



INSTITUTO POLITÉCNICO NACIONAL ESCUELA SUPERIOR DE CÓMPUTO

ESCOM

Trabajo Terminal

**“Sistema embebido como herramienta de apoyo a la
comunicación para pacientes con afecciones del habla”**

2023-A013

Presenta

Luis Eduardo Valle Martínez

Directores

***M. en C. Rodolfo Romero
Herrera***

Dr. Jesús Yaljá Montiel Pérez





**INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO
SUBDIRECCIÓN ACADÉMICA**



No. de TT: 2023-A013

Documento Técnico

“Sistema embebido como herramienta de apoyo a la comunicación para pacientes con afecciones del habla”

Presenta

Luis Eduardo Valle Martínez¹

Directores

M. en C. Rodolfo Romero Herrera

Dr. Jesús Yaljá Montiel Pérez

RESUMEN

En el trabajo se presenta una propuesta de herramienta de apoyo para facilitar la comunicación verbal de pacientes que recientemente han sufrido un traumatismo que ha dañado el lóbulo izquierdo del cerebro, desarrollando como resultado alguna de las siguientes condiciones: afasia, disartria, apraxia.

La herramienta se propone como un sistema embebido conformado por un dispositivo sensor tipo *wearable* para la captación de gestos motrices, y una unidad principal de procesamiento incluyendo un minicomputador Raspberry Pi 4.

Para el reconocimiento de los gestos se plantea y utiliza un modelo propuesto basado en la generación de vectores con características de distancia DTW entre series de tiempo en representación SAX; Utilizándose como modelo clasificador una Máquina de Soporte Vectorial.

¹ lvalle212@gmail.com

Advertencia

“Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional, a partir de datos y documentos con derecho de propiedad y por lo tanto, su uso quedará restringido a las aplicaciones que explícitamente se convengan.”

La aplicación no convenida exime a la escuela su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.

Información adicional sobre este reporte técnico podrá obtenerse en:

La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 55-57296000, extensión 52000.

ÍNDICE

INTRODUCCIÓN	6
CAPÍTULO I: PLANTEAMIENTO DEL TRABAJO	7
Problemática	7
Propuesta de solución	8
Justificación	9
Objetivos	10
CAPÍTULO II: ESTADO DEL ARTE Y METODOLOGÍA	11
Estado del arte	11
Metodología	13
CAPÍTULO III: MARCO TEÓRICO	15
Antecedentes	15
La herramienta embebida	16
Componentes lógicos	17
Arquitectura de la solución	18
CAPÍTULO IV: CÓDIGO MOTRIZ	19
Marco teórico	19
Fonemas del español	21
Análisis del código motriz	22
Clases textuales	23
Elección de los gestos motrices	23
Catálogo de gestos motrices	25
Animación de gestos	27
Fonemas	28
CAPÍTULO V: MODELO PARA RECONOCIMIENTO DE GESTOS MOTRICES	30
Marco teórico	30
DTW	30
PAA	31
SAX	32
DBA	32
Método para el reconocimiento de gestos	33
Fase de preprocesamiento de los datos	33
Fase de entrenamiento	34
Fase de predicción	35
Implementación en la solución	35
Conjunto de datos	37
Herramienta web para la recolección del conjunto de datos	38
Traducción de gestos	40
CAPÍTULO VI: NODO SENSOR	42
Marco teórico del nodo sensor	42
Micro:bit	42
Módulo ESP-01 Wi-Fi	42

Modos de funcionamiento	43
Módulo TP4056	43
Análisis del nodo sensor	43
Diseño del nodo sensor	44
Portabilidad	44
Inicio y finalización del muestreo	45
Comunicación	45
Batería	46
Circuito	47
Carcasa del prototipo	49
Componentes lógicos	51
Creación conjunto de datos	52
Traducción de gestos	53
CAPÍTULO VII: NODO DE PROCESAMIENTO	55
Marco teórico	55
RESTful API	55
Servidor TTS Text-to-Speech	56
Servicios Text-to-Speech en la nube	56
Google Text-to-Speech	56
Voces	56
Precio	56
Desventajas	56
Microsoft Azure Text-to-speech	57
Voces	57
Precio	57
Desventajas	57
Amazon Polly	57
Voces	57
Precio	57
Desventajas	57
Análisis del nodo de procesamiento	58
Diseño del nodo de procesamiento	58
APIs Web	59
API de conjunto de datos	60
API traducción de gestos	61
CONCLUSIONES	64
TRABAJO A FUTURO	66
REFERENCIAS	67
GLOSARIO DE TÉRMINOS	72
APÉNDICE A	73
Características del micro:bit	73
Características del módulo Wi-Fi ESP-01	73
Características del módulo TP4056	74

INTRODUCCIÓN

El lenguaje humano, expresado a nivel oral mediante el habla, se alcanzó en algún punto de la evolución del canal vocal-auditivo en el humano, lo que nos otorgó la posibilidad de crear sonidos y a través de la especialización como especie en la actividad, se asignó una semántica a los sonidos generados permitiendo la asociación a significados. De tal relevancia es el habla como medio para la comunicación, que como recurso evolutivo clave del hombre, ha permitido alcanzar el desarrollo moderno de las áreas del conocimiento humano.

Existen condiciones físicas y mentales que dificultan a una porción de la sociedad desempeñar de forma óptima el lenguaje hablado, siendo 3 condiciones las más comunes y que suelen desarrollarse después de sufrir un accidente cerebrovascular o un traumatismo craneoencefálico, afectando más comúnmente al lóbulo izquierdo pero derivando en afecciones distintas en función de la parte específica lesionada en el cerebro: las afasias, las disartrias y las apraxias.

En este trabajo se estudia y desarrolla una solución que propone una herramienta de naturaleza embebida, además de una metodología y método para el reconocimiento de gestos motrices, que buscan proporcionar apoyo a la comunicación verbal para el tipo de trastornos mencionados a través de la reproducción de gestos motrices que se traducen a discurso hablado.

El documento se divide en 7 capítulos en los que se expone el proceso de investigación, análisis, diseño y desarrollo de la propuesta de solución con la que el autor aspira a obtener a nivel licenciatura el título de Ingeniero en Sistemas Computacionales por parte de la Escuela Superior de Cómputo perteneciente al Instituto Politécnico Nacional.

Los 3 primeros capítulos se utilizan para plantear las bases del trabajo y explicar de manera general el enfoque que se utiliza para abordar la problemática y proponer una solución. Los 4 capítulos subsecuentes explican de forma particular cada uno de los principales componentes y productos resultantes de la solución. Esta organización alternativa del documento se encuentra inspirada en la división de componentes, conveniente en proyectos mixtos de desarrollo de software y hardware, planteando así un orden lógico y secuencial de la explicación para una mejor comprensión del lector acerca de los productos resultantes del trabajo.

La Tabla 1 indica el número de capítulos y el contenido que se desarrolla en este relativo al componente que aborda.

Tabla 1: Organización de los capítulos desde el 4° y las temáticas que desarrollan.

Capítulo	Temática
CAPÍTULO IV	Representación de los gestos, fonemas del español, clases textuales, gestos y código motriz.
CAPÍTULO V	Modelo para el reconocimiento de gestos motrices.
CAPÍTULO VI	Nodo sensor
CAPÍTULO VII	Nodo de procesamiento

CAPÍTULO I: PLANTEAMIENTO DEL TRABAJO

Problemática

Las afecciones del habla en personas que han superado la etapa de niñez (adolescentes, adultos jóvenes, adultos y adultos mayores) suelen originarse principalmente por accidentes o lesiones que dañaron alguna zona de la masa encefálica, más comúnmente afectando las funciones del lenguaje cuando se localiza en el hemisferio izquierdo del cerebro. Estos daños pueden darse por la falta de circulación del torrente sanguíneo, daño directo en las conexiones entre hemisferios, etc[1]. Algunos de los trastornos más comunes y a los que este trabajo busca ofrecer una herramienta de apoyo son:

- Afasias
 - Afasia de Broca[1]: El área de Broca es una región localizada en el lóbulo cerebral izquierdo y está relacionada con el uso del lenguaje. Específicamente la afasia en la que se sufre un daño en esta área, tienen dificultad para la expresión fluida, la pronunciación y modulación del tono de voz. La producción de los sonidos correctos y encontrar las palabras correctas suelen ser trabajos laboriosos, sin embargo la comprensión del habla de otras personas es relativamente buena, por lo que entender textos o lenguaje oral en comparación con su capacidad de hablar y escribir se encuentra mejor conservada.
 - Afasia motora transcortical[1]: Parecida a la afasia de Broca en la dificultad del paciente para la emisión de un lenguaje fluido y coherente conservando una relativa buena comprensión de lenguaje, esta afectación difiere en el hecho de que los pacientes si son capaces de repetir lo que se les dice.
- Apraxia
 - Apraxia bucofacial u orofacial[2]: Incapacidad de realizar movimientos faciales a voluntad, como pasar la lengua por los labios, silbar, toser o guiñar el ojo.
 - Apraxia verbal[1,2]: Dificultad para coordinar los movimientos de la boca y del habla.
- Disartria

Trastorno de la ejecución motora del habla, derivado de un problema neurológico debido a la presencia de un accidente cerebrovascular, traumas craneoencefálicos u otras lesiones cerebrales[1]. Entre los síntomas se tienen el habla entrecortada jadeante, irregular, imprecisa o monótona, lenta, o rápida y "entre dientes", entonación anormal, cambios del timbre de voz, ronquera, babeo o escasez del control de la saliva y la movilidad limitada de la lengua, los labios y la mandíbula.[3,4].

Propuesta de solución

La solución de este trabajo se propone como un prototipo de dispositivo sensor tipo *wearable*, similar a un reloj inteligente en su portabilidad y sencillez de uso, que permite el muestreo de movimientos asociados a un código y que hace posible la conformación de palabras. Mediante comunicación inalámbrica el sensor portable comparte los gestos recolectados y un nodo de procesamiento se encarga de identificar el tipo de gesto realizado, conformar las entradas en palabras y texto para finalmente reproducirlo con voz (Figura 1).

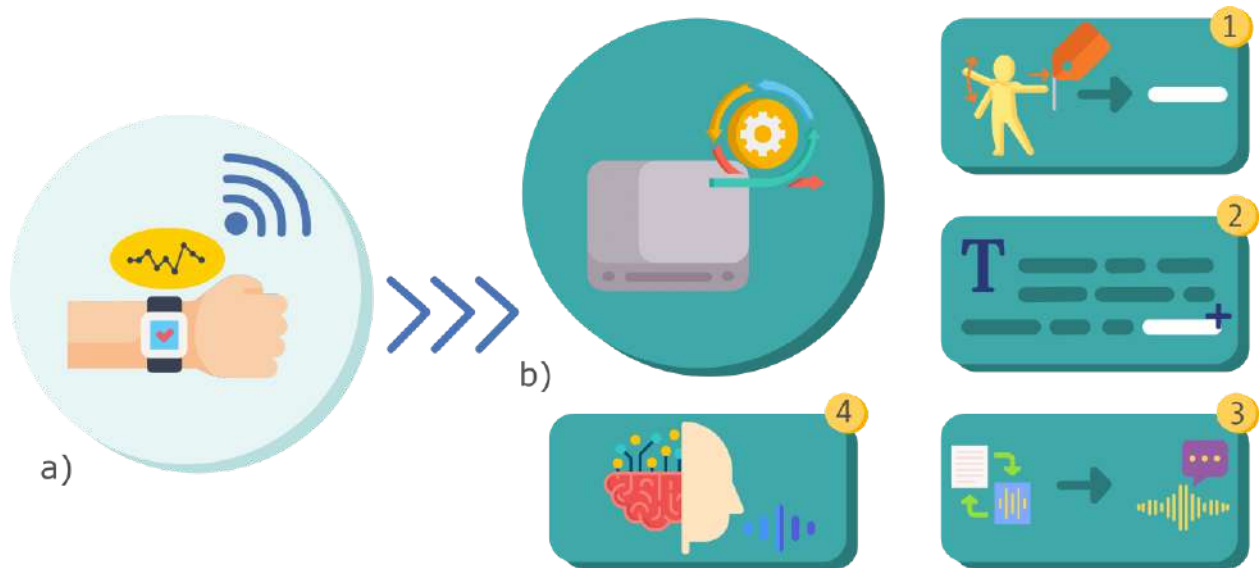


Figura 1: Esquema conceptual de la operación de la solución propuesta.

- a) Nodo sensor recolector del gesto motriz
- b) Nodo de procesamiento
 - 1. Etiquetado o clasificación de los gestos en una clase textual.
 - 2. Conformación de palabras y texto con las clases textuales.
 - 3. Traducción del texto en archivo de voz.
 - 4. Reproducción del mensaje con voz sintetizada.

El prototipo sensor o también llamado nodo sensor, está constituido por 4 elementos primordiales: La unidad de procesamiento principal encargada de orquestar la captura de las mediciones, comunicación y demás funciones del dispositivo; El sensor acelerómetro o IMU (Unidad de Medida Inercial del inglés *Inertial Measurement Unit*) que hace posible medir como una magnitud numérica el cambio de la aceleración sobre el prototipo en los 3 ejes dimensionales; El dispositivo de comunicación inalámbrica (Bluetooth o Wi-Fi) y finalmente la batería como fuente de energía portátil.

El nodo de procesamiento es dentro de la solución el elemento con mayor capacidad computacional, razón por la que los procesos y algoritmos más complejos, como el modelo de aprendizaje máquina clasificador de gestos, son delegados a este nodo limitando las funciones del prototipo a interacción con el usuario.

Los gestos muestreados son descritos mediante un código motriz especialmente diseñado para la solución y que relacionan un movimiento con una clase textual (palabra o fonema). El objetivo de este código es utilizar movimientos cortos y sencillos de reproducir para la conformación de texto.

Finalmente es necesario identificar el nuevo movimiento realizado por el usuario del conjunto completo de gestos del código motriz, y para este fin se desarrolla un método de reconocimiento de gestos de naturaleza motriz, se explica en que consiste en capítulos posteriores y de forma más detallada en el artículo “Clasificador de gestos motrices utilizando vectores de características SAX-DTW” [5].

Justificación

Los pacientes que recientemente han sufrido algún trastorno en el habla a raíz de un accidente se enfrentan a una condición nueva que perjudica su autonomía e integración social derivada de la pérdida del lenguaje y la comunicación verbal como habilidades esenciales en la vida humana. Los casos de personas con estos trastornos encaran una realidad en el que la rehabilitación dependerá de la causa, tamaño y ubicación de la lesión, habiendo casos donde una recuperación funcional de las capacidades comunicativas es imposible y se obliga a la intervención de sistemas aumentativos o alternativos de la comunicación para los pacientes con condiciones agravadas[6].

El conjunto de herramientas tecnológicas con mayor disponibilidad y que pueden adaptarse a este tipo de problemáticas son aquellas basadas en dispositivos y guantes traductores de lenguaje de señas [7,8]. El inconveniente más importante de estas tecnologías surge de la problemática principal de la que parten estos desarrollos y en los que la condición esencial es el dominio de un lenguaje señado, por ejemplo el Lenguaje de Señas Mexicano(LSM), y más sin embargo los sectores de la población que lo practican se ven muy disminuidos en contraste con una mayoría de la sociedad mexicana que no lo practica e ignora casi en su totalidad.

Este trabajo busca entonces proponer una alternativa tecnológica para la comunicación verbal que considere en su diseño algunos de los síntomas de los trastornos mencionados, de manera que sea útil como herramienta de primer contacto con pacientes recientemente afectados y en los que la adopción temporal de un código sencillo para la comunicación es viable; Contrastando el esfuerzo y tiempo requerido para la adquisición de un lenguaje señado como el LSM que equivaldría al aprendizaje de un nuevo idioma.

Una ventaja importante con la que cuenta la propuesta de solución es la escalabilidad y fácil adaptación a dispositivos de comodidad distintos al prototipo *wearable* propuesto. Gracias a la metodología utilizada en el reconocimiento de los gestos y las tecnologías con las que se implementa la solución, se abre la posibilidad de adaptar sin mayor esfuerzo dispositivos comerciales de última generación para cumplir con el rol que desempeña el prototipo sensor, funcionando como una actualización que simplifica la adaptación del usuarios y además provee potencial para agregar más funcionalidades desde el lado del usuario apalancando al gran poder de procesamiento y memoria que incluyen los últimos modelos de relojes inteligentes.

La propuesta de solución integra conocimientos de distintas áreas en el campo tecnológico requeridos para su desarrollo y se enlistan a continuación en la Tabla 2:

Tabla 2: Áreas involucradas en las actividades durante el desarrollo de la solución.

Actividades	Área De Especialidad Involucrada
Diseño y construcción del nodo sensor.	Electrónica digital
Comunicación y programación de las funciones y comportamiento del nodo sensor.	Electrónica digital, Sistemas embebidos
Configuración y comunicación del nodo de procesamiento.	Sistemas embebidos
Programación de servicios y funcionalidades del nodo de procesamiento.	Sistemas embebidos, sistemas web y programación.
Investigación y desarrollo del método para el reconocimiento de gestos.	Aprendizaje máquina, IA y Programación.

Objetivos

Objetivo general

Desarrollar un prototipo embebido útil para pacientes con trastornos del habla como herramienta de apoyo para la comunicación verbal que utiliza gestos motrices para formar palabras que posteriormente son reproducidas mediante voz.

Objetivos específicos

1. Proponer un conjunto de patrones de movimiento como código motriz para su asociación en la conformación de palabras.
2. Implementar un prototipo de nodo sensor tipo *wearable* con el computador micro:bit para el muestreo de gestos motrices.
3. Desarrollar un modelo clasificador utilizando el algoritmo DTW y bibliotecas especializadas para el reconocimiento de gestos motrices.

CAPÍTULO II: ESTADO DEL ARTE Y METODOLOGÍA

Estado del arte

El tipo de desarrollos que abordan una problemática similar a la estudiada aquí, las herramientas traductoras de lengua de señas a voz, cuenta con una amplia bibliografía entre trabajos de titulación, tesis, artículo de investigación etc. Que además no se limitan a trabajos de nivel institucional o nacional, siendo en su cantidad mayor para los trabajos internacionales que proponen una solución propia.

En este sentido dominan 2 técnicas de solución principales que aborda el tema. La primera propone una solución a través de los recursos y técnicas que provee la visión por computadora y análisis de imágenes, sin embargo este tipo de técnicas suelen ser complicadas de implementar pues su puesta en práctica requiere del control parcial o total de algunas variables ambientales, siendo cruciales la iluminación, el contraste con el fondo y la visibilidad. Estas restricciones limitan las soluciones a implementarse únicamente en ambientes controlados, muy lejano de un escenario real cotidiano en el que se plantea utilizar.

Un trabajo de ejemplo que utiliza la visión artificial es el trabajo terminal [7], el cual a través de las imágenes captadas por el dispositivo Kinect se utiliza un modelo de redes neuronales que permiten identificar un número de señas pertenecientes al LSM que son realizadas mediante combinaciones de posiciones en los dedos y mano.

La segunda técnica más socorrida consiste en utilizar un dispositivo hardware que permite sensar y muestrear el movimiento, posición, fuerza, etc. de la mano al realizar un movimiento específico. En este tipo de enfoque, es común la creación de un dispositivo tipo guante que incluya el hardware necesario para el muestreo.

A nivel licenciatura, se tiene un trabajo de titulación que se enfoca en realizar un sistema de apoyo a la comunicación para personas con afecciones en el habla, y que se encarga de realizar una traducción del LSM al español a través de la reproducción sonora de letras y palabras [9].

En el ámbito internacional se han desarrollado trabajos enfocados en la miniaturización y disminución del hardware necesario para la creación de un *wearable* sensor del movimiento de manos y dedos en forma de pulsera[10]. Desarrollado por un equipo de la Universidad de Amrita en la India, ocupan un IMU que alberga un acelerómetro y giroscopio, el cual junto con un arreglo de electrodos *Electromyography* (EMG), les permite a los autores reconocer la contracción de los músculos para el movimiento respectivo de cada dedo.

Otro trabajo realizado en la misma línea de trabajo es el desarrollado en el *National Institute of Technology Pudcherry Karaikal India*[8], que proponen un prototipo de guante con sensores de flexión, acelerómetro y giroscopio. La recopilación de las mediciones se logra con una tarjeta de desarrollo Arduino Nano, de la cual con tecnología Bluetooth se envía a una PC que corre el algoritmo clasificador *Support Vector Machine* (SVM). Como característica adicional este

prototipo tiene la posibilidad de identificar gestos de tanto el Lenguaje de Señas Americano como del Lenguaje de Señas Indio.

En cuanto a los trabajos previos relacionados que respaldan el modelo clasificador de los gestos motrices ‘vectores de atributos SAX-DTW’ [5], se encontraron para la fase de investigación un gran número de artículos útiles que respaldan el desarrollo final del método propuesto, más son 3 los trabajos que proveen inspiración directa por las técnicas y algoritmos empleados, así como sus propios modelos, los cuales ocupa el método original de este trabajo para referencia en la medición y evaluación de desempeño.

El primero de ellos describe el método de referencia aceptado por la comunidad involucrada en la tarea de clasificación de series de tiempo, y que a pesar de su concepto y funcionamiento sumamente sencillo, se mantuvo durante un largo periodo de tiempo como el modelo con una precisión en predicción imbatible. El método descrito en [11] propone la clasificación de secuencias temporales mediante la asignación de clase del vecino más cercano en *K-Nearest Neighbor* (K-NN), con valor $K = 1$, utilizando el algoritmo DTW como función de similitud entre series de tiempo.

El segundo trabajo enfoca su atención en la tarea particular de la clasificación de gestos motrices utilizando dispositivos *wearables* de uso común como el smartwatch *Pebble*. El trabajo de Mezari y Maglogiannis [12] propone una metodología para mejorar la precisión y desempeño del reconocimiento de gestos cuando se utilizan dispositivos básicos como teléfonos inteligentes y *wearables*.

El tercer artículo de importancia para este trabajo es el desarrollado por su autor Kate [13], el cual parte de la idea de explotar la robustez de DTW y propone una metodología para mejorar la precisión en predicción de series de tiempo mediante la conformación de vectores con atributos resultantes de la distancia DTW de una serie con respecto a las instancias de entrenamiento. La gran ventaja que otorga este método con respecto a los 2 anteriores, es la posibilidad de utilizar diferentes y más sofisticados modelos de aprendizaje máquina.

El par de trabajos desarrollados por Mezari, Maglogiannis [12] y Kate [13], sirvieron como inspiración directa para la creación de una metodología híbrida propia y que se incluye en el conjunto de productos resultantes del trabajo.

A continuación se muestra en forma de árbol el número de artículos recopilados y consultados en cada área (Figura 2).

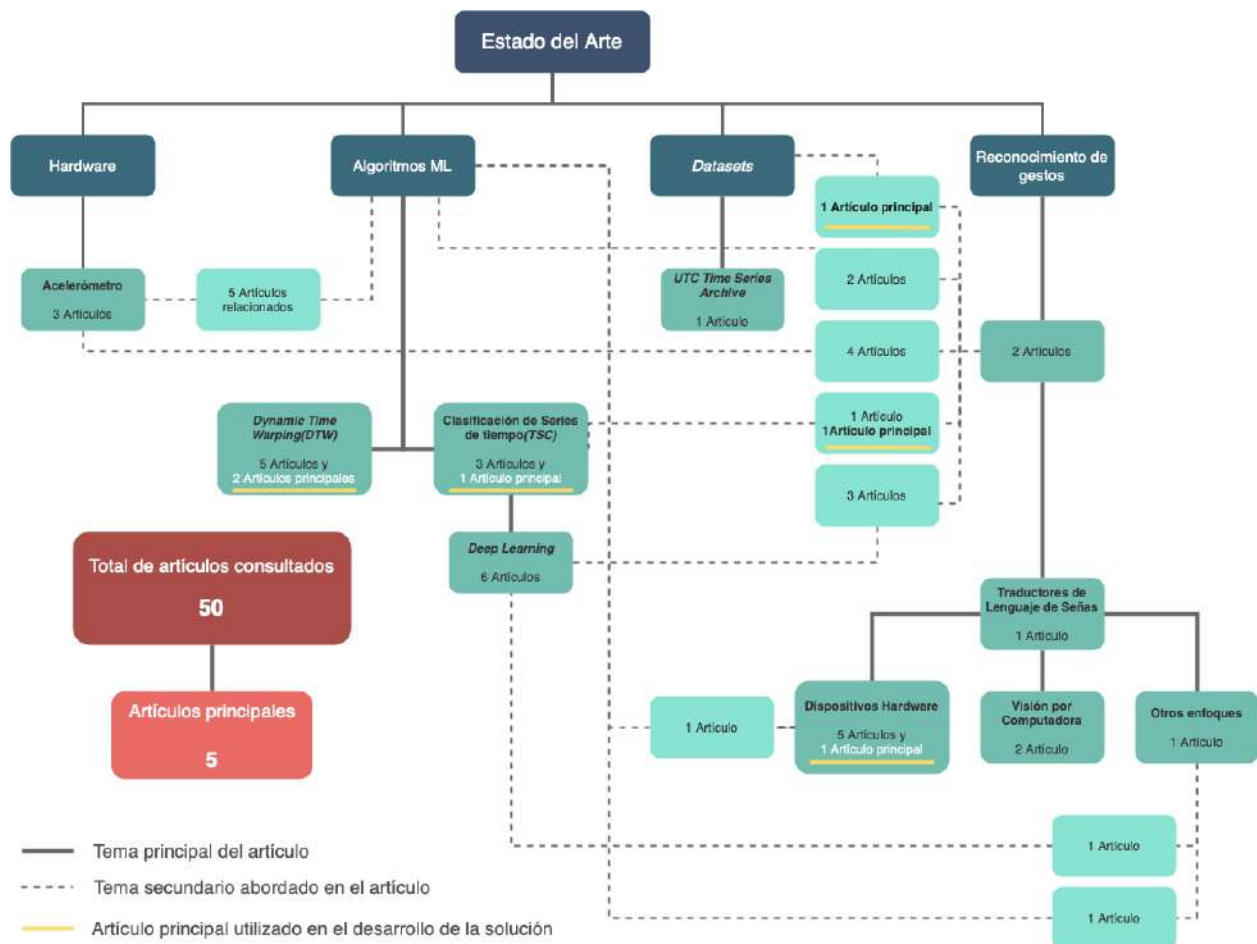


Figura 2: Representación gráfica en forma de árbol de las categorías de trabajos y artículos recopilados y consultados para conformar el estado del arte del trabajo.

Se resalta el número total de consultas como 50, mientras 5 son los trabajos considerados principales (numeraciones resaltadas con subrayado amarillo) ya sea por la metodología, métodos o recursos que aportan e implementan en la solución.

Metodología

El método en V define un procedimiento para el desarrollo de productos para las TIC, siendo importante mencionar que suele ser un estándar para los proyectos de la Administración Federal alemana y defensa. Se trata de un modelo rígido y con una gran cantidad de iteraciones siendo similar al modelo de cascada.

El nombre lo toma de su estructura gráfica que suele asemejarse a una letra V, dividiéndose por su parte izquierda en las actividades que descomponen las necesidades y la creación de las especificaciones del sistema. En su parte derecha, se representa la integración de las piezas y su verificación.

La implementación de este tipo de modelo aporta una serie de ventajas que son importantes como objetivos a alcanzar en el proyecto. Este modelo aporta una mejora y garantía de la calidad gracias a las medidas de control de calidad firmemente integradas.

También minimiza los riesgos del proyecto, permitiendo una detección temprana de las desviaciones y riesgos para la mejora de la gestión de los procesos. Otra gran ventaja es la

reducción de los gastos totales durante el proyecto y sistema de ciclo de vida, logrado gracias al procesamiento transparente a lo largo de todo el ciclo del producto.

Aun con todas estas ventajas este modelo puede ser poco flexible a cambios durante el desarrollo, promoviendo un curso lineal del proyecto, sin embargo si el modelo es entendido y se utiliza correctamente es posible utilizar el modelo V para el desarrollo ágil.

El modelo V define el curso del proyecto en fases individuales cada vez más detalladas(Definición del proyecto):

- **Fase de especificaciones:** Prevé un análisis de las especificaciones del sistema a grandes rasgos
- **Fase funcional:** Se complementa el proyecto con requisitos funcionales y no funcionales para la arquitectura del sistema
- **Fase de diseño:** Se planifican los componentes y las interfaces con un diseño detallado
- **Codificación:** Completadas estas fases inicial el desarrollo de la arquitectura en software

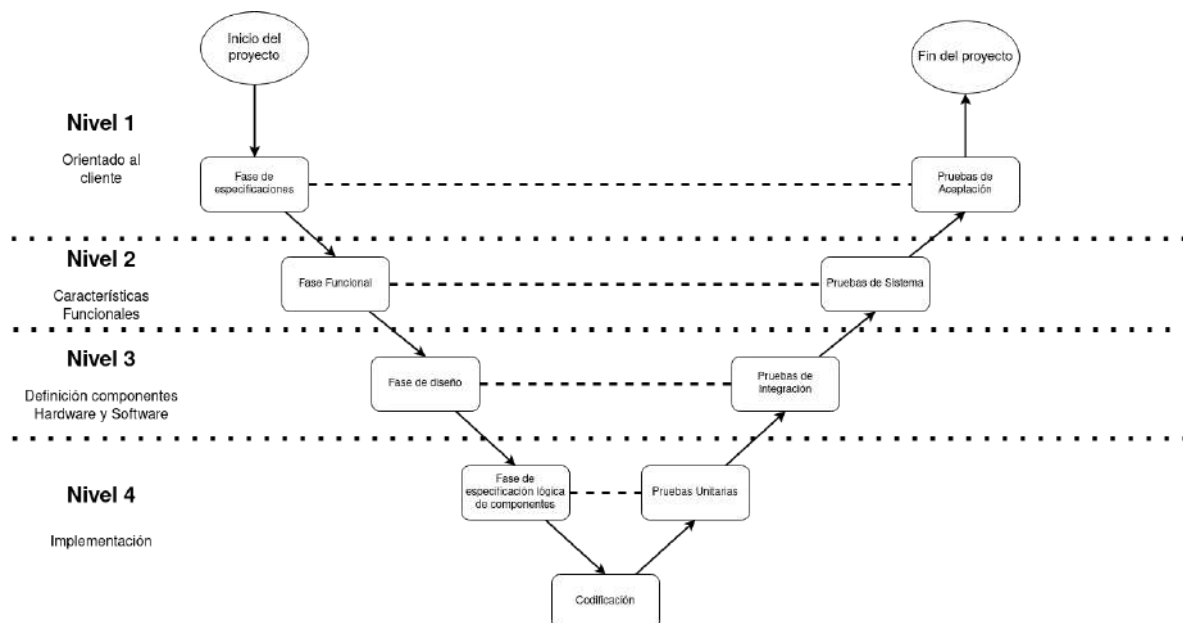


Figura 3: Diagrama mostrando las fases del modelo V.

CAPÍTULO III: MARCO TEÓRICO

Antecedentes

Las herramientas traductoras de lenguaje de señas son adaptables a los trastornos del habla de manera que funcionen como auxiliares o aumentativos para la pronunciación oral a través de una traducción del lenguaje señado a su aproximación en el idioma hablado de origen.

Con 2 técnicas populares, la más similar al desarrollo de este trabajo se encuentran los guantes traductores [8,9], prototipos electrónicos que amalgaman un conjunto de sensores distintos adaptados sobre un guante que hacen posible el muestreo de movimientos de la mano, rotación de muñeca, posición y extensión de los dedos, de tal manera que con la combinación de los valores de las distintas magnitudes medidas es posible identificar las señas realizadas para su traducción.

Los tipos de sensores comúnmente utilizados en estos guantes son los IMU, que permiten medir los cambios en la aceleración del dispositivo (movimiento) y dirección respecto a los ejes cardinales al que apunta el dispositivo (posición); Los sensores de flexión colocados a lo largo de los dedos, permiten medir el grado en el que cada dedo se contrae o extiende, y en respuesta este sensor varía su resistencia requiriendo de una fase de acondicionamiento previo al ingreso del microcontrolador. Finalmente los arreglos de electrodos EMG (Electromiográficos del inglés *Electromyography*) son utilizados para grabar el movimiento de los músculos y parten del principio simple en el que con cada contracción de un músculo una ráfaga eléctrica pequeña se genera y se propaga a través del tejido adyacente[14], lo que permite a este tipo de sistemas identificar el movimiento independiente de cada dedo si se coloca como una banda alrededor de la muñeca.

Surgen 2 grandes inconvenientes derivados de las particularidades de los síntomas experimentados por los pacientes con un trastorno del lenguaje o habla.

El primero de ellos es la complejidad del prototipo que abultado con varios elementos electrónicos en un guante, interfieren con el movimiento natural de la mano del usuario. El segundo es la complejidad de algunas señas y movimientos existentes en los lenguajes señados, los cuales requieren de gestos con posiciones y combinaciones de dedos complejas y precisas. Ambos defectos son especialmente graves en pacientes con apraxias, disartrias y algunos con afasias, los cuales experimentan una disminución parcial en motricidad de algunas partes del cuerpo y extremidades como las manos y dedos.

La herramienta embebida

La herramienta desarrollada intencionalmente limita el número de sensores para el muestreo de los gestos a un acelerómetro o IMU, beneficiando al prototipo con una disminución de la electrónica necesaria y simplificando el ensamble en un diseño poco obstructivo. La principal inspiración para el diseño del prototipo se encuentra en los relojes inteligentes, los cuales por su amplia aceptación se convirtieron en objetos cotidianos con avanzada tecnología en portabilidad.

Se escoge como unidad principal de procesamiento para el prototipo sensor la tarjeta de desarrollo micro:bit, la cual ofrece un buen balance en rendimiento, sensores, tamaño y entorno de desarrollo. De fábrica el micro:bit integra una IMU y un entorno de programación que facilita la recopilación de las mediciones desde el sensor acelerómetro y como una característica extra, nativamente implementa un arreglo de LEDs y un par de botones en la parte frontal que resultan convenientes para utilizarse como una sencilla interfaz con el usuario.

Un par de módulos extra se añaden a la electrónica del prototipo, el primero de ellos se encarga de la comunicación Wi-Fi con el módulo ESP-01 y un módulo para la recarga y descarga segura de la batería de litio con identificador TP4056.

La conexión Wi-Fi, posible por el módulo ESP-01, permite la creación de una API Web que el prototipo sensor consulta compartiendo la serie de tiempo multivariable cada vez que el usuario registra un gesto. Mientras tanto el servicio de la API Web será expuesto por el nodo de procesamiento y esencialmente dispondrá de *endpoints* al que el prototipo podrá consultar para enviar información o recibir datos de configuración.

El nodo de procesamiento incluye una RaspberryPi 4 y es el responsable de ejecutar el método para el reconocimiento del gesto de entrada, concatenar las clases textuales identificadas, consumir y/o proveer el servicio de traducción de texto a voz y finalmente reproducir el archivo de audio resultante.

Para el servicio de traducción se consideran potencialmente 2 servicios: El primero de ellos nombrado TTS, es un proyecto *open source* gratuito del grupo Mozilla que debe ser hospedado localmente, y permite ser consultado a través de una API Web en la red local o una biblioteca especializada. Estas características lo convierten en la opción ideal en situaciones donde no se cuenta con una salida a internet dentro de la red o no se dispone de recursos económicos para la etapa de traducción de texto a voz..

La segunda opción es un servicio de traducción en alguna de las grandes nubes (GCP, Azure, AWS), este tipo de servicios se consumen a través de una API Web o una biblioteca desarrollada por el proveedor, que más sin embargo requisitan el ingreso y disponibilidad del servicio a través de una cuenta en sus plataformas así como una inversión para acceder a sus servicios; A pesar de esto ofrecen una mayor calidad en la sintetización de voz por parte de la IA y proveen una capa gratuita limitada.

Un esquema conceptual del sistema completo se muestra en la Figura 4.

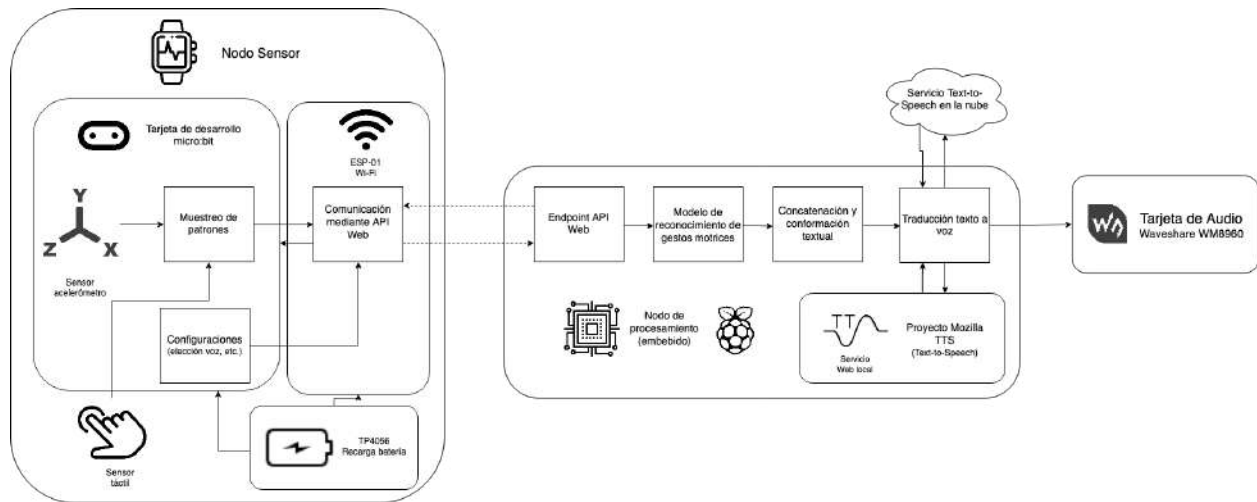


Figura 4: Diagrama conceptual de los componentes y sus principales funciones de la herramienta embebida.

Componentes lógicos

Al igual que sucede con el prototipo hardware, existen los componentes de sistema los cuales describen una parte funcional del sistema, encapsulada, autónoma y que provee 1 o más interfaces para la interacción con otros componentes.

Los diagramas de componentes pertenecientes a la especificación UML 2.0, sirven su propósito principal mostrando las relaciones estructurales existentes entre los componentes de un sistema, permite verificar la implementación de las funcionalidades de un sistema y suele utilizarse para presentar el concepto general de un sistema en construcción.

El siguiente diagrama muestra los componentes lógicos considerados en la solución Figura 5.

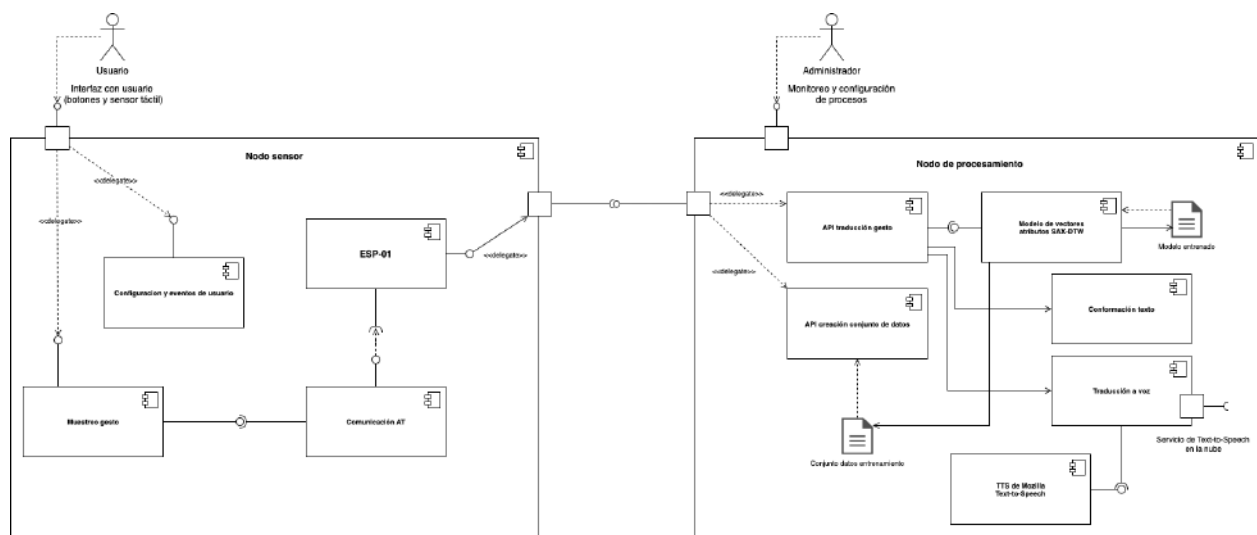


Figura 5: Diagrama de componentes de la solución donde se ilustra la estructura y las relaciones entre estos.

Arquitectura de la solución

La herramienta embebida está concebida para funcionar en un entorno de red local donde un solo nodo de procesamiento es capaz de servir a varios nodos sensores en la tarea de reconocimiento de gestos, conformación de palabras y texto, y la traducción de texto a voz. En este punto del desarrollo se limita la reproducción sonora a través de una única unidad de audio pero se propone como trabajo futuro la implementación de un nodo de audio independiente que pueda servir a cada nodo sensor.

La solución opcionalmente funciona sin conexión a internet pero se provee la alternativa de aumentar la calidad y naturalidad de la voz artificial sintetizada mediante servicios de traducción ofrecidos por las grandes nubes privadas (GCP, Azure, AWS). Estos servicios además de conexión a internet requieren de una cuenta en sus plataforma con una cuenta bancaria registrada, aunque el pago de estos servicios solo se realiza una vez excedida la capa gratuita.

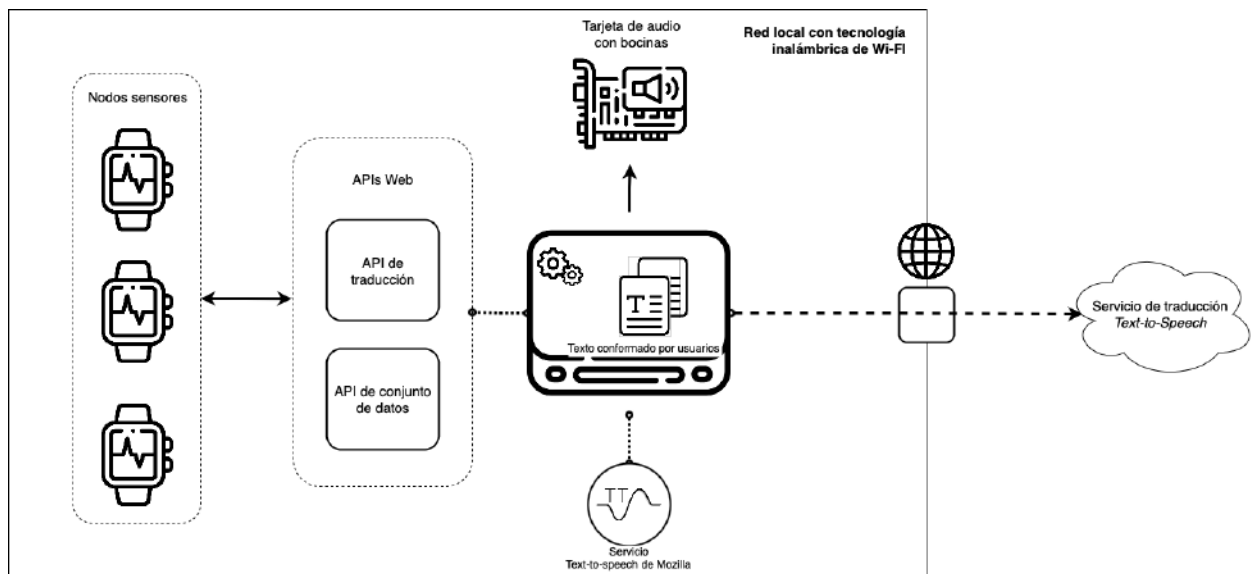


Figura 6: Arquitectura de la solución.

La Figura 6 representa la arquitectura de la solución en una red local. Los nodos sensores se comunican a través de las APIs expuestas por el nodo sensor y consumen los servicios mediante sus *endpoints*. A la vez, el nodo de procesamiento expone además el servicio de *Text-to-Speech* del grupo Mozilla y es capaz de consumir un servicio de traducción ofrecido por una de las grandes nubes, a través de internet. Finalmente el nodo sensor reproduce archivos de audio utilizando el dispositivo dispuesto para este fin.

CAPÍTULO IV: CÓDIGO MOTRIZ

El objetivo de la herramienta es poder utilizar un conjunto de movimientos de tal forma que su conjunción resulte en palabras que pueden ser traducidas desde un mensaje en escrito a una redacción oral utilizando la sintetización de voz provista por herramientas de inteligencia artificial texto a voz, o IA *Text-to-Speech* en inglés. Este capítulo aborda el conjunto de gestos que se utilizan en la solución, se describen las cualidades con las que deben cumplir, como se representan para su reconocimiento, los fonemas del español y finalmente la relación de los gestos con los fonemas y grafías para crear el código motriz.

Marco teórico

El área del reconocimiento de la actividad humana es una rama de la tecnología de actividad humana que permite la detección de actividades simples realizadas por una persona, tal como correr, caminar, subir escaleras, etc.[15], este tipo de estudios y aplicaciones han encontrado excelentes resultados utilizando sensores como el acelerómetro y el giroscopio [15-17] y han derivado en otras aplicaciones que por ejemplo monitorear aspectos de la salud de una persona como la ingesta de píldoras durante un esquema de medicamentos utilizando dispositivos *wearables*[18].

Estos principios se han adaptado e implementado exitosamente en la tarea de reconocimiento de gestos de manos, tal como lo muestra el trabajo [12] donde validan su hipótesis que expone la utilidad de los dispositivos de comodidad para la identificación de ciertos patrones de movimiento que pueden ser utilizados como secuencias instructivas en ciertas aplicaciones.

Tomando inspiración en este último artículo se implementa en la solución de este trabajo el reconocimiento de los gestos motrices utilizando exclusivamente un sensor acelerómetro, hecho que aumenta la efectividad en la tarea de clasificación de gestos y además propicia la simplificación en la experiencia de uso del prototipo por parte del usuario, aumentando de esta manera la adopción de la herramienta por pacientes con trastornos que experimentan síntomas que limitan la motricidad de sus extremidades.

Los gestos se caracterizan por ser de corta duración y fácilmente replicables pues para su realización se ocupan las articulaciones más grandes del brazo (el codo y el hombro) y se intenta de mantener en un mínimo los movimientos que involucran la muñeca. Los movimientos precisos y delicados en los que intervienen los dedos se ignoran completamente.

La descripción de un gesto se realiza mediante series de tiempo multivariantes; Una serie de tiempo es una secuencia de mediciones con una noción de ordenamiento cualquiera, aunque más típicamente la noción es el tiempo y de ahí su nombre.

La aceleración en un módulo IMU se captura en los 3 ejes espaciales y en un tiempo dado donde se consultan las magnitudes del sensor se obtienen 3 medidas, una para el eje X, otra para el eje Y y finalmente otra para el eje Z. La capacidad de un gesto motriz de ser caracterizado por 3 magnitudes distintas pero relacionadas y que refieren al mismo fenómeno, la categoriza como una serie de tiempo multivariable.

Resumiendo: Cada gesto motriz realizado por un usuario será caracterizado por 3 conjuntos de mediciones capturadas en un periodo temporal fijo, las mediciones resultan del cambio de la aceleración en los ejes espaciales y los conjuntos en unión toman por nombre “Serie de tiempo multivariable”.

En la Figura 7 se muestra ejemplificado un movimiento y su serie de tiempo multivariable que lo describe.

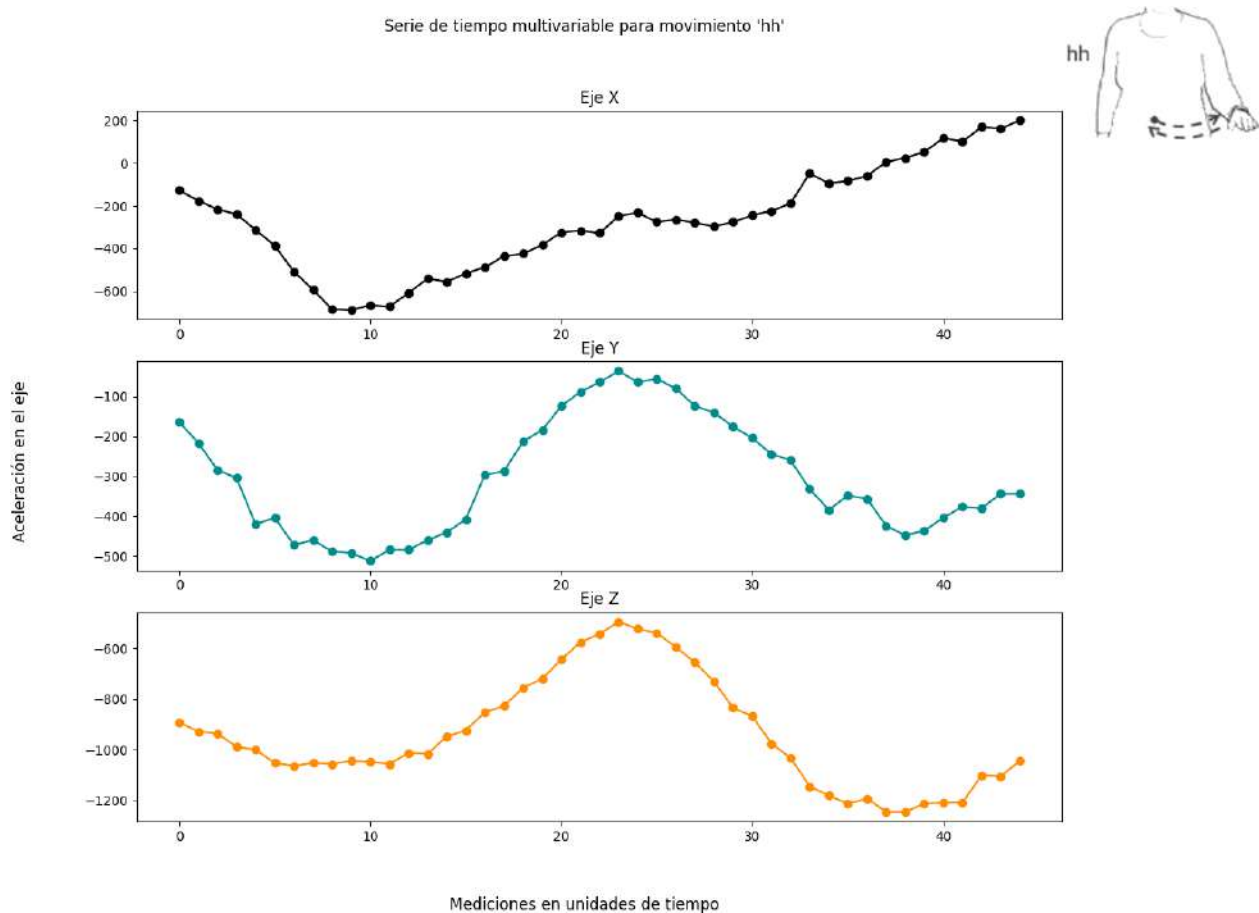


Figura 7: Mediciones del cambio de la aceleración en los 3 ejes espaciales para el movimiento 'hh' descrito en el trabajo [12].

Esta instancia forma parte de un conjunto de datos propio que replica los movimientos del trabajo de [12] y forma parte de los puntos que permanecieron pendientes de desarrollo en el artículo [5]

Fonemas del español

Los fonemas en una lengua son sonidos contrastivos que componen palabras a través de la combinación y orden de esto. Por si solos los fonemas no significan nada y más sin embargo si cambiamos un fonema por otro en una palabra esta se convierte en otra distinta y cambia su significado[19].

Las lenguas del mundo difieren en sus sonidos y fonemas, pero siempre el conjunto de estos es pequeño relativo al vocabulario. En el caso del español se cuentan con 5 fonemas vocálicos y alrededor de 20 consonánticos que dependen de la variedad dialéctica de la zona[19].

La escritura alfabética se basa en la posibilidad de identificar los sonidos contrastivos del idioma, mientras que toma el nombre de ortografía fonémica ideal cuando la relación es de 1 a 1 entre fonema y letra. El español goza de una ortografía con correspondencia casi perfecta con unas poquísimas excepciones[19].

El AFI (Alfabeto Fonético Internacional) es un sistema de notación fonética creado por lingüistas con el propósito de establecer de forma regularizada, precisa y única, la representación de los sonidos del habla de cualquier lengua. Bajo este esquema los fonemas del español son los siguientes (Tabla 3):

Tabla 3: Relación entre los sonidos del español, el símbolo asignado por el AFI al fonema (columna Fonema), las posibles grafías en el idioma español y un ejemplo colocando primero la palabra en español a la que le sigue su representación en símbolos fonéticos del AFI.

Fonema	Grafía	Ejemplos
Vocales		
/a/	<i>a</i>	<i>casa</i> /kása/
/e/	<i>e</i>	<i>mesa</i> /mésa/
/i/	<i>i, y</i>	<i>pino</i> /pino/
/o/	<i>o</i>	<i>copa</i> /kópa/
/u/	<i>u</i>	<i>cuna</i> /kúna/
Consonantes oclusivas		
/p/	<i>p</i>	<i>pelo</i> /pélo/
/b/	<i>b, v</i>	<i>boca</i> /bóka/, <i>vaca</i> /báka/
/t/	<i>t</i>	<i>toro</i> /tóro/
/d/	<i>d</i>	<i>dama</i> /dáma/

Fonema	Grafía	Ejemplos
Consonantes oclusivas		
/k/	<i>c, qu, k</i>	<i>capa</i> /kápa/, <i>queso</i> /késo/, <i>kilo</i> /kílo/
/g/	<i>g, gu</i>	<i>garra</i> /gára/, <i>guerra</i> /géra/
Consonantes africadas		
/tʃ/	<i>ch</i>	<i>chico</i> /tʃíko/
Consonantes fricativas		
/f/	<i>f</i>	<i>foca</i> /fóka/
/s/	<i>s, c(e,i), sc(e,i), z</i>	<i>saco</i> /sáko/, <i>cena</i> /séna/, <i>escena</i> /eséna/, <i>azul</i> /asúl/
/x/	<i>j, g(e,i), x</i>	<i>jota</i> /xóta/, <i>gente</i> /xénte/, <i>mexicano</i> /mexikáno/
/j/	<i>y, ll</i>	<i>yeso</i> /jéso/, <i>llano</i> /jáno/
Consonantes nasales		
/m/	<i>m</i>	<i>mes</i> /més/
/n/	<i>n</i>	<i>nada</i> /náda/
/ɲ/	<i>ñ</i>	<i>año</i> /áño/
Consonantes laterales		
/l/	<i>l</i>	<i>loco</i> /lóko/
Consonantes vibrantes		
/r/	<i>r</i>	<i>coro</i> /kóro/
/r/	<i>rr, r</i>	<i>corro</i> /kóro/, <i>rosa</i> /rósa/, <i>honra</i> /ónra/

Análisis del código motriz

El código motriz funciona como un alfabeto y un sencillo diccionario traductor, con gestos del brazo y manos en vez de caracteres y en el que se explica e ilustra de manera clara la correspondencia entre un movimiento y el fonema o palabra al que es traducido. Estos gestos tienen el objetivo de armar palabras y conformar texto a través de la traducción del movimiento a su clase textual.

Los movimientos deben cumplir con las siguientes cualidades:

- Movimiento amplio y marcado,
- De corta duración con un máximo de 2 segundos,
- Con no más de 2 repeticiones si se conforma por un patrón repetitivo,
- Preferentemente ejecutados en un mismo sentido o un par de cambios en la misma dirección,
- Y deben ser lo suficientemente diferentes entre sí para ser identificados por el modelo de reconocimiento como gestos distintos.

El cumplimiento de estos puntos permite que el gesto sea replicado con naturalidad y fluidez por el usuario, facilita su memorización y ayudan a que los pacientes con limitaciones motrices lo adapten y dominen más rápidamente.

La importancia de la elección de estos gestos para el código se ve reflejado en la facilidad de uso de la herramienta.

Clases textuales

Se considera una clase textual a la grafía asignada por el AFI al fonema utilizado en el texto.

La tarea que se encarga de relacionar la grafía correcta en español al fonema según la palabra, no es trivial pues su relación no es de uno a uno y se requiere conocer la ortografía de la palabra, y en algunos casos su contexto para la desambiguación y asegurar una asignación correcta. En la Tabla 3 se observan ejemplos de fonemas con más de 1 grafía en el español.

El enfoque desarrollado en este trabajo se concentra en la creación de palabras utilizando la grafía determinada por el AFI, lo que la convierte en una relación biyectiva en la que la asignación de etiquetas de clase es directa a una por símbolo (ver las columnas Fonema y Grafía de la Tabla 3). El proceso de corrección ortográfica es tarea del área de Procesamiento de Lenguaje Natural y por el alcance del proyecto se utiliza una biblioteca existente con un modelo pre entrenado en el idioma español.

Elección de los gestos motrices

La mayoría de gestos fueron inspirados de movimientos utilizados por el Lenguaje de Señas Mexicano, simplificando, acortando o alargando algunos de estos para adaptarlos a las cualidades con las que deben cumplir.

Tabla 4: Esta tabla conforma la primer propuesta de código motriz de la solución,

Nombre del gesto	Fonema y grafía asignada	Explicación del movimiento
Saludo	/a/	Partiendo con el brazo al costado, se lleva la punta de los dedos al costado de la sien y se extiende el brazo hacia afuera y adelante de manera que la mano se encuentre ligeramente más alta que el nivel del hombro.
Nombre del gesto	Fonema y grafía	Explicación del movimiento

asignada		
Agradecimiento	/e/	Partiendo con el brazo al costado se toca con la punta de los dedos la barbilla y se extiende completamente el brazo hacia adelante a la altura del pecho.
Elevación	/i/	Partiendo a la altura de la cadera y con la palma hacia arriba, se eleva verticalmente la mano aproximadamente a la altura de la barbilla.
Circunferencia	/o/	Con los dedos extendidos y apuntando hacia arriba se dibuja una circunferencia iniciando el recorrido hacia adentro (hacia al pecho).
Balanceo	/u/	Partiendo de la altura de las costillas con el codo ligeramente más flexionado de 90 grados y los dedos apuntando al frente, se dibuja una depresión hacia la izquierda (como un tazón poco profundo) y se regresa con una depresión hacia la derecha.
Doble golpe	/p/	Con la mano derecha en puño y a la altura del corazón, y la izquierda con la palma hacia arriba a la altura del ombligo, se baja la derecha golpeando dos veces en la mano izquierda y regresa a la altura original.
Indicación	/b/	Con la mano derecha a la altura de la cadera y la izquierda a la altura de la barbilla, se realiza un movimiento como corte vertical en sentidos contrarios, la derecha sube a la barbilla y la izquierda baja a la cadera. Por un instante a medio camino ambas manos se rozan.
Corte	/t/	Con la mano derecha tocando el costado izquierdo de la cadera y la palma viendo hacia abajo, se realiza un corte horizontal hacia la derecha con una pequeña inclinación hacia arriba.
Pensamiento	/d/	Con la palma de la mano derecha entornada a medio camino hacia el pecho y a la altura del corazón, sube tocando con las yemas de los dedos la sección derecha de la frente y luego extiende hacia el frente y ligeramente hacia afuera para terminar con los dedos extendidos apuntando al frente.
Exclamación	/k/	Con las yemas de los dedos tocándose y las palmas mirando hacia el estómago, se abren los brazos en sentido contrario de manera que las palmas se entornan ligeramente hacia arriba y hacia afuera.
Medio círculo	/g/	Con las yemas de los dedos tocándose y las palmas mirando hacia el estómago, se marca un medio círculo con cada mano para finalizar tocándose las yemas de los dedos otra vez a la altura de la barbilla.
Aplauso doble	/tʃ/	Con las manos en los costados se realizan 2 aplausos y se termina el movimiento con las manos en los costados.

Nombre del gesto	Fonema y grafía	Explicación del movimiento
------------------	-----------------	----------------------------

asignada		
Chop chop	/f/	Con la mano a 45 grados y tocando la barbilla se realizan 2 movimientos como cortes hacia al frente y termina la mano en la posición inicial a la altura de la barbilla
Sentimiento	/s/	Tocando con la puntas de los dedos el pecho al nivel del corazón, se marca un círculo iniciando a la izquierda y se termina en el lugar de inicio.
Entrecruzado	/x/	Se dibuja una x partiendo de la altura del hombro derecho, después al costado de la cadera izquierda, sube a la altura del hombro izquierdo y finaliza en el costado derecho de la cadera.
Propio	/j/	Partiendo con el brazo al costado se sube y se toca el pecho a la altura del corazón para finalmente regresar el brazo al costado
Costados	/m/	Con la mano apuntando al frente y a la mitad del pecho, se marca un cuarto de arco hacia la izquierda bajando a la altura de las costillas y se regresa con un medio arco a la derecha que sube a la altura del pecho y finaliza en las costillas derechas.
Agitado	/n/	Con la mano a la altura del hombro derecho, se agita dos veces de derecha a izquierda y de regreso.
Corte medio corazón	/ɲ/	Se parte de la mitad del pecho y se realiza un semicírculo hacia afuera a la altura del inicio de las costilla y el hombro, después de baja con una inclinación de 45 grados hacia el ombligo.
Latido doble	/l/	Con la palma tocando el pecho a la altura del corazón, se despega unos centímetros y se vuelve a tocar en 2 ocasiones.
Meneo	/r/	Con la mano derecha arriba y la izquierda abajo, se giran en sentidos contrarios(hacia afuera y hacia atrás respectivamente) completando una circunferencia y vuelven a la misma altura en la que se empezó.
Impulso	/r/	Partiendo de la mano en el costado, se sube a la altura y pegada al pecho con la palma mirando al frente y se hace un gesto de empujar hacia adelante.

Catálogo de gestos motrices

Los movimientos utilizados para esta solución no forman parte de algún lenguaje o conjunto estándar utilizado por algún grupo en específico, al menos no con las variantes realizadas aunque suelen asemejarse a gestos del LSM. Por esta razón y buscando facilitar el aprendizaje y adopción del código motriz, se desarrolló una herramienta basada en web que vuelve accesible y fácil de consultar los gestos que conforman el código de la solución.

La herramienta web nombrada “Habla”(Figura 8), funciona como un catálogo digital de los gestos motrices e interfaz monitor de la solución. De las tecnologías utilizadas para su desarrollo se destaca la más importante React, un *framework Frontend* basado en Javascript. La filosofía y herramientas disponibles de esta tecnología permiten desarrollar aplicaciones enfocadas en la interacción con el usuario, fluidez y adaptabilidad, por lo que el catálogo puede consultarse en todo tipo de dispositivos.

Se destaca una función adicional la cuál es la completa traducción del contenido de la herramienta a Inglés del Español(Figura 9). Esta traducción se refiere al texto mostrado en la página únicamente y no la solución completa, el modelo se encuentra diseñado y entrenado para funcionar exclusivamente para el Español con dialecto de México.



Figura 8: Carátula de la herramienta en Español

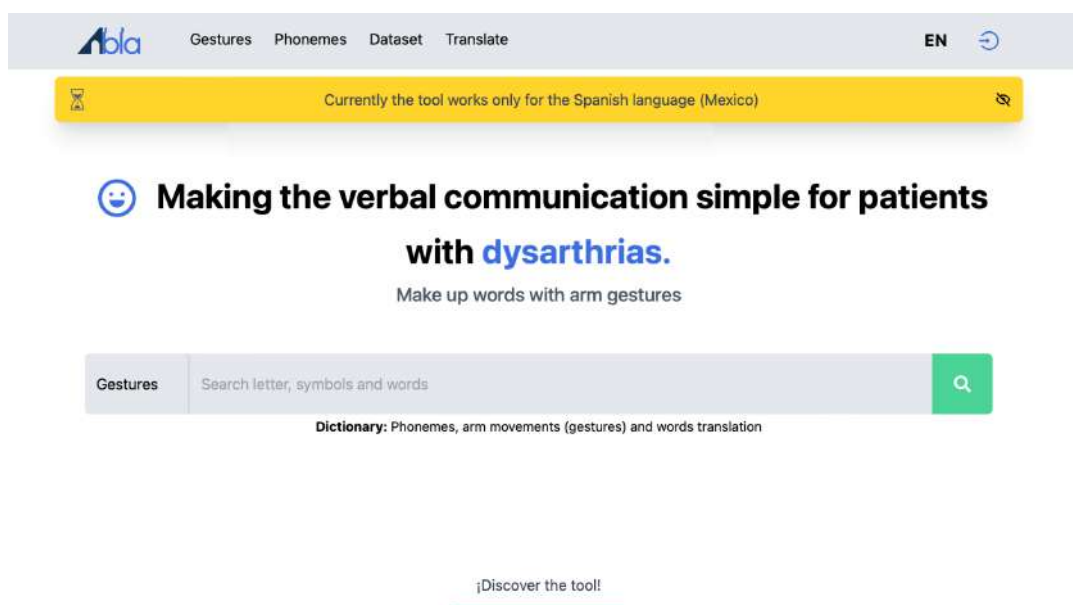


Figura 9: Carátula de la herramienta en Inglés

Animación de gestos

Se identifica que el método más natural para la adopción de los gestos es a través de visualizar el movimiento ayudándose de un modelo que permita replicar su técnica de movimiento. En este apartado se contaba con la alternativa de realizar un conjunto de videos realizando repeticiones del gesto u optar por un modelo virtual para esta tarea. La elección final se decantó por un modelo animado que despliega la técnica idónea para cada gesto, sin vicios o movimientos particulares al usuario que los ejecuta y finalmente el diseño del modelo permite claramente imitar sus movimientos pues no porta accesorios que entorpezcan la visualización de los brazos.

El modelo utilizado es conocido como “Vincent”, un personaje diseñado en 3D como parte de los recursos disponibles en la herramienta de modelado y animación Blender. Se realizó la animación de los 22 gestos interpretados por Vincent como videos cortos donde se ilustra claramente la técnica de cada movimiento, posición inicial, y posición final para cada brazo.

Ubicada en la sección de ‘Gestos’, las animaciones se reproducen en bucle junto a una tarjeta de información que recuerda el fonema al que se asocia, las grafías utilizadas, la dificultad de replicación y la descripción de la ejecución(Figuras 10 y 11)

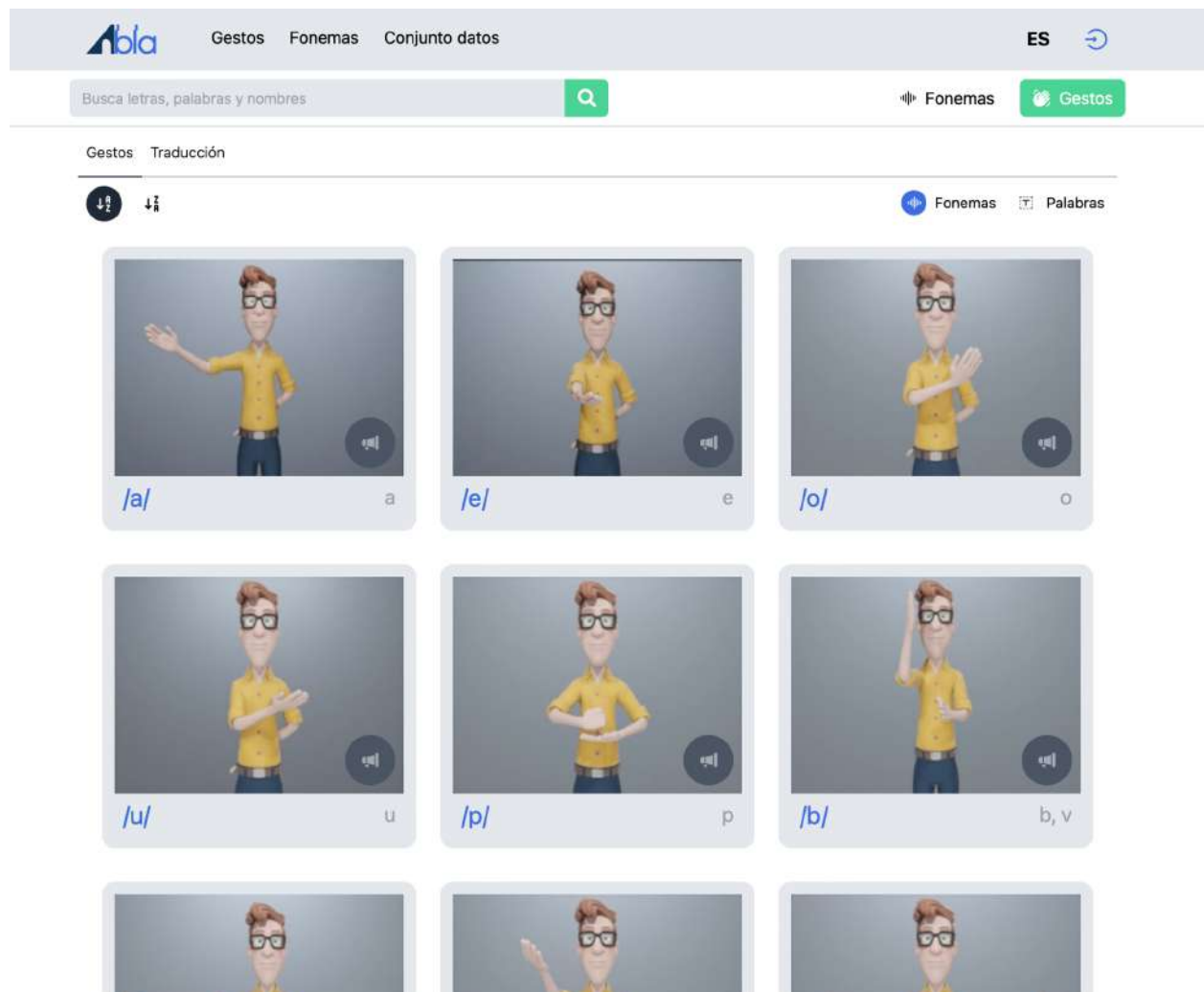


Figura 10: Listado de los gestos motrices en el catálogo digital



Figura 11: Video corto y diseño de la tarjeta informativa utilizada en todos los gestos

Fonemas

Por la importancia que representa el conocer y familiarizarse con los fonemas más que con la representación escrita de las palabras, se añade a la herramienta un apartado de fonemas que permite visualizar como tarjetas de información cada uno de los fonemas que se utilizan como clase textual en la herramienta(Figura 12).

Para cada fonema se especifica la grafía utilizada por el Alfabeto Fonético Internacional(colocado entre diagonales /), las grafías asociadas durante la escritura, el orden asignado por el AFI, y un listado de palabras que ejemplifican el uso del mismo fonema con las diferentes grafías asociadas.

Diccionario de fonemas en el idioma Español

Utilizamos los símbolos del SFI(Sistema Fonético Internacional) para asignar una representación escrita a los fonemas y la pronunciación de una palabra

/x/	x
/d/	d
/k/	c, qu, k
/g/	g, gu
/tʃ/	ch
/f/	f
/s/	s, c(e,i), sc(e,i), z
/x/	j, g(e,i), x
/j/	y, ll
/m/	m
/n/	n
/ɲ/	ñ




/ x /

Orden Consonante Fricativa

Grafías asociadas

j, g(e,i), x

Ejemplos

jota	/ xóta /
gente	/ xénte /
mexicajo	/ mexikáno /

Figura 12: Lista de fonemas y carta lateral con la información de cada fonema seleccionado.

CAPÍTULO V: MODELO PARA RECONOCIMIENTO DE GESTOS MOTRICES

En el artículo “Clasificador de gestos motrices utilizando vectores de atributos distancia DTW sobre series de tiempo en representación SAX”[5] se propone una nueva metodología y método para el reconocimiento de gestos motrices, y además realiza una revisión de las técnicas y algoritmos existentes para esta tarea, desarrollándose de forma más extensa el marco teórico respecto a los algoritmos tratados y las técnicas consultadas y evaluadas.

El contenido teórico aquí expresado se trata de un resumen corto de las herramientas útiles durante la investigación y desarrollo de la metodología y método.

Marco teórico

DTW

Dynamic Time Warping es una técnica bien conocida utilizada para encontrar una alineación óptima entre 2 secuencias dependientes del tiempo dadas y bajo ciertas restricciones[20]. Este algoritmo es útil para medir la similitud entre 2 secuencias que no se alinean exactamente en el tiempo, velocidad o extensión[20]. El algoritmo DTW retorna como resultado, una cantidad como medida de similitud entre 2 series de tiempo y un camino de deformación óptima que asocia elemento a elemento de las 2 secuencias.

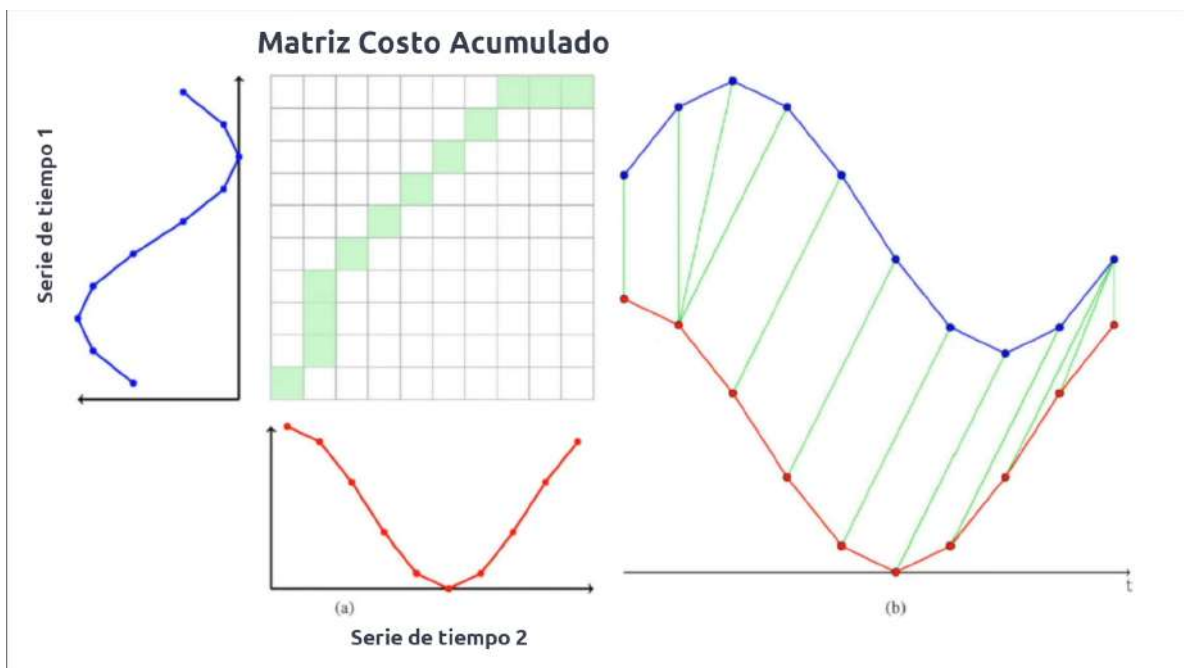


Figura 13: Representación de la matriz de costo acumulado y alineación óptima entre 2 series de tiempo.

DTW es un algoritmo muy robusto que ofrece excelentes resultados en su tarea, y sin embargo suele ser costoso derivado de aplicar programación dinámica en su algoritmo implicado el llenado de una matriz de costo. Surgieron variantes que abordan esta situación y se busca aplicar restricciones que disminuyen la complejidad limitando el número de cálculo a realizar, mientras

que como cualidad añadida se aumenta la precisión de predicción evitando la desviación excesiva de la deformación de la diagonal principal.

La banda Sakoe-Chiba[21], es una de las restricciones más comunes aplicadas a DTW y que asemeja, tal como su nombre lo indica, a una banda de tamaño T que corre a lo largo de la diagonal principal y obliga al cálculo de solo aquellos elementos dentro de los límites en vez de toda la matriz de costos.

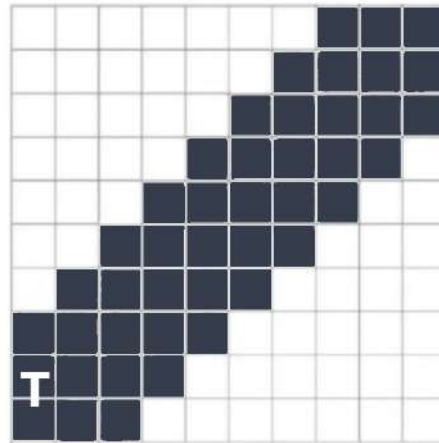


Figura 14: Restricción tipo banda Sakoe-Chiba representada sobre la matriz de costo acumulado.

PAA

Piecewise Aggregate Approximation, es un algoritmo que tiene como idea básica la reducción dimensional de una serie de tiempo de entrada mediante la partición de esta en segmentos del mismo tamaño, sobre cada cual se realiza el cálculo promedio de los valores en el segmento[22].

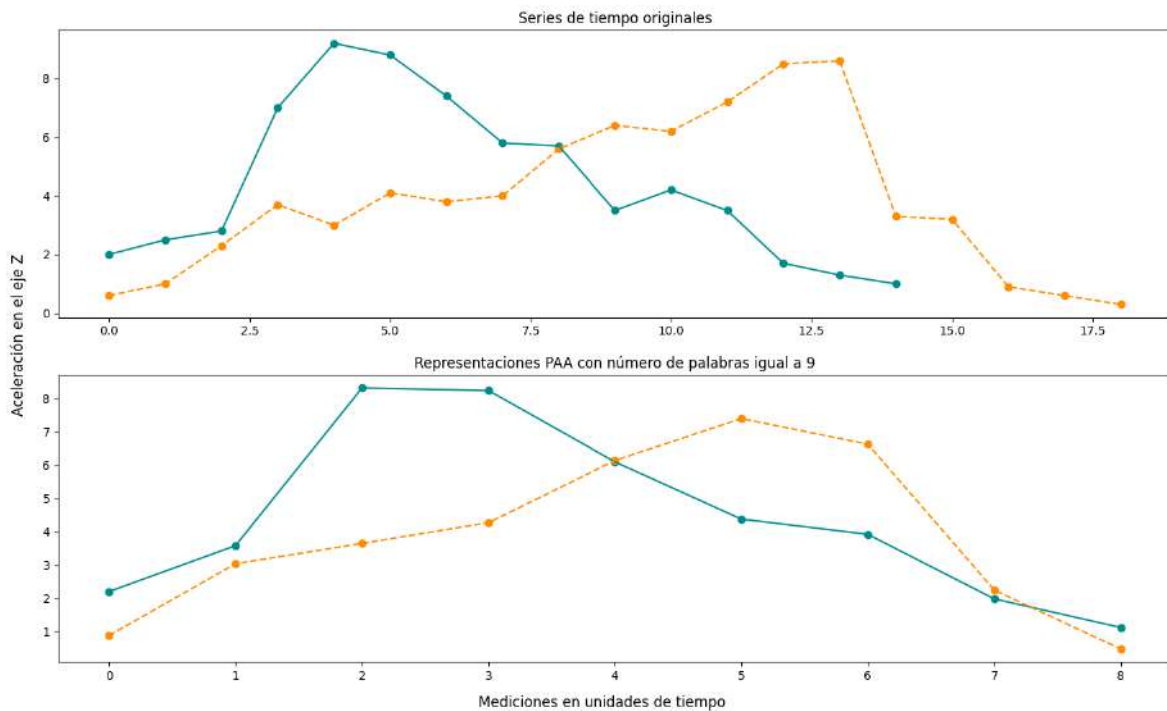


Figura 15: Comparación de 2 series de tiempo con un número distinto de mediciones antes y después de la reducción dimensional a 8 palabras.

El algoritmo PAA al reducir la dimensión con un número fijo de palabras m , mantiene la tendencia y forma de la secuencia por lo tanto la información de la señal.

SAX

Symbolic Aggregate Approximation, es una técnica desarrollada y enfocada en la reducción dimensional de una serie numérica con una serie de tiempo, a un espacio simbólico de ‘palabras’. Dada una serie de tiempo de longitud arbitraria n se realiza la transformación a una cadena de longitud w utilizando un alfabeto $A = a_1, a_2, \dots, a_k$ [23].

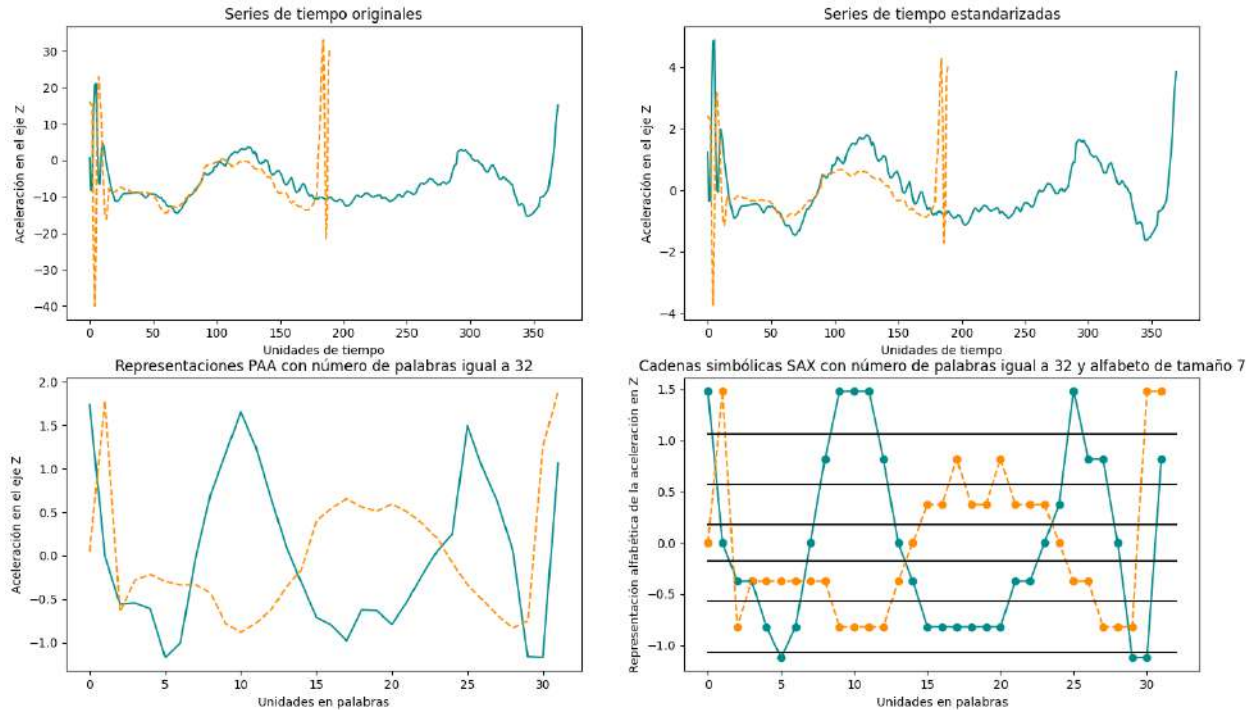


Figura 16: Se representan mediante gráficas, las diferentes fases y tareas del algoritmo SAX para 2 series de tiempo que tienen un número distinto de mediciones y en un rango de valores distinto.

Esta técnica se divide en 2 grandes tareas, la primera de ellas es la reducción dimensional de la serie de tiempo, proceso del que se encarga el algoritmo PAA, y posteriormente se realiza la asignación de símbolos alfabéticos como la discretización de secuencia temporal.

DBA

DTW Barycenter Averaging, es un algoritmo iterativo que utiliza DTW para la alineación de series de tiempo utilizando un promedio envolvente[24]. Tomando un conjunto de series de tiempo se obtiene una secuencia temporal de salida promediada utilizando el algoritmo DBA.

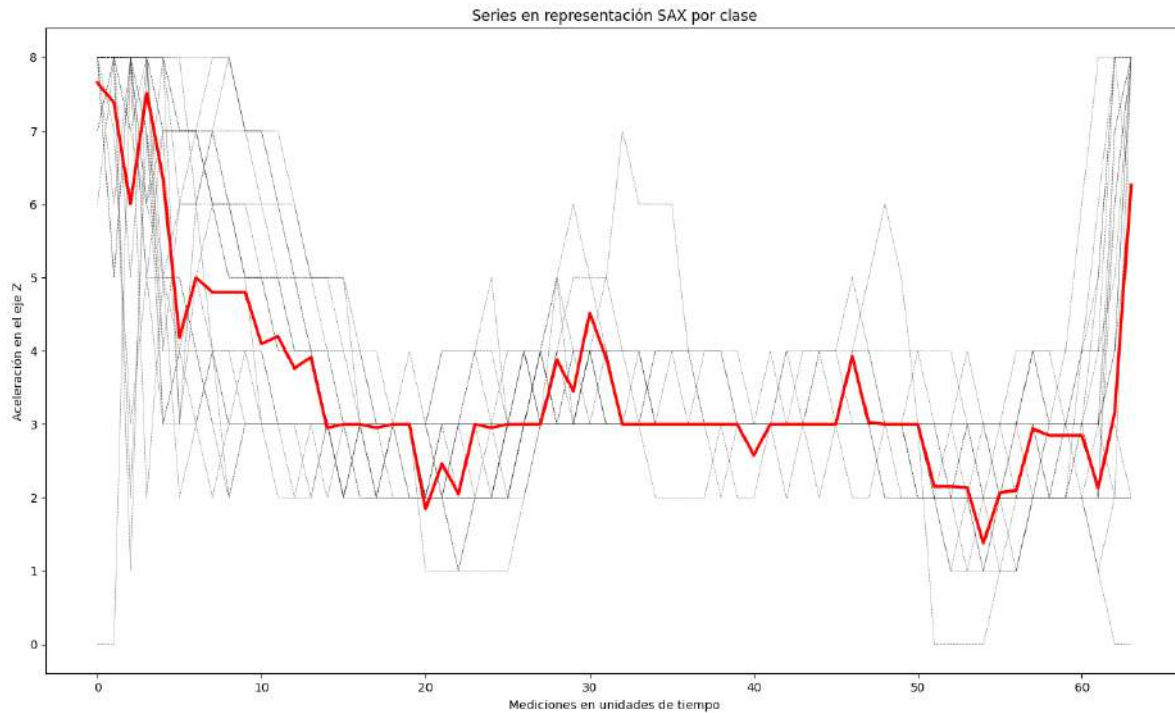


Figura 17: Serie promedio resultante del algoritmo DBA para un conjunto de 5 secuencias en representación SAX con misma clase.

Método para el reconocimiento de gestos

El método **Vectores de atributos SAX-DTW**, es el método propuesto como producto de este trabajo y busca aprovechar la coexistencia de DTW como algoritmo robusto para medir la similitud entre secuencia de tiempo y SAX con la reducción de complejidad y transformación a un espacio simbólico de palabras. Una de las ventajas que provee la creación de vectores descriptivos de una secuencia temporal es que permite utilizar modelos de aprendizaje máquina más sofisticados que el ampliamente utilizado K-NN; En esta metodología se opta por utilizar una máquina de soporte vectorial para la clasificación multiclase.

Fase de preprocesamiento de los datos

La manera más sencilla de obtener una mejora en la precisión de la clasificación para secuencias dependientes del tiempo es la implementación de un método basado en características (Representación estadística o simbólica definida para una serie de tiempo), y aprovechando este hecho se requiere la transformación de los datos en un espacio alternativo donde las características discriminatorias pueden ser más fácilmente detectadas que si se compara con un clasificador más complejo que permanece operando en el contexto temporal[25].

La totalidad de series de tiempo manejadas, incluyendo las instancias que se utilizarán para entrenar el modelo clasificador para la predicción de la etiqueta de clase, es a través de su representación en cadenas simbólicas SAX, que provee además de una transformación a un espacio alternativo de dominio simbólico, una reducción dimensional que se traduce en una simplificación de la complejidad. Esto implica que ninguna de las etapas siguientes pueda realizarse sin antes transformar las secuencias en cadenas simbólicas con número de palabras

definido s (con $s = 64$ en [5]) y su discretización en el dominio de valores con un alfabeto de tamaño fijo a (con $a = 9$ en [5]).

Pensando en aprovechar todo el potencial de las similitudes calculadas por DTW, se escoge un alfabeto numérico $A = [0, \dots, a - 1]$ en las representaciones SAX, permitiendo a su vez utilizar sin realizar una previa modificación a los símbolos, la distancia DTW.

Fase de entrenamiento

Cuando se involucra un método de Aprendizaje Máquina como SVC, el modelo debe pasar por una etapa de entrenamiento antes de poder ofrecer la tarea de predicción, situación que no sucede con el algoritmo k-NN de los métodos referencia, pues realmente la etapa de entrenamiento y predicción sucede en un proceso integral.

La creación de los vectores de atributos en este método es esencial, no solo por la posibilidad que ofrece de aplicarse como entrada a modelos robustos de aprendizaje máquina, pero además diverge del trabajo[26] en el que se inspira y se toma una interpretación alternativa que impacta positivamente en las etapa de entrenamiento y predicción, pero siendo de mayor relevancia su ventaja durante la predicción; Disminuyendo significativamente el tiempo de cálculo y complejidad requerido para la creación del vector de instancias por cada secuencia temporal.

Tomando un enfoque alternativo a la creación del vector de atributos, antes de pasar siquiera a esta etapa, se requiere crear una cadena simbólica representante de cada clase existente en el conjunto de datos, y como requisito se exige la estratificación del conjunto para mantener la equiprobabilidad de las clases durante el entrenamiento.

La fabricación del conjunto de cadenas representantes se logra mediante el promedio de todos los vectores existentes en el conjunto de datos que pertenecen a una misma clase, resultando el método más efectivo para lograr un representante fidedigno de cada etiqueta el algoritmo DBA[24].

Este conjunto de representantes resultante se convierte a partir de su cálculo en el conjunto de cadenas simbólicas más importantes del modelo. El hecho de que mediante la similitud DTW de estos representantes con los vectores destinados al entrenamiento y posteriormente los nuevos patrones a ser predecidos, exige la preservación del conjunto durante la vida útil del modelo entrenado. A pesar de tratarse de un algoritmo determinístico que sugiere el recálculo del conjunto de representantes frente a la pérdida de este, si la elección de la serie de tiempo abreviada inicial I se obtiene por un proceso aleatorio, un segundo cálculo del conjunto de representantes significa desechar el modelo entrenado para volver a crear las instancias de entrenamiento con este nuevo conjunto.

Creado el conjunto de cadenas representantes de cada clase, se calculan los vectores de atributos distancia DTW entre la instancia de entrenamiento y los representantes de clase(es importante el orden en que se ingresan ambas cadenas simbólicas, pues la operación de similitud DTW no es conmutativa), alimentando el modelo SVC para iniciar la etapa de entrenamiento.

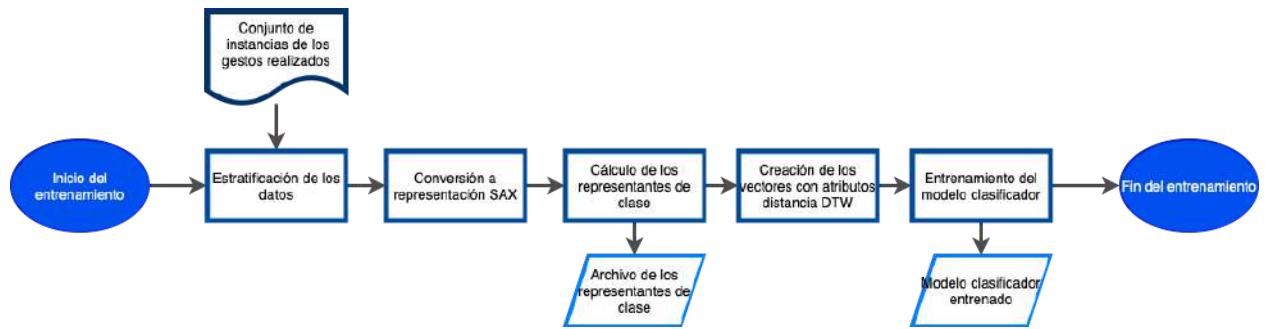


Figura 18: Diagrama de flujo del proceso de entrenamiento para el modelo Vectores de atributos SAX-DTW.

Fase de predicción

Muestreado, filtrado y preprocesado un patrón de movimiento, el primer paso será representarlo mediante SAX en una cadena simbólica con el alfabeto numérico A definido previamente.

Como cadena simbólica, ahora esta secuencia puede ser usada para conformar un vector de atributos distancia DTW utilizando el mismo conjunto de representantes de clase usado para fabricar los vectores de entrenamiento.

Finalmente el vector resultante puede ser alimentado al modelo SVC previamente entrenado para la obtención de la etiqueta de clase resultante del proceso de predicción.

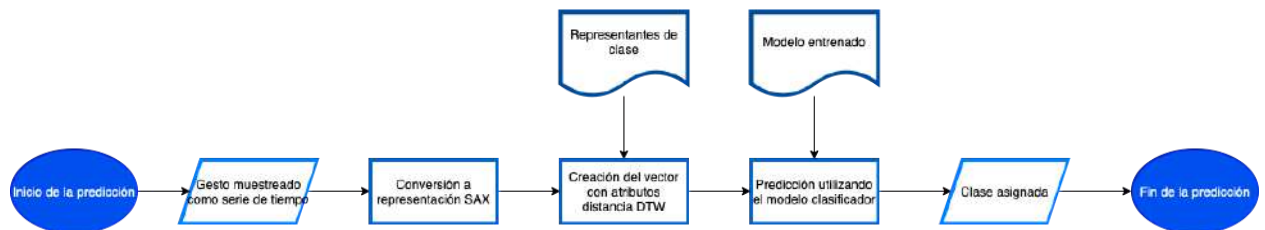


Figura 19: Diagrama de flujo del proceso de predicción para el modelo Vectores de atributos SAX-DTW.

Implementación en la solución

Para la implementación del modelo en cualquier solución se cuenta con 2 alternativas de entrenamiento, entrenamiento de modelo por usuario y el entrenamiento de un modelo único utilizando los gestos de todos los usuarios. Para cada una de las alternativas se tienen ventajas y desventajas(Tabla 5)

Tabla 5: Comparativa entre las alternativas de entrenamiento para el modelo de la solución

Modelo por usuario	Modelo único
Bien adaptado a la técnica particular del usuario para cada gesto	Sensible a la técnica de cada usuario para los gestos
Se ocupa un menor número de instancias para el entrenamiento	Se ocupa un mayor número de instancias de entrenamiento alimentado por todos los usuarios muestreados

Modelo por usuario	Modelo único
Forzosamente se requiere realizar una etapa de recolección de instancias de entrenamiento con el usuario que utilizará la herramienta	Modelo escalable para ser utilizado en una herramienta de disposición instantánea para todo tipo de usuario
Se consigue una precisión en predicción excelente con poco entrenamiento y sin modificaciones de parámetros	Para conseguir una predicción buena es necesario una mayor cantidad de gestos de entrenamiento y ajuste de parámetros del modelo

En la solución se opta por realizar el entrenamiento de un modelo por cada usuario con lo que se logra una precisión en el rango de 88.64% - 100%, y aunque para cada usuario la precisión de su modelo será distinta, la precisión promedio general para el modelo utilizado ya durante la tarea de traducción es del 99.43%. A partir de esta consideración, es obligatorio que nuevos usuarios creen un conjunto de datos con repeticiones de los gestos para igualmente obtener su configuración entrenada, difícilmente se podría adaptar el modelo de otro usuario conservando la misma precisión de predicción.

Independiente a la opción de entrenamiento que se elija el método para ser eficiente y preciso en su tarea, requiere de la afinación de los hiper parámetros que particularmente se desempeñan mejor para cada aplicación. Los valores encontrados que proveen los mejores resultados para la solución se listan en la Tabla 6.

Tabla 6 Valores que proveen los mejores resultados para la solución

Parámetro	Valor	Método utilizado
Periodo de muestreo	$p = 37 \text{ ms}$ (frecuencia $\sim 27\text{Hz}$)	Mediante experimentación partiendo en referencia a un muestreo realizado previamente para el conjunto de datos en el artículo[5].
Tamaño de ventana DTW	$w = 9$	Utilizando el método de optimización de tamaño de ventana DTW[41]
Número de palabras	$nw = 22$	En función del gesto registrado con el menor número de muestreos
Tamaño de alfabeto	$a = 9$	Mediante experimentación partiendo en referencia a los parámetros de la implementación en [5].

Para la solución implementada se encontró que el óptimo periodo de muestreo fue de 37ms, una disminución de aproximadamente 25% partiendo de los 50ms utilizados durante la conformación del conjunto de datos en la implementación del artículo. La disminución del periodo se realizó buscando obtener un mayor número de muestras por segundo para cada gesto en mira a aumentar la precisión durante la predicción.

A pesar de haber partido de esta hipótesis, una disminución del 50%(25ms o 40Hz) presentaba 2 problemas principales, el primero relacionado con la memoria de la placa micro:bit, la cual tiende agotarse y generar un error para gestos que duran durante su muestreo más de 3 segundo. El segundo problema se encuentra relacionado al ajuste de parámetros del modelo, con esta tasa de muestreo no fue posible encontrar una combinación en la que la precisión durante el

entrenamiento fuera mayor a 65.4%, una precisión pésima en contraste con la configuración final.

Conjunto de datos

El proceso de recolección de los datos consistió en una sesión con el voluntario o la voluntaria, que realizó un total de 20 repeticiones por cada uno de los 22 gestos, dividido en un par de conjuntos con 10 repeticiones por cada gesto, para permitir un descanso antes de completar el conjunto de datos. Se recolectaron mediciones de 8 voluntarios de ambos sexos (5 mujeres y 3 hombres) en un rango de edad amplio de 18 a 57 años.

Cabe destacar que el total de los voluntarios cuentan con la capacidad de generación del lenguaje y del habla intacta, así como limitaciones motrices de importancia inexistentes, por lo que según el alcance del proyecto y los objetivos planteados, los conjuntos de datos recolectados cumplen con la función de probar el correcto funcionamiento de la herramienta embebida y demostrar la mejora en tiempo de ejecución y precisión para el modelo de reconocimiento propuesto frente a los existentes en el estado del arte[12,13,25]; De esta manera se confirma la viabilidad del proyecto para buscar, como una tarea futura, un despliegue de la solución realizando la recolección de los conjuntos de datos con pacientes diagnosticados con alguno de los 3 principales trastornos.

Al iniciar la sesión con los voluntarios, se provee una breve explicación de la propuesta de solución, los objetivos del proyecto y la forma en que funciona la herramienta a través del dispositivo sensor y hasta la conformación del texto a voz.

Posteriormente se explica la finalidad de su aportación a través del grabado de las repeticiones de los gestos, el entrenamiento del modelo y la capacidad de este de reconocer el tipo de gesto efectuado gracias a la etapa de entrenamiento.

Previo al inicio de un muestreo que funge como preparación para el voluntario, se externan algunas indicaciones y se toman ciertas consideraciones:

- Se explica el funcionamiento del sensor táctil en el prototipo para indicar el inicio y fin del muestreo de un gesto.
- El muestreo de los gestos los debe realizar de pie el voluntario.
- La técnica del gesto realizado por cada voluntario debe ser lo más parecido al movimiento mostrado por el modelo animado sin obligar al voluntario a adaptar perfectamente los movimientos (técnica individual).
- Se solicita al voluntario que la velocidad y extensión de cada gesto lo realice de forma que su ejecución se sienta natural.
- Se pide al voluntario mantener consistencia en la técnica durante todas las repeticiones de cada gesto.

Con la finalidad de explicar de manera precisa la técnica deseada de cada gesto, se utiliza la herramienta web para tomar un conjunto de preparación, compuesto por una única repetición por gesto, que le permite al voluntario adaptar sus movimientos, desarrollar su técnica y practicar la coordinación del inicio y fin del muestreo con el sensor táctil del prototipo.

Durante el muestreo de los 2 conjuntos se ocupa la herramienta web como interfaz para el usuario permitiendo la visualización de los parámetros más importantes para el voluntario, mientras que el monitoreo de la persona que lo aplica se logra utilizando la interfaz de línea de comandos de la API y endpoints de administración.

Herramienta web para la recolección del conjunto de datos

El agregar esta sección a la herramienta web implica proveer al voluntario de una interfaz que hace intuitivo el proceso de recolección(Figura 21), pues se añaden acciones desencadenadas por eventos (como la recepción de un gesto muestreado) que constantemente actualiza la interfaz y da seguridad al voluntario de la replicación correcta de cada gesto.

Se facilita la creación de un nuevo conjunto desde la herramienta “Habla” al ingresar el nombre del nuevo conjunto de datos(Figura 20) Posteriormente la aplicación web se comunica a través de sockets con la API de datos para mantener actualizada la interfaz cada vez que cambia el estado del proceso de recolección.

Una vez completada la conexión con la API y confirmada la creación del conjunto, la herramienta espera a la conexión del prototipo con el micro:bit para iniciar el proceso de recolección.

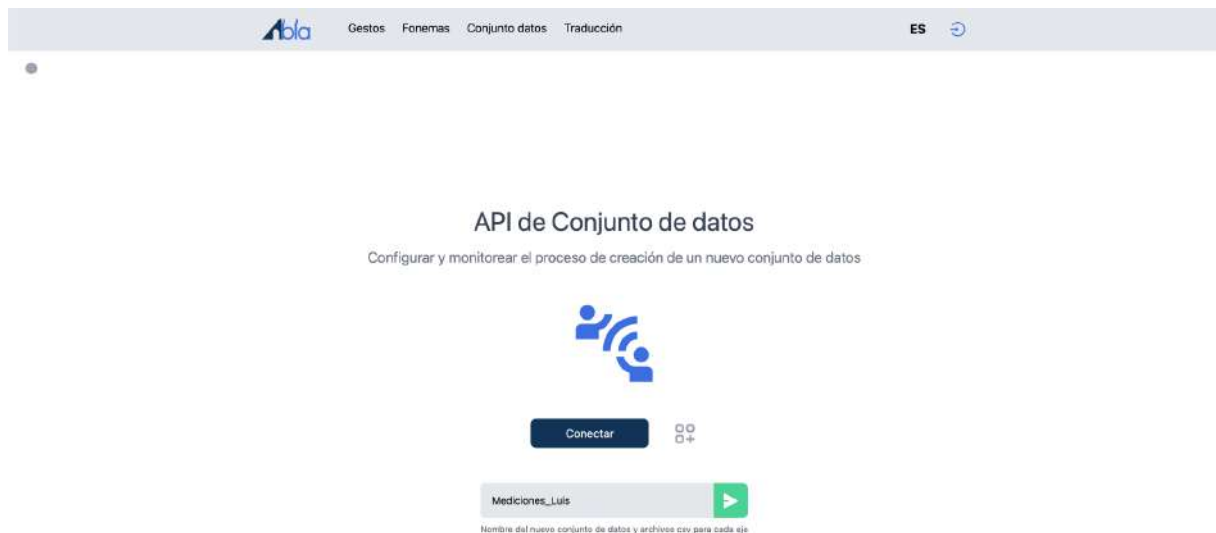


Figura 20: Pantalla inicial para conectar la interfaz a un conjunto de datos en proceso de recolección o la creación de un nuevo conjunto de datos.



Figura 21: Elementos en la interfaz mostrada durante todo el proceso de recolección al voluntario:

1. Indicador del estado de la conexión con la API (verde indica conectado)
2. Se visualiza el ícono de micro:bit cuando se reconoce la conexión de un prototipo sensor.
3. En la cabecera de la carta principal se agrega el nombre del gesto y del lado derecho el nombre del conjunto de datos en recolección.
4. Video en loop de la animación del gesto correspondiente.
5. Fonema asociado, grafía o grafías del fonema y descripción del gesto.
6. Indica el número de repeticiones realizadas y el número de gesto actual.

En la Tabla 7 se muestran los resultados obtenidos después del entrenamiento y evaluación del modelo utilizando de manera individual el conjunto de datos recolectado para cada voluntario.

Tabla 7: Resultados del modelo de reconocimiento para cada voluntario

Voluntario	Rango mediciones por gesto	Rango precisión
Alondra	29 - 57	97.73% - 100%
Yared	31 - 69	88.64% - 97.73%
Adriana	30 - 51	97.73% - 100%
Patricia	25 - 46	95.45% - 100%
Elizabeth	30 - 56	90.9% - 97.73%
Claudia	27 - 43	93.18% - 100%
Eduardo	27 - 48	95.45% - 100%
Luis	23 - 40	93.18 - 100%

Traducción de gestos

Imitando lo que sería la recopilación de las repeticiones de los gestos de un nuevo usuario, se utiliza el conjunto de datos de los voluntarios para generar un modelo particular que será utilizado durante la traducción del usuario.

En resultados previos como la Tabla 7, se muestra un rango de precisión, este rango se obtiene de tomar la menor y la mayor precisión conseguida durante una iteración de 10 entrenamientos y evaluaciones. Una iteración de este tipo consiste en utilizar el conjunto de datos y particionarlo de manera que se obtienen 18 instancias de entrenamiento por cada gesto y 2 instancias de evaluación por cada gesto, este proceso de selección se realiza de manera aleatoria cada vez.

Una vez terminadas estas iteraciones se guarda únicamente la configuración que obtuvo la precisión de predicción máxima en cada conjunto de datos y se registra para poder ser utilizada como el modelo de reconocimiento durante la tarea de traducción de gestos.

Al igual que sucedió con la tarea de recopilación del conjunto de datos, se añade a la herramienta web un apartado dedicado a la funcionalidad principal del sistema, la traducción de los gestos motrices a texto y finalmente a voz.

La interfaz inicial de este apartado solicita identificar el modelo predictivo que se utilizará para la sesión siguiente(Figura 22), y una vez ingresado se realiza la conexión con la API de traducción a través de sockets para realizar la configuración del lado del servidor con el modelo escogido.

Mientras no sea reconocido que un dispositivo sensor se ha conectado con la API de traducción, es posible realizar peticiones de reproducción en voz ingresando directamente los fonemas. Una vez se identifica la solicitud de conexión del micro:bit la interfaz tan solo permite visualizar los fonemas que se van ingresando a través de los gestos muestreados por el prototipo y su posible representación escrita en el español(Figura 23)

Adicionalmente se añade a la herramienta un *widget* que sirve como apoyo a la conformación de las palabras. Este *widget* permite que se escriba con fonemas la palabra que se desea formar, y utilizado de manera independiente a la palabra en actual conformación, recuerda el gesto a realizar mostrando la animación cuando se da click a cada uno de los fonemas.

Es posible también utilizarlo en conjunto con la palabra actual en conformación, si se identifica que el fragmento de palabra generado con los gestos es una subcadena de la palabra de fonemas en el *widget* de ayuda, se adelanta de manera automática mostrando la animación del siguiente gesto a realizar.

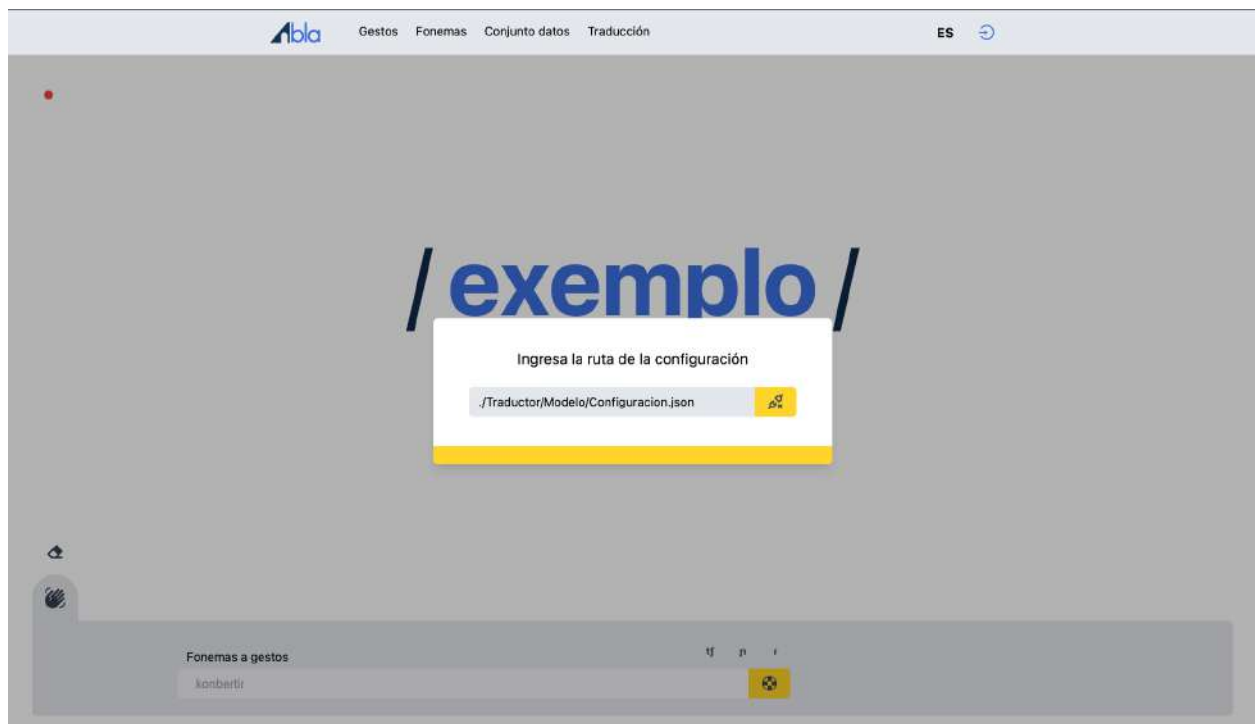


Figura 22: Pantalla inicial donde se identifica el modelo que se utilizará para el reconocimiento de los gestos durante la sesión

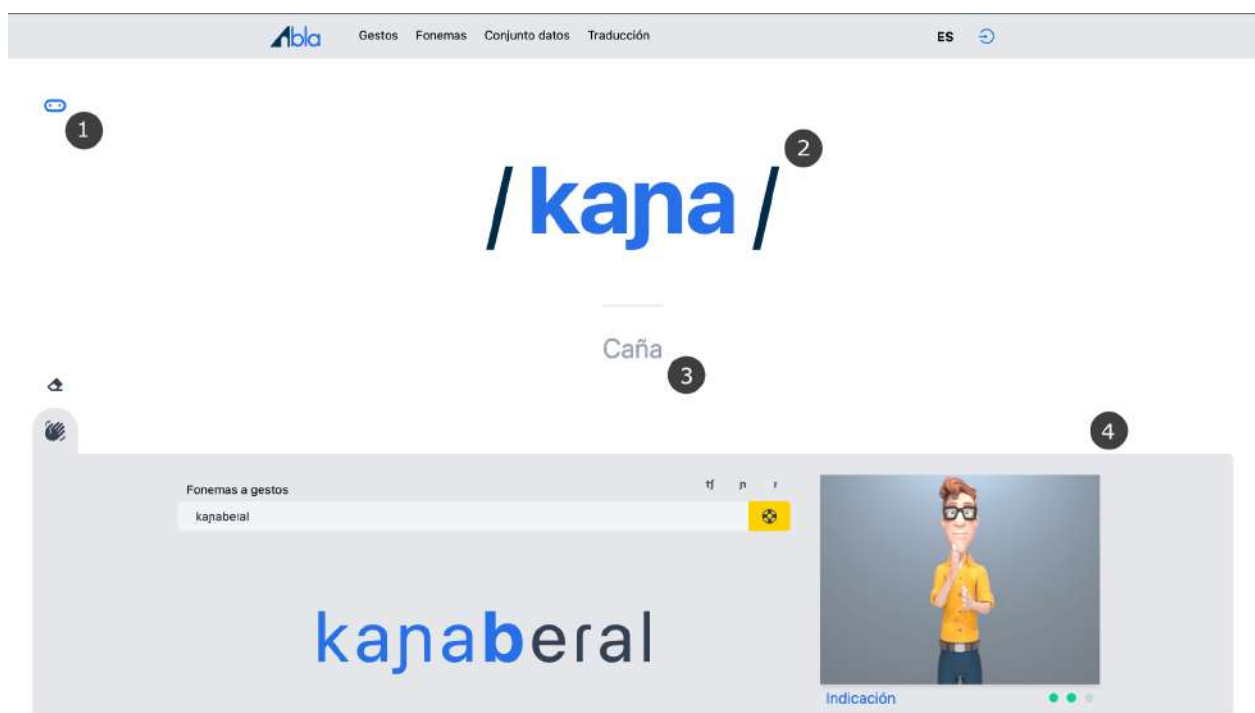


Figura 23: Apartado de traducción en la herramienta “Habla”. El prototipo microbit se encuentra conectado y la palabra en conformación corresponde a ‘Caña’ que es una subcadena de ‘Cañaveral’ por lo que el *widget* avanza automáticamente mostrando el gesto siguiente que debería realizarse para formar la palabra buscada.

Los elementos en la interfaz son los siguientes:

1. Se mantiene en gris cuando no se reconoce una conexión y pasa a color azul cuando hay un prototipo conectado.
2. Palabra de cadenas en conformación mediante los gestos muestreados.
3. Posible representación escrita correcta para la cadena de fonemas
4. *Widget* de ayuda que muestra el siguiente gesto a realizar según el fonema activo escogido.

CAPÍTULO VI: NODO SENSOR

El nodo sensor es el dispositivo que funciona como interfaz de la herramienta con el usuario, permite grabar los movimientos realizados y realizar actividades relacionadas a la traducción y muestreo. Este capítulo aborda inicialmente algunos conceptos y componentes hardware utilizados, se realiza el análisis que define las funciones necesarias en el prototipo y finalmente se diseña el prototipo y la placa para conexión de los componentes.

Marco teórico del nodo sensor

Micro:bit

El micro:bit es un computador de tamaño de bolsillo ideado como un sistema embebido basado en tecnología ARM y hardware de código abierto diseñado por la BBC, principalmente enfocado para uso educacional en el Reino Unido[27] pero que ha extendido su mercado para ser implementado en diversos proyectos embebidos por su integración sencilla, la capacidad de ser completamente programable, entradas y salidas, sensores integrados y componentes como un arreglo de leds y par de botones para la interacción con el usuario.

De los sensores que incluye internamente el micro:bit, el acelerómetro[27,28], incluido en el mismo encapsulado IMU con la brújula, permite la detección de movimiento en términos de la fuerza de aceleración en los 3 ejes, X, Y y Z, retornando las mediciones en mili-gs.

El rango de valores en el acelerómetro se definen entre +2g a -2g, registrándose en una escala de valores entre -1024 a 1024, pero es capaz de identificar los gestos de 3g, 6g y 8g, si es sometido a tales aceleraciones.

En el Apéndice A pueden encontrarse las características técnicas del dispositivo.

Módulo ESP-01 Wi-Fi

El módulo ESP-01 es una placa de desarrollo que incorpora el SoC ESP8266 en un encapsulado de menor tamaño y microcontrolador de 32 bits MCU de ultra baja potencia, un transceptor Wi-Fi con pila TCP/IP con soporte del estándar IEEE802.11, de tal forma que se trata de una solución autocontenida de conectividad Wi-Fi que lo habilita para añadir módulos a un dispositivo de red existente o crear un controlador de red separado[29].

Por defecto el módulo viene cargado con el *firmware* AT para controlar y programar el dispositivo microcontrolador, aunque su *firmware* puede ser cambiado para aprovechar la capacidad completa del microcontrolador.

En el Apéndice A pueden encontrarse las características técnicas del dispositivo.

Modos de funcionamiento

- *Access Point* o AP:
 - Refiere al modo de funcionamiento de un dispositivo de red para actuar como un transmisor central y receptor de señales de radio inalámbricas, específicamente de dispositivos Wi-Fi, y se trata de pequeños dispositivos físicamente que recuerdan de manera cercana los *routers* caseros de banda ancha[30].
 - En el módulo ESP-01 la activación de este modo permite crear una red inalámbrica Wi-Fi a la que otros dispositivos pueden conectarse para interactuar con la placa de desarrollo ESP-01, los dispositivos agregados a la red o el dispositivos conectado al módulo.
- *Station* o STA:
 - En el modo estación sencillamente los módulos actúan como dispositivos que se desean agregar a una red inalámbrica existente y administrada por un dispositivo de red externo, tal como puede ser un *router* casero de banda ancha.
- AP-STA
 - Tal como se infiere del nombre, la tarjeta de desarrollo actúa desarrollando los procesos y el comportamiento de ambos modos al mismo tiempo.

Módulo TP4056

El módulo TP4056 es un cargador de batería Li-Ion lineal autónomo de 1A con termorregulación en sop-8, capaz de cargar y descargar, baterías de celda única de manera segura[31]. Protege las baterías de ser sobrecargadas cuando permanecen alimentadas excediendo el ciclo de carga de estas, o de descarga excesiva, condición que es igualmente peligrosa para las baterías de litio pudiendo inclusive arruinar por completo el funcionamiento de la batería.

En el Apéndice A pueden encontrarse las características técnicas del dispositivo.

Análisis del nodo sensor

El prototipo del nodo sensor debe cumplir con los siguientes requerimientos funcionales de manera que su construcción y ensamble incluya todas las características necesarias para utilizar la herramienta:

- El usuario podrá realizar el gesto motriz y ser muestreado por el nodo sensor a través de las mediciones de la aceleración en los 3 ejes dimensionales.
- El usuario será capaz de indicar el muestreo de un gesto permaneciendo en contacto con el sensor táctil del prototipo.
- El usuario será capaz de indicar el fin del muestreo de un gesto interrumpiendo el contacto táctil con la placa.
- El nodo sensor debe comunicar un gesto muestreado al nodo de procesamiento.

- El micro:bit debe comunicarse con el módulo ESP-01 para la conexión Wi-Fi a la red del nodo de procesamiento, permitiendo de esta manera el envío y recepción de solicitudes HTTP.
- El usuario debe tener la facilidad de cargar la pila del nodo con un cable de entrada común tal como la micro usb.

Los requerimientos no funcionales del nodo sensor van a permitir mantener la calidad y usabilidad del prototipo construido:

- Los gestos deben poder ser muestreados a una tasa de muestreo de 27Hz.
- La comunicación entre el nodo sensor y el nodo de procesamiento debe utilizar una tecnología inalámbrica.
- El nodo sensor debe ser fácilmente portable como un *wearable* y no impedir la sensibilidad y movimiento natural del brazo y mano del usuario.
- La placa táctil debe ser colocada de manera que su contacto y separación requiera el menor esfuerzo, se encuentre en un lugar accesible y sea ajustable a preferencia del usuario.
- La carga de la batería debe poderse realizar utilizando una fuente de alimentación portable común o un regulador de voltaje para dispositivos como celulares, que entreguen una salida de 5V en tensión.
- La matriz de LED debe poder indicar con símbolos el estado y errores para el conocimiento del usuario de manera sencilla.
- El envío del gesto muestreado no debe tardar más de 750 ms desde que se terminó de realizar el gesto hasta la recepción en el nodo de procesamiento.

Diseño del nodo sensor

El componente protagónico del prototipo es el micro:bit, dispositivo que aporta al nodo gran poder de procesamiento con su procesador ARM, además de portabilidad y reducido tamaño. La integración de la IMU facilita la recuperación de las mediciones en la aceleración a través de las funciones programadas en la biblioteca microbit de los diferentes entornos disponibles, mientras la integración nativa de una matriz de LED posibilita el despliegue de símbolos que indiquen de manera sencilla e intuitiva, el estado del sistema, configuración o advierta de errores durante la ejecución de algún proceso.

Portabilidad

Buscando imitar las prestaciones en portabilidad y comodidad de un dispositivo *wearable*, se propone un diseño que asemeja el dispositivo a un reloj inteligente y en el que la cara frontal del micro:bit cumple como carátula o pantalla.

El tamaño del micro:bit, aunque reducido, es de dimensiones mayores al estándar de los relojes inteligentes en el mercado, razón por la que los demás dispositivos y la placa del circuito se integran en capas que incrementan la altura pero se evita extender el ancho o largo del prototipo.

Inicio y finalización del muestreo

La propuesta inicial consideraba un sensor de pulso para marcar el inicio y final del muestreo de un gesto, sin embargo movimientos involuntarios fácilmente podrían desencadenar lecturas falsas y se descartó la idea.

El sensor implementado forma parte de un sencillo circuito táctil que aprovecha la capacidad resistiva disponible de forma nativa para los pins grandes 0, 1 y 2 del microbit de manera que el muestreo se realiza mientras el usuario hace contacto con la placa metálica. El tacto resistivo en el micro:bit funciona detectando un cambio en la resistencia cuando una señal eléctrica circula a través de un material conductor como parte de un circuito. Utilizando un resistor *pullup* débil en los pines del 0 al 2, el voltaje del pin se iguala a la tierra o disminuye a 0V cuando a través del tacto se completa el circuito [32].

Para este fin se disponen de 2 terminales, el primero de ellos se añade en la parte posterior del prototipo de manera que al asegurar con la correa el nodo, permanece todo el tiempo en contacto con la piel del usuario a la altura de la muñeca. La segunda terminal se añade en una cinta destinada a portarse como anillo y que puede colocarse a conveniencia del usuario. Esta cinta se recomienda utilizar en el dedo índice con la placa mirando al costado de la mano, de esta manera con solo juntar el pulgar al índice se hace contacto con la placa y se indica el inicio de un muestreo. Una imagen de las terminales se muestran en la Figura 24.

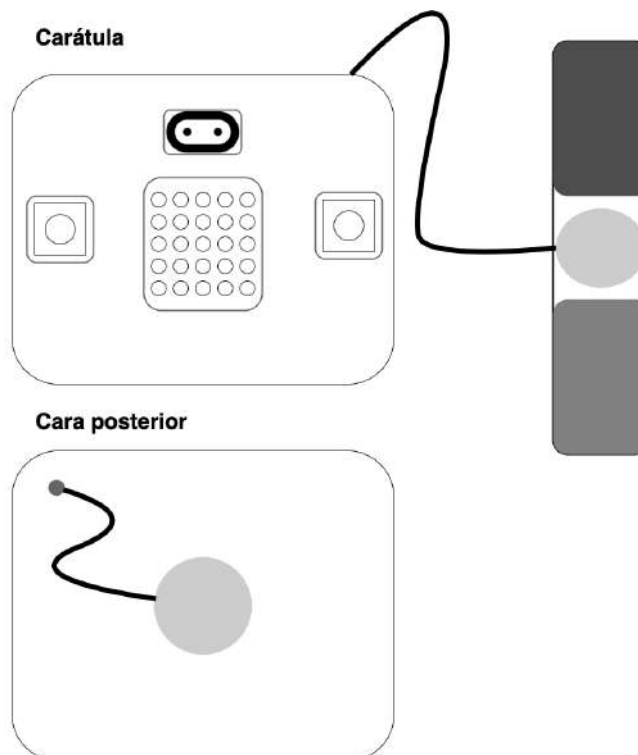


Figura 24: Terminales que conforman el sensor táctil (placas en gris claro). La cinta ubicada al lado de la carátula ocupa un par de velcros a los lados para poder colocarse como anillo (rectángulos de gris oscuro).

Comunicación

De forma nativa la tarjeta micro:bit incluye 2 métodos de comunicación inalámbrica. El primero de ellos es radiofrecuencia y es comúnmente útil en casos de uso de comunicación entre placas de desarrollo exclusivamente. El segundo medio, más versátil al primero, es el Bluetooth *Smart*

Technology, tecnología considerada en la propuesta de solución inicial, sin embargo durante la investigación y experimentación esta comunicación nativa se descarta y es solucionada implementando la conexión por Wi-Fi.

Para la memoria flash del dispositivo micro:bit, la pila del protocolo Bluetooth es demasiado pesada como para ejecutar cualquier programa cuando se desarrolla en Micro Python, razón por la que no es una función disponible y se optó por experimentar con otras plataformas. Utilizando TypeScript, la comunicación Bluetooth fue exitosa pero limitada, las capacidades del dispositivo una vez más tan solo permitían la interacción con un único recurso a la vez, y aunque efectivamente se pueden recuperar las mediciones del acelerómetro, se pierda la capacidad de recibir eventos desde el usuario, mediante botones presionados por ejemplo, o comunicar el estado a través del arreglo de LEDs.

La alternativa de comunicación elegida fue mediante Wi-Fi con la implementación del módulo ESP-01. Esta decisión permite adoptar una arquitectura basada en eventos que facilita la lógica y forma de comunicación.

Entre la tarjeta micro:bit y el módulo Wi-Fi, se utiliza una comunicación serial de manera que las terminales transmisoras y receptoras se conectan para cada uno, pines fijos 2(Tx) y 7(Rx) en el ESP-01, y los pines grandes 0(Tx) y 1(Rx) en el micro:bit configuradas a través de código.

La ejecución de instrucciones en el módulo ESP-01 se consigue utilizando comandos AT[33,34], los cuales proveen todas las funciones requeridas para asignar un modo de funcionamiento(AP, STA, AP-STA), conexión y desconexión a una red, configuración y datos de la red, información de dispositivos, conexiones TCP y UDP, asignación y exposición de servicios (por ejemplo HTTP en puerto 80), así como recepción y envío de solicitudes HTTP[33,35].

Batería

Se integra al nodo una batería que provee al prototipo de autonomía para la realización de gestos sin complicaciones y restricciones de cables de alimentación. La batería utilizada es una batería reciclada con capacidad de hasta 4.2V tipo recargable Li-ion.

Para la carga y descarga segura de la batería se incluye el módulo TP4056, que además provee de una entrada micro usb, un tipo de conector común aunque el estándar actualmente lo ha reemplazado el tipo C, aún con eso gran cantidad de dispositivos de modelos anteriores o de gamma de entrada, siguen incluyendolo como entrada para alimentación, por lo que es seguro que cualquier usuario cuente con uno de estos para la recarga de la batería.

Finalmente la carga se logra utilizando una fuente de energía de reserva externa o la mayoría de reguladores de uso común que están destinados a cargar dispositivos móviles y portátiles.

A continuación se muestra una imagen que representa las capas de los componentes que conforman al prototipo sensor (Figura 25).

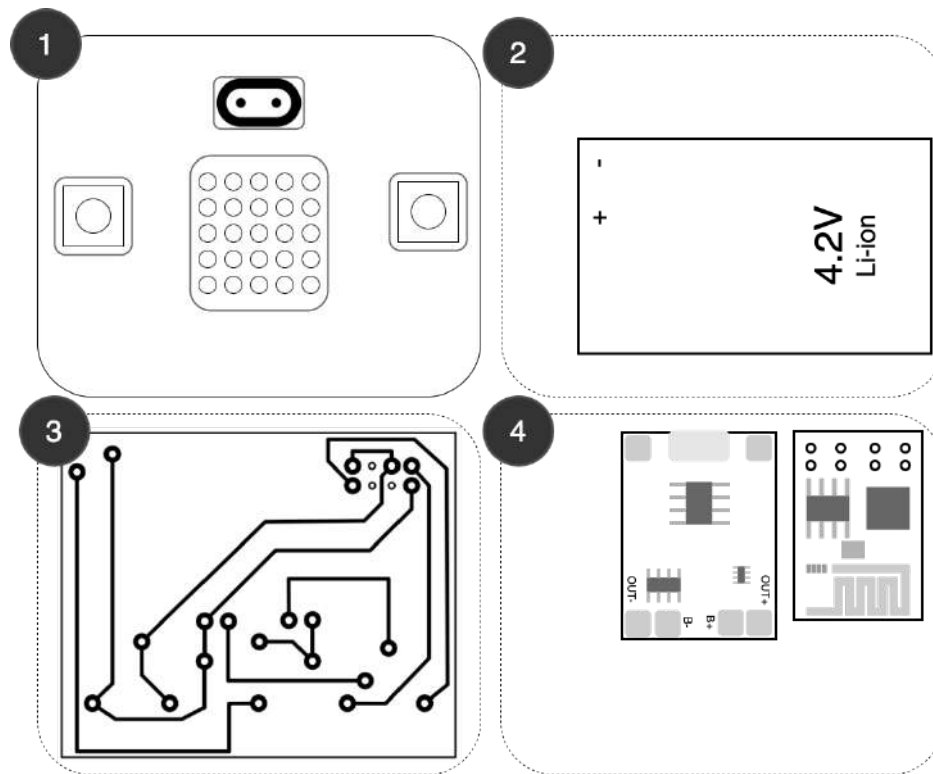


Figura 25: Representación de los componentes en capas del prototipo sensor.

1. Carátula del prototipo sensor y micro:bit
2. Batería de litio tipo Li-ion de 4.2V
3. Placa impresa de circuito
4. (De izquierda a derecha) Módulos TP4056 y ESP-01

Circuito

El circuito diseñado permite la alimentación de los dispositivos y la comunicación serial entre el módulo ESP-01 y la micro:bit.

El diseño se digitaliza utilizando el software EasyEDA, herramienta que posteriormente permitirá la creación del circuito para impresión profesional en PCB(Figura 26).

Los componentes hardware que conforman al circuito son los siguientes:

- Tarjeta de desarrollo micro:bit
- Módulo de conectividad WiFi ESP-01
- Módulo de carga lineal para baterías Li-ion de una celda con protección TP4056
- Mini Switch
- Batería de litio recargable de 3.7V

A continuación se muestra el esquemático del circuito diseñado:

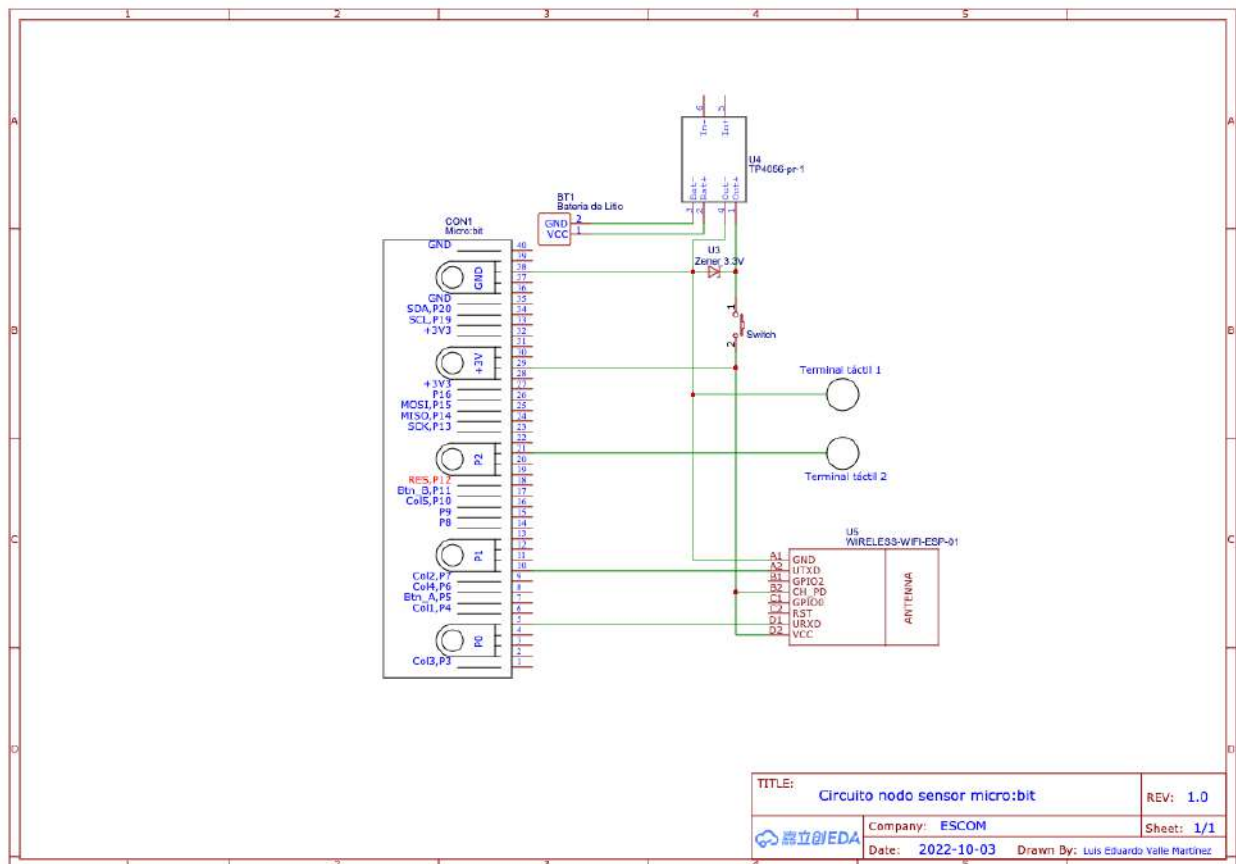


Figura 26: Esquema de las conexiones entre los componentes que conforman el nodo sensor.

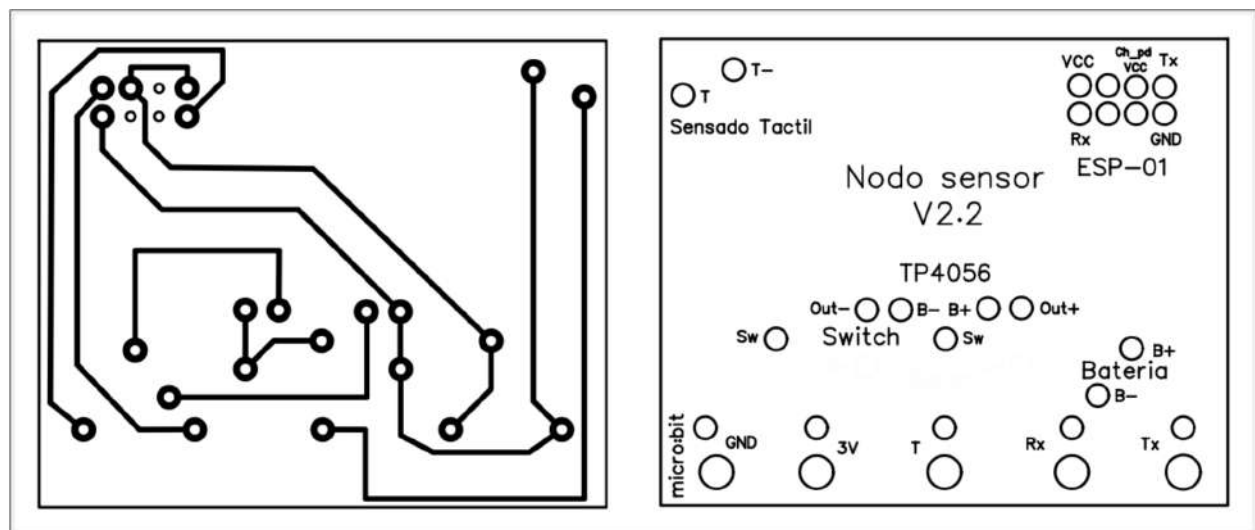


Figura 27: Imagen del circuito diseñado para la posterior transferencia a la placa fenólica o impresión en PCB.

El recuadro de la izquierda son las pistas del circuito sobre cobre y las caras en las que se sueldan los componentes. El recuadro en la derecha se imprime en la cara posterior y sirve como referencia para localizar la posición de los elementos (Se encuentra en modo espejo para obtener una impresión correcta).

Carcasa del prototipo

Buscando cumplir con el cometido de asemejar la apariencia y portabilidad del nodo sensor al de un reloj inteligente, se diseñó e imprimió una carcasa que tiene las siguientes características:

- Ofrecer estructura, soporte y protección a los componentes del nodo
- Realizar una disposición de los componentes de manera que se ocupe el menor volumen posible
- Permitir el montaje/desmontaje del dispositivo de manera sencilla y sin ocupar pegamento, tan solo ensamblando los componentes de la carcasa
- Realizar un diseño que pudiera ser impreso utilizando una impresora 3D

Durante la etapa del diseño de la carcasa se analizó la disposición óptima de los componentes para facilitar el soldado, se disminuyera el volumen ocupado y se hicieran disponible todos los puertos utilizados: Carga del dispositivo, transmisión de programas a la placa de desarrollo, encendido y apagado del dispositivo, y las terminales táctiles para el muestreo de los gestos.

De esta fase se obtuvieron las proyecciones ortogonales(Figura 28) que sirven para definir las dimensiones de las partes de la carcasa una vez se encuentra como modelo tridimensional.

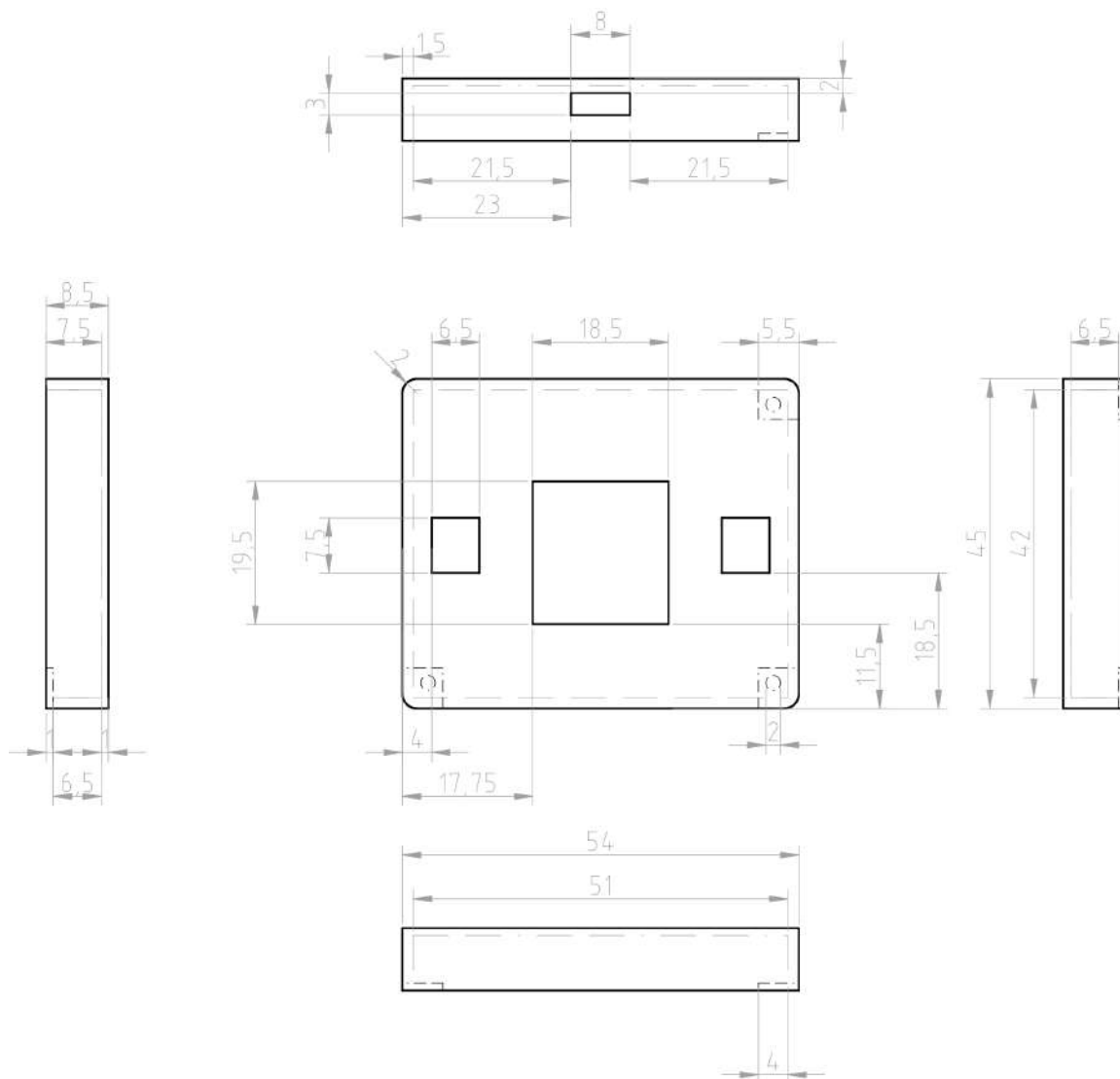


Figura 28: Proyecciones ortogonales de la pieza carátula de la carcasa del prototipo

Utilizando el programa para modelado y animación en 3 dimensiones, Blender, se concluyó un modelo compuesto por 3 piezas(Figura 29) independientes que se ensamblan juntas utilizando la geometría de las partes y 3 tornillos pequeños.

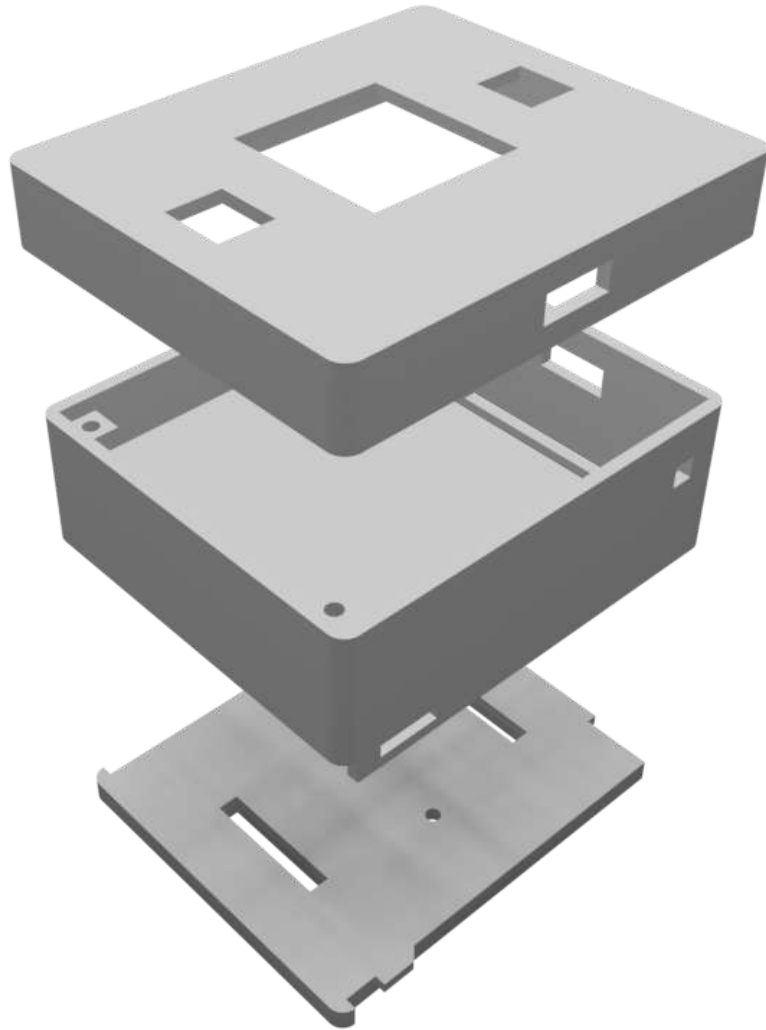


Figura 29 Vista del modelo tridimensional de la carcasa por pieza

Tabla 8: Piezas que conforman la carcasa del prototipo

Pieza	Dimensiones	Descripción
Pieza 1 (Carátula)	47x55x8.5mm	Sirve como protección a la placa de desarrollo además de que la sostiene en lugar. Cuenta con unos soportes con los que se une con el cuerpo utilizando los tornillos.
Pieza 2 (Cuerpo)	47x55x19.5mm	Utilizando los soportes se fija la placa PCB y se une con la carátula. Cuenta con orificios al tamaño de los puertos que se exponen(micro-usb para carga, terminal táctil + y switch). Separa la placa PCB de la batería.
Pieza 3 (Tapadera)	47x55x3mm	Se ensambla con 2 pestañas que embonan en ranuras en el cuerpo. Cuenta con una cuenca para colocar la placa que funciona como terminal - del sensor táctil y un par de ranuras para las correas.

Una vez ensamblado el dispositivo, colocadas las terminales del sensor táctil y las correas para sujetarlo, tiene la siguiente apariencia en la Figura 30.

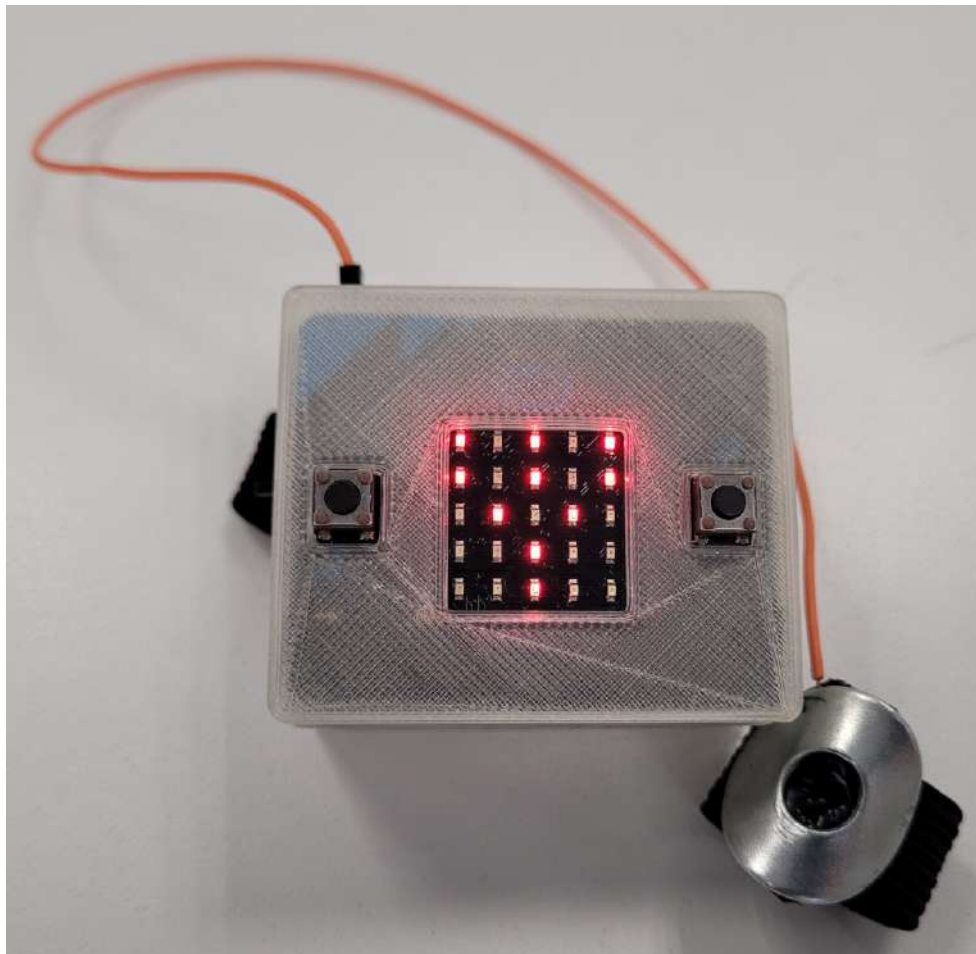


Figura 30: Prototipo sensor ensamblado con la carcasa y terminales táctiles

Componentes lógicos

Para el prototipo sensor se desarrollaron 2 componentes lógicos principales correspondientes a las 2 tareas de la herramienta, la recolección de repeticiones para el conjunto de datos y el muestreo de gestos para la traducción a fonemas, palabras y finalmente voz.

Existen distintas opciones para programar el comportamiento de la tarjeta micro:bit, la escogida para el desarrollo de la solución fue el ambiente que provee micro:bit en conjunto con Microsoft nombrado ‘Microsoft MakeCode for micro:bit’, es un ambiente de desarrollo basado en una plataforma web que ya incluye la biblioteca de funciones del micro:bit y la compilación y procesamiento del programa de forma que únicamente se descarga el archivo hexadecimal que se transfiere a la tarjeta de desarrollo.

Utilizando TypeScript para el desarrollo de los componentes se presentó un primer gran reto ocasionado por la inexistente documentación para conseguir la comunicación entre el módulo ESP-01 y la tarjeta de desarrollo. Se tomó como referencia el ejercicio desarrollado en [33], un servidor web que permite prender y apagar un led conectado al micro:bit, para el desarrollo de los componentes lógicos del prototipo.

Utilizando comandos AT se configuran las acciones del ESP-01 desde la tarjeta de desarrollo y logra de esta forma establecer la conexión con un punto de acceso de red local (router o módem casero) y la emisión de peticiones HTTP a las API de datos y de traducción.

Creación conjunto de datos

Se encarga de:

1. Realizar la conexión a la LAN inalámbrica provista por un router o módem casero
2. Acordar la conexión con la API de datos para retomar o crear un nuevo conjunto de datos
3. Muestrear los gestos del usuario y
4. Comunicarlos a la API a través de una petición POST al endpoint '/datos-api/dataset'.

Dado que es posible que surjan errores como agotar la memoria del micro:bit por un gesto demasiado largo, por defecto el dispositivo se reinicia y se recupera el conjunto de datos en proceso desde el último gesto recopilado, evitando así perder las repeticiones ya captadas del usuario.

Durante todo el proceso se muestra en el arreglo de LED's diferentes iconos y símbolos para indicar el estado del proceso en apoyo a la herramienta gráfica web. A continuación se describen los iconos y símbolos mostrados junto con su significado (Tabla 9)

Tabla 9: Iconos y símbolos del programa de creación de conjuntos de datos del prototipo sensor

Icono o símbolo	Etapas del proceso	Significado
Tridente	Configuración de la conexión	Se muestra el tridente durante la configuración del ESP-01 y la conexión a la red local. De ser exitosa la conexión a la red no tarda más de 6 segundos en desaparecer.
Fantasma	Establecimiento de conexión con la API	Aparece durante la etapa de acuerdo entre la tarjeta de desarrollo y la API de datos. De ser exitoso el primer contacto el fantasma desaparece en máximo 5 segundos. De otra forma se muestra el tipo de error y se intenta de nuevo el primer contacto.
E0	Establecimiento de conexión con la API y comunicación de gesto muestreado	Error en la solicitud para abrir una conexión TCP con el endpoint de la API
E1	Establecimiento de conexión con la API y comunicación de gesto muestreado	Error o tiempo agotado en la preparación para enviar la solicitud HTTP
E2	Establecimiento de conexión con la API y comunicación de gesto muestreado	Tiempo agotado para la recepción de respuesta desde el servidor
E3	Establecimiento de conexión con la API y comunicación de gesto muestreado	Error durante el cierre de la conexión con la API

Icono o símbolo	Etapas del proceso	Significado
Dígito	Muestreo de gestos	Muestra el primer dígito de la clase del gesto actualmente en muestreo. Cuando se comunica de manera exitosa este dígito es igual o mayor a 0, de otra forma se muestra -1 en caso de error.
Cara sonriente	Muestreo de gestos	Indica el fin del muestreo y recopilación de repeticiones del conjunto de datos
:(021	Muestreo de gestos	Indica que la memoria de programa del micro:bit se agotó. Obliga por defecto el reinicio del programa.

Traducción de gestos

Se encarga de:

1. Realizar la conexión a la LAN inalámbrica provista por un router o módem casero
2. Acordar la conexión con la API de datos para iniciar la conformación de palabras
3. Muestrear el gesto del usuario y
4. Comunicarlo a la API a través de una petición POST al endpoint '/traducir/gesto.
5. O eliminar el último fonema muestreado
6. O iniciar la reproducción con voz inmediatamente ignorando que se agote el tiempo para su reproducción automática

Durante todo el proceso se muestra en el arreglo de LED's diferentes iconos y símbolos para indicar el estado del proceso en apoyo a la herramienta gráfica web. A continuación se describen los íconos y símbolos mostrados junto con su significado (Tabla 10):

Tabla 10: Iconos y símbolos del programa de traducción de gestos del prototipo sensor

Icono o símbolo	Etapas del proceso	Significado
Tridente	Configuración de la conexión	Se muestra el tridente durante la configuración del ESP-01 y la conexión a la red local. De ser exitosa la conexión a la red no tarda más de 6 segundos en desaparecer.
Craneo	Establecimiento de conexión con la API	Aparece durante la etapa de acuerdo entre la tarjeta de desarrollo y la API de traducción. De ser exitoso el primer contacto el craneo desaparece en máximo 5 segundos. De otra forma se muestra el tipo de error y se intenta de nuevo el primer contacto.
E0	Establecimiento de conexión con la API y comunicación de gesto muestreado	Error en la solicitud para abrir una conexión TCP con el endpoint de la API

Icono o símbolo	Etapas del proceso	Significado
E1	Establecimiento de conexión con la API y comunicación de gesto muestreado	Error o tiempo agotado en la preparación para enviar la solicitud HTTP
E2	Establecimiento de conexión con la API y comunicación de gesto muestreado	Tiempo agotado para la recepción de respuesta desde el servidor
E3	Establecimiento de conexión con la API y comunicación de gesto muestreado	Error durante el cierre de la conexión con la API
Fonema	Muestreo de gestos	Muestra el fonema reconocido por el modelo de predicción. Cuando se comunica de manera exitosa este muestra un carácter alfabético, de otra forma se muestra -1 en caso de error.
Check	Eliminar último fonema y reproducción de voz inmediata	Indica que la acción fue realizada exitosamente. De otra manera se muestra el -1
:(021	Muestreo de gestos	Indica que la memoria de programa del micro:bit se agotó. Obliga por defecto el reinicio del programa.

CAPÍTULO VII: NODO DE PROCESAMIENTO

Cuando se adapta una arquitectura basada en eventos se consideran primordialmente 2 actores en el sistema, los productores de eventos que suelen ser dispositivos de sensado y recopilación de información en un sistema IoT, y los consumidores de eventos los cuales sirven peticiones de los primeros mediante funciones y algoritmos que requieren una mayor infraestructura y poder computacional.

El nodo de procesamiento cumple con el rol de consumidor de eventos pues se encarga de proveer servicios a los nodos sensores que sería imposible para ellos ejecutar de valerse tan solo de su poder computacional.

En el capítulo se estudian los conceptos clave que deben considerarse para implementar la arquitectura, se realiza un análisis de las funciones que debe desempeñar el nodo y se diseñan los componentes lógicos que se encargarán de procesar las peticiones de los nodos sensores.

Marco teórico

RESTful API

La API o *Application Programming Interface*, es una interfaz de programación en la que se definen unas reglas específicas para lograr la comunicación entre 2 sistemas software. La exposición de esta interfaz, cumple con la necesidad de comunicar una aplicación a otra que provee un conjunto de rutinas, funciones y procedimientos o métodos[36].

El término REST proviene de la arquitectura de software conocida como *Representational State Transfer*, la cual impone condiciones sobre el funcionamiento de una API, especificando parámetros requeridos durante una consulta, diferentes tipos de procedimientos o métodos y puntos de anclaje o *endpoints*[36].

La arquitectura REST trabaja sobre el protocolo HTTP, de manera que sus implementaciones heredan los métodos principales de HTTP (GET, POST, PUT, etc.). Así mismo se cuenta con un componente llamado *HTTP Status Code*, que informa al cliente o consumidor de la API que debe hacer con la respuesta a través de un código de estado HTTP[35].

Una API Restful entonces, es una interfaz entre 2 sistemas computacionales que intercambian información a través de internet utilizando el protocolo HTTP y la arquitectura REST imponiendo condiciones de consulta. Esta interfaz permite a dispositivos conectados a una red, solicitar como clientes, servicios con procesos especializados o específicos a un servidor.

Servidor TTS *Text-to-Speech*

TTS: *Text-to-Speech* de la fundación Mozilla[37], es una biblioteca para generación avanzada de texto a habla, construida en lo último de investigación y fué diseñada para lograr el mejor equilibrio entre la facilidad de entrenamiento, rapidez y calidad.

TTS viene con modelos pre entrenados, herramientas para medir la calidad de un conjunto de datos y es ya utilizado con más de 20 lenguajes para productos y proyectos de investigación.

Servicios *Text-to-Speech* en la nube

La manera más sencilla y más común para realizar la transformación de un texto a un audio con habla, es el consumo de un servicios de las grandes nubes existentes en el ecosistema de la web. Entre las tecnologías existentes se mencionan las 3 opciones más destacadas por sus características, modificación de parámetros y costo.

Google *Text-to-Speech*

El servicio disponible en GCP ofrece voces muy convincentes[38] que asemejan al habla natural de los humanos. Cuenta con soporte para español, aunque solo en su versión neutra, sin embargo esta se adapta fácilmente al acento y estilo de habla en el centro del país.

Voces

- Provee 3 voces convincentes en idioma Español con acento neutro.
- Cuenta con 1 voz femenina en español y las restantes 2 son masculinas.
- Soporta SSML para texto enriquecido.
- Con la modificación de parámetros es posible lograr voces para:
 - Hombre y mujer joven,
 - Hombre de edad mediana.

Precio

Al igual que sucede con las demás grandes nubes, su servicio ofrece una capa gratis para voces estándar y voces estilizadas *WaveNet*(modelo mejorado que otorga naturalidad a la forma y tono en que se interpreta la voz).

- Desde 0 hasta 4 millones de caracteres en voces estándar.
- Desde 0 hasta 1 millón de caracteres en voces *WaveNet*.

Una vez superada la capa gratis el costo se añade por cada millón de caracteres extra:

- Cargo de \$4 USD para voces estándar.
- Cargo de \$16 USD para voces *WaveNet*.

Desventajas

- La voz femenina, en comparación con las masculinas, suena menos natural
- La modificación de parámetros no permite conseguir fácilmente distintas voces con un tono natural en rangos de edad distintos.
- Los caracteres del SSML y espacios también se cuentan dentro de la contabilización para cotización del servicio.

Microsoft Azure Text-to-speech

Azure dispone de un servicio para sintetización de texto a voz utilizando una tecnología de voces neurales que ofrece resultados naturales y convincentes[39].

Voces

- Ofrece voces convincentes en idioma Español y que además categoriza como con acento de México.
- Se encuentran disponibles un total de 8 voces femeninas y 7 masculinas.
- Con la modificación de parámetros es posible lograr voces para:
 - Hombre joven y de mediana edad.
 - Mujer joven y niña pre puberta.

Precio

Al igual que sucede con las demás grandes nubes, su servicio ofrece una capa gratis considerando una solicitud simultánea cada mes.

- Desde 0 hasta 500 mil caracteres.

Una vez superada la capa gratis el costo se añade por cada millón de caracteres extra:

- Cargo de \$16 USD.

Desventajas

- Aunque naturales las voces con el modelo neural, siguen siendo inferiores a la experiencia que ofrecen las *WaveNet* de GCP.

Amazon Polly

Al igual que sucede con las 2 nubes anteriores, Polly ofrece voces con tecnología basada en redes neuronales, logrando resultados con voces naturales[40].

Voces

- Ofrece voces en idioma Español categorizadas con acento neutral y mexicano.
- Se encuentra disponible 1 voz femenina para el acento de México.
- Están disponibles 2 voces femeninas con Español de acento neutro y 1 voz masculina.

Precio

El costo para el servicio que AWS ofrece, es una capa gratis renovada mensualmente que depende de si se consumen voces estándar o voces neurales.

- Desde 0 hasta 5 millones de caracteres en voces estándar.
- Desde 0 hasta 1 millón de caracteres en voces neurales.

Una vez superada la capa gratis el costo se añade por cada millón de caracteres extra:

- Cargo de \$4 USD para voces estándar.
- Cargo de \$16 USD para voces neurales.

Desventajas

- Las voces son medianamente convincentes y no compiten con la naturalidad de las de Azure y GCP.

- Únicamente 2 voces femeninas son neurales y por lo tanto más naturales al hablar de una persona.

Análisis del nodo de procesamiento

Las operaciones y procesos de mayor complejidad computacional son delegadas al nodo de procesamiento cumpliendo los siguientes requisitos:

- El nodo de procesamiento debe ser capaz de recopilar los gestos muestreados
- La comunicación con el nodo sensor debe realizarse de manera inalámbrica.
- El nodo de procesamiento debe habilitar una interfaz que permita recibir información en tiempo asíncrono desde el nodo sensor después de un evento.
- El nodo de procesamiento debe ofrecer una interfaz para las configuraciones de la herramienta.
- El nodo de procesamiento debe ofrecer una interfaz para el registro de movimientos como instancias de ejemplo durante la etapa de construcción del conjunto de datos.
- El nodo de procesamiento debe ofrecer una interfaz para la traducción de los gestos pertenecientes al código motriz.
- El nodo de procesamiento debe ser capaz de implementar la metodología y modelo para el reconocimiento de gestos.
- El nodo de procesamiento debe procesar y conformar las clases textuales en texto.
- El nodo de procesamiento debe hospedar un servicio local de traducción de texto a habla.
- El nodo de procesamiento debe ser capaz de reproducir la grabación oral del texto conformado por el usuario.

Los requerimientos catalogados como no funcionales que aseguran la calidad y desempeño óptimo del nodo de procesamiento son:

- El tiempo de traducción que transcurre desde que se recibe el gesto como secuencia temporal hasta la reproducción sonora no debe ser mayor a 1.5 s.
- El nodo de procesamiento debe incluir al computador Raspberry Pi 4 en el proceso.
- El nodo de procesamiento debe incluir información relevante en una interfaz para el monitoreo del estado de la herramienta en todos los procesos.

Diseño del nodo de procesamiento

La arquitectura que propone una API RESTful es ideal para sistemas IoT y embebidos que desarrollan su modelo de negocio alrededor de la ejecución de procesos por eventos, desencadenados ya sea por una acción del usuario, una condición ambiental o del sistema, o la interacción con otros sistemas en tiempos asíncronos.

APIs Web

El esquema de comunicación entre el nodo sensor y de procesamiento sufrió de varios cambios durante el análisis. En una primera instancia la comunicación se pensaba realizar a través de Bluetooth y con el dispositivo micro:bit con el rol de servidor, este tipo de arquitectura obligaba al nodo de procesamiento de contar con un programa cliente que constantemente se encontrara consultando los servicios de la tarjeta de desarrollo, lo cual es en extremo inconveniente y además desecha la posibilidad de escalar o actualizar el número y tipo de dispositivos (un reloj inteligente por ejemplo).

La integración del módulo Wi-Fi en el prototipo dio paso a integrar una arquitectura más adecuada y que además es ampliamente utilizada en la industria, las arquitecturas basadas en eventos para dispositivos de IoT (*Internet of Things*) es una propuesta de solución a la naturaleza asíncrona de este tipo de soluciones, donde un conjunto dispositivo recolectores y sensores de datos opera a través de interacciones con el usuario, interacciones con otros sistemas o condiciones ambientales; Todas estas ocurren sin que un sistema sea capaz de predecir la naturaleza y momento en que sucederán, por lo tanto los procesos se desencadenan a partir de que un evento sucede y tan solo en ese instante es que consume el servicio requerido.

La solución implementa esta arquitectura a través de proveer una API Web que ofrece un conjunto de *endpoints* que el dispositivo sensor puede consumir cada vez que lo requiera. Este enfoque permite centralizar el procesamiento en un dispositivo *backend* completamente transparente a los sistemas que lo consumen, las acciones que se realicen cada vez que se consulta cierto *endpoint* no incunben a estos pero se tiene una expectativa bien definida de lo que resultará, pues la comunicación mediante la API representa un contrato entre las partes donde explícitamente se definen los parámetros que recibe el servicio y lo que este otorga de vuelta.

El esquema típico de una API es *stateless* o sin estado, lo que indica que el servidor que expone los servicios no almacena ningún dato, información o resultado calculado, sin embargo las APIs de la solución son *stateful* pues debido a las limitaciones de memoria del prototipo sensor, las palabras y texto que construye el usuario se almacenan en el mismo servidor.

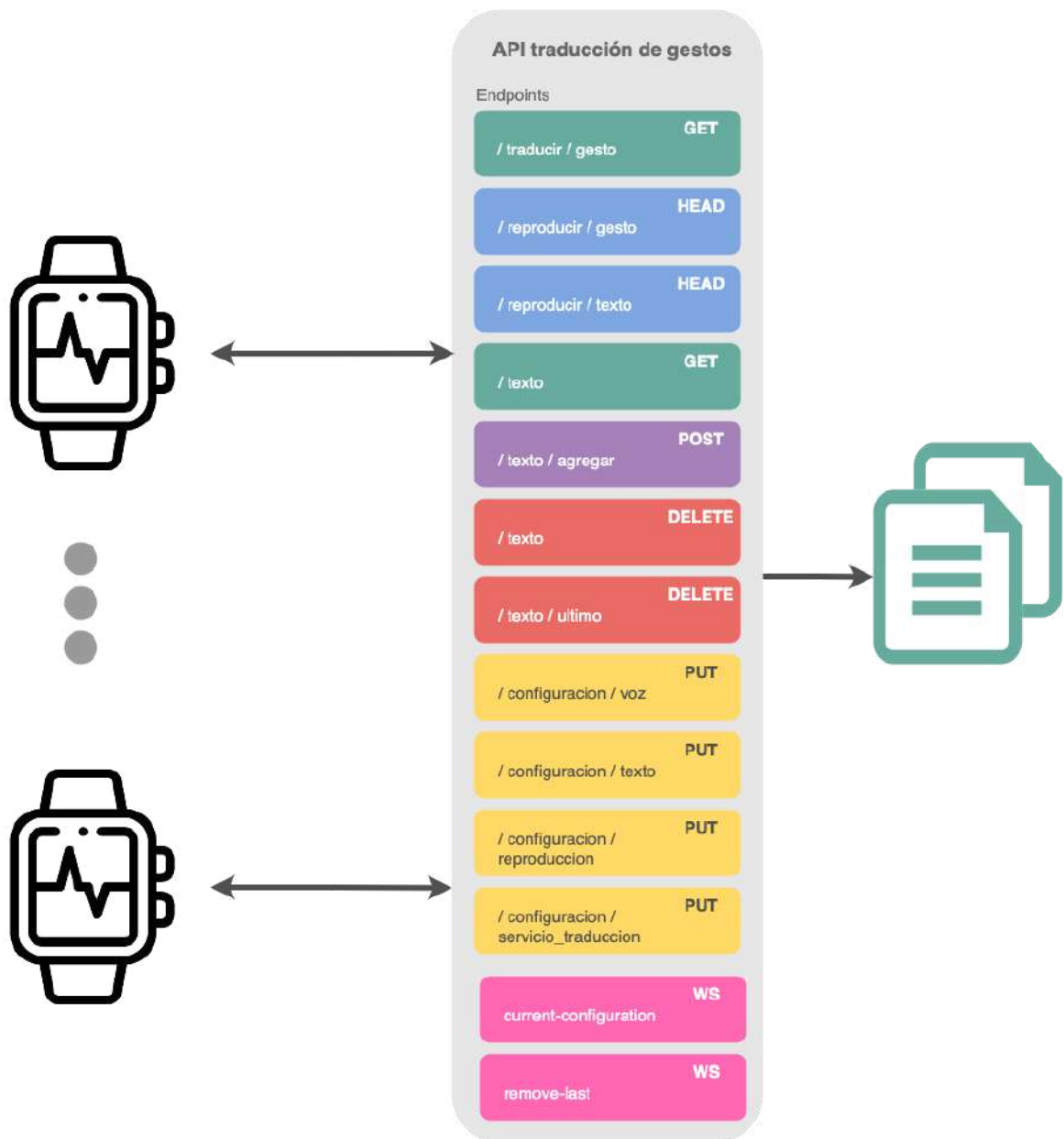


Figura 31: Diagrama de la arquitectura de comunicación a través de una API; La API representada aquí es la de traducción de gestos.

API de conjunto de datos

Esta primera API se desarrolla con el objetivo de ofrecer los servicios de recolección de gestos durante la fase de creación del conjunto de datos propio, posteriormente a la recolección estos datos serán utilizados para el entrenamiento del modelo clasificador.

Adicional a los end-points típicos para consulta del micro:bit u otro cliente a través de los métodos de petición HTTP, se añade aquellos para la comunicación bidireccional de los sockets establecidos con la interfaz web. La tecnología de los sockets es Socket.IO, una biblioteca que abstrae la complejidad de los sockets, y que fácilmente se implementa con FastAPI para el intercambio de mensajes.

Esta API incluye los siguientes puntos de anclaje y el servicio que provee (Tabla 11).

Tabla 11: Se mencionan los puntos de anclaje, el tipo de método y el servicio que ofrece cada *endpoint* de la API de conjuntos de datos.

Endpoint	Método HTTP	Servicio
/datos-api/dataset	GET	Cuando se solicita este servicio se inicia un nuevo conjunto de datos en el que se almacenarán los siguientes gestos.
/datos-api/configuración	PUT	Permite configurar el nombre del conjunto, definir el gesto y repetición desde el que se quiere iniciar el muestreo
/datos-api/dataset	POST	Este método recibe las mediciones independientes de cada eje espacial y los integra como una nueva instancia del gesto actual que se muestrea.
/datos-api/dataset	DELETE	Elimina la última instancia registrada
/datos-api/dataset/guardar	PUT	Guarda en archivo csv las instancias recopiladas hasta el momento
/dataset/prueba	POST	Recibe el gesto muestreado por el nodo sensor y notifica de su correcta recepción.
/datos-api/dataset/retomar	GET	Comparte la configuración actual del conjunto de datos que se graba(clase de gesto actual e índice de última medición guardada). En ocasiones el dispositivo sensor lanza un error que requiere el reinicio del prototipo, con este método se retoman las grabaciones.
/datos-api/dataset/grabado	GET	Servicio para administradores - Retorna el estado actual del conjunto de datos y las mediciones guardadas hasta el momento.
current-configuration	WS	Sirve como primer contacto para la creación o recuperación de la configuración de un conjunto de datos

API traducción de gestos

Esta segunda API se enfoca en proveer los servicios necesarios durante el uso de la herramienta con los usuarios. Los servicios de la API requieren de previamente contar con un modelo entrenado, el servicio de TTS corriendo en la red local o las credenciales de la cuenta en la plataforma de nube privada para el servicio de traducción.

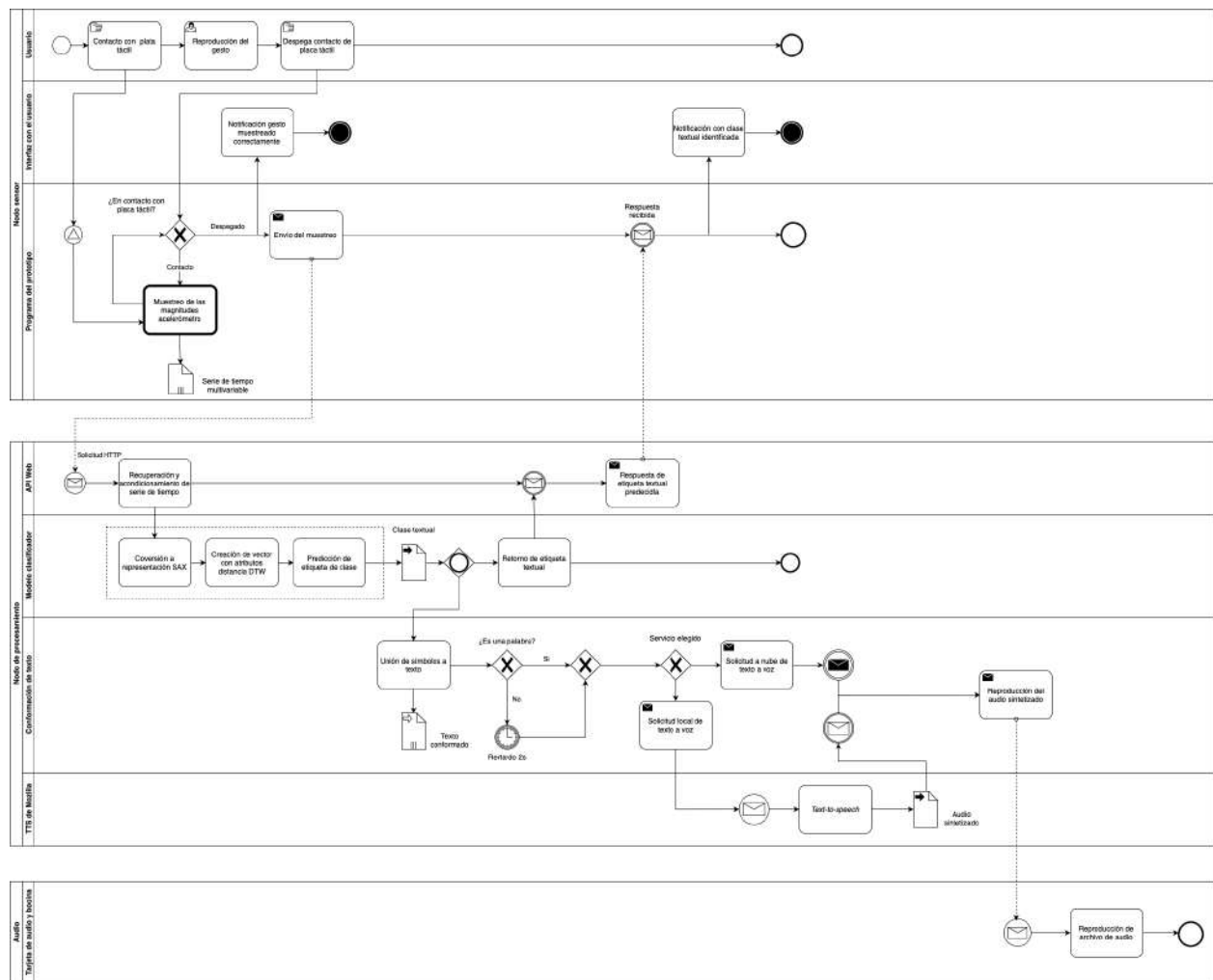
Al igual que sucede con la API de gestos, se añade a los end-points típicos aquellos para la comunicación bidireccional de los sockets con la interfaz web.

Los servicios que ofrece se mencionan a continuación (Tabla 12):

Tabla 12: Se mencionan los puntos de anclaje, el tipo de método y el servicio que ofrece cada *endpoint* de la API de traducción.

<i>Endpoint</i>	Método HTTP	Servicio
/traducir/gesto	POST	Identifica el gesto mediante la predicción de su clase y agrega el fonema al proceso de conformación de palabras
/configuracion	PUT	Permite configurar el modelo para la traducción
/reproducir/gesto	HEAD	Reproduce a través de la tarjeta de audio y las bocinas un gesto o clase textual de entrada.
/reproducir/texto	HEAD	Reproduce a través de la tarjeta de audio y las bocinas una cadena de entrada.
/reproducir/palabra	PUT	Obliga a reproducir la palabra en actual conformación sin esperar los 5 segundos por defecto entre fonema y fonema
/fonemas/gestos	HEAD	Imprime una lista de los nombres de los gestos que corresponden a los fonemas pasados como parámetro
/fonemas/palabras	GET	Regresa la posible correcta forma escrita de la palabra de fonemas
/raspberry/limpiar	PUT	Mediante comunicación con la raspberry se encarga de borrar la palabra actual en conformación
/raspberry/ultimo	DELETE	En comunicación con la raspberry se encarga de eliminar el último carácter recibido
/raspberry/modificar_ espera	PATCH	Modifica el tiempo de espera entre fonema y fonema para reproducir la palabra
/raspberry/cerrar	PUT	En comunicación con la raspberry termina el servidor
current-configuration	WS	Sirve como primer comunicación a través del socket y permite definir la configuración entrenada para la predicción de los gestos
remove-last	WS	Elimina el último fonema de la palabra en conformación

Figura 32: Diagrama BPMN para el proceso de conformación de palabras y reproducción con voz.



CONCLUSIONES

El objetivo principal del trabajo es el desarrollo de una propuesta de solución, a través de una herramienta embebida, que pruebe ser competente para buscar en una etapa posterior la implementación como una herramienta de apoyo a la comunicación que facilite la conformación de palabras y su reproducción con voz, utilizando únicamente gestos del brazo para los pacientes que sufren de trastornos del habla y el lenguaje como las afasias, apraxias y disartrias.

La naturaleza embebida del proyecto obligó a una extensa investigación inicial en búsqueda de no únicamente herramientas relacionadas, trabajos existentes o referencias para entender el contexto de la problemática, también era necesario encontrar, o como sucede con la solución, proponer un método de reconocimiento de gestos con una buena precisión en la predicción pero prestando especial atención a los recursos de procesamiento limitados y un rápido tiempo de predicción cercano al tiempo real.

De esta etapa de investigación, entre los documentos consumidos, se resaltan un par de trabajos[12, 13] que tuvieron la mayor influencia como referentes e inspiración para el desarrollo del método propio “SAX-DTW Features”, el cual logró superar en precisión de predicción y tiempo de entrenamiento y predicción, a los métodos de referencia con una precisión del 94.06% (3.73 puntos porcentuales más que el segundo más cercano) en tan solo 5.181 segundos(una mejora del 248%) para un conjunto de datos con 120 instancias de entrenamiento y 150 instancias de prueba. Implementado en la solución del trabajo, el modelo alcanza una precisión promedio de 99.43% de precisión con un tiempo de entrenamiento de 100ms y un tiempo de predicción de 19ms para conjuntos de datos con 396 instancias de entrenamiento y 44 instancias de prueba.

El método de reconocimiento de gestos se desarrolló en el marco del congreso CORE 2022 organizado por el CIC, para el cual se generó un artículo que fue publicado en la revista de investigación científica *Research in Computing Science* del Instituto Politécnico Nacional, con el nombre “Clasificador de gestos motrices utilizando vectores de atributos distancia DTW sobre series de tiempo en representación SAX”, fungiendo como una revisión del estado del arte en la problemática de reconocimiento de gestos y la teoría detrás de la propuesta de nuevo método; En el mismo congreso, en el mes de Septiembre del año 2022, se llevó a cabo una ponencia en la que se presentaron los resultado del trabajo contrastado con los métodos de referencia.

Acotado de manera particular a la problemática que aborda este documento, se generó una lista de gestos nombrados en conjunto como “Código motriz”, que permiten asociar un movimiento sencillo con un fonema del español, de esta forma su concatenación permite la creación de palabras según su fonética, que se traduce a texto y permiten ser reproducidas como voz mediante la sintetización de una inteligencia artificial.

Se definió una infraestructura basada en eventos para la herramienta embebida, que compuesta por 2 nodos principales, permite al usuario realizar el gesto y obtener como respuesta el fonema, la palabra y/o la reproducción en voz de su entrada.

Enfocado a la interacción directa con el usuario y como interfaz de la herramienta, se desarrolló el nodo sensor, un prototipo parecido a un reloj inteligente, que permite grabar el movimiento del usuario y comunicarlo a un nodo de procesamiento a través de una interfaz API que permite inferir el gesto realizado por el usuario y realizar el procedimiento completo hasta la reproducción de la palabra creada.

Adicional a los productos listados en el planteamiento de la solución, se desarrolló una aplicación web como una herramienta virtual llamada “Habla”, la cual pone a disposición de los usuarios el código motriz completo, acompañado de animaciones del gesto para su sencilla asimilación, un listado de los fonemas del español que se encuentran asociados a los gestos, y un par de apartados desarrollados específicamente como interfaces para las 2 principales tareas de la solución, la grabación de las repeticiones para la conformación de un conjunto de datos por usuario, y la tarea de traducción de un gesto muestreado en tiempo real hasta su reproducción con voz.

De estas últimas interfaces se utiliza la tecnología de sockets lo que permite visualizar al usuario en todo momento con precisión el estado de la tarea que está realizando. Algunas acciones propias del administrador, como eliminar reproducciones grabadas, no se facilitan a través de esta interfaz pues su finalidad es limitarse únicamente en mantener al usuario consciente del estado del proceso y los eventos que desencadena el propio usuario.

Integrados los diferentes componentes que se desarrollaron, se obtiene una herramienta funcional, optimizada para la tarea que desempeña, con una excelente precisión en el reconocimiento de los gestos efectuados por el usuario y que cumple en su totalidad con las metas que se plantearon al inicio del desarrollo en forma de objetivos. Adicionalmente, la adopción de la herramienta aumenta su porcentaje de éxito gracias a la facilidad de aprendizaje e interacción intuitiva que provee la herramienta web, que además es gráficamente atractiva, minimalista en diseño y se enfoca en dar la mejor experiencia al usuario por su fluidez y adaptabilidad a diferentes dispositivos.

TRABAJO A FUTURO

En esta sección se agregan las funciones, componentes y demás tareas que pueden fungir como una actualización, una mejora o un enfoque alternativo que valdría la pena explorar en la solución pero debido al alcance del trabajo, presupuesto o tiempo, no fue posible integrarlas al producto final.

Durante el desarrollo del trabajo se identificaron algunas oportunidades para mejorar la herramienta embebida. La primera de estas se deriva del alcance del trabajo, el cual tiene el objetivo principal de desarrollar la herramienta pero no se considera una fase de implementación en producción, condición que requeriría forzosamente de un alternativa al esquema actual.

En la arquitectura utilizada existe un único nodo de procesamiento que es capaz de servir y atender las peticiones de múltiples nodos sensores dentro de una red local. Sería de esperarse entonces que si múltiples usuarios pueden utilizar la herramienta al mismo tiempo todos estos puedan utilizar un dispositivo con parlantes distinto para comunicar sus mensajes, y sin embargo de primera mano esta dificultad no se resuelve con el esquema actual pues la tarjeta de audio y las bocinas forman parte del nodo de procesamiento.

La propuesta que surge es la creación de un tercer nodo encargado únicamente de la reproducción de los archivos de audio para cada nodo sensor. Este nodo debería existir en misma cantidad que nodos sensores y además asegurar la comunicación mediante Wi-Fi para la recuperación del archivo de audio.

La segunda situación ofrece una oportunidad para mejorar internamente los procesos de producción de texto y posiblemente agregar nuevas funcionalidades a la herramienta. Partiendo de que la relación entre los fonemas y las grafías según la gramática del español, no es una relación biyectiva, entonces no existe un algoritmo sencillo que permita esta transformación.

La idea de propuesta considera utilizar técnicas de Procesamiento de Lenguaje Natural para identificar la grafía que sustituye correctamente al fonema en cierta palabra. A esta tarea se añade la complejidad de palabras homófonas (misma pronunciación pero significado y ortografía distinta) las cuales obligan a incluir una desambiguación del contexto para colocar la ortografía correcta.

Otra propuesta que surge de aplicar técnicas de Procesamiento de Lenguaje Natural puede ser el análisis de sentimiento de tal forma que se tenga la posibilidad de cambiar los parámetros de la traducción del texto a voz para que esta corresponda mejor con el contenido del mensaje y la forma en que se expresa.

Como última idea surge la evolución de la herramienta al adaptar dispositivos *wearables* comerciales como nodos sensores, esto requeriría desarrollar los componentes lógicos del muestreo en el nuevo dispositivo y en el lenguaje correspondientes, y más sin embargo fuera de esto la adaptación sería sumamente sencilla y traería numerosas ventajas frente al prototipo de la solución (mejor adaptación con los usuario, mejor interfaz de notificaciones, adición de nuevas funcionalidades, etc.).

REFERENCIAS

1. Sandín, B., & Belloch, A. (2008). Manual de Psicopatología, Vol. I. Edición revisada y actualizada (B. Sandín & A. Belloch, Eds.; Vol. 1º). McGraw-Hill Interamericana de España S.L.
2. National Institute on Deafness and Other Communication Disorders. (2022, July 8). Apraxia. NINDS. Consultado en Octubre 23, 2022, de <https://espanol.ninds.nih.gov/es/trastornos/apraxia>
3. American Speech-Language-Hearing Association. (n.d.). La Disartria. ASHA. Consultado en Julio 8, 2022, de <https://www.asha.org/public/speech/Spanish/La-Disartria>.
4. Huang, J. (2021, Diciembre). Disartria - Enfermedades cerebrales, medulares y nerviosas - Manual MSD versión para público general. MSD Manuals. Retrieved October 23, 2022, from <https://www.msmanuals.com/es/hogar/enfermedades-cerebrales,-medulares-y-nerviosas/disfunci-ón-cerebral/disartria>
5. Valle L. E., Romero R., Yaljá J., (2022, 26 al 30 de Septiembre). Clasificador de gestos motrices utilizando vectores de atributos distancia DTW sobre series de tiempo en representación SAX [Presentación de artículo]. CORE 2022, Ciudad de México, México.
6. Bolaños, N. P., Garcés, J. S. B., & Tavera, S. Y. G. (2017). Características neuropsicológicas y neurofisiológicas de los distintos tipos de Afasias. Revista chilena de neuropsicología, 12(2), 38-42.
7. E. D. Jiménez Carbajal, G. E. Rivera Taboada, "Sistema de comunicación auditiva para personas con problemas del habla", Tesis para título de licenciatura, ESCOM, Ciudad de México, México, 2013.
8. M. M. Chandra, S. Rajkumar and L. S. Kumar, "Sign Languages to Speech Conversion Prototype using the SVM Classifier,"TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), 2019, pp. 1803-1807, doi: 10.1109/TENCON.2019.8929356.
9. C.J.G.AyalaAburto,"Guante traductor de señas para sordomudos" Tesis título licenciatura, ESIME, unidad Azcapotzalco. Ciudad de México, México, 2018.
10. D. Vishal, H. M. Aishwarya, K. Nishkala, B. T. Royan and T. K. Ramesh, "Sign Language to Speech Conversion,"(en ingles) 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2017, pp. 1-4, doi: 10.1109/ICCIC.2017.8523832.

11. Minnar, A. (2014, April 16). Time Series Classification and Clustering with Python. Alex Minnaar. Retrieved Consultado en Octubre 23, 2022, de <http://alexminnaar.com/2014/04/16/Time-Series-Classification-and-Clustering-with-Python.html>
12. Mezari, Antigoni & Maglogiannis, Ilias. (2017). Gesture recognition using symbolic aggregate approximation and dynamic time warping on motion data. 342-347. 10.1145/3154862.3154927.
13. Kate, R.J. Using dynamic time warping distances as features for improved time series classification. Data Min Knowl Disc 30, 283–312 (2016). <https://doi.org/10.1007/s10618-015-0418-x>.
14. Hashimoto Y., Ushiba J., Kimura A., Liu M. & Tomita Y. (2010). Correlation between EEG-EMG coherence during isometric contraction and its imaginary execution. Acta Neurobiol. Exp. 70, 76–85.
15. Hardiyanti, N., Lawi, A., & Aziz, F. (2018, November). Classification of human activity based on sensor accelerometer and gyroscope using ensemble SVM method. In 2018 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT) (pp. 304-307). IEEE.
16. Wunderlich, K., & Abdelfattah, E. (2018, November). Human activity and posture classification using wearable accelerometer data. In 2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)(pp. 77-81). IEEE.
17. Eskofier, B. M., Musho, E., & Schlarb, H. (2013, May). Pattern classification of foot strike type using body worn accelerometers. In 2013 IEEE International Conference on Body Sensor Networks (pp. 1-4). IEEE.
18. Moccia, S., Solbiati, S., Khornegah, M., Bossi, F. F., & Caiani, E. G. (2022). Automated classification of hand gestures using a wristband and machine learning for possible application in pill intake monitoring. Computer Methods and Programs in Biomedicine, 219, 106753.
19. Hualde, J. I. (2013). Los sonidos del español: Spanish Language edition. Cambridge University Press.
20. M Müller. "Dynamic Time Warping," Information Retrieval for Music and Motion, 4. Alemania: Springer, 2007, pp. 69-73.
21. Sakoe, H., & Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, 26(1), 43–49. doi:10.1109/tassp.1978.1163055.

22. Krish, V. (2018, Febrero). Piecewise Aggregate Approximation. Vignesh Krishnamoorthy. Consultado en Mayo 10, 2022, de <https://vigne.sh/posts/piecewise-aggregate-approx/>
23. Krish, V. (2018, Junio). Symbolic Aggregate Approximation. Vignesh Krishnamoorthy. Consultado en Mayo 10, 2022, de <https://vigne.sh/posts/symbolic-aggregate-approximation/>
24. Petitjean, Francois & Ketterlin, Alain & Gancarski, Pierre. (2011). A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*. 44. 678-.10.1016/j.patcog.2010.09.013.
25. Lines, J., Bagnall, A. Time series classification with ensembles of elastic distance measures. *Data Min Knowl Disc* 29, 565–592 (2015). <https://doi.org/10.1007/s10618-014-0361-2>.
26. Kate, R.J. Using dynamic time warping distances as features for improved time series classification. *Data Min Knowl Disc* 30, 283–312 (2016). <https://doi.org/10.1007/s10618-015-0418-x>.
27. Seneviratne, P. (2018). Beginning BBC micro:bit: A Practical Introduction to micro:bit Development. Apress. <https://doi.org/10.1007/978-1-4842-3360-3>.
28. BBC micro:bit. (2017). Accelerometer — UCL BBC micro:bit tutorial. UCL BBC micro:bit tutorial. Recuperado Octubre 23, 2022, de <https://microbit-challenges.readthedocs.io/en/latest/tutorials/accelerometer.html>
29. ESP-01 WiFi Module: Version 1.0 (De Shenzhen Anxinke Technology CO;LTD). (2015). AI-Thinker team. Recuperado 23 de Octubre de 2022, de <https://pdf1.alldatasheet.com/datasheet-pdf/view/1179098/ETC2/ESP-01.html>
30. Mitchell, B. (2021, September 10). What Is a Wireless Access Point? Lifewire. Consultado en October 23, 2022, de <https://www.lifewire.com/wireless-access-point-81654>.
31. Utmel Electronic. (2021, October 18). TP4056 Standalone Linear Li-Lon Battery Charger: Datasheet, Schematics and Current. Utmel. Constulado en October 24, 2022, de <https://www.utmel.com/components/tp4056-standalone-linear-li-lon-battery-charger-datasheet-schematics-and-current?id=689>.
32. Micro:bit. (2021, July 19). Touch sensing on the micro:bit : Help & Support. Micro:bit support. Consultado en Octubre 24, 2022, de <https://support.microbit.org/support/solutions/articles/19000116318-touch-sensing-on-the-micro-bit>.

33. Wang, A. (2019, March 30). WiFi Web Server on BBC micro:bit and ESP-01 (ESP8266). Hackster.io. Consultado Octubre 24, 2022, from <https://www.hackster.io/alankrantas/wifi-web-server-on-bbc-micro-bit-and-esp-01-esp8266-498e0d>.
34. Microcontrolles Lab. (2016). ESP8266 AT Commands Explained with Examples. Microcontrollers Lab. Consultado en Octubre 24, 2022, de <https://microcontrollerslab.com/esp8266-at-commands/>
35. MDN Mozilla. (2022, October 11). HTTP Messages - HTTP | MDN. MDN Web Docs. Consultado en Octubre 24, 2022, de <https://developer.mozilla.org/en-US/docs/Web/HTTP/Messages>
36. Jin, B., Shevat, A., & Sahni, S. (2018). Designing Web APIs: Building APIs That Developers Love. O'Reilly Media.
37. Fundación Mozilla. (2021, Febrero 12). mozilla/TTS: Deep learning for Text to Speech (Discussion forum: <https://discourse.mozilla.org/c/tts>). GitHub. Consultado en Noviembre 10, 2022, de <https://github.com/mozilla/TTS>.
38. Google Cloud. (n.d.). Text-to-Speech: Lifelike Speech Synthesis. Google Cloud. Consultado en Noviembre 10, 2022, de <https://cloud.google.com/text-to-speech/>.
39. Microsoft Azure. (n.d.). Text to Speech – Realistic AI Voice Generator. Microsoft Azure. Consultado en Noviembre 10, 2022, de <https://azure.microsoft.com/en-us/products/cognitive-services/text-to-speech/#features>.
40. AWS. (n.d.). Text to Speech Software – Amazon Polly – Amazon Web Services. Amazon Polly. Consultado en Noviembre 10, 2022, de <https://aws.amazon.com/polly/>.
41. Dau, H.A., Silva, D.F., Petitjean, F. et al. Optimizing dynamic time warping's window width for time series data mining applications. Data Min Knowl Disc 32, 1074–1120 (2018). <https://doi.org/10.1007/s10618-018-0565-y>.

GLOSARIO DE TÉRMINOS

DTW	Siglas en inglés de <i>Dynamic Time Warping</i> (Deformación Temporal Dinámica), y que hace referencia a un algoritmo extensamente utilizado para encontrar la alineación óptima entre 2 secuencias temporales.
IMU	Siglas en inglés para <i>Inertial Measurement Unit</i> (Unidad de Medida Inercial), y que hace referencia a un dispositivo electrónico que mide y reporta la fuerza, velocidad angular e inclusive la orientación de un cuerpo específico utilizando una combinación de acelerómetros, giroscopios y algunas veces magnetómetros.
K-NN	Siglas en inglés de <i>K-Nearest Neighbors</i> (K Vecinos cercanos), que nombra a un modelo estadístico utilizado en el área de inteligencia artificial como el modelo más simple para la tarea de clasificación.
LSM	Siglas para el Lenguaje de Señas Mexicano; Se trata de un lenguaje señado utilizado por personas con discapacidad auditiva en México.
PAA	Siglas en inglés de <i>Piecewise Aggregate Approximation</i> , nombra a un algoritmo que se enfoca en la reducción dimensional de una secuencia temporal.
SAX	Siglas en inglés de <i>Symbolic Aggregate Approximation</i> , hace referencia a un algoritmo utilizado para reducir dimensionalmente y discretizar una serie de tiempo a un número de palabras definido dentro de un rango de símbolos alfabéticos.
SSML	Siglas en inglés para <i>Speech Synthesis Markup Language</i> , y que hace referencia a un lenguaje de marcado utilizado para especificar cómo se realizará la sintetización de un texto de entrada a discurso utilizando un servicios de texto a voz.
SVM	Siglas en inglés para <i>Support Vector Machine</i> (Máquina de Soporte Vectorial), el cual es modelo estadístico categorizado como modelo de aprendizaje supervisado en el área de inteligencia artificial, y es utilizado principalmente para la asignación de una etiqueta de clase a un vector de entrada.
AFI	Siglas para el Alfabeto Fonético Intencionalidad, el cual presenta un marco estandarizado para los sonidos de los lenguajes a nivel global y el cual asigna un símbolo específico a cada sonido.
IoT	Siglas en inglés de <i>Internet-of-Things</i> (Internet de las cosas), y que hace referencia a los sistemas de dispositivos electrónicos interconectados que son capaces de recopilar y transferir datos a través de una red sin la intervención humana.

APÉNDICE A

Características del micro:bit

El dispositivo micro:bit es programado a través de un ambiente de edición basado en web que provee varios editores para la creación de programas. El editor soporta Typescript, Micro Python y *Blocks* (un lenguaje de programación visual). Una vez escrito el programa, se envía a un servidor para la compilación y después de la descarga, el archivo *hex* puede ser arrastrado a la memoria del micro:bit.

Esta placa incluye algunas de las características relevantes listadas a continuación[27]:

- Microprocesador ARM de 32 bits Cortex M0 CPU
- Matriz LED de 5x5 color rojo: Cada LED puede ser programado individualmente y permite mostrar patrones animados, texto deslizando y caracteres alfanuméricos.
- 2 botones programables: El micro:bit es capaz de detectar si los botones se encuentran presionados o no, lo que posibilita su programación para ejecutar acciones, enviar información a otro dispositivo, etc.
- Bluetooth *Smart Technology*: Anteriormente llamado Bluetooth de baja energía permite la comunicación inalámbrica con otros dispositivos Bluetooth para intercambiar datos y comandos.
- Entradas y Salidas(I/O) en forma de anillo: Cada I/O es programable ya sea análogamente o digitalmente, de forma que sensores externos pueden conectarse para realizar obtener mediciones, actuar de acuerdo a estas o compartirlo con otro dispositivo. Así mismo a estas entradas y salidas en forma de anillo se les pueden enviar comandos para el control, por ejemplo, de un robot o un motor.
- Controlador micro usb: Controlado por un procesador externo, expone el micro:bit a otras computadoras como una tarjeta de memoria, de manera que archivos pueden ser arrastrados a este para ser almacenados en la memoria del micro:bit.

Características del módulo Wi-Fi ESP-01

A continuación se enlistan características importantes del módulo Wi-Fi[29]:

- Estándar 802.11 soportado
- Pila de protocolo TCP/IP integrado
- Integrado ADC de 10 bits
- Wi-Fi de 2.4 GHz, soporta WPA/WPA2
- Tensión de alimentación de 3.3 V
- Comunicación tipo interfaz Serial y UART
- Protocolos de red: IPv4, TCP, UDP y HTTP

Características del módulo TP4056

Por defecto, la corriente de carga que permite es de hasta 1A, aunque puede ajustarse añadiendo un resistor externo. El voltaje de carga máximo es de 4.2V, aunque permite un voltaje de entrada desde 4.5V hasta 5.5V.

La tarjeta cuenta con 6 entradas descritas de la siguiente manera[31]:

- *OUT +*: Corresponde a la salida positiva de voltaje de la batería. Esta terminal se conecta al circuito que está destinado a ser energizado por la batería.
- *B+*: Salida de voltaje positiva de la que el cable USB, o las entradas, cargará la batería. Conectado al positivo de la batería.
- *B-*: Salida de voltaje negativa de la que el cable USB, o las entradas, cargará la batería. Conectado al negativo de la batería.
- *OUT -*: Corresponde a la salida de voltaje negativo de la batería. Esta terminal se conecta a la tierra del circuito que está destinado a ser energizado por la batería.
- *IN+*: Debe ser provisto de +5V si no se cuenta con cable micro USB.
- *IN-*: Debe ser provisto a la tierra si no se cuenta con cable micro USB.

Finalmente en la placa se agregan 2 led: El primero de color azul indica cuando no se encuentra conectada una pila o esta se encuentra completamente cargada. Mientras que el segundo led de color rojo prende cuando la batería se encuentra en su ciclo de recarga.