial del riesgo de STC.
3. **Verificación de sensores:** Permite realizar un análisis rápido para comprobar que los sensores se encuentren correctamente colocados.

El diseño gráfico emplea una paleta de colores con degradados en tonos azules y blancos, promoviendo una experiencia visual agradable y profesional. Además, los iconos complementarios facilitan la identificación rápida de cada funcionalidad, optimizando la usabilidad de la aplicación.

Figura 44*Pantalla principal*

Nota. Autoría propia.

3.12.2.2 Pantalla de registro.

La pantalla de registro está diseñada para la recopilación estructurada de información esencial de los pacientes. Su principal objetivo es permitir el almacenamiento y seguimiento de los diagnósticos realizados, facilitando la trazabilidad del progreso del paciente dentro del sistema. Toda la información ingresada es almacenada en la base de datos para su posterior consulta y análisis.

La interfaz presenta un diseño intuitivo y minimalista, con un encabezado que destaca la acción principal: "Registrar", seguido de una breve instrucción que orienta al usuario en el proceso de ingreso de datos como se muestra en la figura 47.

Los campos de entrada definidos para esta pantalla incluyen:

Nombre y apellido: Identificación del paciente.

Cédula: Documento de identidad para garantizar registros únicos.

Teléfono: Medio de contacto adicional.

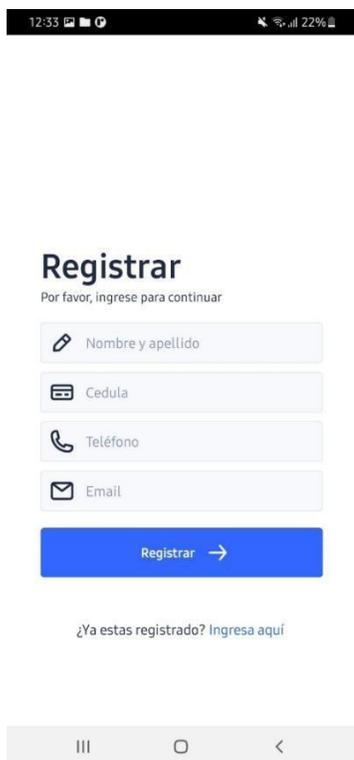
Email: Información para comunicación y recuperación de datos.

El sistema implementa validaciones en tiempo real que restringen el avance del usuario hasta que se cumplan los requisitos de cada campo, aplicando un modelo de *feedback preventivo* mediante indicadores visuales. Esta capa de validación asegura la integridad de los datos antes de su almacenamiento en la base de datos, minimizando errores de captura.

En la parte inferior de la interfaz, se dispone de un botón de acción destacado con la etiqueta "Registrar", cuya función es almacenar los datos ingresados en la base de datos. Además, se incluye un enlace con la opción "¿Ya estás registrado? Ingresa aquí", lo que permite a los usuarios existentes acceder a su información sin generar registros duplicados y almacenar un nuevo diagnóstico en su historial. El identificador único de usuario es la cédula con la cual se registra.

Figura 45

Pantalla de registro



The screenshot shows a mobile application registration screen. At the top, there is a status bar with the time 12:33, signal strength, Wi-Fi, and battery level at 22%. The main heading is 'Registrar' in bold black text, followed by the instruction 'Por favor, ingrese para continuar'. Below this are four input fields: 'Nombre y apellido' (with a pencil icon), 'Cedula' (with an ID card icon), 'Teléfono' (with a phone icon), and 'Email' (with an envelope icon). A blue button labeled 'Registrar' with a right-pointing arrow is positioned below the fields. At the bottom of the form area, there is a link: '¿Ya estas registrado? Ingresar aquí'. The bottom of the screen shows a standard Android navigation bar with three icons: a square, a circle, and a triangle.

Nota. Autoría propia.

Una vez que se completa el registro se presenta un breve tutorial de 3 slides guía para el usuario de los requerimientos preliminares que se debe cumplir antes de comenzar con un nuevo diagnóstico. Posteriormente se presenta una configuración del comienzo de análisis para poder iniciar e indicar al dispositivo que comience con la toma de muestras y realice la predicción del nivel de dolor o fatiga del paciente como se evidencia en la figura 48.

Figura 46

Guía para el usuario



Nota. Autoría propia.

3.12.2.3 Pantalla de resultados.

La pantalla de resultados se presenta una vez que el dispositivo ha completado exitosamente el proceso de predicción, basado en la recopilación y análisis de datos obtenidos a través de los sensores integrados. Esta funcionalidad permite proporcionar un diagnóstico objetivo en función de los valores capturados, ofreciendo así una evaluación clara.

Para la visualización de los resultados, se ha optado por mostrar el diagnóstico generado por el sistema junto con los valores promedio de los sensores, conforme a lo descrito en secciones anteriores. Esta presentación de datos permite una mejor comprensión del estado actual del paciente y sugiere una correlación directa con las mediciones realizadas.

Adicionalmente, para cada diagnóstico se incluyen recomendaciones generales, orientadas a guiar al usuario sobre las acciones a tomar para mejorar su condición o prevenir un agravamiento de su estado, especialmente en lo referente a la fatiga o el dolor en la muñeca como se muestra en la figura 49.

En la parte superior derecha de la pantalla se presenta un botón de acceso al historial del paciente donde el estado resultante junto con los datos son almacenados junto con su fecha y hora del diagnóstico como se muestra en la figura 50, esta funcionalidad únicamente cuando se ha realizado el análisis mediante el ingreso de paciente, en caso del análisis rápido este queda deshabilitado.

En la parte inferior de la pantalla se presenta un botón con la opción de realizar un nuevo análisis y se repite el proceso descrito anteriormente.

Figura 47

Pantalla de resultados



Nota. Autoría propia.

Figura 48

Historial del paciente



The screenshot shows a mobile application interface with a status bar at the top displaying the time 12:34 and battery level 22%. Below the status bar is a navigation bar with a back arrow and the title "Historial". Underneath is a table with five columns: Fecha, Hora, Estado, EMG, and GSR. The table contains three rows of data.

Fecha	Hora	Estado	EMG	GSR
2025-02-12	00:31:02	2	421	170
2025-02-12	00:31:18	2	421	170
2025-02-12	00:34:23	2	421	170

Nota. Autoría propia.

3.13 Backend

La aplicación, como se menciona anteriormente, firebase no trabaja con un backend centralizado, ya que utiliza un esquema de comunicación entre clientes (Aplicación – ESP32). Sin embargo, Firebase Realtime Database cumple un rol fundamental en la gestión y sincronización de datos en tiempo real, funcionando como un backend basado en la nube.

De ese modo, la aplicación publica ciertos valores en la base de datos, los cuales son procesados posteriormente por el ESP32 antes de ser almacenados nuevamente en Firebase los nuevos valores de los sensores. De este modo, aunque no se cuenta con un servidor dedicado para la lógica, la arquitectura del sistema se apoya en Firebase para la gestión eficiente de la información y la sincronización entre clientes.

Capítulo IV: Pruebas de Funcionamiento

Este capítulo presenta la cuarta fase de la metodología Bottom-Up, enfocada en la verificación integral del sistema desarrollado. En esta etapa se lleva a cabo la evaluación de los parámetros y resultados obtenidos por la red neuronal entrenada, complementada con representaciones gráficas que ilustran su rendimiento. Asimismo, se detalla el proceso de integración total de los componentes de hardware y software, junto con la ejecución de pruebas funcionales diseñadas para evaluar el desempeño del sistema en condiciones reales. Estas pruebas se centraron particularmente en el módulo de prediagnóstico de molestias asociadas al Síndrome del Túnel Carpiano (STC), y fueron realizadas con la colaboración de personal administrativo del Departamento Seguridad, Gestión y Riesgos y la Facultad de Ingeniería y Ciencias Aplicadas.

4.1 Evaluación del modelo de red neuronal

En esta sección se presentan los parámetros utilizados durante el proceso de entrenamiento del modelo, los cuales fueron analizados en función de los distintos resultados obtenidos, con el objetivo de seleccionar la configuración más adecuada para la obtención del modelo final. Se comienza detallando la configuración del hardware empleada para las fases de entrenamiento, validación y prueba de la red neuronal artificial (RNA).

Cabe destacar que el desarrollo y entrenamiento del modelo se realizó a través de la plataforma Jupyter Notebook, ejecutada desde la distribución Anaconda, lo cual implica que todos los procesos computacionales se llevaron a cabo utilizando exclusivamente los recursos locales del equipo. La Tabla 19 resume las especificaciones del entorno computacional utilizado.

Tabla 19

Especificaciones del hardware utilizado durante el entrenamiento del modelo

COMPONENTE	ESPECIFICACIÓN TÉCNICA
Modelo del equipo	HP 250 15.6 inch G9 Notebook PC
Procesador (CPU)	12th Gen Intel® Core™ I5-1235U 1.30GHz
Memoria RAM	16,0 GB
Unidad de almacenamiento	500GB
Sistema operativo	Windows 11
Unidad de procesamiento gráfico (GPU)	Intel® UHD Graphics
Arquitectura del sistema	64 bits

A continuación, en la tabla 20 se presentan los parámetros utilizados para definir las características mínimas que debe tener un dispositivo para que la aplicación, desarrollada en formato APK, funcione correctamente. Estos requisitos técnicos permiten establecer un estándar que facilite futuras instalaciones, actualizaciones y asegure la compatibilidad del sistema.

Tabla 20

Especificaciones del dispositivo móvil

Componente	Especificación Mínima Recomendada
Sistema Operativo	Android 11 o superior
Procesador (CPU)	Octa-core, al menos 2.0 GHz
RAM	4 GB
Batería	3400 mAh
Conectividad	4G LTE, Wi-Fi 5, Bluetooth 5.0, GPS
Almacenamiento	32 GB

Teniendo en cuenta las especificaciones de hardware utilizadas se determina las configuraciones de entrenamiento, y valores de precisión y pérdida obtenidos.

4.1.1. Configuración y parámetros de entrenamiento del modelo.

Dado que el conjunto de datos cuenta con 300 muestras, la implementación de un modelo *Multilayer Perceptron* (MLP) resulta apropiada debido a la naturaleza tabular de los datos. En este tipo de aplicaciones, las variables involucradas son relativamente pocas y cada muestra puede representarse mediante un vector de baja dimensión. Esta característica permite que los modelos MLP sean entrenados de manera eficaz utilizando conjuntos de datos reducidos. La razón principal es que este tipo de redes contiene una menor cantidad de parámetros, lo cual disminuye el riesgo de sobreajuste (*overfitting*) y reduce la necesidad de grandes volúmenes de datos para alcanzar una adecuada capacidad de generalización.

Aunque el conjunto de datos original cuenta con 300 muestras, se seleccionaron únicamente 150 para el entrenamiento del modelo, eligiendo aquellas que representan de manera más diversa y significativa los distintos patrones presentes. Esta estrategia busca reducir la complejidad computacional y evitar el sobreajuste, ya que en los datos completos predominan las muestras con valores normales, lo cual podría hacer que el modelo se vuelva demasiado general y no aprenda adecuadamente a distinguir los casos relevantes de dolor muscular.

Al entrenar con un subconjunto representativo, se logra capturar la variabilidad real del fenómeno que se desea predecir, evitando la redundancia de valores similares que no aportan nueva información. Así, se mejora la capacidad de generalización del modelo sin sacrificar precisión ni eficiencia.

En la Tabla 21 se presentan los parámetros considerados para el modelo de red neuronal, donde varían configuraciones y el número de épocas utilizadas durante el entrenamiento.

Tabla 21

Parámetros para la red neuronal

CONFIGURACIÓN DE ENTRENAMIENTO –CLASIFICACION MULTICLASE			
MODELO - MLP (SECUENCIAL)			
Número de épocas	100 (con <i>EarlyStopping</i>)	200 (con <i>EarlyStopping</i>)	300 (con <i>EarlyStopping</i>)
	Valor	Valor	Valor
Tamaño del lote	32	32	32
Optimizador	Adam	Adam	Adam
Tasa de aprendizaje	0.001	0.001	0.001
Función de pérdida	Categorical Crossentropy	Categorical Crossentropy	Categorical Crossentropy
Métrica de evaluación	Categorical Accuracy	Categorical Accuracy	Categorical Accuracy
Regularización	<i>Dropout</i> (0.3 en dos capas)	<i>Dropout</i> (0.3 en dos capas)	<i>Dropout</i> (0.3 en dos capas)
Arquitectura	Dense(64) → Dropout → Dense(32) → Dropout → Dense(Softmax)	Dense(64) → Dropout → Dense(32) → Dropout → Dense(Softmax)	Dense(64) → Dropout → Dense(32) → Dropout → Dense(Softmax)

Activaciones	ReLU (ocultas), Softmax (salida)	ReLU (ocultas), Softmax (salida)	ReLU (ocultas), Softmax (salida)
Llamada de retorno	<i>EarlyStopping</i> (patience=20)	<i>EarlyStopping</i> (patience=20)	<i>EarlyStopping</i> (patience=20)
Método de escalado	MinMaxScaler	MinMaxScaler	MinMaxScaler

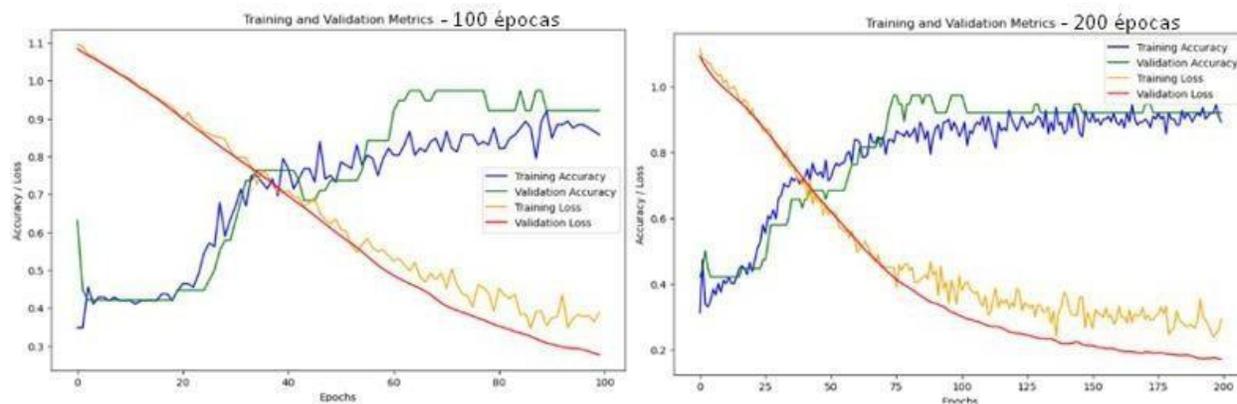
Como se pudo evidenciar se comparan tres configuraciones del modelo con diferentes cantidades de épocas (100, 200 y 300), todas utilizando la técnica de *EarlyStopping* con el objetivo de mitigar el sobreajuste. Si bien la arquitectura y los hiperparámetros principales se mantienen constantes, la variación en el número de épocas permite evaluar el efecto del entrenamiento prolongado sobre el rendimiento del modelo.

En la figura 51 se presentan los gráficos correspondientes al proceso de entrenamiento del modelo con 100, 200 y 300 épocas, donde se representan las métricas de precisión y pérdida tanto para el conjunto de entrenamiento como para el de validación. En el primer caso (a), con 100 épocas, se evidencia que el modelo apenas inicia su fase de aprendizaje, ya que las curvas aún presentan oscilaciones y no alcanzan una estabilidad clara, lo que sugiere que el entrenamiento es insuficiente para lograr una convergencia adecuada. En el segundo caso (b), con 200 épocas, se observa una mejora significativa: las curvas de validación se estabilizan alrededor de las épocas 150 a 190, lo cual indica que el modelo alcanza un rendimiento óptimo dentro de ese intervalo, con una pérdida de validación que desciende de forma sostenida y una precisión que se mantiene alta y estable, reflejando una adecuada capacidad de generalización. Finalmente, en el tercer caso (c), al extender el entrenamiento hasta 300 épocas, las métricas de validación muestran una ligera mejora adicional, especialmente en la pérdida, sin que se presenten indicios de sobreajuste, lo que refuerza la solidez del modelo y confirma que sigue generalizando correctamente incluso con un

mayor número de iteraciones. En conjunto, el análisis comparativo de los tres escenarios demuestra que el modelo alcanza su punto de mayor rendimiento entre las 150 y 190 épocas, aunque puede beneficiarse moderadamente de una extensión del entrenamiento sin comprometer su desempeño.

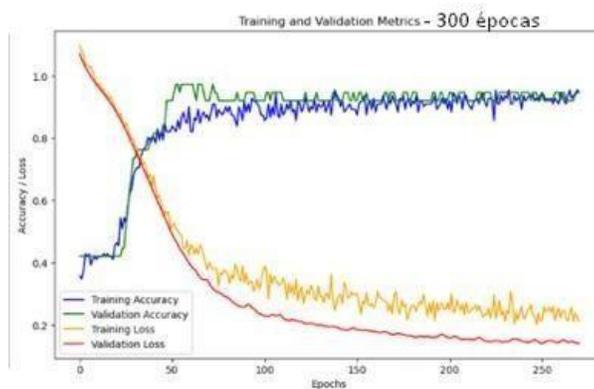
Figura 49

Gráfico de curvas de entrenamiento y perdida



a) 100 épocas

b) 200 épocas



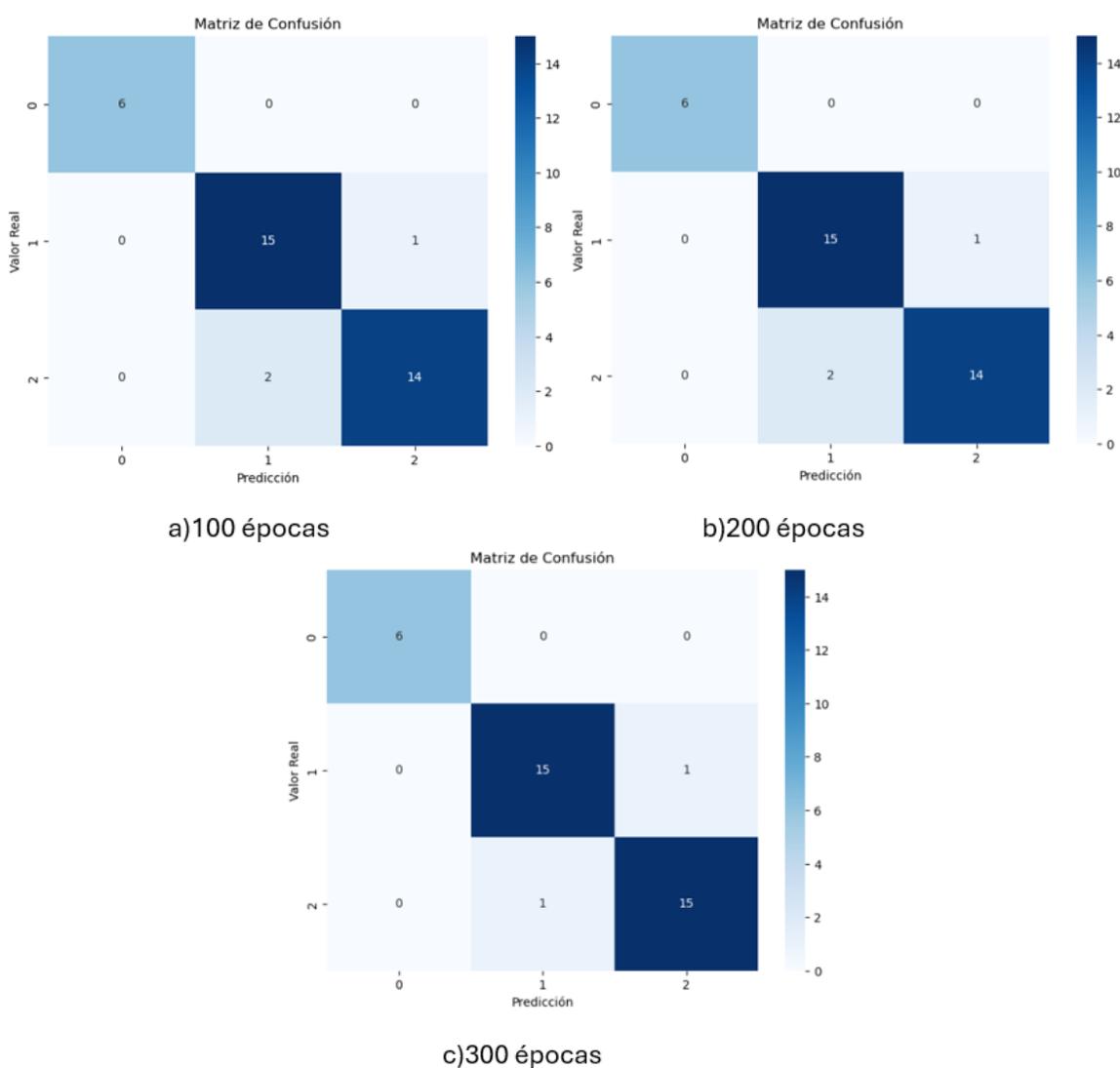
c) 300 épocas

Nota. Autoría propia.

A continuación, en la Figura 52, se presentan las matrices de confusión obtenidas de los modelos entrenados. En los tres casos, se observa un buen desempeño general, con clasificaciones mayoritariamente correctas en las tres clases. Sin embargo, el modelo entrenado con 300 épocas muestra una leve mejora, reduciendo los errores de clasificación en la tercera clase (etiqueta 2), donde solo se registra una predicción incorrecta, frente a lo observado en los modelos.

Figura 50

Matrices de confusión de red neuronal.



Nota. Autoría propia.

La elección del modelo entrenado con 300 épocas responde a su rendimiento consistentemente superior en comparación con las otras configuraciones evaluadas. Este modelo alcanzó la mayor precisión (94.73 %) y la menor pérdida (14.07 %), lo cual indica un aprendizaje más profundo de los patrones presentes en los datos. Además, la matriz de confusión correspondiente muestra una mejora en la clasificación de la tercera clase, reduciendo los errores cometidos en configuraciones anteriores. Aunque el entrenamiento se detuvo de forma anticipada en la época 240 por medio del criterio *EarlyStopping*, este hecho confirma que el modelo logra converger de manera eficiente antes de agotar el número máximo de iteraciones, optimizando el uso de los recursos computacionales. En conjunto, estos resultados justifican la selección del modelo de 300 épocas como el más adecuado para la implementación en el dispositivo diseñado.

En la Tabla 22 se presentan los valores obtenidos de precisión y pérdida tanto para el conjunto de entrenamiento como para el conjunto de validación, lo que permite una comparación objetiva del comportamiento del modelo bajo cada configuración.

Tabla 22

Valores de entrenamiento RNA.

VALORES DE ENTRENAMIENTO RNA		
Nº Épocas	Precisión	Pérdida
100	92.11%	27.65%
200	92.11%	17.09%
300	94.73%	14.07%

A partir de los resultados presentados, se observa que el modelo alcanza su mejor desempeño con 300 épocas configuradas, logrando una precisión del 94.73 % y una pérdida del

14.07 %. Aunque las configuraciones de 100 y 200 épocas también presentan un rendimiento aceptable (92.11 % de precisión en ambos casos), la disminución progresiva en la pérdida indica que un mayor número de épocas favorece el aprendizaje del modelo, sin evidencias de sobreajuste. No obstante, debido al uso del mecanismo *EarlyStopping*, el entrenamiento correspondiente a las 300 épocas se detuvo anticipadamente en la época 240, al no observarse mejoras significativas en la pérdida de validación. Esto sugiere que el modelo converge antes de alcanzar el número máximo de épocas, optimizando así el proceso de entrenamiento.

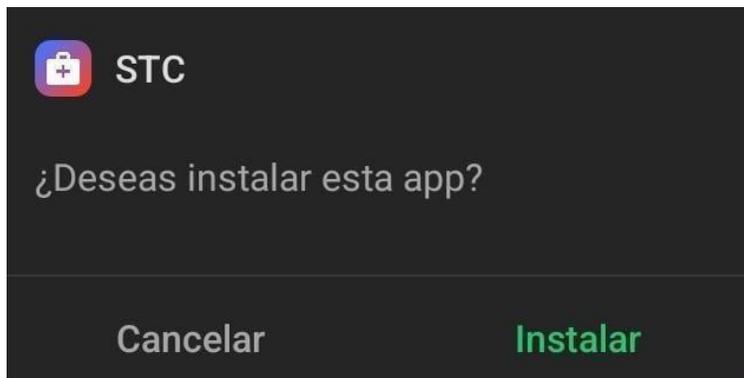
4.2 Prueba de funcionamiento del sistema.

Para la correcta utilización del dispositivo, el primer paso consiste en la instalación del software de administración, el cual habilita las funciones descritas en la sección 3.4.2. Este procedimiento requiere el uso de un dispositivo móvil con sistema operativo Android, que disponga de conexión a internet y permita la creación de un punto de acceso (hotspot) o la función de compartir datos móviles.

La instalación del archivo ejecutable (APK) correspondiente al programa de administración se realiza mediante el procedimiento estándar del sistema operativo, como se ilustra en la Figura 53. Es importante destacar que, en caso de que el dispositivo no tenga habilitada la opción para instalar aplicaciones provenientes de fuentes externas (es decir, fuera de la tienda oficial), será necesario activar dicha opción desde la configuración del sistema.

Figura 51

APK de la aplicación móvil.



Nota. Autoría propia.

Una vez finalizado el proceso de instalación del programa de administración, se procede a compartir una red inalámbrica desde el dispositivo móvil, utilizando las credenciales predeterminadas: nombre de red (SSID) "STC-WIFI" y contraseña numérica del 1 al 8. A continuación, se enciende el dispositivo electrónico que será gestionado por el sistema.

Es importante señalar que esta red es configurada por defecto. Por lo tanto, en caso de que el sistema sea restablecido (ya sea mediante el botón físico de reinicio o a través de la opción correspondiente en la aplicación) será necesario repetir este procedimiento inicial de configuración de red.

Una vez que el dispositivo ha sido encendido, y si se desea conectar a una red local existente distinta a la predeterminada, es posible modificar los parámetros de conexión ingresando al apartado "Configuración de red". Desde allí, se pueden introducir las nuevas credenciales de acceso, tal como se ilustra en la Figura 54.

Figura 52

Interfaz de configuración de red.



Nota. Autoría propia.

Una vez completadas las acciones descritas anteriormente, se procede a la colocación de los sensores de electromiografía superficial (EMG) en la región del antebrazo, tal como se indica en la Figura 55. La disposición de los electrodos debe realizarse respetando una separación adecuada entre ellos, con el propósito de garantizar una captación representativa y precisa del nivel general de fatiga muscular. Esta medición es relevante en contextos de actividad laboral que impliquen movimientos repetitivos o esfuerzos sostenidos, donde se busca evaluar el impacto fisiológico de dichas tareas sobre la musculatura del miembro superior.

Figura 53

Colocación de sensor de electromiografía.

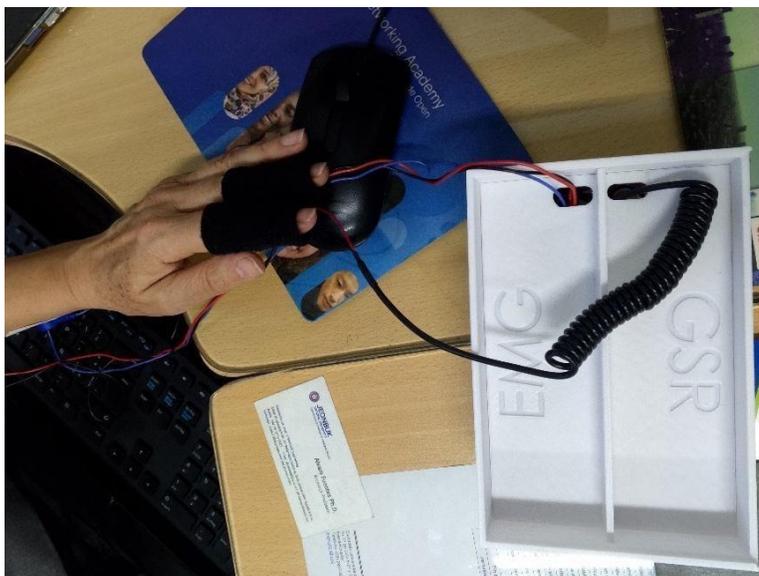


Nota. Autoría propia.

Adicionalmente, el sensor de respuesta galvánica de la piel (GSR) se posiciona en los dedos índice y medio de la mano dominante del usuario, conforme a lo ilustrado en la Figura 56. Esta localización fue seleccionada debido a su efectividad en la detección de variaciones fisiológicas asociadas a la exposición prolongada a posturas estáticas o a elevados niveles de tensión muscular vinculadas estrechamente con la aparición de sintomatología del STC.

Figura 54

Colocación de sensor de respuesta galvanica



Nota. Autoría propia.

Con los sensores conectados se sugiere realizar una verificación inicial de funcionamiento. Esta prueba rápida, que no implica el registro de datos, permite confirmar que los sensores están operativos y adecuadamente conectados. En caso de detectarse alguna anomalía en la conexión o funcionamiento del dispositivo, el sistema genera una notificación de error, como se muestra en la Figura 57. Esta validación preliminar contribuye a garantizar la fiabilidad de los datos que serán recopilados al realizar los análisis posteriores.

Figura 55

Pantalla de validación de sensores



Nota. Autoría propia.

Para la realización del análisis fisiológico, es necesario ingresar ciertos datos personales del usuario, tales como nombre completo, número de cédula, número de teléfono y dirección de correo electrónico. Esta información tiene como finalidad mantener un registro individualizado de cada participante y permitir la generación de un historial con los resultados obtenidos en cada sesión de análisis, como se ejemplifica en la Figura 58. En caso de que el usuario ya esté registrado en el sistema, el ingreso se realiza mediante su identificador único, correspondiente a su número de cédula.

Figura 56

Interfaz de registro



The screenshot shows a mobile application registration screen. At the top, there is a status bar with the time '9:00 p.m.' and various system icons. Below the status bar is a back arrow icon. The main content area is titled 'Registrar' in bold, followed by the instruction 'Por favor, ingrese para continuar'. There are four input fields: 'Nombre y apellido' with a pencil icon, 'Cedula' with an ID card icon, 'Teléfono' with a phone icon, and 'Email' with an envelope icon. Below these fields is a blue button labeled 'Registrar' with a right-pointing arrow. At the bottom of the form area, there is a link that says '¿Ya estas registrado? Ingresa aqui'. The bottom of the screen shows a standard Android navigation bar with three icons: a hamburger menu, a circle, and a back arrow.

Nota. Autoría propia.

Una vez ingresadas las credenciales, el sistema despliega un breve tutorial informativo, así como un aviso que recuerda mantener los sensores conectados durante el proceso de análisis. Como se observa en la Figura 59, se presenta una pantalla con los resultados preliminares de un prediagnóstico del estado actual del usuario. En la parte superior de la interfaz, se encuentra accesible la sección de historial, donde se almacenan automáticamente los resultados de cada análisis, junto con la fecha y hora correspondiente.

Figura 57

Interfaz de resultados



Nota. Autoría propia.

Como parte del diseño funcional de la aplicación, se incorporó un sistema de alerta que monitorea el estado del usuario en función de los resultados obtenidos durante los análisis. En caso de que se registren tres lecturas consecutivas clasificadas como "estado de fatiga" como lo muestra la figura 60, la aplicación genera un mensaje emergente con el objetivo de advertir al usuario sobre la posible persistencia de síntomas relacionados con el síndrome del túnel carpiano. Este mensaje no realiza un diagnóstico médico, pero actúa como una recomendación preventiva, sugiriendo acudir a un profesional de la salud para una evaluación más detallada.

Figura 58

Mensaje informativo después de 3 lecturas consecutivas de estado de Fatiga



Nota. Autoría propia.

En cuanto a la administración de usuarios, se puede acceder a dicha funcionalidad desde la esquina superior derecha de la pantalla principal. Al seleccionar esta opción, se requiere el ingreso de credenciales administrativas predeterminadas. Como se ilustra en la Figura 61, se despliega una interfaz que permite visualizar todos los usuarios registrados, así como crear nuevos perfiles, modificar la información existente o eliminar usuarios según sea necesario.

Figura 59*Interfaz del administrador*

Nota. Autoría propia.

Finalmente, desde la sección de historial (ya sea a través de la interfaz de administración o tras la finalización de un análisis) se habilita la opción de generar un archivo en formato PDF como lo muestra la figura 62. Este documento contiene un registro cronológico detallado de los análisis realizados por el usuario, lo cual facilita el seguimiento longitudinal de su condición fisiológica.

Figura 60

Resultados del usuario en formato pdf



RESULTADOS

Dispositivo electrónico de apoyo a la prevención del STC

Cédula de identificación del paciente: 1050447026

Nombre y apellido: Anderson Pérez

Email: andersonbaneado7@gmail.com

Telefono: 0985667214

Fecha: 28/5/2025

Clave de Valores de Estado	
Estado	Resultado
0	Normal
1	Tenso
2	Fatiga

Recomendaciones:

- Ejercicios de fortalecimiento: Aprieta una pelota de goma suave para fortalecer los músculos de la mano.
- Estiramientos: Realiza estiramientos suaves.
- Masajes: Masajea suavemente la muñeca.
- Rotaciones de muñeca: Gira suavemente las muñecas en ambas direcciones.

Resultados

Fecha	Hora	Estado	EMG (μ V)	GSR (Ω)
2025-05-25	20:44:29	2	237	303
2025-05-25	20:51:20	2	300	403

Nota. Autoría propia.

Para finalizar la verificación del correcto funcionamiento del hardware, se consideran varios factores que se presentan en la Tabla 23 . Estos permiten evaluar si cada componente del sistema realiza adecuadamente su tarea. Los resultados de estas pruebas permiten confirmar que los elementos están bien integrados en el sistema, ya que, si alguno no cumple correctamente su función, podría indicar que no está bien conectado o acoplado.

Tabla 23*Pruebas de operatividad el hardware*

PRUEBAS DE HARDWARE	
Parámetro	Cumplimiento
Los microcontroladores operan de manera adecuada.	SI
El sensor de electromiografía (EMG) presenta un funcionamiento correcto.	SI
El sensor de respuesta galvánica de la piel (GSR) funciona correctamente.	SI
El sistema recibe y procesa los datos de forma adecuada.	SI

Autoría propia.

4.3 Pruebas de campo del sistema.

Con el objetivo de validar el funcionamiento del dispositivo desarrollado, se llevaron a cabo pruebas de campo que incluyeron la aplicación de prediagnósticos a un grupo de usuarios. Esta etapa experimental permitió verificar tanto la operatividad técnica del dispositivo como la utilidad de los datos generados para la detección temprana de posibles alteraciones fisiológicas asociadas a fatiga y estrés muscular.

4.3.1. Participantes y ejecución de pruebas.

El proceso de validación se llevó a cabo con la colaboración de personal administrativo del Departamento Seguridad, Gestión y Riesgos y la Facultad de Ingeniería y Ciencias Aplicadas. La selección de los participantes se realizó en función de su exposición a tareas laborales que implican posturas prolongadas o movimientos repetitivos, características comunes en entornos de oficina y asociadas a un mayor riesgo de desarrollar Síndrome del Túnel Carpiano (STC). Las pruebas se ejecutaron en un entorno controlado y durante el horario laboral habitual de los sujetos, garantizando condiciones realistas y mínimamente invasivas.

Durante un periodo de dos semanas, cada participante fue evaluado en cuatro sesiones distribuidas en dos jornadas por semana. En cada jornada se realizaron dos sesiones en un mismo día, una en la mañana y otra en la tarde con el objetivo de descartar la influencia de factores como la respuesta emocional o el nivel de estrés. Este esquema se replicó en un día distinto de la semana siguiente para asegurar un seguimiento continuo y representativo.

En cada sesión se llevó a cabo la conexión de los sensores biomédicos, la verificación de su correcto funcionamiento, el registro de las credenciales personales del usuario y la ejecución del procedimiento de prediagnóstico. Los datos obtenidos fueron almacenados automáticamente en el historial individual de cada participante, lo que permitió un seguimiento detallado y preciso del estado fisiológico durante todo el periodo de análisis

Cabe señalar que, en cumplimiento de los principios éticos de investigación, no se divulgarán datos personales ni imágenes de los participantes en la redacción o presentación de los resultados, dado que la actividad tiene exclusivamente fines académicos de validación.

Una vez definido el grupo de usuarios para el proceso de pruebas, se procedió a la evaluación mediante el dispositivo en distintos contextos temporales, distribuidos en diferentes días y franjas horarias. Con base en la información proporcionada por los participantes y conforme al criterio técnico-médico de la experta en fisioterapia, se planificaron las sesiones en jornadas que suelen implicar una mayor carga laboral, como los lunes cuando se asignan nuevas tareas y viernes cuando se busca cerrar las actividades pendientes de la semana, así como en días con menor exigencia operativa. Esta planificación permitió obtener registros fisiológicos representativos de diversos niveles de exigencia física y mental, en función de la carga de trabajo experimentada durante la jornada.

Esta etapa experimental permitió validar tanto la operatividad técnica del dispositivo como la utilidad de los datos generados para la detección temprana de posibles alteraciones fisiológicas asociadas a la fatiga muscular y otras manifestaciones vinculadas con el esfuerzo laboral.

Tras completar exitosamente el protocolo de pruebas durante un periodo de 10 días, con los participantes indicados en la Tabla 25, se recopilaron 60 registros en las fechas planificadas. Los resultados obtenidos reflejan una amplia variedad de respuestas fisiológicas, las cuales fueron clasificadas en tres categorías: estado relajado o normal, estado de tensión, y estado de astenia o fatiga. Esta clasificación sirvió como base para el análisis detallado del impacto de la carga laboral sobre las condiciones fisiológicas de los usuarios.

Tabla 24

Pruebas de funcionamiento en los usuarios

PRUEBAS DE FUNCIONAMIENTO EN LOS USUARIOS					
USUARIO	FECHA	HORA	ESTADO	EMG	GSR
Usuario 1	5/5/2025	10:07:15	Normal	300	114
		14:16:12	Normal	283	99
Usuario 2	5/5/2025	10:25:42	Normal	316	101
		14:34:55	Tenso	296	222
Usuario 3	5/5/2025	11:02:11	Normal	294	63
		15:12:43	Fatiga	289	235
Usuario 4	6/5/2025	10:01:33	Normal	289	99
		14:20:38	Tenso	313	178
Usuario 5	6/5/2025	10:11:04	Normal	313	138
		14:38:44	Normal	262	171

Usuario 6	6/5/2025	11:17:22	Normal	314	118
		15:28:09	Tenso	299	178
Usuario 7	7/5/2025	10:00:56	Normal	341	92
		14:18:40	Normal	278	142
Usuario 8	7/5/2025	10:27:47	Normal	341	88
		14:39:52	Normal	372	108
Usuario 9	7/5/2025	11:14:35	Normal	396	87
		15:24:28	Tenso	306	200
Usuario 10	8/5/2025	10:05:25	Normal	352	118
		14:16:48	Normal	305	139
Usuario 11	8/5/2025	10:36:18	Normal	290	57
		14:33:01	Normal	266	119
Usuario 12	8/5/2025	11:27:30	Normal	345	131
		15:08:19	Normal	335	151
Usuario 13	9/5/2025	10:10:50	Normal	329	115
		14:19:29	Normal	310	147
Usuario 14	9/5/2025	10:32:42	Normal	303	178
		14:44:11	Fatiga	295	286
Usuario 15	9/5/2025	11:13:18	Normal	299	175
		15:17:56	Normal	354	172
Usuario 7	12/5/2025	10:01:09	Normal	301	44
		14:16:33	Fatiga	337	365
Usuario 2	12/5/2025	10:21:37	Normal	307	161
		14:35:58	Normal	315	119
Usuario 12	12/5/2025	11:07:52	Normal	287	141

		15:25:15	Fatiga	304	247
Usuario 5	13/5/2025	10:04:28	Normal	333	164
		14:20:21	Normal	295	129
Usuario 14	13/5/2025	10:26:44	Normal	308	140
		14:37:46	Tenso	294	205
Usuario 9	13/5/2025	11:19:06	Normal	307	126
		15:15:32	Normal	311	137
Usuario 1	14/5/2025	10:09:11	Normal	356	136
		14:17:20	Normal	318	151
Usuario 8	14/5/2025	10:30:27	Normal	307	194
		14:39:43	Tenso	302	218
Usuario 13	14/5/2025	11:16:58	Normal	304	90
		15:11:09	Normal	320	157
Usuario 6	15/5/2025	10:06:42	Normal	347	194
		14:19:10	Normal	297	151
Usuario 11	15/5/2025	10:31:56	Normal	273	109
		14:41:38	Tenso	296	178
Usuario 3	15/5/2025	11:28:45	Normal	317	157
		15:22:27	Normal	297	184
Usuario 4	16/5/2025	10:03:38	Normal	273	162
		14:18:03	Fatiga	396	301
Usuario 15	16/5/2025	10:29:03	Normal	325	109
		14:36:30	Normal	284	110
Usuario 10	16/5/2025	11:22:17	Normal	250	61
		15:06:44	Tenso	286	222

A partir de las pruebas realizadas se evidencia que los usuarios evaluados presentan distintos estados de respuesta fisiológica, influenciados por diversos factores como la carga laboral, la organización de las actividades y otros elementos relacionados como el aumento de la frecuencia cardíaca, una leve variación en la temperatura corporal, tensión muscular, irritabilidad o dificultad para concentrarse. Estas condiciones pueden provocar alteraciones en el estado emocional del usuario, lo cual incide tanto en su rendimiento laboral como en su bienestar general. En particular, se registraron 46 estados dentro del rango normal, lo que representa el 78.33% del total; 8 estados de tensión, correspondientes al 13.33%; y 6 estados de fatiga, que equivalen al 8.34% como se detallan en la Tabla 26.

Tabla 25

Resultados de la evaluación fisiológica: estado normal, tensión y fatiga

	N	%
NORMAL	47	78,33%
TENSO	8	13,33%
FATIGA	5	8,34%

A continuación, se presentan los resultados de la evaluación según la influencia de los días de la semana. Se observa que, al inicio o fin de la semana, el 8.33% de los resultados evidencian niveles elevados de fatiga, lo cual podría estar relacionado con una mayor presencia de estresores en esos días, en comparación con los días intermedios. Estos datos muestran que, aunque el estado fisiológico normal es el más común, tanto el inicio como el final de la semana pueden impactar negativamente en el bienestar físico y emocional de algunas personas. Por otro lado, el 28.33 % de los participantes no presenta niveles significativos de fatiga, lo que podría atribuirse a una

adecuada capacidad para gestionar las tareas asignadas y mantener un equilibrio saludable entre la vida laboral y personal.

Así mismo se muestra que, durante los días entre semana, predomina el estado Normal, con un 50 %, lo que refleja una mejor respuesta fisiológica en comparación con el inicio o el final de la semana. El estado Tenso se presenta en un 10%, mientras que no se registran casos de Fatiga en estos días. Estos resultados muestran que la ausencia de fatiga también podría estar relacionada con un manejo más eficiente del tiempo o con una adecuada adaptación al ritmo semanal del trabajo, a continuación en la tabla 27 se muestra los resultados obtenidos.

Tabla 26

Resultados de la evaluación según la influencia de los días

	INICIO O FIN DE SEMANA	DÍAS ENTRE SEMANA
NORMAL	28.33%	50%
TENSO	3.33%	10%
FATIGA	8.33%	0

Una vez realizadas las respectivas pruebas con el dispositivo, se procede en este apartado a realizar una evaluación complementaria mediante métodos tradicionales, utilizando escalas de medición de la salud. En este caso, se empleará el Cuestionario de Boston, una herramienta ampliamente validada y reconocida para valorar el Síndrome del Túnel Carpiano (STC).

Las escalas utilizadas para medir la salud no solo sirven para evaluar el estado físico de los pacientes, sino que también resultan clave para conocer la percepción individual de los trabajadores expuestos a posibles sobrecargas musculares locales y síntomas relacionados con el síndrome del túnel carpiano (STC). Estas herramientas desempeñan un papel crucial en la

prevención, ya que permiten detectar de manera oportuna señales de tensión en el sistema musculoesquelético.

El Cuestionario de Boston como se evidencia en la figura 63 es una de las medidas más utilizadas para evaluar tanto la gravedad de los síntomas como el impacto funcional del STC. Este instrumento contempla aspectos sintomáticos y funcionales, y ha sido referido en la literatura con distintos nombres, como Carpal Tunnel Questionnaire, Brigham and Women's Carpal Tunnel Questionnaire o Levine Questionnaire. Su validez ha sido comprobada en diversos estudios y se encuentra disponible en varios idiomas, incluyendo el polaco, español y persa. (Ulbrichtová et al., 2019)

Figura 61

Cuestionario de Boston

1 ¿Cómo es de grave la molestia en la mano o el dolor en la muñeca durante la noche?	7 ¿Tiene debilidad en la mano o en la muñeca?
<input type="checkbox"/> 1. No tengo molestias durante la noche.	<input type="checkbox"/> 1. No hay debilidad
<input type="checkbox"/> 2. Dolor leve	<input type="checkbox"/> 2. Debilidad leve
<input type="checkbox"/> 3. Dolor moderado	<input type="checkbox"/> 3. Debilidad moderada
<input type="checkbox"/> 4. Dolor intenso	<input type="checkbox"/> 4. Debilidad severa
<input type="checkbox"/> 5. Dolor muy severo	<input type="checkbox"/> 5. Debilidad muy severa
2 ¿Con qué frecuencia le despiertan las molestias durante una noche en las últimas dos semanas?	8 ¿Tiene sensación de hormigueo en la mano?
<input type="checkbox"/> 1. Nunca	<input type="checkbox"/> 1. No hay sensación de hormigueo
<input type="checkbox"/> 2. Una vez	<input type="checkbox"/> 2. Leve hormigueo
<input type="checkbox"/> 3. Dos o tres veces	<input type="checkbox"/> 3. Hormigueo moderado
<input type="checkbox"/> 4. Cuatro o cinco veces	<input type="checkbox"/> 4. Grave hormigueo
<input type="checkbox"/> 5. Más de cinco veces	<input type="checkbox"/> 5. Hormigueo muy severo
3 ¿Suele tener dolor en la mano o en la muñeca durante el día?	9 ¿Cómo es de grave es el adormecimiento (pérdida de sensibilidad) o sensación de hormigueo durante la noche?
<input type="checkbox"/> 1. Nunca tengo dolor durante el día	<input type="checkbox"/> 1. No tengo entumecimiento u hormigueo en la noche
<input type="checkbox"/> 2. Tengo un dolor leve durante el día	<input type="checkbox"/> 2. Leve
<input type="checkbox"/> 3. Tengo dolor moderado durante el día	<input type="checkbox"/> 3. Moderado
<input type="checkbox"/> 4. Tengo un dolor intenso durante el día	<input type="checkbox"/> 4. Grave
<input type="checkbox"/> 5. Tengo un dolor muy intenso durante el día	<input type="checkbox"/> 5. Muy grave
4 ¿Con qué frecuencia tiene dolor en la mano o en la muñeca durante el día?	10 ¿Cuántas veces el entumecimiento u hormigueo en la mano le despierta durante una noche típica en las últimas dos semanas?
<input type="checkbox"/> 1. Nunca	<input type="checkbox"/> 1. Nunca
<input type="checkbox"/> 2. Una o dos veces al día	<input type="checkbox"/> 2. Una vez
<input type="checkbox"/> 3. de tres a cinco veces al día	<input type="checkbox"/> 3. Dos o tres veces
<input type="checkbox"/> 4. Más de cinco veces al día	<input type="checkbox"/> 4. Cuatro o cinco veces
<input type="checkbox"/> 5. El dolor es constante.	<input type="checkbox"/> 5. Más de cinco veces
5 ¿Cuánto tiempo, en promedio, tiene un episodio de dolor durante el día?	11 ¿Tiene dificultad para la captación y uso de objetos pequeños como llaves o plumas?
<input type="checkbox"/> 1. Nunca tengo dolor durante el día.	<input type="checkbox"/> 1. No tengo dificultad
<input type="checkbox"/> 2. Menos de 10 minutos	<input type="checkbox"/> 2. Leve dificultad
<input type="checkbox"/> 3. 10 a 60 minutos	<input type="checkbox"/> 3. Dificultad moderada
<input type="checkbox"/> 4. Más de 60 minutos	<input type="checkbox"/> 4. Dificultad severa
<input type="checkbox"/> 5. El dolor es constante durante todo el día	<input type="checkbox"/> 5. Dificultad muy severa
6 ¿Tiene entumecimiento (pérdida de sensibilidad) en la mano?	
<input type="checkbox"/> 1. No	
<input type="checkbox"/> 2. Presenta entumecimiento leve	
<input type="checkbox"/> 3. Entumecimiento moderado	
<input type="checkbox"/> 4. Tengo entumecimiento grave	
<input type="checkbox"/> 5. Tengo entumecimiento muy grave	

Chi Pearson (p<0,001)

A los voluntarios de la Universidad Técnica del Norte se les abordó de manera respetuosa y cordial, proporcionándoles una explicación detallada sobre el cuestionario, sus características y los objetivos de la investigación que se evidencian en la tabla 28. Se garantizó la confidencialidad de la información, brindando a cada participante seguridad y el compromiso de mantener su anonimato en todas las etapas del estudio. Durante el proceso de recolección de datos, se obtuvieron los siguientes resultados, considerando que la gravedad de los síntomas se clasifica en una escala del 1 al 5, donde 1 representa síntomas leves y 5 los más graves.

Tabla 27*Cuestionario realizado a voluntarios*

Descripción de síntomas	GRAVEDAD DE SÍNTOMAS									
	1		2		3		4		5	
	N	%	N	%	N	%	N	%	N	%
Gravedad nocturna	1	6,67	11	73,33	1	6,67	1	6,67	1	6,67
Despertar reciente por molestias	14	93,33	1	6,67	0,00	0,00	0,00	0,00	0,00	0,00
Dolor diurno	13	86,67	2	13,37	0,00	0,00	0,00	0,00	0,00	0,00
Frecuencia dolor diurno	14	93,33	1	6,67	0,00	0,00	0,00	0,00	0,00	0,00
Tiempo dolor diurno	12	80	3	20	0,00	0,00	0,00	0,00	0,00	0,00
Pérdida de sensibilidad	13	86,67	1	6,67	0,00	0,00	1	6,67	0,00	0,00
Debilidad	13	86,67	2	13,33	0,00	0,00	0,00	0,00	0,00	0,00
Hormigueo	14	93,33	1	6,67	0,00	0,00	0,00	0,00	0,00	0,00
Pérdida sensibilidad / hormigueo nocturno	13	86,67	1	6,67	0,00	0,00	0,00	0,00	1	6,67

Despertar reciente, pérdida sensibilidad u hormigueo	14	93,33	1	6,67	0,00	0,00	0,00	0,00	0,00	0,00
Dificultad funcional	13	86,67	2	13,33	0,00	0,00	0,00	0,00	0,00	0,00

En los resultados obtenidos que se muestran en la tabla 29, únicamente se registraron dos síntomas con un nivel de gravedad 3, siendo la fatiga el más alto reportado, con un 2,42 %. En cuanto a los síntomas moderados, clasificados en el nivel 2, el estado tenso alcanzó un 6,06 %. Finalmente, se observó que la gran mayoría de los registros, un 91,52 %, corresponde a una sintomatología normal, lo que indica una baja incidencia de malestar significativo entre los participantes.

Tabla 28

Resultados obtenidos del test de Boston

	N	%
NORMAL (1)	151	91.52%
TENSO (2.3)	10	6.06 %
TAFIGA (4.5)	4	2,42 %

4.4 Comparación entre el dispositivo y el Cuestionario de Boston.

Con el propósito de evaluar la confiabilidad del Dispositivo Electrónico como Apoyo a la Prevención del Síndrome del Túnel Carpiano (STC) en comparación con un instrumento subjetivo de autopercepción en este caso, el Test de Boston se realizó una comparación porcentual entre los

tres estados reportados: normal, tenso y fatiga como se muestra en la tabla 30. Esta comparación tuvo como objetivo identificar el grado de concordancia entre los datos registrados por el dispositivo y las percepciones subjetivas expresadas por los participantes mediante autoevaluaciones.

Al analizar los resultados obtenidos en diferentes momentos de la semana tanto al inicio y final de la jornada laboral como durante los días intermedios se identificaron patrones consistentes que permiten inferir una adecuada validez del sistema de monitoreo automático implementado.

Ambos métodos reconocen al estado Normal como el más frecuente, aunque con distinta magnitud. Las discrepancias observadas justifican el uso de métodos estadísticos para determinar el grado de concordancia entre ambas fuentes de información.

Para analizar la similitud de las mediciones, se calculó la diferencia absoluta porcentual entre el dispositivo y el test para cada estado.

$$\textit{Diferencia Absoluta} = |P_{\textit{dispositivo}} - P_{\textit{test}}| \quad (4)$$

Esta ecuación es utilizada frecuentemente en análisis de error y validación de instrumentos, permite cuantificar el grado de desacuerdo entre dos mediciones.

Tabla 29

Tabla comparativa entre el dispositivo y el cuestionario de Boston

ESTADO	DISPOSITIVO	TEST	DIFERENCIA ABSOLUTA
NORMAL	78.33	91,52	13.19
TENSO	13.33	6,06	7.27
FATIGA	8.34	2,42	5.92

4.5 Cálculo del índice de concordancia

Con el fin de obtener un valor global de coincidencia entre los métodos, se utilizó el siguiente cálculo:

$$\text{Índice de concordancia} = 100\% - \frac{\sum_{i=1}^n |P_{\text{dispositivo},i} - P_{\text{test},i}|}{n} \quad (5)$$

Donde:

- **n** es el número de categorías (en este caso, 3)
- **P** representa los porcentajes por cada categoría.
- **Índice $\geq 90\%$** → Alta concordancia.
- **Índice 70–89%** → Concordancia moderada.
- **Índice $< 70\%$** → Baja concordancia.

Aplicando la fórmula:

$$\frac{13.19 + 7.27 + 5.92}{3} = 8.79\% \quad (6)$$

$$\text{Índice de concordancia} = 100\% - 8.79\% = 91.21\%$$

El índice de concordancia calculado, del **91.21%**, indica una **alta coincidencia** entre las mediciones realizadas por el dispositivo y el test. Esto valida la confiabilidad del dispositivo como herramienta objetiva de evaluación del estado físico-emocional.

4.6 Viabilidad Técnica Económica

En esta sección se describen los principales costos asociados al desarrollo e implementación del proyecto, clasificados en cuatro categorías: hardware, software,

infraestructura y desarrollo. Los costos de hardware corresponden a los componentes físicos del dispositivo, mientras que los de software incluyen programas y herramientas utilizadas. La infraestructura abarca los recursos computacionales utilizados, y el desarrollo contempla la mano de obra técnica empleada durante todo el proceso.

4.6.1. *Costo General de dispositivo.*

Para estimar el costo total del desarrollo del dispositivo, como se detalla en la Tabla 31, se han considerado tanto los componentes de hardware empleados como los gastos derivados de las distintas etapas de diseño, implementación y pruebas del sistema. Además, se incluyen los recursos utilizados en infraestructura y procesos de manufactura, con especial énfasis en aquellos relacionados con la fabricación del case del dispositivo.

Tabla 30

Costo General del dispositivo

COSTO DEL SISTEMA		
HARDWARE		
COMPONENTE	CANTIDAD	SUBTOTAL (USD)
Grove - EMG Detector	1	\$ 28.00
Grove - GSR sensor	1	\$ 10.90
Grove Base Shield V2.0 for	1	\$ 3.50
Arduino		
Seeeduino V4.3 (ATMega328P)	1	\$ 7.60
Batería 3.7V 3300mA	2	\$ 12
Switch	1	\$ 0.30
Elevador de voltaje	1	\$ 2
Cargador de batería	1	\$ 1

Porta baterías	1	\$ 1
TOTAL		\$ 66.3
SOFTWARE		
SOFTWARE	CANTIDAD	SUBTOTAL
Arduino IDE	1	\$ 0
Anaconda (Jupyter)	1	\$ 0
Visual Studio Code	1	\$ 0
React Native	1	\$0
TOTAL		\$ 0
INFRAESTRUCTURA		
INFRAESTRUCTURA	CANTIDAD	SUBTOTAL
Case 3D del dispositivo	1	\$ 16
TOTAL		\$ 16
TOTAL (HARDWARE + SOFTWARE + INFRAESTRUCTURA)		\$ 82.3

Cabe destacar que el monto total estimado de **\$82,3 USD** corresponde exclusivamente al costo directo de los componentes materiales y la infraestructura necesaria para el desarrollo del prototipo. Esta cifra no contempla los gastos asociados a la importación de componentes, tales como sensores biomédicos o microcontroladores, los cuales pueden variar significativamente en función del Courier de preferencia, impuestos aduaneros y tiempos de entrega.

Una vez desglosados los componentes utilizados en la tabla de costos anterior, en la Tabla 32 se presentan de forma general los costos totales asociados al desarrollo del dispositivo. Esta tabla resume los gastos estimados considerando tanto los materiales como los elementos

electrónicos involucrados en su construcción, permitiendo así obtener una visión clara del presupuesto necesario para su implementación.

Tabla 31

Resumen del costo general del dispositivo

COSTO GENERAL DE DISPOSITIVO	
Tipo de Costo	Subtotal (USD)
Costo de hardware	\$ 66.3
Costo de software	\$ 0
Costo de infraestructura	\$ 16
TOTAL (USD)	\$ 82.3

4. Conclusiones y Recomendaciones

4.4.1 Conclusiones

En base a la fundamentación teórica y al desarrollo del sistema, se concluye que fue esencial consolidar un marco teórico robusto que abordara tanto los aspectos médicos y ergonómicos del síndrome del túnel carpiano incluyendo sus causas, síntomas y métodos de prevención como los principios fundamentales de las redes neuronales. Esta investigación integral permitió comprender la problemática desde una perspectiva multidisciplinaria, brindando una base sólida para el diseño de un sistema capaz de detectar señales tempranas asociadas al STC mediante técnicas de inteligencia artificial.

El sistema de detección desarrollado cumple con los requerimientos funcionales establecidos, actuando como una herramienta de prediagnóstico no invasiva, capaz de identificar patrones de molestias físicas asociadas al síndrome del túnel carpiano. El uso de redes neuronales

permitió que el modelo aprendiera y clasificara señales fisiológicas relevantes, demostrando un alto nivel de precisión en la identificación de señales tempranas derivadas de movimientos repetitivos.

Se logró diseñar una aplicación visual intuitiva, que facilita la interacción del usuario con el sistema. Este componente fue clave para asegurar una experiencia de uso accesible, permitiendo que el dispositivo sea manejado incluso por usuarios sin conocimientos técnicos. La claridad de la visualización y la retroalimentación inmediata contribuyen a una mejor comprensión de los datos obtenidos y su interpretación para la toma de decisiones preventivas.

Durante la fase de pruebas e implementación en un entorno real, el sistema demostró ser funcional, confiable y adaptable. La precisión alcanzada durante el entrenamiento, validada en condiciones reales, respalda su aplicabilidad en escenarios laborales donde se realizan movimientos repetitivos, generando un impacto positivo en la prevención del STC.

A pesar de ejecutarse en un entorno de hardware limitado sin GPU dedicada, el sistema demostró que es posible implementar modelos de redes neuronales con eficiencia. La estructura ligera del modelo MLP permitió alcanzar una precisión del 94.73 % y una pérdida mínima del 14.07 %, deteniéndose el entrenamiento en la época 240 mediante la técnica de *EarlyStopping*. Estos resultados reflejan una excelente capacidad de generalización sin signos de sobreajuste, lo cual valida la viabilidad de este tipo de soluciones en plataformas de bajo costo y recursos limitados.

El dispositivo logró integrar de manera eficaz sensores fisiológicos como el EMG (electromiografía) y GSR (respuesta galvánica de la piel), lo que permitió la adquisición y análisis de datos en tiempo real sobre el estado físico del usuario. Gracias al procesamiento de estos datos mediante redes neuronales entrenadas con etiquetas adecuadas, el sistema pudo clasificar con

precisión los niveles de molestias físicas derivadas de movimientos repetitivos. Esta integración tecnológica representa un avance significativo para la salud ocupacional, ya que ofrece una herramienta de monitoreo y prediagnóstico accesible y de bajo costo.

4.4.2 Recomendaciones

Se recomienda investigar e implementar técnicas avanzadas de procesamiento de señales fisiológicas (como filtrado adaptativo o transformadas de wavelet), así como ajustar los hiperparámetros del modelo (tasa de aprendizaje, regularización, tamaño del batch) con el fin de reducir la pérdida de validación y mejorar la precisión en la detección.

Se sugiere realizar el entrenamiento y ajuste de modelos en plataformas que brinden soporte completo para Python, como Google Colab o entornos locales con Anaconda, ya que facilitan el uso de bibliotecas especializadas como TensorFlow y Keras, permitiendo un desarrollo más eficiente y escalable.

Es recomendable contar con la supervisión de especialistas en salud ocupacional o fisioterapia para asegurar que el etiquetado de los datos sea preciso y representativo.

Se recomienda implementar el dispositivo como parte de un programa de seguimiento de la salud física en trabajadores administrativos, permitiendo detectar signos tempranos de fatiga o sobrecarga muscular y prevenir la aparición del síndrome del túnel carpiano en el mediano y largo plazo.

Bibliografía

- Anaconda. (2024). *Las mejores herramientas de ciencia de datos en 2024*.
<https://www.anaconda.com/guides/data-science-tools>
- Arduino. (2017). *¿Qué es Arduino?* <https://www.arduino.cc/>
- Arduino. (2025). *Software*. <https://www.arduino.cc/en/software/>
- Arthritis Research UK. (s/f). *Carpal tunnel syndrome*.
https://www.csp.org.uk/system/files/2_carpal_tunnel.pdf
- Asociación Internacional de Ergonomía. (2000). *¿Qué es la ergonomía?* <https://iea.cc/about/what-is-ergonomics/>
- Ayala, S. (2018). *PREVALENCIA DE SÍNDROME DE TÚNEL CARPIANO EN PUESTOS ADMINISTRATIVOS*. <http://localhost:8080/xmlui/handle/123456789/2824>
- Balbastre, M., Andani, J., Garrido, R., & López, A. (2016). Análisis de factores de riesgo laborales y no laborales en Síndrome de Túnel Carpiano (STC) mediante análisis bivariante y multivariante. *Revista de la Asociación Española de Especialistas en Medicina del Trabajo*, 25(3), 113–194.
- Banda, H. (2014). *Inteligencia Artificial: Principios y Aplicaciones*.
<https://www.researchgate.net/publication/262487459>
- Bayo, J. (2023). *Propuesta de uso del paradigma de coreografía de procesos para crear sistemas de eSalud en entornos heterogéneos*. Universitat Politècnica de València.
- Benítez, R., Cencerrado, A., Gerard, B., & Kanaan, E. S. (2018). *Inteligencia artificial avanzada*. Universitat Oberta de Catalunya.
- Bolaños, A. (2015). *Incidencia del síndrome del túnel carpiano por movimientos repetitivos en cajeras, que acuden al Centro de Rehabilitación Athrez de la ciudad de Latacunga*. Universidad Técnica de Ambato.

- Buechley, L., Eisenberg, M., Catchen, J., & Crockett, A. (2008). The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. *Conference on Human Factors in Computing Systems*, 423–432.
- Cepeda, J., Meijome, X., & Santillán, A. (2012). Innovaciones en salud y tecnologías: las cosas claras. *Revista Enfermería CyL*, 4(1), 28–32.
- Chamorro J. (s/f). *Sensores Biofísicos*.
- Chavarría, R. (1989). NTP 242: Ergonomía: análisis ergonómico de los espacios de trabajo en oficinas. *Instituto nacional de seguridad e higiene en el trabajo*, 7.
- Código del trabajo, Pub. L. No. Registro Oficial Suplemento 167 de 2005 (2005).
- Confederación Regional de Organizaciones Empresariales de Murcia. (s/f). *Prevención de Riesgos Ergonómicos*. <https://portal.croem.es/prevergo/formativo/5.pdf>
- Constitución de la República del Ecuador (2008).
- Díaz, K. (2023). *Redes neuronales artificiales aplicada al estudio de perfiles de carga eléctrica en alimentadores primarios de una arquitectura de distribución* [Universidad Técnica de Cotopaxi]. <http://repositorio.utc.edu.ec/handle/27000/10338>
- Espressif Systems. (2025). *ESP32*. <https://www.espressif.com/en/products/socs/esp32>
- Fernández, M. (2011). *Aplicación de lenguas electrónicas en la monitorización de procesos de biosorción de metales* [Universitat Politècnica de Catalunya]. <http://hdl.handle.net/2099.1/12244>
- Fidalgo, M., & Nogareda, C. (2001). NTP 602 : El diseño ergonómico del puesto de trabajo con pantallas de visualización : el equipo de trabajo. *Instituto Nacional de Seguridad e Higiene en el Trabajo.*, 1–8.

http://www.insht.es/InshtWeb/Contenidos/Documentacion/FichasTecnicas/NTP/Ficheros/601a700/ntp_602.pdf

FisioPro. (2018). *La importancia de una buena postura.*

<https://www.fisioterapiaprofesional.com/importancia-de-una-buena-postura/>

Fuel, Y. (2017). *Conocimientos, actitudes y prácticas del síndrome de túnel del carpo en el personal administrativo Universidad Técnica del Norte, Ibarra 2017* [Universidad Técnica del Norte]. [https://repositorio.utn.edu.ec/bitstream/123456789/7340/1/06 ENF 838 TRABAJO DE GRADO.pdf](https://repositorio.utn.edu.ec/bitstream/123456789/7340/1/06_ENF_838_TRABAJO_DE_GRADO.pdf)

Gavilema, J., & Mullo, E. (2014). *Departamento de eléctrica y electrónica* [Universidad de las Fuerzas Armadas ESPE]. <http://repositorio.espe.edu.ec/bitstream/21000/9158/1/T-ESPEL-ENI-0333.pdf>

Ghasemi-rad, M., Nosair, E., Vegh, A., Mohammadi, A., Akkad, A., Lesha, E., Hossein, M., Sayed, D., Davarian, A., Maleki-Miyandoab, T., & Hasan, A. (2014). A handy review of carpal tunnel syndrome: From anatomy to diagnosis and treatment. *World Journal of Radiology*, 6(6), 284–300. <https://doi.org/10.4329/wjr.v6.i6.284>

Gorostiza, J. (2015). *Sensibilidad y especificidad de las pruebas de Phalen y Tinel para el diagnóstico del síndrome del túnel del carpo Una revisión sistemática* [Universidad de Cantabria]. <https://repositorio.unican.es/xmlui/handle/10902/15592>

Gualotuña, D. (2022). *Investigación bibliográfica sobre la incidencia del Síndrome del Túnel Carpiano debido al uso excesivo del mouse y teclado en oficinistas, en edades comprendidas entre 20 a 60 años* [Universidad Central del Ecuador]. <https://www.dspace.uce.edu.ec/server/api/core/bitstreams/a8518bd5-9032-4614-8efb-5d2c1327807c/content>

- Guevara, R. (2008). *Sentencias básicas usadas en la programación de computadores*. Fondo Editorial ITM.
- Ilunga, M., & Stephenson, D. (2005). *Infilling streamflow data using feed-forward back-propagation (BP) artificial neural networks: Application of standard BP and pseudo MacLaurin power series BP techniques*. <http://www.wrc.org.za>
- Instituto de Salud Pública Chile. (2016). *Guía ergonomía trabajo oficina uso PC*. <http://www.ispch.cl/saludocupacional>,
- Jarrín, J. (2020). *Sistema de detección del ángulo articular en los movimientos de miembro superior para evaluación en fisioterapia visión artificial* [Universidad Técnica del Norte]. <http://repositorio.utn.edu.ec/handle/123456789/10295>
- Kafuru, A., Rojas, J., Castañeda, J., Ospina, A., & Rincon, D. (2016). Manejo endoscópico del síndrome de túnel del carpo, una experiencia de 5 años. *Revista de la Universidad Industrial de Santander. Salud*, 48(2), 206–212. <https://doi.org/10.18273/revsal.v48n2-2016005>
- Lepage, C. (2016). *Aplicaciones actuales de la inteligencia artificial y su uso con la tecnología IBM Watson* [Universidad Nacional Mayor de San Marcos]. <https://cybertesis.unmsm.edu.pe/item/80dd6767-88a4-4e0d-859f-b6a50a4c5ade>
- Liu, C. W., Chen, T. W., Wang, M. C., Chen, C. H., Lee, C. L., & Huang, M. H. (2003). Relationship between Carpal Tunnel Syndrome and Wrist Angle in Computer Workers. *Kaohsiung Journal of Medical Sciences*, 19(12), 617–622. [https://doi.org/10.1016/s1607-551x\(09\)70515-7](https://doi.org/10.1016/s1607-551x(09)70515-7)
- McMutual. (2017). *Prevención de riesgos laborales en oficinas y despachos*. https://prevencion.mc-mutual.com/c/document_library/get_file?folderId=0&name=49663

- MedlinePlus. (2023). *Síndrome del túnel carpiano*.
<https://medlineplus.gov/spanish/ency/article/000433.htm>
- Microsoft. (2022). *Visual Studio*.
- Muñoz, R., & Pauline, V. (2010). *Aplicación y comparación de la metodología de diseño Top Down y Bottom Up*.
- Naciones Unidas. (2018). *La Agenda 2030 y los Objetivos de Desarrollo Sostenible: una oportunidad para América Latina y el Caribe*.
- Naranjo, J. (2016). *Determinantes de exposición a movimientos repetitivos y la presencia de Síndrome de túnel carpiano en el personal administrativo de una clínica de la ciudad de Riobamba en el año 2015* [Universidad Tecnológica Equinoccial].
<http://repositorio.ute.edu.ec/handle/123456789/16422>
- Organización Internacional del Trabajo. (2009). *Documento de información técnica sobre las enfermedades que plantean problemas para su posible inclusión en la lista de enfermedades profesionales que figura como anexo de la Recomendación sobre la lista de enfermedades profesionales, 2002 (núm. 194)*.
- Organización Panamericana de la Salud. (s/f). *Objetivos de Desarrollo Sostenible*. OPS.
<https://www.paho.org/es/temas/objetivos-desarrollo-sostenible>
- Pajankar, A. (2017). *Raspberry Pi Image Processing Programming*. Apress.
- Possidoni, C. (2015). Seguridad y soberanía alimentaria: ejes para pensar la realidad de salud de los pueblos originarios. En *XVII Congreso Latinoamericano de Nutrición (SLAN)* (Vol. 65, Número 1, pp. 32–33).
- Pozo, M. (2018). *Estrategias preventivas que contribuyan a la Reducción del síndrome de túnel carpiano en el personal Administrativo del G.A.D Municipal “San Pedro de Huaca”*

[Universidad Regional Autónoma de los Andes].

<https://dspace.uniandes.edu.ec/bitstream/123456789/8950/1/TUTENF039-2018.pdf>

Resolución No. C.D. 513. Reglamento del Seguro General de Riesgos del Trabajo (2016).

Rosado, B. (2023). *Síndrome del Túnel Carpiano. Revisión sistemática de la Literatura*

[Universidad Católica de Cuenca].

<https://dspace.ucacue.edu.ec/server/api/core/bitstreams/b96a3e87-a052-48c7-885e-17956a82d744/content>

Salas, D., & Díaz, L. (2016). *Factores de riesgo asociados a alteraciones osteomusculares de la muñeca en trabajadores del área administrativa de una entidad promotora de salud del departamento de Córdoba durante el año 2016* [Universidad Libre Seccional Barranquilla].

<https://repository.unilibre.edu.co/bitstream/handle/10901/10679/1.143.116.684.pdf?sequence=1&isAllowed=y>

Sarango, D. (2019). *Incidencia de las enfermedades profesionales en el Ecuador 2015-2017*

[Universidad Internacional SEK]. <http://repositorio.uisek.edu.ec/handle/123456789/3537>

Seeed Studio. (2024a). *Grove - GSR Sensor*. https://wiki.seeedstudio.com/Grove-GSR_Sensor/

Seeed Studio. (2024b). *Seeeduino V4.3 (ATMega328P)*. [https://www.seeedstudio.com/Seeeduino-](https://www.seeedstudio.com/Seeeduino-V4-2-p-2517.html?srltid=AfmBOop92Y3RXjR_qWbQjKXFqKgN-57ucTFRSnbI-Wr1m6sScKoccMT5)

[V4-2-p-2517.html?srltid=AfmBOop92Y3RXjR_qWbQjKXFqKgN-57ucTFRSnbI-](https://www.seeedstudio.com/Seeeduino-V4-2-p-2517.html?srltid=AfmBOop92Y3RXjR_qWbQjKXFqKgN-57ucTFRSnbI-Wr1m6sScKoccMT5)

[Wr1m6sScKoccMT5](https://www.seeedstudio.com/Seeeduino-V4-2-p-2517.html?srltid=AfmBOop92Y3RXjR_qWbQjKXFqKgN-57ucTFRSnbI-Wr1m6sScKoccMT5)

Seeed Studio. (2025). *Grove - Detector EMG*. [https://wiki.seeedstudio.com/Grove-](https://wiki.seeedstudio.com/Grove-EMG_Detector/)

[EMG_Detector/](https://wiki.seeedstudio.com/Grove-EMG_Detector/)

Stallman, R. (2004). *Software libre para una sociedad libre*. GNU.