

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE OCCIDENTE

Departamento de Procesos Tecnológicos e Industriales

Sustentabilidad y tecnología

**PROYECTO DE APLICACIÓN PROFESIONAL (PAP)
DESARROLLO TECNOLÓGICO PARA LA SUSTENTABILIDAD AMBIENTAL,
ENERGÉTICA Y ALIMENTARIA**



**ITESO, Universidad
Jesuita de Guadalajara**

**4D08 - DESARROLLO TECNOLÓGICO PARA LA SUSTENTABILIDAD
AMBIENTAL, ENERGÉTICA Y ALIMENTARIA I**

Sistema de adquisición de datos biomecánicos

PRESENTAN

Programas educativos y Estudiantes

Ing. en Sistemas Computacionales Lilia Arceli Lobato Martínez

Ing. en Sistemas Computacionales Alonso De Aguinaga Muro

Profesor PAP: M.I.M Fermín Castro Aragón

Tlaquepaque, Jalisco, mayo de 2022

Contenido

REPORTE PAP	2
Presentación Institucional de los Proyectos de Aplicación Profesional	2
Resumen	0
1. Ciclo participativo del Proyecto de Aplicación Profesional.....	1
1.1 Entendimiento del ámbito y del contexto	2
1.2 Caracterización de la organización.....	7
1.3 Identificación de la(s) problemática(s).....	8
1.4. Planeación de alternativa(s).....	9
1.5. Desarrollo de la propuesta de mejora	18
1.6. Valoración de productos, resultados e impactos	52
1.7. Bibliografía y otros recursos	68
1.8. Anexos generales.....	69
2. Productos	70
3. Reflexión crítica y ética de la experiencia.....	73
3.1 Sensibilización ante las realidades	73
3.2 Aprendizajes logrados	74
3.3 Inventario de competencias Inicial e Inventario de competencias Final	75

REPORTE PAP

Presentación Institucional de los Proyectos de Aplicación Profesional

Los Proyectos de Aplicación Profesional (PAP) son experiencias socio-profesionales de los alumnos que desde el currículo de su formación universitaria- enfrentan retos, resuelven problemas o innovan una necesidad sociotécnica del entorno, en vinculación (colaboración) (co-participación) con grupos, instituciones, organizaciones o comunidades, en escenarios reales donde comparten saberes.

El PAP, como espacio curricular de formación vinculada, ha logrado integrar el Servicio Social (acorde con las Orientaciones Fundamentales del ITESO), los requisitos de dar cuenta de los saberes y del saber aplicar los mismos al culminar la formación profesional (Opción Terminal), mediante la realización de proyectos profesionales de cara a las necesidades y retos del entorno (Aplicación Profesional).

El PAP es un proceso acotado en el tiempo en que los estudiantes, los beneficiarios externos y los profesores se asocian colaborativamente y en red, en un proyecto, e incursionan en un mundo social, como actores que enfrentan verdaderos problemas y desafíos traducibles en demandas pertinentes y socialmente relevantes. Frente a éstas transfieren experiencia de sus saberes profesionales y demuestran que saben hacer, innovar, co-crear o transformar en distintos campos sociales.

El PAP trata de sembrar en los estudiantes una disposición permanente de encargarse de la realidad con una actitud comprometida y ética frente a las disimetrías sociales. En otras palabras, se trata del reto de “saber y aprender a transformar”.

El Reporte PAP consta de tres componentes:

El primer componente refiere al ciclo participativo del PAP, en donde se documentan las diferentes fases del proyecto y las actividades que tuvieron lugar durante el desarrollo de este y la valoración de las incidencias en el entorno.

El segundo componente presenta los productos elaborados de acuerdo con su tipología.

El tercer componente es la reflexión crítica y ética de la experiencia, el reconocimiento de las competencias y los aprendizajes profesionales que el estudiante desarrolló en el transcurso de su labor.

Resumen

El presente PAP es la primera entrega del proyecto iniciado en el semestre de primavera 2022 con el que se aborda una nueva línea de investigación referente a sistemas de adquisición de datos biométricos. El objetivo diseñar un sistema biomecánico para la adquisición de datos de movimiento de rodilla. Para el logro de este propósito, se plantearon cinco objetivos específicos: Realizar una investigación de la literatura del censado de datos de movimiento en la articulación., diseñar el sistema de hardware enfocado en la instrumentación de los sensores, realizar un sistema de captura de información de los sensores de movimiento, establecer un sistema de ecuaciones para calcular los ángulos y la aceleración del sistema y, finalmente, realizar una simulación de los ángulos de movimiento de la rodilla.

Para lo anterior se estudiaron los diferentes sistemas de captura de movimiento y se caracterizó el sistema inercial como base para crear un sistema para la captura de movimientos. Asimismo, se estudiaron las diferentes alternativas para la implementación de cada uno de los componentes y se documentó su interacción. Se realizó un estudio de costo, precisión y facilidad de uso para definir los mejores componentes y arquitectura para obtener datos de sensores, procesarlos y desplegarlos.

Se logró crear un prototipo funcional para la captura de datos de movimiento usando 2 sensores MPU6050 y una tarjeta de desarrollo ESP8260 con conexión al servicio AWS IoT para guardar los datos. Además, se generó un tablero que muestra los datos medidos en tiempo real y una simulación.

1. Ciclo participativo del Proyecto de Aplicación Profesional

El PAP es una experiencia de aprendizaje y de contribución social integrada por estudiantes, profesores, actores sociales y responsables de las organizaciones, que de manera colaborativa construyen sus conocimientos para dar respuestas a problemáticas de un contexto específico y en un tiempo delimitado. Por tanto, la experiencia PAP supone un proceso en lógica de proyecto, así como de un estilo de trabajo participativo y recíproco entre los involucrados.

Se realizará un sistema de adquisición de datos biomecánicos. Esto para poder apoyar el desarrollo de los exoesqueletos de rehabilitación, permitiendo hacer análisis de datos y simulaciones del funcionamiento, esto con el fin de mejorar el funcionamiento de estos.

Se cuenta con la colaboración de los departamentos del DESI y DPTI del ITESO. Por parte del DESI, contamos con el apoyo de las instalaciones de laboratorio de electrónica y, por parte del DPTI, con la tutoría de Fermín Castro Aragón como asesor de proyecto.

Objetivo general

Diseñar un sistema biomecánico para la adquisición de datos de movimiento de rodilla.

Objetivos específicos

- Realizar una investigación de la literatura del censado de datos de movimiento en la articulación.
- Diseñar el sistema de hardware enfocado en la instrumentación de los sensores.
- Realizar un sistema de captura de información de los sensores de movimiento. (base de datos)
- Establecer un sistema de ecuaciones para calcular los ángulos y la aceleración del sistema.
- Realizar una simulación de los ángulos de movimiento de la rodilla.

1.1 Entendimiento del ámbito y del contexto

En la última década se ha desarrollado una rama en la robótica especialmente diseñada para la rehabilitación de pacientes con enfermedades degenerativas o personas que pertenecen al grupo vulnerable de la 3ra edad [1]. Esta área de estudio usa plataformas robóticas en conjunto con planes de ejercicios de rehabilitación de la mano con un fisioterapeuta.

Los planes de rehabilitación consistente en la repetición constante de uno o varios movimientos centrándose en obtener mayor fuerza a un grupo de músculo, desarrollar movilidad y evitar la distrofia del paciente. En las primeras fases de la terapia, estos movimientos suelen necesitar de un fisioterapeuta que fuerce el movimiento o asista en él. Esto representa un problema ya que el fisioterapeuta debe de realizar grandes esfuerzos que dependen del peso del paciente y de su nivel de movilidad para lograr completar la terapia. Se pueden dar casos donde sea necesario contar con más de un fisioterapeuta o donde las sesiones deban de acortarse en tiempo, pero aumentar en periodicidad.

Para un paciente, ver que su fisioterapeuta está en constante sobre esfuerzo, la necesidad de un mayor número de personal para lograr los ejercicios o tener que cumplir con un mayor número de sesiones, pueden derivar en un problema de desmotivación en el paciente y, consecuentemente, abandono del proceso [2].

Las plataformas robóticas pueden ayudar a mitigar estos problemas. La plataforma se encarga de forzar los movimientos en el paciente, de esta manera, el fisioterapeuta puede llevar a cabo sus sesiones sin necesidad de apoyo humano adicional ni de aumento de frecuencia [3].

Para lograr este objetivo, se deben de crear sistema que permitan captar los movimientos naturales de una articulación humana y, basados en estos, diseñar las trayectorias deseadas para la plataforma de rehabilitación. Esta técnica es sumamente utilizada ya que el rendimiento de los algoritmos utilizados de forma natural por los humanos supera a los presentes en los robots a pesar de la gran destreza que los algoritmos de control artificial pueden proporcionar.

Captación de movimientos

Como se muestra en la figura 1, existen varias técnicas para capturar el movimiento humano: sistemas ópticos, inerciales, electromagnéticos y electromecánicos. Cada uno de estos sistemas es capaz de registrar características del cuerpo humano y reconstruir posturas corporales en movimiento (posiciones y ángulos de articulaciones), difieren en su método de captura, materiales, costos y aplicaciones.



Figura 1- Clasificación de sistemas de captura de movimiento [4].

Los sistemas de captura de movimiento electromecánicos usan sensores mecánicos, el proceso consiste en portar un traje especial adaptable al cuerpo del paciente. Estos trajes son generalmente estructuras rígidas compuestas de barras metálicas o plásticas, unidas mediante potenciómetros colocados en las principales articulaciones [4].

Los sistemas de captura electromagnéticos disponen de una colección de sensores que miden la relación espacial con un transmisor cercano. Los sensores se colocan en el cuerpo y se conectan a una unidad electrónica central; están constituidos por tres espiras ortogonales que miden el flujo magnético, determinando posición y orientación del sensor [4].

Los sistemas de captura de movimiento inerciales, se colocan sensores inerciales en distintas partes del cuerpo (acelerómetros triaxiales y giroscopios).

Los sistemas ópticos utilizan los datos recogidos por sensores de imagen para inferir la posición de un elemento en el espacio, utilizando una o más cámaras sincronizadas para proporcionar proyecciones simultáneas [4].

Al tratarse de una aplicación biomédica, se busca un sistema con un alto grado de precisión y baja invasión del usuario. Debe permitir un constante análisis, bajo una diversa gama de escenarios y a un costo accesible. Basado en estos requerimientos, se pretende hacer una comparativa en las siguientes 5 categorías:

- Precisión: Exactitud de los datos obtenidos.
- Movilidad: Facilidad de transporte del sistema, tanto a la reubicación total del sistema como a recalibración.
- Precio: Coste monetario del sistema
- Robustez: Sensibilidad de las medidas.
- Integración: Facilidad con la que se pueden pasar los datos recolectados a una plataforma robótica o una plataforma de simulación.

La tabla 1 muestra la escala Likert, donde 1 es ‘Nada satisfactorio’ y 5 es ‘Extremadamente Satisfactorio’ y toman datos del artículo “Sistemas de captura y análisis de movimiento cinemático humano: una revisión sistemática” [5] para los sistemas electromagnéticos, ópticos e inerciales y datos del artículo “Sistemas de Captura de Movimiento - EXOCAP” para los sistemas electromecánicos, electromagnéticos y ópticos [6].

Tabla 1 - Comparativa entre los sistemas de captura de movimiento

	Precisión	Movilidad	Precio	Robustez	Integración
Sistema electromecánico	5	1	4	5	3
Sistema electromagnético	3	3	2	3	3
Sistema óptico	5	1	4	5	5
Sistema inercial	3	4	2	4	5

Estos requerimientos nos dejan con 2 alternativas viables: sistema óptico y el sistema inercial.

Sistema de captura de movimiento óptico

El funcionamiento de estos sistemas se basa en el reconocimiento de puntos específicos del cuerpo humano por medio de varias cámaras en diferentes posiciones. Cada una de las cámaras envía una lectura de las coordenadas del punto, combinándolas, se puede triangular la posición de cada punto y mapear en un objeto 3D.

El número de cámaras es un factor sumamente importante, mientras más cámaras se tengan, mayor será la calidad y resolución del modelo capturado, esto también repercute en el coste económico y computacional por lo que se debe de elegir cuidadosamente. Para aplicaciones de captura de movimiento humano, el mínimo se suele situar entre 6-8 cámaras [4].

Dentro de los sistemas ópticos se pueden clasificar en dos categorías, los sistemas con marcadores y sin marcadores.

Sistema de captura de movimiento óptico con marcadores

Los sistemas basados en marcadores usan puntos físicos que dan contraste de luminosidad a comparación del entorno en el que se encuentran. Estos marcadores pueden ser pasivos, como en la figura 2, o activos.



Figura 2 - Sistema de captura óptico con marcadores pasivos [7].

Los marcadores pasivos consisten en objetos reflejantes adheridos a la superficie del cuerpo a medir. Se emite una luz infrarroja y se recoge el reflejo de los marcadores con las cámaras [8].

Los marcadores activos emiten algún tipo de señal por lo que requieren un equipo de telemetría o cámara especial para capturar estas señales [8].

Sistema de captura de movimiento óptico sin marcadores

Estos sistemas no usan marcadores, recogen las imágenes directamente y las procesan para crear una simulación del cuerpo humano. Hacen uso de algoritmos de reconocimiento, partiendo el cuerpo en partes y analizando así su movimiento [9].

Sistema de captura de movimiento inercial

Los sistemas inerciales IMU (unidades de medición inerciales) se basan en el uso de acelerómetros, giroscopios y magnetómetros para medir la aceleración lineal, velocidad angular y campos magnéticos.

Estos sistemas son comparativamente más económicos y flexibles ya que los sensores son de menor precio, de tamaño pequeño, fáciles de transportar y no se limitan a un espacio de trabajo designado [5].

1.2 Caracterización de la organización

Este PAP se desarrolla dentro de ITESO (Instituto Tecnológico y de Estudios Superiores de Occidente) localizado en la Zona Metropolitana de Guadalajara, Jalisco.

Se cuenta con la colaboración de los departamentos del DESI (Departamento de Electrónica, Sistemas e Informática) .

Por parte del DESI, contamos con el apoyo de las instalaciones de laboratorio de electrónica y, por parte del IDPTI, con la tutoría de Fermín Castro Aragón como asesor de proyecto.

Participan como autores:

- Lilia Arceli Lobato Martínez tomando el rol de investigador principal, líder del área de electrónica tomando como responsabilidad la instrumentación, filtrado de señales y prototipado del sistema y líder del área de sistemas computacionales tomando responsabilidad sobre las bases de datos, métodos de transmisión de datos y simulaciones con los datos obtenidos.
- Alonso De Aguinaga Muro apoyando en el área de sistemas computacionales enfocada en generar simulaciones con los datos obtenidos.

1.3 Identificación de la(s) problemática(s)

Cerca del 15% de la población mundial presenta alguna enfermedad crónico-degenerativa que deriva en algún problema de movilidad disminuida, ya sea por accidentes, enfermedad, discapacidad o lesión [10]. Estas situaciones pueden causar dificultades para realizar tareas cotidianas, y en casos extremos, causar que sea imposible hacerlas.

Las maneras en las que se busca solucionar o disminuir el impacto que tienen estas situaciones es por medio del uso de prótesis y rehabilitación física, así como el uso de herramientas externas como sillas o bastones. Las terapias de rehabilitación tienden a ser el método más utilizado por su amplia gama de usos, flexibilidad y accesibilidad, sin embargo, suelen ser un proceso largo, tedioso y cansado tanto para el paciente como para el fisioterapeuta.

En el proceso de rehabilitación, los especialistas utilizan una amplia variedad de ejercicios, apoyándose de tecnologías como las pulsaciones eléctricas, vibraciones, monitoreo de señales nerviosas o prótesis exoesqueléticas especializadas. Estas últimas son utilizadas como plataformas que asisten en los ejercicios de rehabilitación física ya que permiten que el paciente pueda realizar rutinas específicas, aligeran el peso del paciente y dan soporte adicional.

A pesar de ser un método que cada día se acerca más a la movilidad y sensación natural del cuerpo humano, tiene la tendencia de generalizar los movimientos y rutinas careciendo de la personalización que un fisioterapeuta le da a un paciente.

Es de suma importancia contar con sistemas que permitan monitorear en tiempo real las trayectorias de movimiento, su velocidad, aceleración y fuerza. Estos datos pueden ayudar a resolver algunos de los problemas de la terapia por exoesqueletos. Mediante la captación de ciertos movimientos se puede lograr una mayor granularidad en su observación por parte del fisioterapeuta, lo que lleva a tener mejores resultados en su terapia ya que pueden mostrar error en la técnica de los ejercicios, detectar patrones de intensidad o zonas de sobreesfuerzo.

1.4. Planeación de alternativa(s)

Antes de poder comenzar con la planeación del proyecto es importante elegir el método de captura de movimiento óptimo y apegado a nuestros objetivos. En secciones anteriores limitamos los sistemas a: ópticos e inerciales.

Aunque los sistemas basados en la visión pueden medir con fiabilidad y mayor precisión los movimientos humanos si se aplican suficientes marcadores y son observados por las cámaras distribuidas, estos sistemas sufren de oclusión, requieren una configuración de laboratorio fija o dedicada, un proceso de calibración tedioso y un cálculo intensivo, entre otras dificultades. Por esto, los sistemas basados en sensores inerciales son una mejor opción para este proyecto ya que producen mediciones de precisión aceptables de las aceleraciones y otras variables en función de la instrumentación, son ligeros y están disponibles para la comunicación inalámbrica [11].

Un inconveniente importante de los sensores inerciales es la presencia del error acumulativo (*drifting*), sin embargo, se van a proponer y posteriormente implementar un sistema de filtros para tratar dicho problema.

Con esto en mente y usando la tabla 2, se decidió utilizar un sistema basado en sensores inerciales.

Tabla 2- Características de los sistemas de captura de movimiento [5]

Tecnología	Fortalezas	Debilidades
Sistemas ópticos con marcadores (<i>Vicon</i>)	<ul style="list-style-type: none"> • Alta precisión • Procesamiento de datos • Uso en animación 3D • Uso en realidad virtual y aumentada • Abarca gran cantidad de aplicaciones • Uso en estudios de alta complejidad 	<ul style="list-style-type: none"> • Alto tiempo de adecuación del espacio y el sistema • Alto costo del equipo • Oclusiones de marcadores • Poco usado en estudios deportivos • Poca robustez • Requiere de un espacio delimitado
Sistemas ópticos sin marcadores (<i>Kinect</i>)	<ul style="list-style-type: none"> • Bajo costo • Fácil manipulación • Procesamiento de datos • Alta precisión en parámetros espaciotemporales • Uso en tele rehabilitación 	<ul style="list-style-type: none"> • Baja precisión en parámetros angulares • Poco uso en estudios de alta complejidad
IMU y sistemas inerciales	<ul style="list-style-type: none"> • Alto uso en estudios deportivos • Alta portabilidad • Fácil manipulación • Apto para realizar estudios fuera de laboratorios • Bajo costo del equipo 	<ul style="list-style-type: none"> • Baja precisión • Poco uso en estudios de alta complejidad

Se proponen dos arquitecturas, la primera para el proceso de desarrollo y la segunda como prototipo final.

En la figura 3 se muestra la arquitectura a alto detalle para el desarrollo. Se hace uso de un microcontrolador ATmega328 ya que permite ver datos de forma sencilla por medio de una interfaz serial en tiempo real y generar gráficas, se puede programar de forma sumamente rápida usando la IDE de Arduino por lo que es posible trabajar de manera modular sin preocupación por el número de periféricos o consumo de energía.

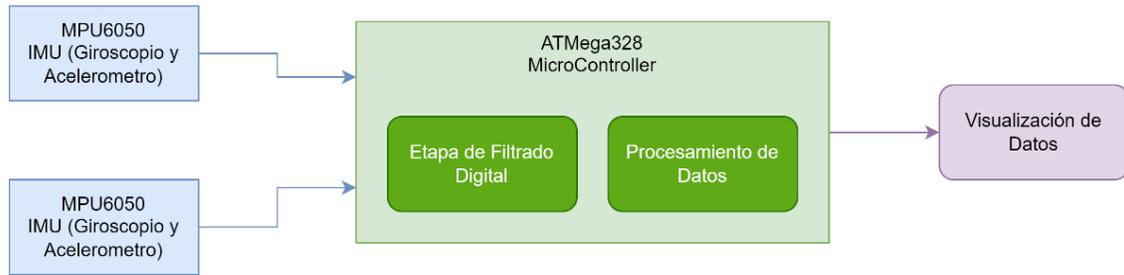


Figura 3 - Arquitectura de desarrollo a alto nivel

En la figura 4 se muestra la arquitectura a alto detalle propuesta para el prototipo final. Se hace uso de un ESP8260 como microcontrolador; con este componente se reduce significativamente el tamaño del prototipo, una característica deseable en dispositivos que van a estar directamente sobre el cuerpo humano. La arquitectura se utilizará únicamente para el prototipo final ya que no permite ver los datos adquiridos por los sensores de forma sencilla; se tiene que mandar la información a un dominio y cuentan con retraso significativo. El proceso de programación es considerablemente más complejo ya que no cuenta con IDE nativa de Arduino y se requiere crear una circuitería adicional.

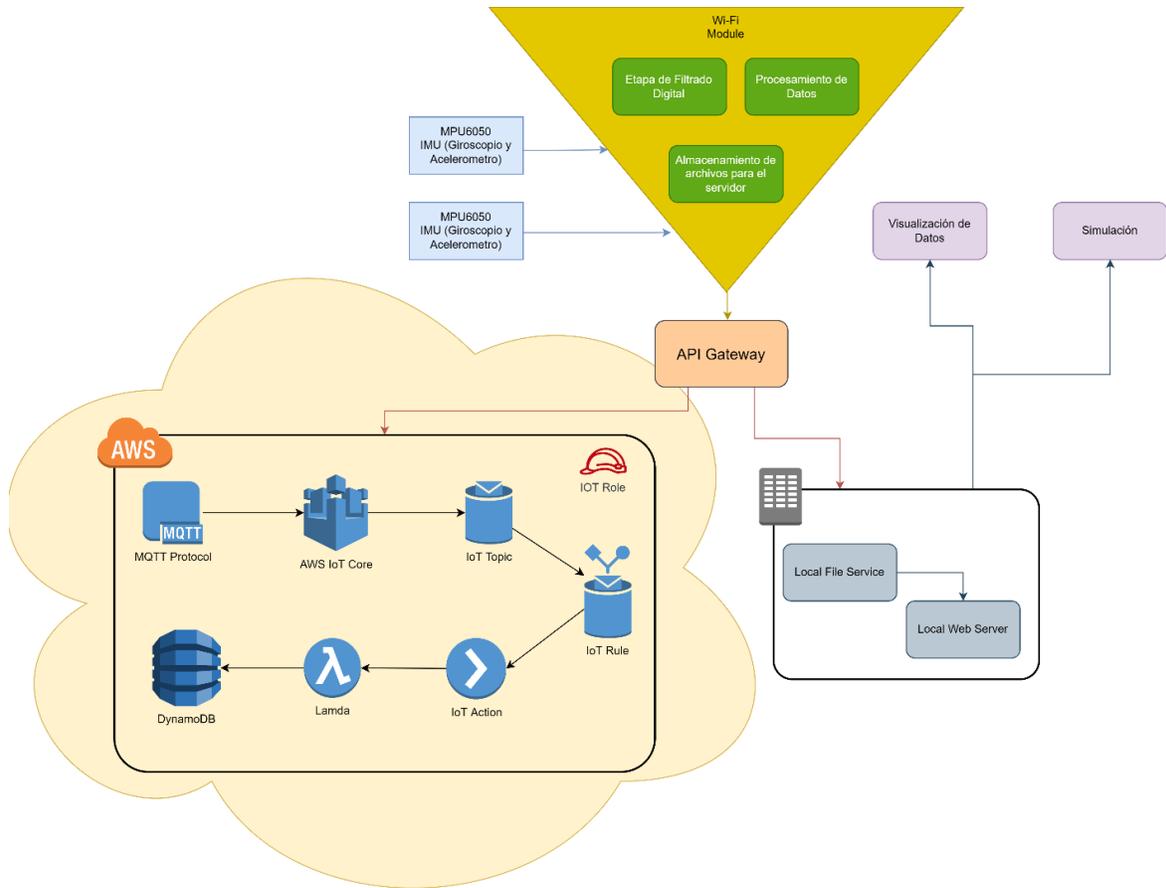


Figura 4- Arquitectura a alto nivel para el prototipo final

Cada uno de los bloques de la figura 4 se eligió por las siguientes razones:

Microcontrolador ATmega328

Se utilizará el microcontrolador ATmega328 para rápido prototipado ya que es el que utiliza un Arduino UNO. Cuenta con entradas analógicas, digitales y puertos dedicados para la comunicación serial necesaria para los módulos Wi-Fi. Las características del microcontrolador se muestran en la tabla 3.

Tendremos etapas de filtrado digital por lo que es necesario se tomó en cuenta la capacidad de memoria de 16K bits, la memoria Flash y memoria dinámica.

Tabla 3 - Características ATmega328

Características	Valores
Rango de temperatura	-40°C a 85°C
Voltaje de operación	1.8 – 5.5V
Precisión	8 bits

Microcontrolador ESP8266

Dado que estamos limitados por el espacio, un ESP8266 es la opción más compacta para realizar la comunicación a una base de datos. Este componente será nuestro Gateway para la base de datos. Contiene 2 puertos GPIO y se comunica por UART (Universal Asynchronous Receiver-Transmitter). Las características del microcontrolador ESP8266 son mostradas en la tabla 4.

Tabla 4 - Características ESP8266

Características	Valores
Rango de temperatura	-40°C a 85°C
Voltaje de operación	3.3V
Precisión	8 bits

Sensores inerciales

Se realizó una comparativa entre dos sensores inerciales comerciales, la BMI160, y la MPU 6050, mostrada en la tabla 5. Tras analizar ambos sensores decidimos utilizar la MPU6050 por su facilidad de uso, precio y librerías existentes con la IDE de Arduino.

Tabla 5 - Tabla comparativa entre las IMU comerciales BMI160 y MPU6050

IMU	BMI160	MPU6050
Error Giroscopio	Error $\pm 0.02\%/K$	Error $\pm 3\%$ at $25^{\circ}C$
Error Temperatura	Error $\pm 2^{\circ} K$	Error $\pm 1^{\circ}C$
Error Acelerómetro	Error $\pm 0.03\%/K$	Error $\pm 3\%$ at $25^{\circ}C$
Exactitud Giroscopio	$0.01\%/V$	Exactitud $\pm 20^{\circ}/seg$ at $-40^{\circ}C$ to $+85^{\circ}C$
Exactitud Temperatura	Exactitud $\pm 2 K$	Exactitud $\pm 1^{\circ}C$
Exactitud Acelerómetro	$0.01\%/V$	Exactitud $\pm 0.02\%/^{\circ}C$ at $-40^{\circ}C$ to $+85^{\circ}C$
Precisión Giroscopio	$0.01\%/V$	Precisión $\pm 20^{\circ}/seg$ at $-40^{\circ}C$ to $+85^{\circ}C$
Precisión Temperatura	Precisión $\pm 2 K$	Precisión $\pm 1^{\circ}C$
Precisión Acelerómetro	$0.01\%/V$	Precisión $\pm 0.02\%/^{\circ}C$ at $-40^{\circ}C$ to $+85^{\circ}C$
Resolución Giroscopio	16.4 LSB/($^{\circ}/s$), 32.8 LSB/($^{\circ}/s$), 65.6 LSB/($^{\circ}/s$), 131.2 LSB/($^{\circ}/s$), 262.4 LSB/($^{\circ}/s$)	131, 65.5, 32.8, and 16.4 LSB/($^{\circ}/s$) factor de escalamiento de sensibilidad
Resolución Temperatura	$0.002 K/LSB$	$340 LSB/^{\circ}C$
Resolución Acelerómetro	17039 LSB/g, 8520 LSB/g, 4260 LSB/g, 2130 LSB/g	16,384 LSB/g, 8,192 LSB/g, 4,096 LSB/g, and 2,048 LSB/g
Sensibilidad	Factor de escalamiento de sensibilidad pre-calibrado	Factor de escalamiento de sensibilidad pre-calibrado
Cifras Significativas	16 bits	16 bits
Rango de Medición Giroscopio	Escalas de rango: $125^{\circ}/s$, $250^{\circ}/s$, $500^{\circ}/s$, $1000^{\circ}/s$, and $2000^{\circ}/s$	Escalas de rango: ± 250 , ± 500 , ± 1000 , and $\pm 2000^{\circ}/sec$
Rango de Medición Temperatura	$-40^{\circ}C$ to $+85^{\circ}C$	$-40^{\circ}C$ to $+85^{\circ}C$
Rango de Medición Acelerómetro	Escalas de rango: $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$	Escalas de rango: $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$
Escala Giroscopio	$^{\circ}/seg$	$^{\circ}/seg$
Escala Temperatura	Kelvin	Celsius
Escala Acelerómetro	g ($9.81m/s$)	g ($9.81m/s$)
Linealidad	Mayor linealidad a $25^{\circ}C$	Mayor linealidad a $25^{\circ}C$
Zona muerta	No hay zonas muertas en el rango	No hay zonas muertas en el rango
Voltaje de alimentación	1.2V to 3.6 V	2.375V-3.46V
Consumo de corriente	990uA	3.9mA
Precio	\$35 mxn	\$80 mxn
Tipo de Sensor	Digital, I2C y SPI	Digital, I2C
Datasheet	https://www.mouser.com/ds/2/783/BST-BMI160-DS000-07-786474.pdf	https://www.invensense.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf

Siguiendo el mapa de registros y descripciones de MPU-6000/MPU-6050 [12] los datos requeridos para el proyecto se encuentran en los siguientes bancos de registros mostrados en las figuras 5 y 6:

Acelerómetro

Type: Read Only

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
3B	59	ACCEL_XOUT[15:8]							
3C	60	ACCEL_XOUT[7:0]							
3D	61	ACCEL_YOUT[15:8]							
3E	62	ACCEL_YOUT[7:0]							
3F	63	ACCEL_ZOUT[15:8]							
40	64	ACCEL_ZOUT[7:0]							

Figura 5 - Mapa de registros MPU 6050, sección Acelerómetro [12]

Giroscopio

Type: Read Only

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
43	67	GYRO_XOUT[15:8]							
44	68	GYRO_XOUT[7:0]							
45	69	GYRO_YOUT[15:8]							
46	70	GYRO_YOUT[7:0]							
47	71	GYRO_ZOUT[15:8]							
48	72	GYRO_ZOUT[7:0]							

Figura 6 - Mapa de registros MPU 6050, sección Acelerómetro [12]

Posición de los sensores sobre el cuerpo

Para la adquisición de datos de movimiento de rodilla se plantea colocar 2 sensores, por la parte superior de la rodilla derecha y por la parte inferior de la rodilla izquierda, como se muestra en la figura 7. Este método nos permite calcular posición y velocidad en cada nodo y el ángulo relativo a la posición inicial; adicionalmente, al tener dos puntos de referencia, podemos calcular el ángulo relativo entre cada sensor y verificar la posición en grados de la rodilla.

Otra ventaja a este acomodo es su facilidad de escala, es posible replicar con facilidad el sistema en la otra extremidad para obtener los datos de movimiento de ambas rodillas o de extender a 4 sensores sobre una sola pierna para cubrir tobillo y cadera.

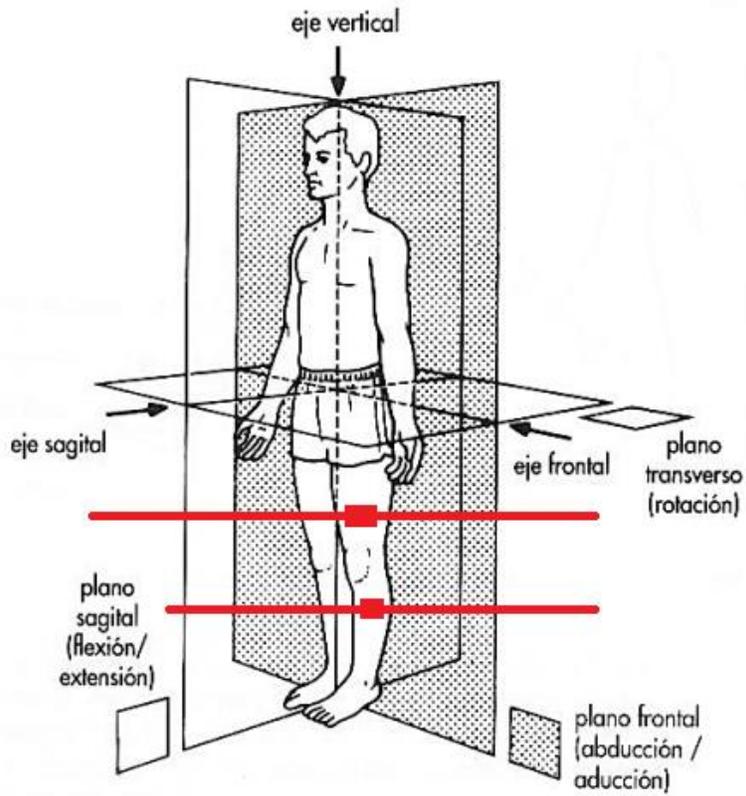


Figura 7 - Posición inicial de los sensores para la adquisición de datos del movimiento de 1 rodilla [13]

Calendario de Actividades

El proyecto se pretende realizar con la distribución de actividades mostradas en la tabla 6, en un plazo de 16 semanas establecidas por calendario oficial de ITESO para el periodo Otoño 2022.

Tabla 6 - Calendario de Actividades [Anexo 1]

No	Actividad	Recursos	Encargado	Req	Inicio	Final	ENERO		FEBRERO				MARZO				ABRIL		MAYO	
							1	2	3	4	5	6	7	8	9	10	11	SS	12	13
1	Apectos Teóricos				M1	M15														
1,1	Elección de proyecto		Lilia Lobato		M1	M1														
1,2	Entrega 1 RPAP		Lilia Lobato, Asesor	1,1	M4	M4														
1,3	Definición de alcance		Lilia Lobato y Alonso Muro	1,1	M2	M3														
1,4	Entrega 2 RPAP		Lilia Lobato, Asesor	1,2	M8	M8														
1,5	Marco teórico		Lilia Lobato	1,2	M5	M8														
1,6	Arquitectura a alto nivel		Lilia Lobato	1,5	M7	M7														
1,7	Entrega 3 RPAP		Lilia Lobato, Asesor	1,4	M10	M16														
2	Software				M8	M15														
2,1	Ángulos del acelerómetro		Lilia Lobato	1,6	M8	M8														
2,2	Ángulos del giroscopio		Lilia Lobato	2,1	M8	M8														
2,3	Relación entre los ángulos		Lilia Lobato	2,2	M9	M9														
2,4	Orientación de 1 sensor	Sensores	Lilia Lobato	2,3	M10	M11														
2,5	Cálculo del ángulo medido entre dos sensores	Sensores	Lilia Lobato	2,4	M12	M12														
2,6	Interfaz gráfica, visualización de datos	Prototipo funcional	Lilia Lobato	4,1	M13	M14														
2,7	Simulación	Prototipo funcional	Lilia Lobato	4,1	M15	M15														
3	Hardware				M7	M10														
3,1	Validación de sensores	Prototipo inicial	Lilia Lobato	1,6	M7	M7														
3,2	Recolección de datos de los sensores	Prototipo inicial	Lilia Lobato	2,4	M8	M8														
3,3	Etapas de filtrado digital	Prototipo inicial	Lilia Lobato	2,2	M9	M9														
3,4	Proceso de calibración de sensores	Prototipo inicial	Lilia Lobato	2,5	M10	M10														
4	Desarrollo del Prototipo				M7	M12														
4,1	Prototipo en protoboard	Prototipo inicial	Lilia Lobato	3,4	M7	M10														
4,2	Prototipo en PCB	Prototipo funcional	Lilia Lobato	4,1	M11	M12														
4,3	Acople a piernas	Prototipo 3D	Lilia Lobato	3,5	M9	M11														
5	Validación				M7	M15														
5,1	Validación de arquitectura	Sensores	Lilia Lobato, Asesor	1,6	M7	M8														
5,2	Validación de acople de sensores a la pierna	Prototipo 3D	Lilia Lobato	4,3	M11	M12														
5,3	Validación de cálculos	Prototipo inicial	Lilia Lobato	2,5	M10	M11														
5,4	Validación de recolección de datos	Prototipo inicial	Lilia Lobato	2,5	M9	M9														
5,5	Validación de marcha	Prototipo funcional	Lilia Lobato, Asesor	4,2	M12	M15														
6	Presentación Final				M16	M16														
6,1	Creación de material audiovisual	Prototipo funcional y documentación aprobada	Lilia Lobato, Asesor	1,7	M16	M16														

1.5. Desarrollo de la propuesta de mejora

Para la captación de movimientos, se midieron los ángulos girados por la rodilla. Para ello se calcularon las orientaciones de los segmentos, es decir, de un nodo arriba de la rodilla y de un nodo por debajo de la rodilla. Con la orientación de dos nodos se pudo obtener el ángulo de la rodilla.

El desarrollo del sistema se realizó progresivamente, empezando con 1 sensor directo a un ATmega328, continuando con la incorporación del módulo wifi y acabando con 2 sensores directo a un ESP8266 con interacción con una base de datos.

Giroscopio digital

El sensor MPU-6050 es una unidad de medición inercial (IMU) de seis grados de libertad (6DOF) fabricado por *Invensense*, que combina un acelerómetro de 3 ejes y un giroscopio de 3 ejes, como muestra la figura 8.

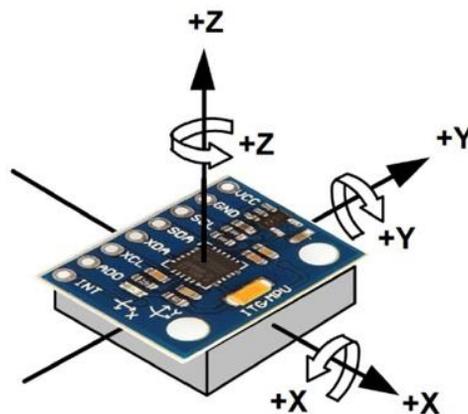


Figura 8 - Orientación de MPU6050 [14]

La comunicación puede realizarse tanto por SPI como por bus I2C, por lo que es sencillo obtener los datos medidos. Para este proyecto utilizamos la IMU por medio de I2C ya que pudimos obtener los datos de múltiples sensores sin aumentar el número de periféricos.

La MPU6050 dispone de conversores analógicos digitales (ADC) de 16 bits. El rango del acelerómetro puede ser ajustado a $\pm 2g$, $\pm 4g$, $\pm 8g$, y $\pm 16g$, el del giroscopio a ± 250 , ± 500 , ± 1000 , and $\pm 2000^\circ/\text{sec}$.

Las conexiones necesarias para obtener la información son SDA y SCL, las cuales son mostradas en la figura 9.

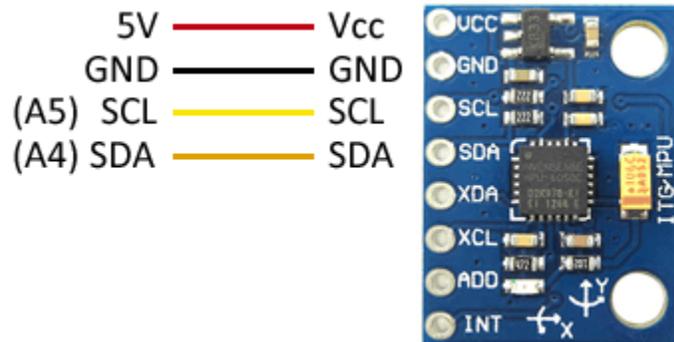


Figura 9 - Conexiones del MPU6050 [15]

Adicionalmente, como se requirió conectar más de un sensor usando la misma conexión I2C, se programó cada dispositivo con una dirección I2C diferente, esto es posible multiplexando al puerto AD0, pero, como solamente necesitamos 2 sensores nos fue posible conectar el puerto AD0 a 5V en un solo sensor como se muestra en la tabla 7.

Tabla 7 - Direcciones I2C

Puerto AD0	Dirección I2C
AD0 = HIGH (3.3V – 5V)	0x69
AD0 = LOW (GND – Sin Conexión)	0x68

Las conexiones con Arduino (ATmega328) quedaron como la figura 10, usamos los puertos A4 y A5 para las salidas del protocolo I2C y se usó el mismo voltaje de alimentación de la tarjeta para los giroscopios digitales.

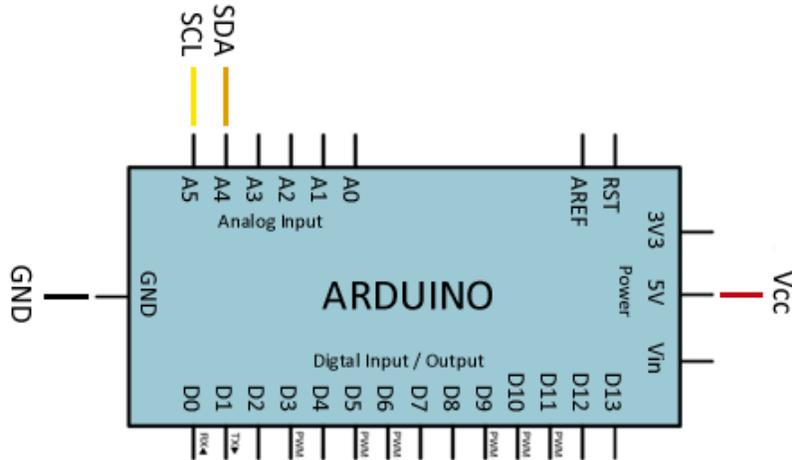


Figura 10 - Implementación de comunicación SPI con Arduino [15]

El diagrama de las figuras 1y figura 9 resultaron en el circuito de la figura 11. En esta se puede observar la mitad de la arquitectura de desarrollo propuesta en la figura 3 que consiste en 2 sensores MPU6050 conectados en configuración maestro-esclavo. Con este circuito se realizó la etapa de calibración e implementación de cálculos para obtener los ángulos.

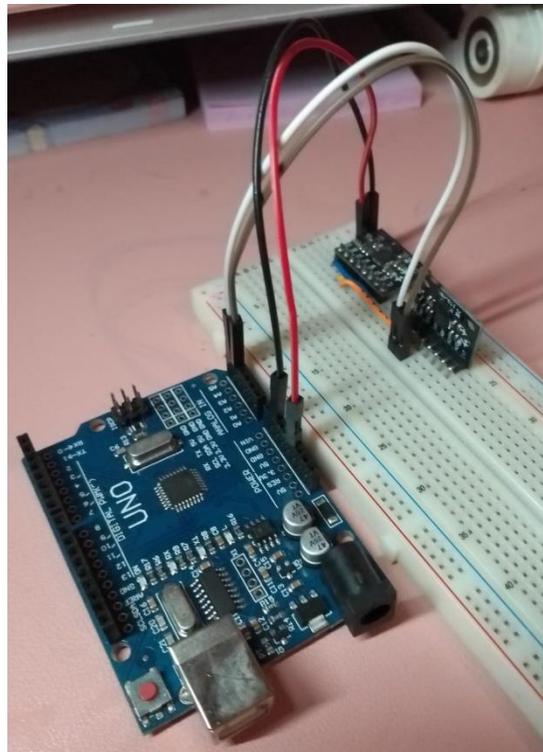


Figura 11 - Implementación de la arquitectura de desarrollo

Para realizar la lectura del MPU-6050 usamos la librería desarrollada por Jeff Rowberg disponible en el siguiente enlace de GitHub:

<https://github.com/jrowberg/i2cdevlib/tree/master/Arduino/MPU6050>

Para implementar la comunicación I2C, usamos la librería I2Cdev desarrollada por el mismo autor ya que mejora la comunicación I2C. La podemos encontrar en el siguiente enlace de GitHub:

<https://github.com/jrowberg/i2cdevlib/tree/master/Arduino/I2Cdev>

Esta decisión se tomó ya que si no se utilizaran estas librerías se tendrían que realizar las transacciones de manera bloqueantes lo cual no hubiera permitido hacer transmisiones de datos de forma eficiente.

La segunda razón fue la reducción de error y ruido que tienen las lecturas del MPU-6050. Para eliminarlas se debe implementar un filtro Kalman que ya viene en esta librería, esto hizo la interfaz aún más fácil de utilizar, de bajo costo en procesamiento y con una mejor precisión.

La información se leyó del banco de direcciones de memoria en la MPU, no utilizamos la unidad DMP (Procesador digital de movimiento) ya que eso aumentaba el tiempo de procesamiento de las lecturas y el error.

Microcontrolador ESP8266

Para cargar el programa lo haremos a través del puerto serial, lo que supone que utilizaremos los pines RX y TX para transmitir los datos a la memoria Flash mostrada en la figura 12, donde se almacenará el sketch.

Existen dos modos de operación y se configuran a través de los puertos GPIO0 y GPIO2: el modo Flash y el modo UART.

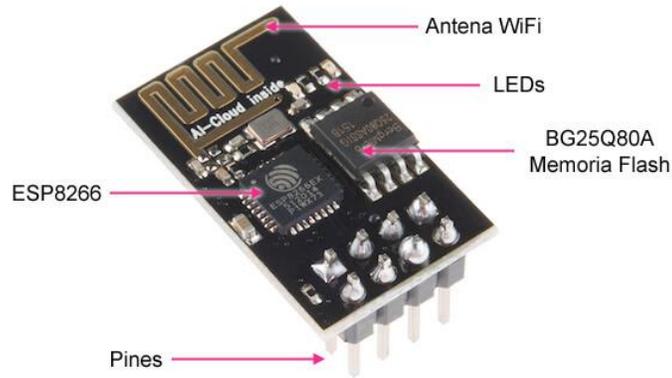


Figura 12 - Componentes del microcontrolador ESP8266 [16]

- Modo UART, carga de programa en la memoria: Cuando queremos cargar un programa en el ESP8266 debemos encenderlo o reiniciarlo teniendo el pin GPIO0 a nivel bajo y el GPIO2 a nivel alto.
- Modo Flash: ejecución de programa: Para ejecutar el programa se debe de poner el GPIO0 y el GPIO2 a nivel alto (3,3 V).

Como se observa en la tabla 8, se debe de crear una circuitería cambiando el estado de ciertos puertos de forma que permita cambiar fácilmente entre estas opciones.

Tabla 8 - Opciones de encendido para ESP8266-01

GPIO0	GPIO2	Modo	Comentarios
Alto (3.3 V)	Alto (3.3 V)	Flash	El dispositivo va a comenzar a ejecutar el programa guardado en la memoria SPI Flash
Bajo (0 V)	Alto (3.3 V)	UART	Se puede realizar la programación por medio de los puertos TX y RX

Para utilizar la IDE de Arduino se descargaron los Cores de tarjetas adicionales de la familia ESP del siguiente enlace:

http://arduino.esp8266.com/stable/package_esp8266com_index.json

En el gestor de tarjetas se instaló el ESP8266 by ESP8266 Community, como se muestra en la figura 13.

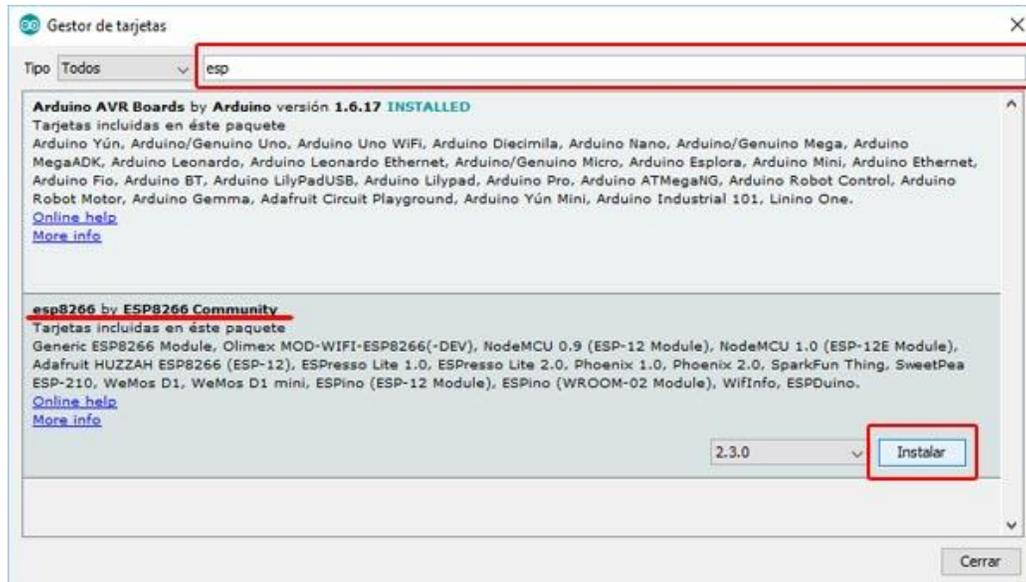


Figura 13 - Gestor de tarjetas para interactuar con un ESP8266

Una vez que instalamos la librería de la placa, se puede elegir entre los diferentes tipos de módulos ESP que se nos ofrecen: para utilizar el ESP8266 seleccionamos Generic ESP8266 module. Este menú es mostrado en la figura 14.

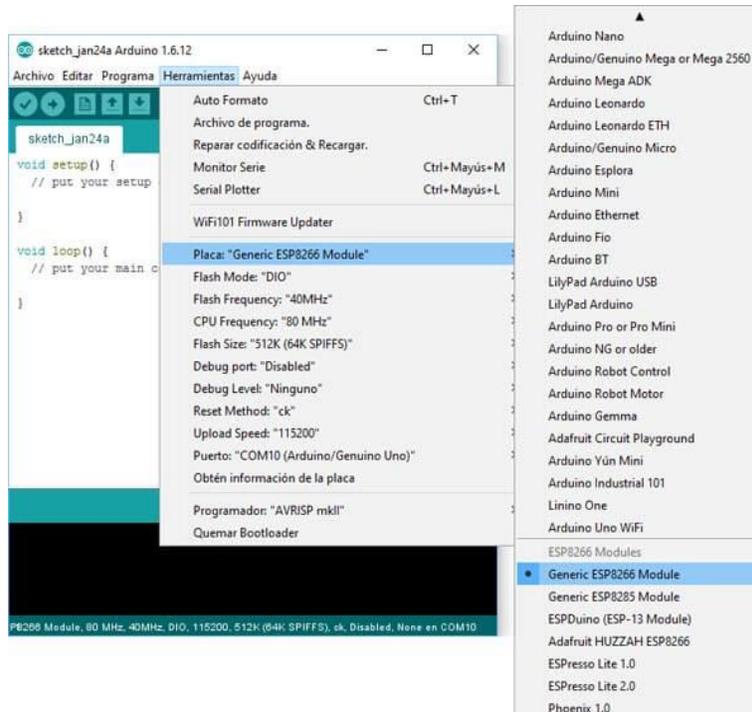


Figura 14 - Configuración de la IDE de Arduino para programar un ESP8266

Con la IDE configurada, se puede programar el eSP8266 conectando un Arduino como en la figura 15. El Arduino va a funcionar como puente serial por lo que debe estar en modo de programador.

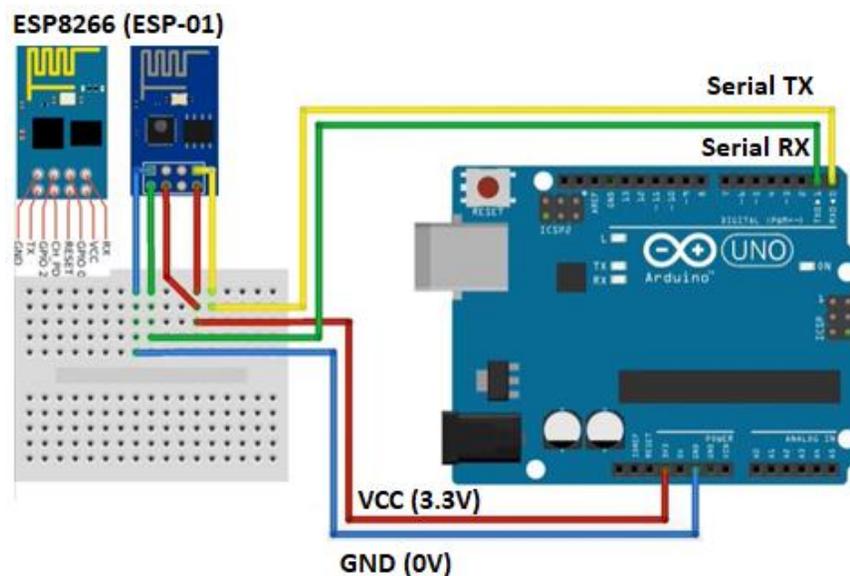


Figura 15 - Pinout y conexión a Arduino de un ESP8266 [16]

Una alternativa es utilizar un módulo adaptador TTL (transistor-transistor logic) de USB a UART con conexiones como en la figura 16.

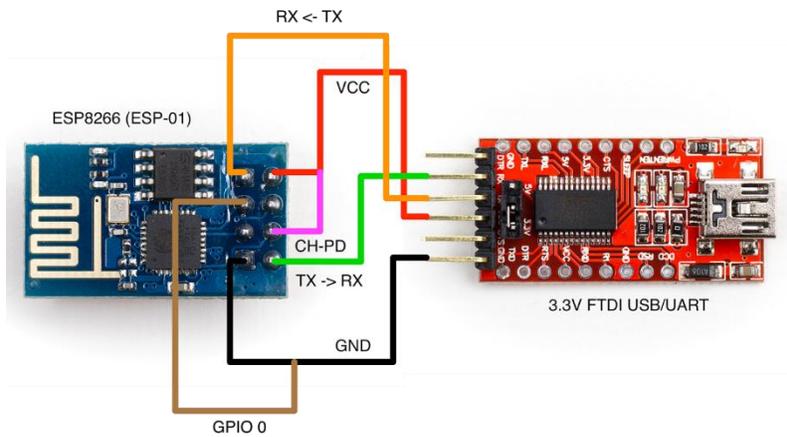


Figura 16 - Conexiones entre ESP8266 a TTL

Tras programar el ESP8266, nos es posible conectar la IMU6050 siguiendo el diagrama de la figura 17. Con esta configuración, se conectan los puertos GPIO0 y GPIO1 como SDA y SCL para comenzar con la recolección de datos.

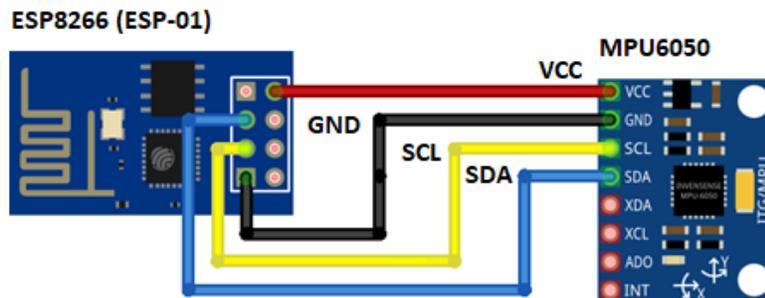


Figura 17 - Circuito inicial para la recolección de datos del MPU6050 por medio del ESP32 [16]

El problema con utilizar los puertos GPIO0 y GPIO1 como bus I2C es que algunos dispositivos, especialmente los que cuentan con baterías externas, pueden quedarse atascados y mantener el bus inactivo. Esto hace que el ESP8266 se reinicie y entre accidentalmente al

modo de carga de programa, por lo que se debe de aislar el bus hasta que el ESP8266 pase por su etapa de arranque [17].

Este problema se puede evitar si se utilizan los puertos de TX y RX como bus I2C y resistencias de pull up como en la figura 18. Con este circuito logramos lo siguiente:

- GPIO1 (TX) se utiliza como la línea de datos, porque siempre obtendrá alguna salida de depuración en GPIO1 al momento de encender el ESP8266. No hay manera de suprimir esta salida, pero RX se mantendrá alta por lo que ninguno de estos datos se sincronice con los esclavos.
- Al programar el ESP8266, la línea RX se conecta a la salida del programador. Al final de la programación, el ESP8266 se reinicia y la resistencia de protección 330 ohm evita que RX cortocircuite la unidad.
- Las resistencias del bus I2C proporcionan una protección similar para el TX y RX sobre cortocircuitos por el tráfico del bus I2C

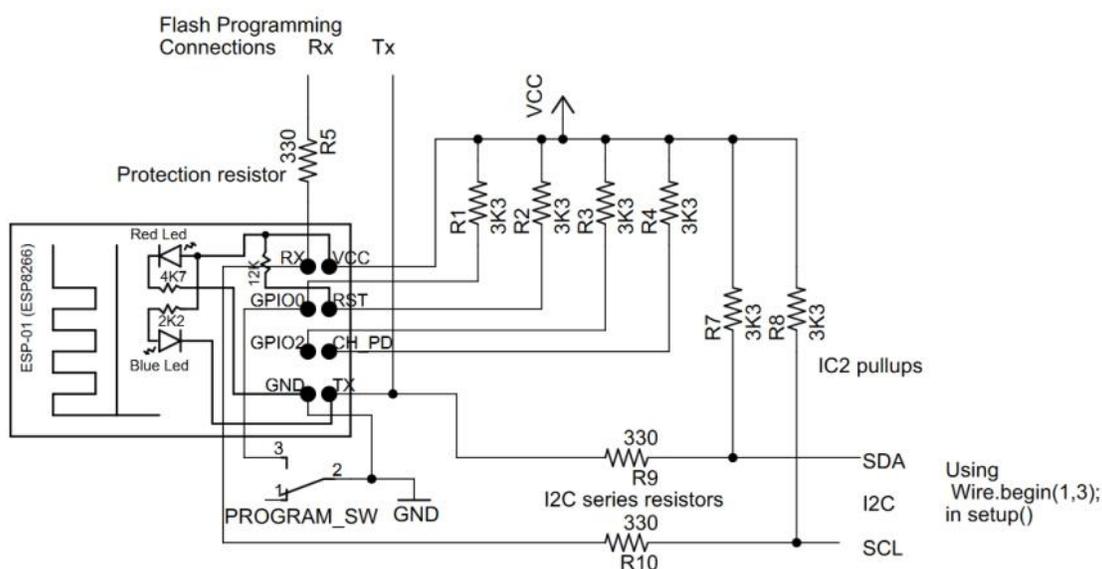


Figura 18 - Circuito para usar TX y RX del ESP8266 como bus I2C [17]

Como el ESP8266 es parte de la arquitectura final, realizamos un un PCB (Printed Circuit Board) en la plataforma *EasyEDA* adicionando un switch de reprograma al sistema y un botón para reiniciar el circuito como se muestra en la figura 19.

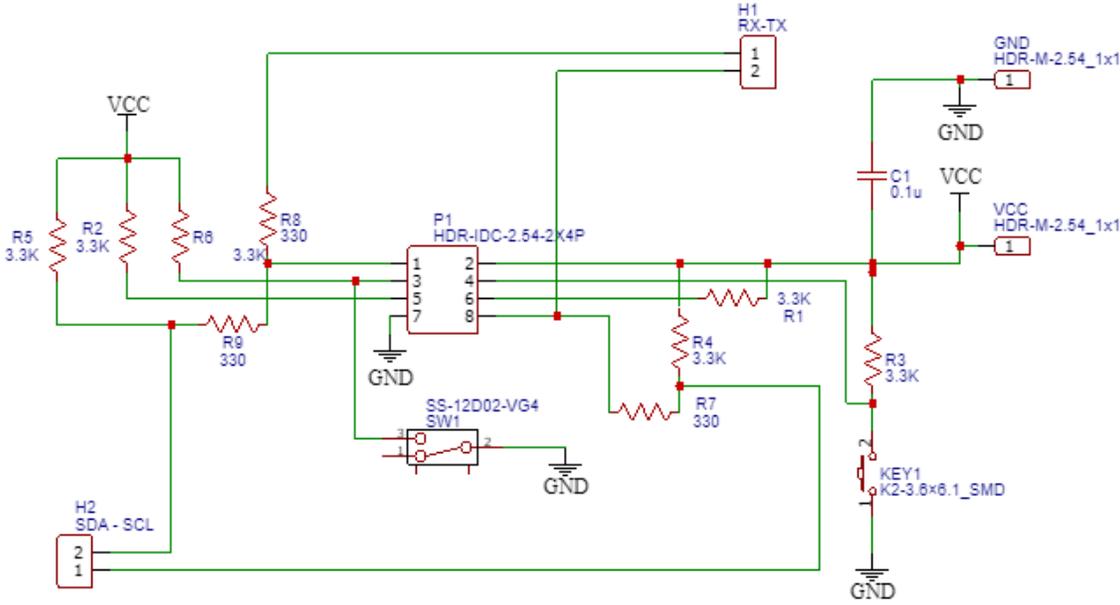


Figura 19 - Esquemático del prototipo final

Con este esquemático creamos el PCB mostrado en la figura 20.

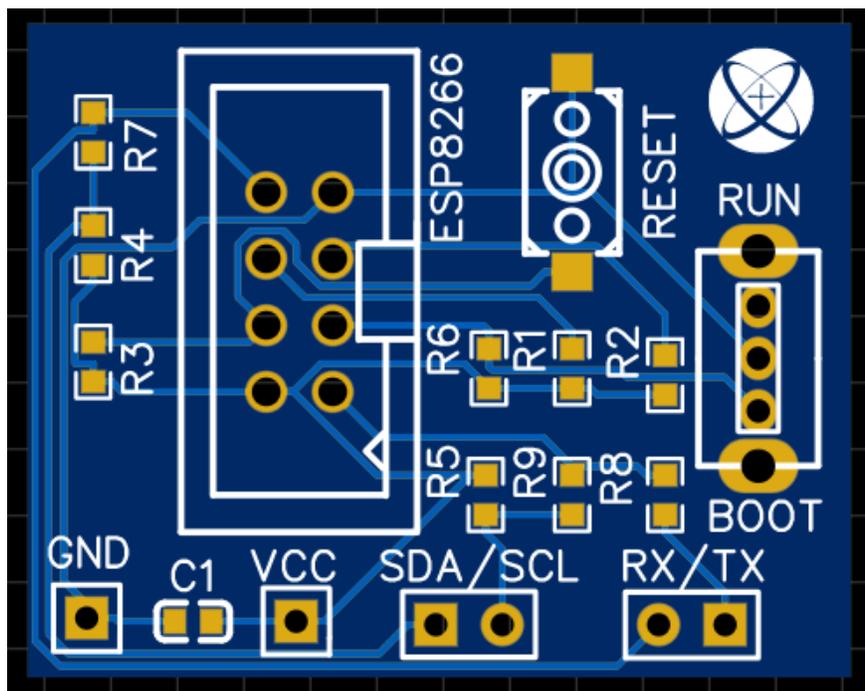


Figura 20 - PCB layout

Tras impresión y soldado de componentes tendríamos el resultado final de la figura 21.

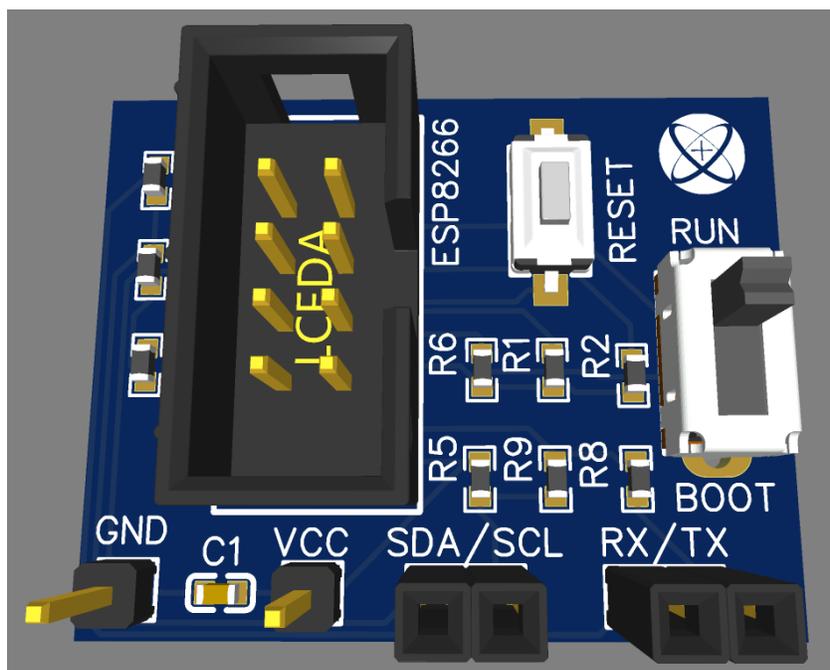


Figura 21 - 3D layout del PCB y componentes

Por cuestiones de tiempo y restricciones aduanales, tomamos la decisión de montar el circuito de la figura 19 en una tarjeta perforada dando como resultado el prototipo final de la figura 22.

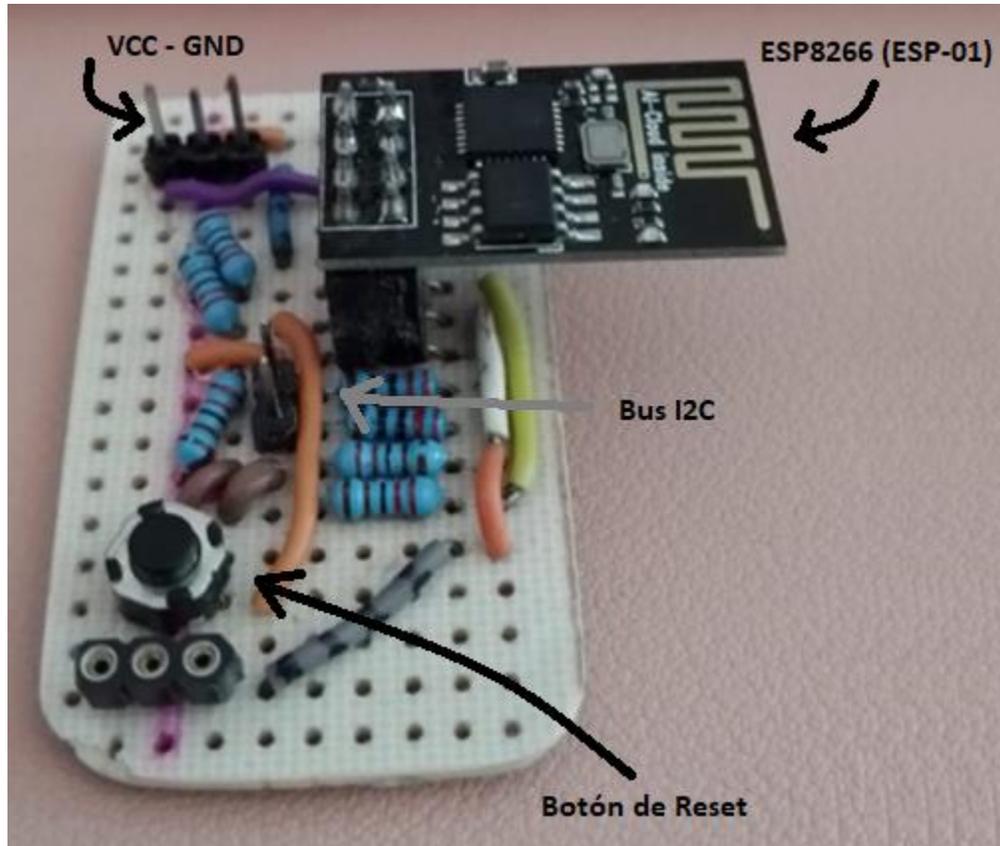


Figura 22 - Prototipo final montado en tarjeta perforada

Calibración IMU

Si una MPU6050 se comportara como un sensor ideal, no sufriría de cambios de temperatura y al encontrarse en una posición neutral y en un lugar donde la fuerza de gravedad es exactamente 1g; esta se podría considerar como perfectamente calibrada y tendrá un valor de 0 para todas las aceleraciones y desplazamientos, excepto para la aceleración Z, en la que tendrá 16384 (es decir, 2^{14}) tal como se muestra en la columna “Valor Central” de la tabla 9.

Tabla 9 - Tabla de valores extremos del MPU6050

Variable	Valor mínimo	Valor Central	Valor Máximo
Lectura MPU6050	-32768	0	+32767
Aceleración	-2g	0g	+2g
Velocidad Angular	-250°/s	0°/s	+250°/s

Los dispositivos reales no funcionan como uno ideal por lo que tuvimos que realizar un proceso de calibración por medio de valores de offset para compensar los valores regresados y acercarnos lo más posible al valor central de la tabla 9.

Para obtener estos valores de offset se utilizó el ejemplo IMU_ZERO de la librería MPU6050 la cual tras poner el sistema sobre una superficie plana y horizontal tomó entre 1000 a 10000 muestras dependiendo de la precisión que se quiere manejar y el tiempo que se quiera dedicar al proceso de calibración. Esta precisión se decidió por medio de un proceso empírico.

Con cada una de estas muestras se va a encontrar el offset deseado siguiendo los siguientes parámetros:

- Encontrando 2 estimaciones, una demasiado baja y otra demasiado alta
- Cerrando estos valores hasta que el diferencial de offset no pueda ser más pequeño.

Al terminar el algoritmo se muestra una línea con los valores en formato de diccionario donde el valor de la izquierda es el offset calculado y el de la derecha la lectura de la IMU con dicho offset en la posición en la que se encuentra.

El código completo para realizar la calibración se encuentra en los anexos. A continuación, vamos a documentar una ejecución de calibración sobre un sensor MPU6050, es importante que al calibrarlo se encuentre en una superficie plana y no se mueva en lo absoluto durante todo el proceso (aproximadamente 5min).

Tras inicializar el bus de I2C y asegurarnos de estar conectados a un sensor, comenzamos a calcular el offset usando un PID que varía sus parámetros tras N lecturas. En la figura 23 se muestran los valores crudos del giroscopio y acelerómetro tras correr el PID por 1000 lecturas.

```
PID tuning Each Dot = 100 readings
>.....>.....
at 600 Readings

//          X Accel      Y Accel      Z Accel      X Gyro      Y Gyro      Z Gyro
// OFFSETS  -707.00000,  124.00000,  1527.00000, -47.00000,  15.00000,  57.00000

>.>.700 Total Readings

//          X Accel      Y Accel      Z Accel      X Gyro      Y Gyro      Z Gyro
// OFFSETS  -707.00000,  122.00000,  1527.00000, -47.00000,  14.00000,  58.00000

>.>.800 Total Readings

//          X Accel      Y Accel      Z Accel      X Gyro      Y Gyro      Z Gyro
// OFFSETS  -705.00000,  122.00000,  1527.00000, -46.00000,  15.00000,  57.00000

>.>.900 Total Readings

//          X Accel      Y Accel      Z Accel      X Gyro      Y Gyro      Z Gyro
// OFFSETS  -705.00000,  124.00000,  1527.00000, -47.00000,  15.00000,  57.00000

>.>.1000 Total Readings

//          X Accel      Y Accel      Z Accel      X Gyro      Y Gyro      Z Gyro
// OFFSETS  -707.00000,  122.00000,  1527.00000, -47.00000,  15.00000,  57.00000

Any of the above offsets will work nice
```

Figura 23 - Calibración por PID en 1000 lecturas

Ahora comenzamos a cerrar los valores obtenidos hasta encontrar el offset. Al final de la figura 24 se muestran los mejores offset, en esta corrida obtuvimos un offset de [-1, -10, 16383] para el acelerómetro y de [-3, 0, -1] para el giroscopio.

```

Lets proof the PID tuning using another method:
averaging 1000 readings each time
expanding:
.... XAccel          YAccel          ZAccel          XGyro          YGyro          ZGyro
[0,0] --> [6359,6362] [0,0] --> [-1139,-1134] [0,0] --> [2846,2849] [0,0] --> [186,186] [0,0] --> [-58,-58] [0,0] --> [-227,-228]
.... [-1000,0] --> [-2634,6361] [0,1000] --> [-1139,7986] [0,1000] --> [2838,11720] [-1000,0] --> [-3812,183] [0,1000] --> [-58,3938] [0,1000] --> [-228,3767]
.... [-1000,0] --> [-2638,6364] [0,1000] --> [-1141,7987] [0,2000] --> [2844,20596] [-1000,0] --> [-3813,181] [0,1000] --> [-54,3938] [0,1000] --> [-223,3768]

closing in:
.. XAccel          YAccel          ZAccel          XGyro          YGyro          ZGyro
[-1000,-500] --> [-2638,1864] [0,500] --> [-1141,3424] [1000,2000] --> [11720,20596] [-500,0] --> [-1813,181] [0,500] --> [-54,1942] [0,500] --> [-223,1774]
.. [-750,-500] --> [-387,1864] [0,250] --> [-1141,1143] [1500,2000] --> [16155,20596] [-250,0] --> [-814,181] [0,250] --> [-54,941] [0,250] --> [-223,772]
.. [-750,-625] --> [-387,731] [125,250] --> [-7,1143] [1500,1750] --> [16155,18375] [-125,0] --> [-314,181] [0,125] --> [-54,441] [0,125] --> [-223,271]
.. [-750,-687] --> [-387,172] [125,187] --> [-7,555] [1500,1625] --> [16155,17258] [-62,0] --> [-61,181] [0,62] --> [-54,188] [0,62] --> [-223,21]
.. [-718,-687] --> [-93,172] [125,156] --> [-7,283] [1500,1562] --> [16155,16706] [-62,-31] --> [-61,62] [0,31] --> [-54,64] [0,31] --> [-104,21]
.. XAccel          YAccel          ZAccel          XGyro          YGyro          ZGyro
[-718,-702] --> [-93,50] [125,140] --> [-7,135] [1500,1531] --> [16155,16421] [-62,-46] --> [-61,2] [15,31] --> [0,64] [46,62] --> [-44,21]
.. [-710,-702] --> [-25,50] [125,132] --> [-7,64] [1515,1531] --> [16278,16421] [-54,-46] --> [-29,2] [15,23] --> [0,30] [54,62] --> [-13,21]
.. [-710,-706] --> [-25,11] [125,128] --> [-7,28] [1523,1531] --> [16350,16421] [-50,-46] --> [-14,2] [15,19] --> [0,15] [54,58] --> [-13,2]
.. [-708,-706] --> [-8,11] [125,126] --> [-7,10] [1523,1527] --> [16350,16388] [-48,-46] --> [-6,2] [15,17] --> [0,7] [56,58] --> [-4,2]
averaging 10000 readings each time
..... [-707,-706] --> [-7,11] [125,126] --> [-8,10] [1525,1527] --> [16367,16388] [-47,-46] --> [-3,2] [15,16] --> [0,3] [57,58] --> [-1,2]
..... XAccel          YAccel          ZAccel          XGyro          YGyro          ZGyro
[-707,-706] --> [-6,11] [125,126] --> [-9,10] [1526,1527] --> [16383,16388] [-47,-46] --> [-3,2] [15,16] --> [0,3] [57,58] --> [-1,2]
..... [-707,-706] --> [-7,10] --> [-5,11] [125,126] --> [-10,10] [1526,1527] --> [16383,16388] [-47,-46] --> [-3,2] [15,16] --> [0,3] [57,58] --> [-1,2]
----- done -----

```

Figura 24 - Valores finales de offset

Orientación de 1 Sensor

Para determinar la distribución de un nodo respecto a una base en su espacio tridimensional tuvimos que establecer su posición y orientación para lo que se requirieron 3 parámetros para la posición relativa y 3 para su orientación relativa.

La razón por la que necesitamos ambos parámetros es porque al combinarlos, compensan las limitaciones del otro:

- Los acelerómetros no tienen deriva a medio o largo plazo ya que realizan la medición absoluta del ángulo que forma el sensor con la dirección vertical, marcada por la gravedad. Sin embargo, se ven influenciados por los movimientos del sensor y el ruido por lo que no son fiables a corto plazo.
- Los giroscopios funcionan bien para movimientos cortos o bruscos, pero al usar giroscopios con el tiempo se va acumulando error y son sumamente susceptibles al ruido por lo que a medio o largo plazo tienen deriva.

Al combinar las mediciones del acelerómetro y del giroscopio podemos obtener mediciones de la orientación con mayor precisión que si se usaran por separado. Para realizar esto, usamos un filtro complementario el cual puede definirse en la ecuación 1.

$$\theta = A * (\theta_{previo} + \theta_{giroscopio}) + B * \theta_{acelerómetro}$$

Ecuación 1 - Formulación del filtro complementario

A y B son constantes que al sumarse deben dar 1. A es la gravedad por lo que será equivalente a 0.98 y B será 0.02. Estos valores son reemplazados en la ecuación 2.

$$\theta = 0.98 * (\theta_{previo} + \theta_{giroscopio}) + 0.02 * \theta_{acelerómetro}$$

Ecuación 2 - Formulación del filtro complementario con constantes

Este filtro funciona como un filtro pasa altas para la medición del giroscopio y un filtro pasa bajas para la medición del acelerómetro. Es decir, la señal del giroscopio mantiene sus características positivas de corto plazo y la señal del acelerómetro mantiene sus características positivas de corto plazo.

Para implementar el filtro complementario es necesario obtener el ángulo de inclinación del giroscopio y del acelerómetro. Para calcular la orientación relativa se puede optar por varios sistemas:

Tabla 10 - Sistemas para calcular la orientación relativa de un sensor.

Sistemas para el cálculo de ángulos
Cálculo de las matrices de rotación
Cálculo de los cuaterniones
Cálculo de los ángulos de Euler
Cálculo de los ángulos de Tait-Bryan

Ángulo del acelerómetro

Optamos por calcular los ángulos de Tait-Bryan ya que su fórmula es intuitiva y sencilla, tienen la ventaja de ser conmutativos, es decir, la orientación final obtenida es independiente del orden en que apliquemos la rotación. Lamentablemente, este método tiene una desventaja denominada bloqueo cardán, la cual nos lleva a perder 1 grado de libertad ya que cuando dos de los ejes (X, Y) giran sobre el mismo eje (Z), este último no puede calcularse.

La aceleración puede expresarse en 3 ejes: X, Y y Z donde uno de ellos será aproximadamente 9.8 m/s^2 (gravedad de la tierra). Gracias a este valor se puede calcular el ángulo de inclinación respecto al eje X o eje Y.

Si la IMU está completamente alineada con el suelo, como en la figura 25, el eje Z marcará 9.8, y los otros dos ejes marcarán 0.

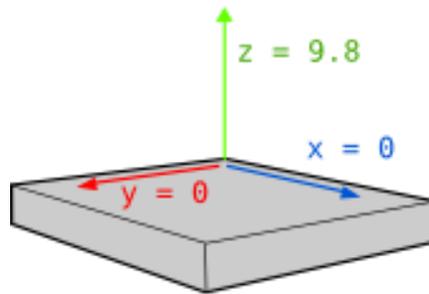


Figura 25 - IMU alineada al suelo [14]

Si giramos la IMU 90 grados, como en la figura 26, el eje X marcará 9, y los otros marcarán 0.

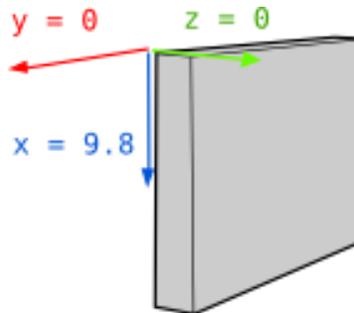


Figura 26 - IMU a 90 grados de inclinación [14]

Con esto y aplicando trigonometría es posible calcular el ángulo de inclinación con las ecuaciones 3 y 4.

$$\text{Ángulo } Y = \tan^{-1}\left(\frac{x}{\sqrt{y^2 + z^2}}\right)$$

Ecuación 3 - Ángulo de inclinación sobre eje Y

$$\text{Ángulo } X = \tan^{-1}\left(\frac{y}{\sqrt{x^2 + z^2}}\right)$$

Ecuación 4 - Ángulo de inclinación sobre eje X

Dado que el ángulo se calcula a partir de la gravedad, no es posible calcular el ángulo Z. Para esta aplicación específica pudimos prescindir del eje Z por lo que este cálculo es suficiente.

Tras obtener los valores lo único que resta es hacer una regla de 3 de forma que nos dé un rango de 180 grados a -180 grados.

Ángulo del giroscopio

El giroscopio nos entrega la velocidad angular por lo que, para calcular el ángulo actual, necesitamos integrar la velocidad y conocer el ángulo inicial tal como se muestra en las ecuaciones 5 y 6.

$$\text{Ángulo } Y = \theta_{y0} + \omega_y \Delta t$$

Ecuación 5 - Ángulo de inclinación sobre eje Y

$$\text{Ángulo } X = \theta_{x0} + \omega_x \Delta t$$

Ecuación 6 - Ángulo de inclinación sobre eje X

Desplegado de datos por servicio web usando un ESP8266

Creamos un tablero de instrumentación desde un servidor web a través de una conexión HTTP con el ESP8266 donde se despliegan las lecturas del sensor MPU6050. Las lecturas consistirán en los ángulos (x, y) utilizando eventos enviados por el servidor (SSE) que actualizará automáticamente las lecturas del sensor en todos los clientes web conectados. Además, contiene una animación en 3D de la orientación de cada sensor utilizando la biblioteca three.js.

Para hacer que la implementación del servidor web sea más cómodo y práctico, utilizamos la función de ESP8266 LittleFS. Con esta es posible guardar los archivos HTML, CSS y JavaScript en la memoria Flash del ESP8266 para construir el servidor web sin tener que utilizar ninguna memoria externa.

La interfaz mostrada en la figura 27 consiste en recuadros para las lecturas de los ángulos de cada sensor y una representación tridimensional de la orientación.

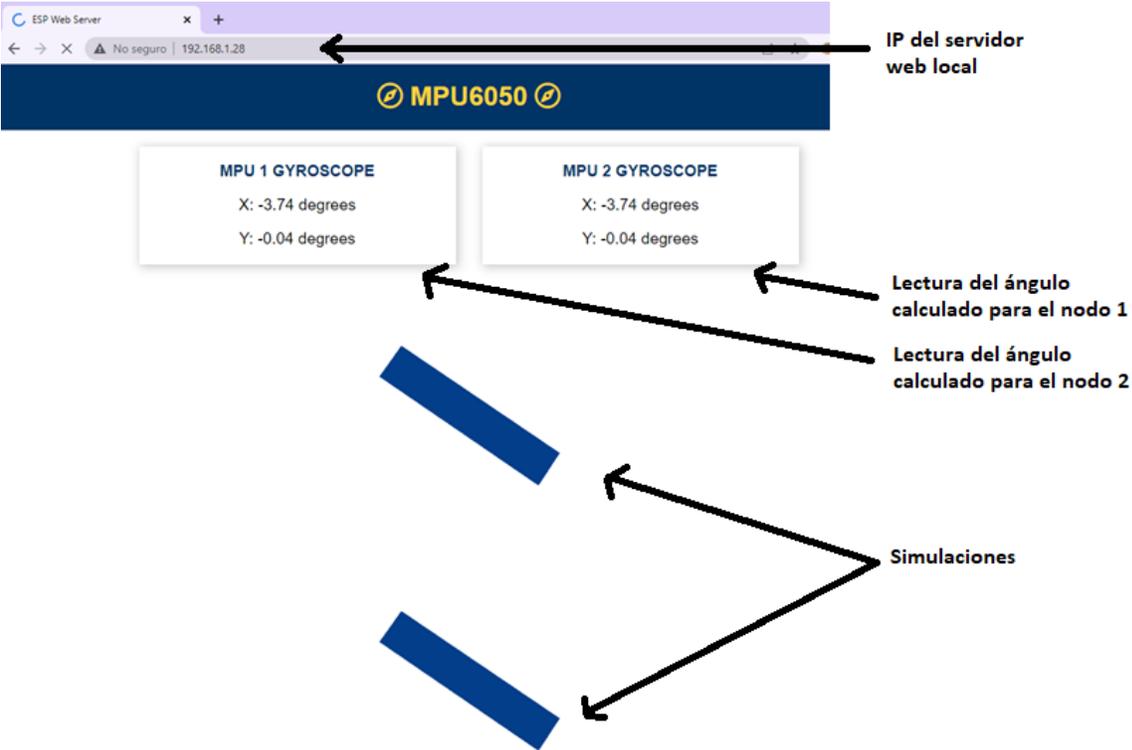


Figura 27 – Interfaz Final

Las lecturas del giroscopio se actualizarán automáticamente cada 1 segundo. Para lograr esto necesitamos las siguientes librerías:

- Biblioteca ESPAsyncWebServer
- Librería ESPAsyncTCP
- Biblioteca Arduino_JSON

La librería ESPAsyncWebServer nos ayudará a crear nuestro servidor asíncrono web fácilmente; incorporando la librería ESPAsyncTCP como dependencia. Estas dos librerías no están disponibles en el gestor de librerías de Arduino por lo que se tienen que instalar desde sus respectivos repositorios.

Para instalar la librería Arduino_JSON simplemente la buscamos en el gestor de bibliotecas de Arduino, como se demuestra en la figura 28.



Figura 28 - Libreria para mandar mensajes JSON

Sistema de Archivos utilizando LittleFS

Creamos cuatro archivos diferentes: HTML, CSS, JavaScript y Arduino sketch. Estos deben estar en una carpeta de proyecto como se muestra en la figura 29.

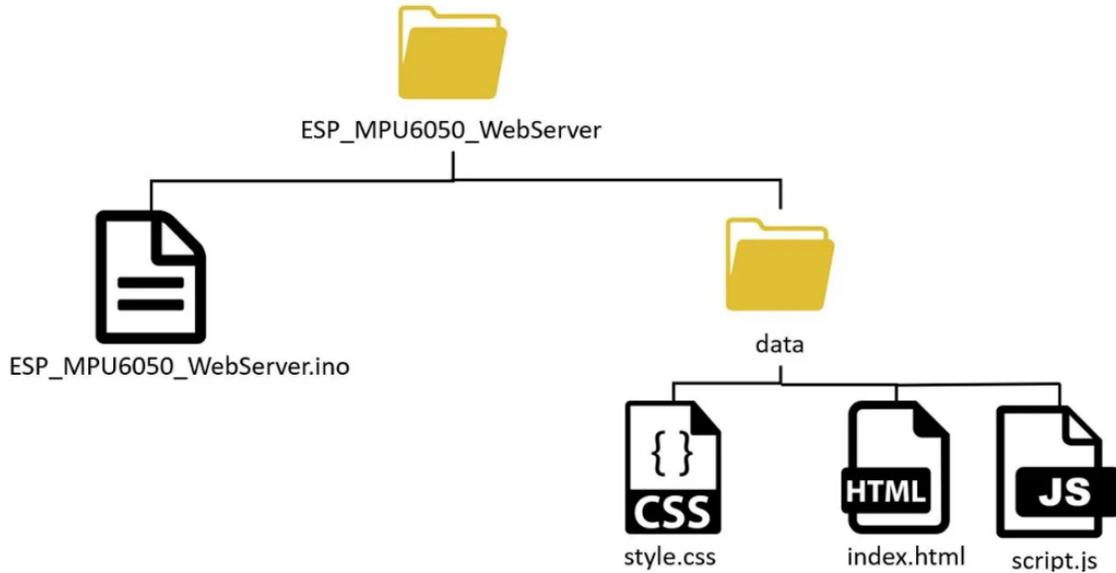
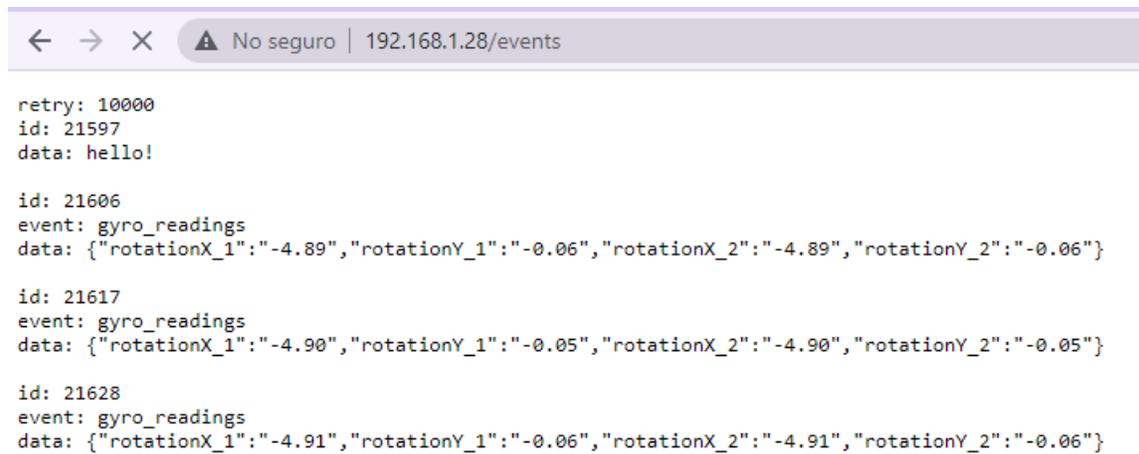


Figura 29 - Sistema de archivos a guardar dentro del ESP8266 [18]

- El archivo HTML especifica los elementos como el título de la página web, las cajas, y la representación en 3D de nuestro módulo MPU6050.
- En el archivo CSS le daremos estilos a la página web de forma que sea presentable.
- El archivo JavaScript será responsable de manejar los eventos SSE y la manipulación de objetos 3D.

Recepción y manejo de eventos SSE

Al iniciar la conexión EventSource con el servidor y podemos manejar los eventos que se reciban al servidor, como se muestra en la figura 30.



```
retry: 10000
id: 21597
data: hello!

id: 21606
event: gyro_readings
data: {"rotationX_1": "-4.89", "rotationY_1": "-0.06", "rotationX_2": "-4.89", "rotationY_2": "-0.06"}

id: 21617
event: gyro_readings
data: {"rotationX_1": "-4.90", "rotationY_1": "-0.05", "rotationX_2": "-4.90", "rotationY_2": "-0.05"}

id: 21628
event: gyro_readings
data: {"rotationX_1": "-4.91", "rotationY_1": "-0.06", "rotationX_2": "-4.91", "rotationY_2": "-0.06"}
```

Figura 30 - Mensajes JSON recibidos en el manejador de eventos

Para poder manejar los eventos, definimos una fuente con la URL /events. Esta URL enviará las actualizaciones a nuestra página web que podemos identificar usando la función `addEventListener()`. Tendrá como argumentos el nombre del evento y la función a llamar.

En el evento 'gyro_readings', cada 500 milisegundos el servidor web recibirá nuevas lecturas de los sensores. Cuando se detecta este evento, se imprimen los datos en la consola del navegador, despliega el valor en los recuadros y actualiza la posición del objeto 3D.

AWS IoT

AWS IoT proporciona una comunicación segura y bidireccional entre dispositivos conectados a Internet, como sensores, actuadores, microcontroladores integrados o aparatos inteligentes, y la nube de AWS. Esto hace posible que se recopile datos de telemetría de múltiples dispositivos, se almacenen y analicen [19].

Servicios utilizados:

- Lambda
- DynamoDB
- IoT Core
- MQTT Topic
- IoT Rule
- IoT Action

Para el proyecto creamos un dispositivo dentro del núcleo de AWS IoT, generando certificados y políticas. Usando la IDE de Arduino y el protocolo MQTT conectamos el ESP8266 con AWS IoT usando las credenciales de AWS IoT Core cargadas directamente en la memoria Flash.

Tras realizar las lecturas de los sensores, vamos a almacenarlos en Amazon DynamoDB utilizando AWS IoT Core (MQTT) mandando los datos en formato JSON que contengan el `mac_Id` de ESP8266 y la posición del sensor.

Todo esto se representa con el siguiente diagrama de arquitectura mostrado en la figura 31.

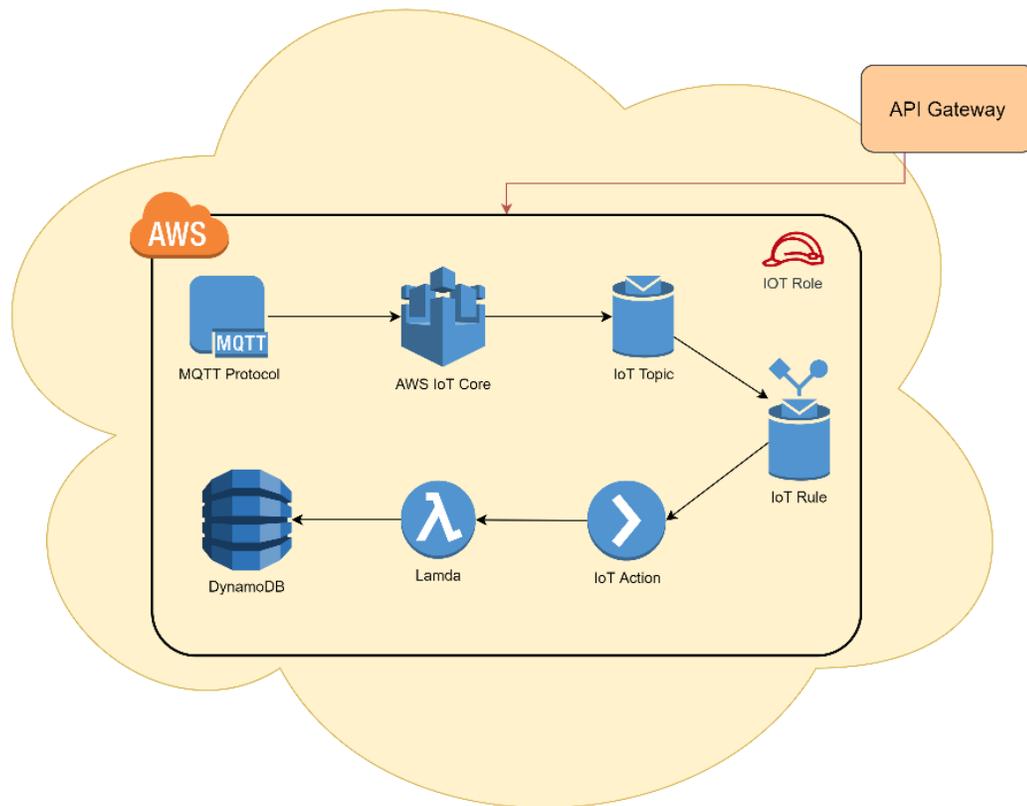


Figura 31 - Arquitectura de servicios de AWS

Antes de comenzar a detallar el proceso de implementación, es necesario entender algunos conceptos.

Dispositivo

Un dispositivo, en el contexto del Internet de las cosas (IoT), es una entidad u objeto físico que tiene un identificador único, un sistema integrado y la capacidad de transferir datos a través de una red [19].

Certificado de AWS IoT

Un dispositivo debe tener un certificado para autenticarse con AWS IoT. Los certificados X.509 se utilizan para autenticar el dispositivo, este certificado representa que el dispositivo en particular pertenece a nuestra cuenta de AWS. Existen dos métodos principales [19] para codificar los datos de los certificados:

- DER = Codificación binaria de los datos del certificado
- PEM = La codificación en base64 del certificado codificado en DER, con una cabecera y líneas de pie de página añadidas.

DER (Distinguished Encoding Rules)

DER es un subconjunto de la codificación BER que proporciona una forma de codificar un valor ASN 1 [19]. DER está pensado para situaciones en las que se necesita una codificación única, como en la criptografía, y garantiza que una estructura de datos que necesita ser firmada digitalmente produce una representación serializada única.

PEM (Privacy-enhanced Electronic Mail)

PEM es la abreviatura de Privacy Enhanced Mail (RFC 1421 - RFC 1424), una de las primeras normas para proteger el correo electrónico (IRTF, IETF). Es un certificado DER codificado en US-ASCII por base64, solicitud de certificado o PKCS#7, encerrado entre delimitadores PEM típicos [19].

Política en AWS IoT

Las políticas de AWS IoT se utilizan para autorizar a un dispositivo a realizar operaciones de AWS IoT, como suscribirse o publicar en temas de MQTT [20].

Regla de AWS IoT

Una regla de AWS IoT consiste en una sentencia SQL SELECT, un filtro de temas y una acción de regla. Los dispositivos envían información a AWS IoT mediante la publicación de mensajes en temas MQTT. La sentencia SQL SELECT permite extraer datos de un mensaje MQTT entrante [20]. La regla se activa cuando se recibe un mensaje MQTT que coincide con el filtro de temas en un tema.

Acciones de AWS IoT

Las acciones permiten tomar la información extraída de un mensaje MQTT y enviarla a otro servicio de AWS. Se definen para servicios de AWS como Amazon DynamoDB, AWS Lambda, Amazon SNS y Amazon S3 [21].

Creación de una cosa/dispositivo en el núcleo de AWS IoT

Como muestra la figura 32, en el panel de navegación dentro de AWS IoT Console, elegimos Manage, luego Register a thing.

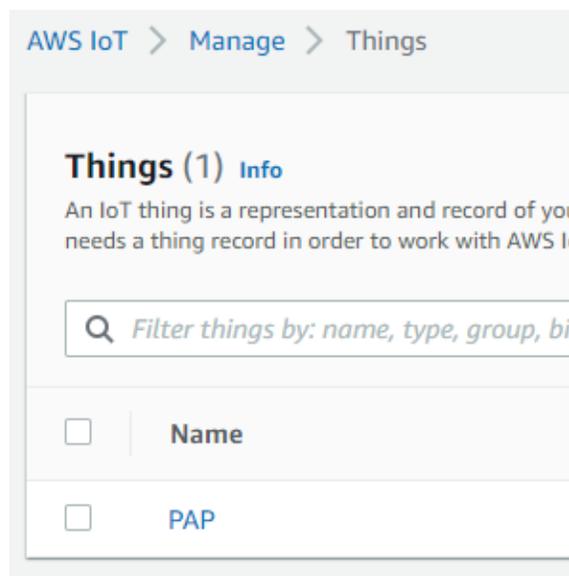


Figura 32 - Dispositivo registrado "PAP"

Convertimos los certificados a formato DER antes de cargarlos a la memoria flash usando OpenSSL. En la carpeta con los certificados, usaremos los siguientes comandos para convertir los archivos [20].

- > openssl x509 -in xxxxxxxxxxx-certificate.pem.crt -out cert.der -outform DER
- > openssl rsa -in xxxxxxxxxxx-private.pem.key -out private.der -outform DER
- > openssl x509 -in AmazonRootCA1.pem -out ca.der -outform DER

Creación de políticas

En el panel de navegación de la consola de AWS IoT, elegimos Secure, y luego Políticas. La figura 35 muestra la política de privacidad para el dispositivo.

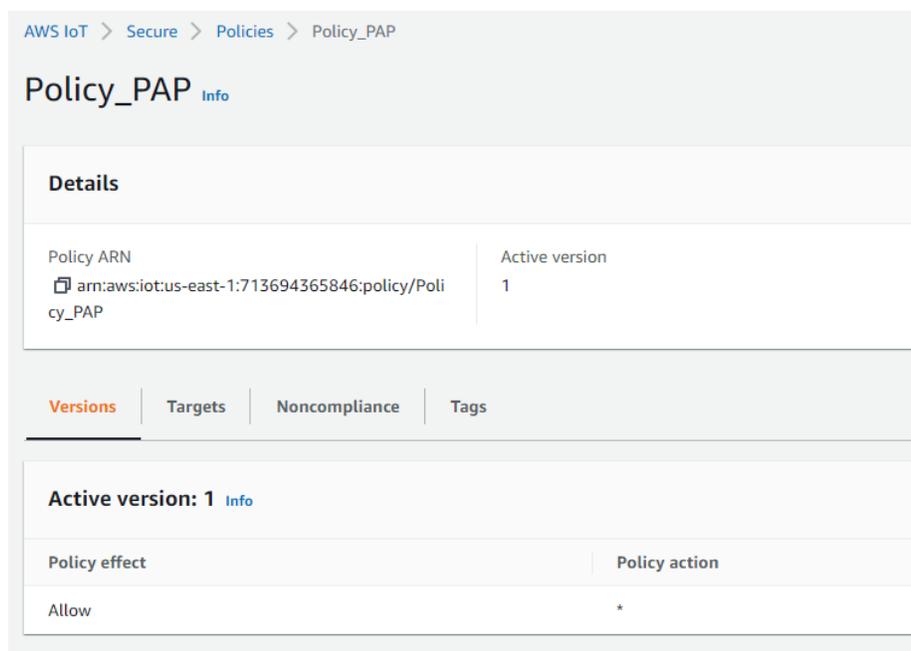


Figura 35 - Política de privacidad para el dispositivo PAP

En Acción colocamos “iot:* “ y en Resource “ ARN, * “

Esta política permite que el dispositivo realice todas las acciones de AWS IoT en todos los recursos de AWS IoT. Esta configuración es demasiado permisiva y debemos cambiar en un entorno de producción para reducir posibles amenazas a la seguridad y uso de recursos.

Creación del endpoint para ESP8266

En el código del ESP8266 cambiamos el `AWS_endpoint` mostrado en la figura 36 —que es la dirección del broker MQTT— por el de la cuenta AWS en una región específica.

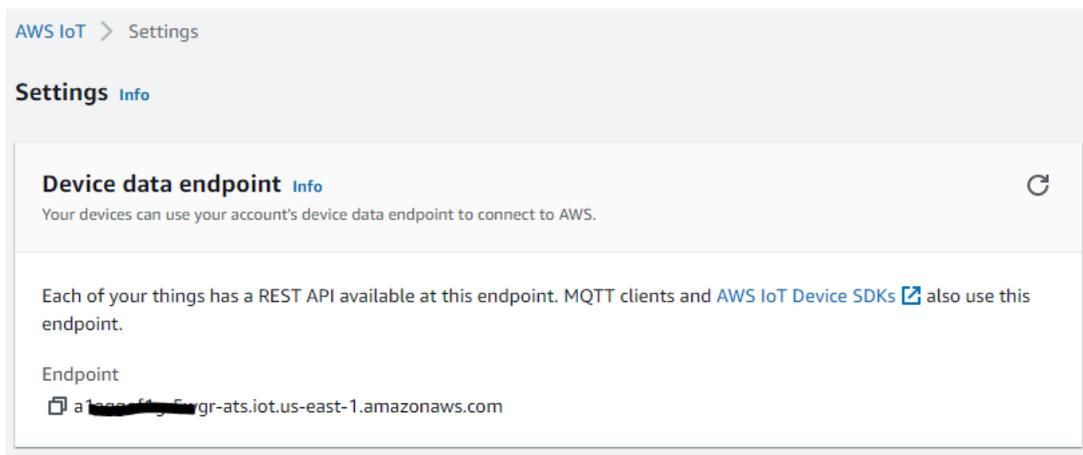


Figura 36 - Endpoint de la cuenta de AWS IoT

Publicar a un Topic desde ESP8266

Dentro del código del ESP8266 nos aseguramos de mandar los mensajes al tópic "outTopic", como muestra la figura 37.

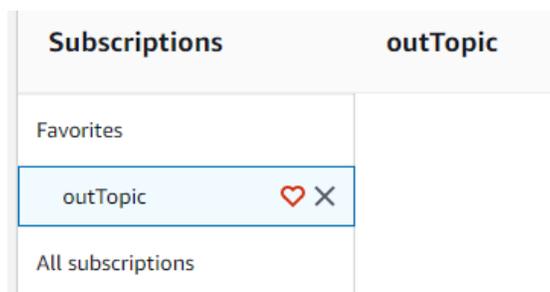


Figura 37 - Suscripción al tópic "OutTopic" desde la consola de AWS IoT

Creación de una regla para Amazon DynamoDB.

Las reglas de DynamoDB permiten tomar información de un mensaje MQTT entrante y escribirla en una tabla de DynamoDB.

En la consola de AWS IoT, en el panel de navegación, seleccionamos Act. En la página Reglas, elegimos Crear una regla. La figura 38 muestra la regla creada.

The screenshot displays the AWS IoT console interface for a rule named 'PAP_rule'. The breadcrumb navigation at the top reads 'AWS IoT > Act > Rules > PAP_rule'. Below the title 'PAP_rule' with an 'Info' link, there is a 'Details' section. This section includes a 'Description' field with a hyphen, an 'ARN' field with the value 'arn:aws:iot:us-east-1:713694365846:rule/PAP_rule', and a 'Status' field with a green checkmark and the text 'Active'. To the right, the 'Topic' field is set to 'outTopic' with a link icon, and the 'Basic ingest topic' is '\$aws/rules/PAP_rule'. Below the details is the 'SQL statement' section, which shows the SQL query 'SELECT *,timestamp() AS ts FROM 'outTopic'' and the 'SQL version' '2016-03-23'. At the bottom, there are tabs for 'Actions', 'Error action', and 'Tags'. The 'Actions' tab is active and shows one action: 'DynamoDBv2' with the action type 'Split message in'.

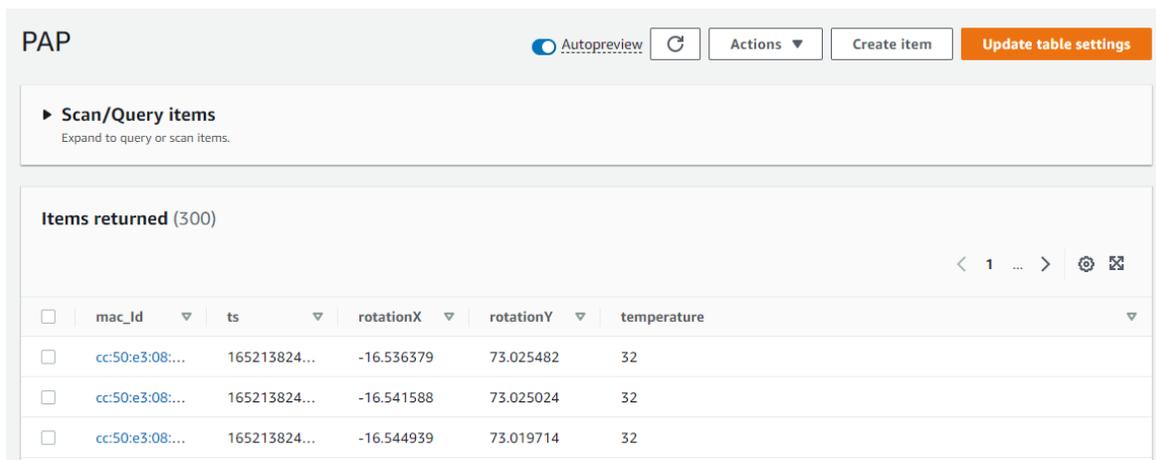
Figura 38 - Regla en AWS IoT para mandar los datos del tópic a una tabla DynamoDB

En Rule query statement, introducimos el siguiente query statement: `select *, timestamp() AS ts from 'outTopic'`

"select *" especifica que se quiere enviar todo el mensaje MQTT que activó la regla & "timestamp() AS ts" especifica que "timestamp()" se copiará en "ts" "from 'outTopic'" indica al motor de reglas que active esta regla cuando se reciba un mensaje MQTT cuyo tema coincida con este filtro de temas.

Creación de una tabla en Amazon DynamoDB

Creamos una base de datos en DynamoDB y la asociamos a la regla de DynamoDB que acabamos de crear. La figura 39 muestra la tabla en DynamoDB.



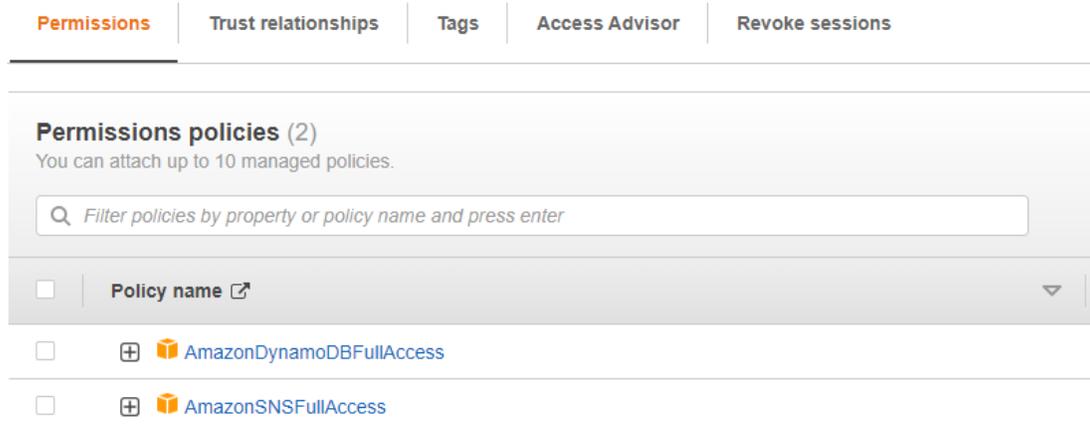
The screenshot shows the Amazon DynamoDB console interface for a table named 'PAP'. At the top, there are buttons for 'Autopreview', 'Actions', 'Create item', and 'Update table settings'. Below this, there is a section for 'Scan/Query items' and a table of 'Items returned (300)'. The table has the following columns: mac_id, ts, rotationX, rotationY, and temperature. Three items are listed:

	mac_id	ts	rotationX	rotationY	temperature
<input type="checkbox"/>	cc:50:e3:08:...	165213824...	-16.536379	73.025482	32
<input type="checkbox"/>	cc:50:e3:08:...	165213824...	-16.541588	73.025024	32
<input type="checkbox"/>	cc:50:e3:08:...	165213824...	-16.544939	73.019714	32

Figura 39 - Tabla en DynamoDB para guardar la posición de los sensores

Creación de un nuevo rol de AWS

Por último, todos los servicios deben estar administrados por un rol que solamente tenga acceso a los servicios elegidos. Con esto minimizamos el alcance y gravedad de un ataque. La figura 40 muestra un resumen del rol creado.



The screenshot shows the AWS IAM console interface for a role. The 'Permissions' tab is selected, showing 'Permissions policies (2)'. Below this, there is a search bar and a list of two policies:

	Policy name
<input type="checkbox"/>	AmazonDynamoDBFullAccess
<input type="checkbox"/>	AmazonSNSFullAccess

Figura 40 - Rol específico para el dispositivo "PAP" con acceso solamente a los servicios conectados

Propuestas de Hardware

Para obtener las mediciones con mayor exactitud se requiere que el sensor se encuentre colocado en las posiciones de la figura 7 de forma que sea sencillo de ponerse, que no se resbale mientras se camina y que no cause molestias a la persona que los esté usando.

Se diseñó el modelo 3D, mostrado en la figura 41, que cumple con las características antes mencionadas.

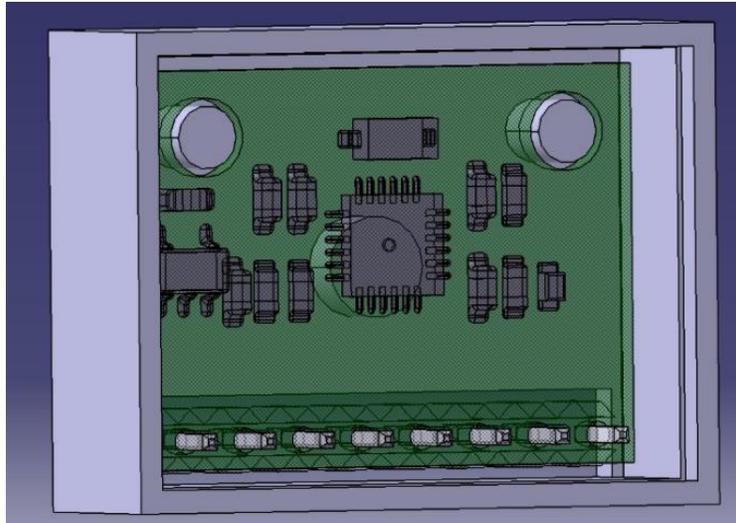


Figura 41 - Caja para colocar el sensor MPU6050

Está diseñado de forma que tenga el tamaño exacto de la IMU, tiene en sus costados unas agarraderas, como muestra la figura 42, por las que va a colocarse un elástico de forma que pueda colocarse y quitarse con facilidad, sea reutilizable para varios tamaños de piernas y no cause incomodidad.

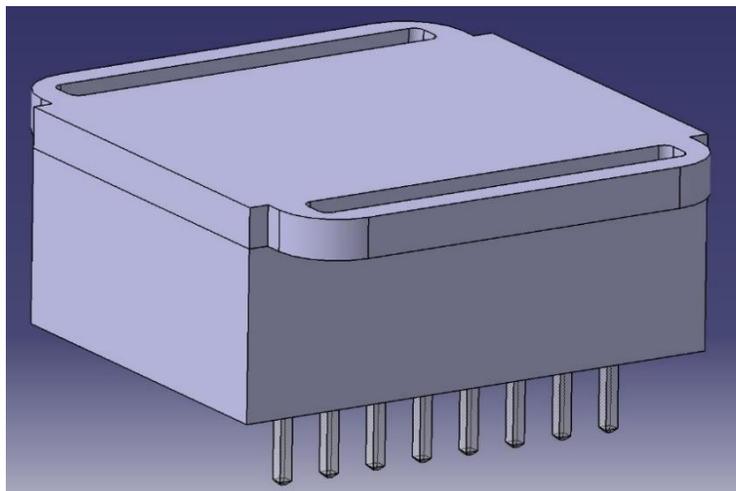


Figura 42 - Caja completa para colocar el sensor MPU6050

Tras imprimirlos podemos colocar el sensor MPU6050 cuidando que la orientación en ambos nodos sea la misma. Por la abertura del contenedor podemos sacar el cableado necesario que interconectará los nodos. La figura 43 muestra la forma en la que se debe orientar el ensamble.

La tapa cuenta con dos agarraderas, pasamos un elástico y en sus orillas colocamos pedazos de velcro para que sea sencillo colocar y quitar el nodo.

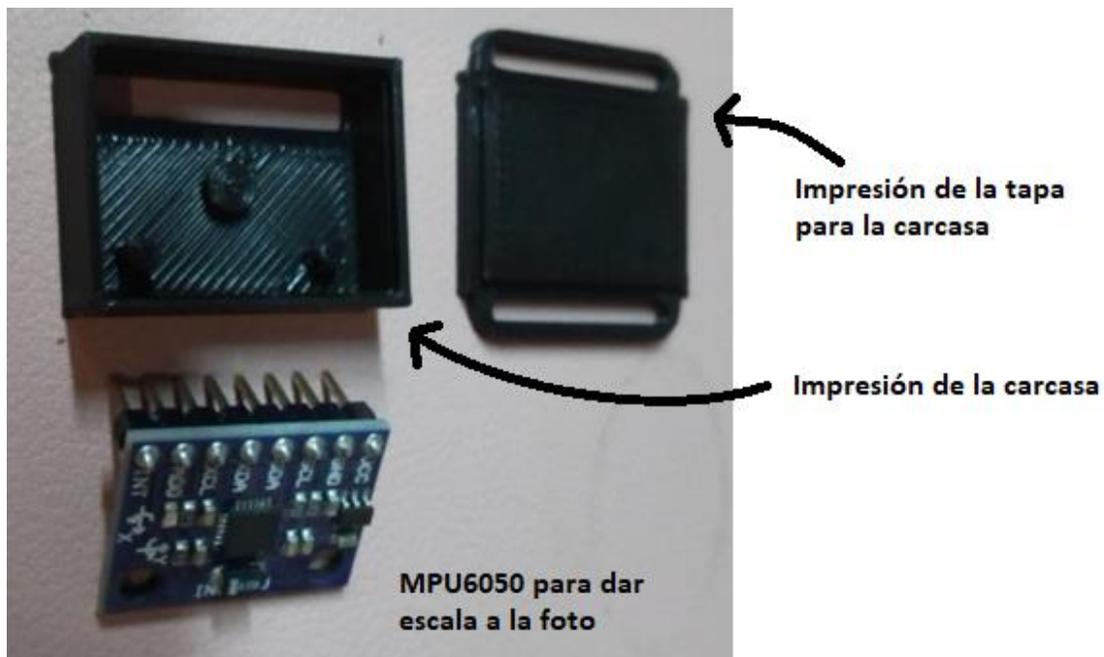


Figura 43 - Cotenedor para el seno MPU6050 impreso

Una característica importante del proyecto es su modularidad, facilidad de replicación y capacidad de llevarse a diferentes entornos. Por lo que decidimos crear una caja que permita guardar el módulo ESP8266, su respectivo PCB con el objetivo de llevarla sobre el cuerpo.

El diseño cuenta con una abertura para colocar el ESP8266 por fuera de la caja y no tengamos obstrucciones para la señal Wifi, como muestra la figura 44. En la parte superior tenemos un gancho para colocar una porta tarjetas y colocar sobre el cinturón.

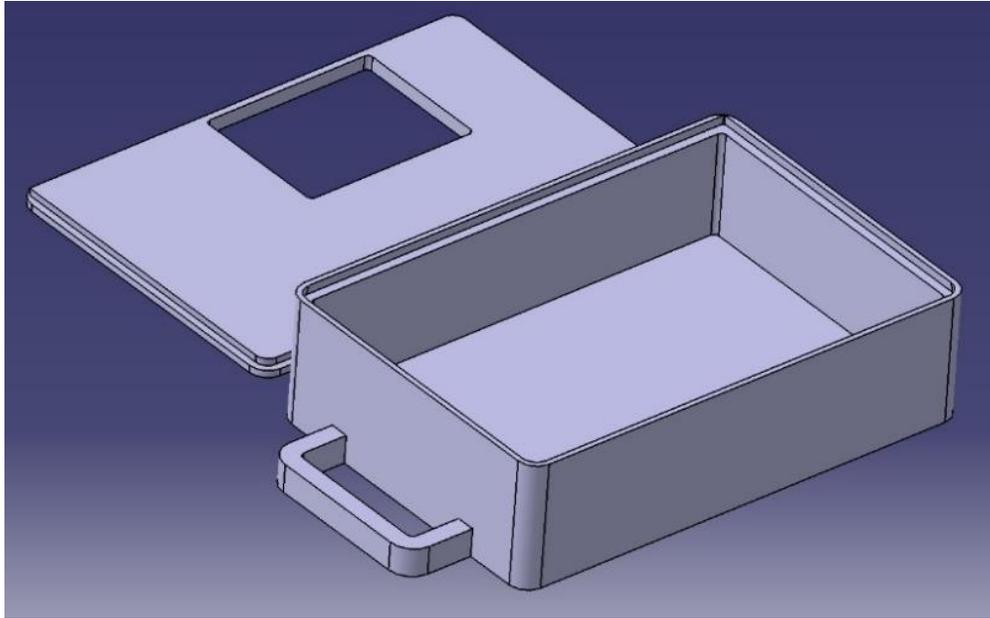


Figura 44 - Modelo 3D para la caja que contendrá el ESP8266 y su respectivo PCB

Tras imprimirse, montamos los PCB en ellas, sacamos los cables necesarios y conectamos entre ellos. La figura 45 muestra los contenedores impresos.

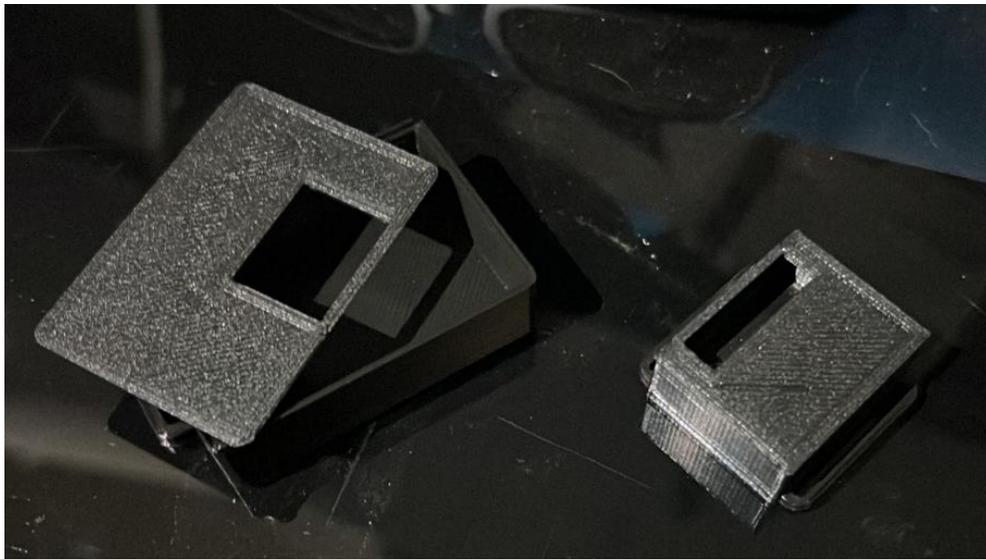


Figura 45 - Impresión de los modelos 3D sobre filamento negro

1.6. Valoración de productos, resultados e impactos

La experiencia PAP nos permitió situarnos bajo el contexto de un investigador y arquitecto de soluciones por lo que pudimos poner en práctica lo que hemos aprendido en nuestras respectivas carreras. El reto fue integrar distintas habilidades, como extrapolar los servicios en la nube a la conexión Wifi de una tarjeta de desarrollo o la instrumentación y el prototipado siempre teniendo en mente los objetivos. Nuestros conocimientos de electrónica digital, prototipado enfocado en dispositivos de IoT, despliegue de datos en tableros e instrumentación ayudaron a que el proyecto se realizara eficientemente.

Para validar como se dio respuesta a la problemática planteada en el capítulo 1 y la calidad de los resultados, vamos a presentar los resultados por separado de las áreas más importantes del prototipo final y una demostración del prototipo final funcionando.

Validación del cálculo de ángulos

Comenzamos comprobando que el ruido innato del sensor se elimine tras pasar por el filtro complementario. En la figura 46, tenemos el sensor MPU6050 sobre una superficie plana sin movimiento.

Los datos en color naranja muestran los valores brutos sobre el ángulo del eje X, los datos en rojo muestran los valores tras ser filtrados. Es claro que el ruido natural del sensor se elimina sin problema alguno.

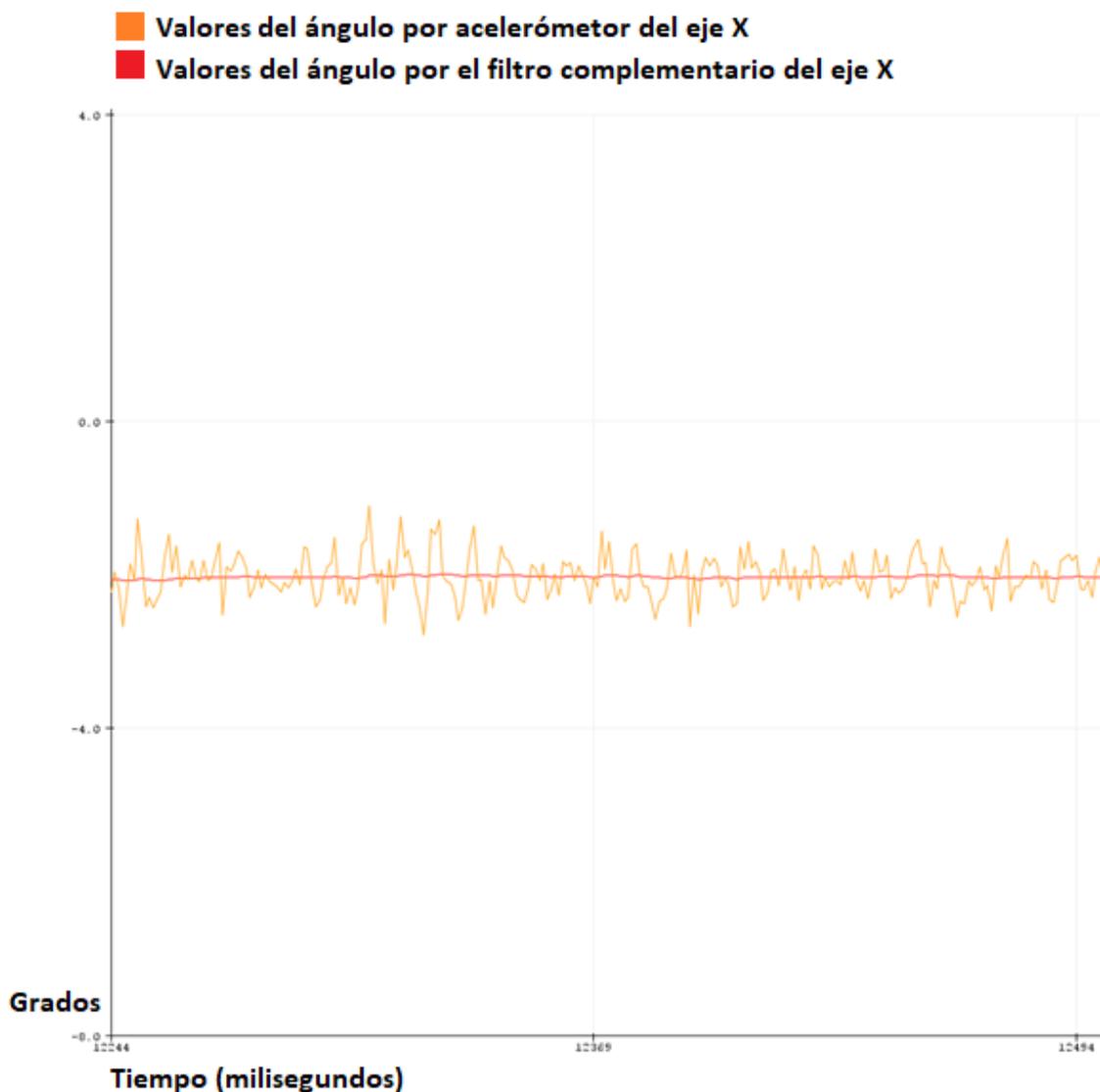


Figura 46 - Valores brutos contra valores filtrados cuando el sensor se encuentra sin movimiento en una superficie plana

Para caracterizar el filtro complementario ponemos como entrada un escalón unitario, con esto se pretende medir su tiempo de estabilización, es decir, cuanto se tarda el sistema en alcanzar su estado estacionario.

En la figura 47 vemos que el escalón comienza aproximadamente a los 5060 milisegundos de haber comenzado a graficar. La salida del filtro se estabiliza a los 5332 milisegundos, con esto podemos asumir que tenemos un tiempo de estabilización de 272 milisegundo aproximadamente.

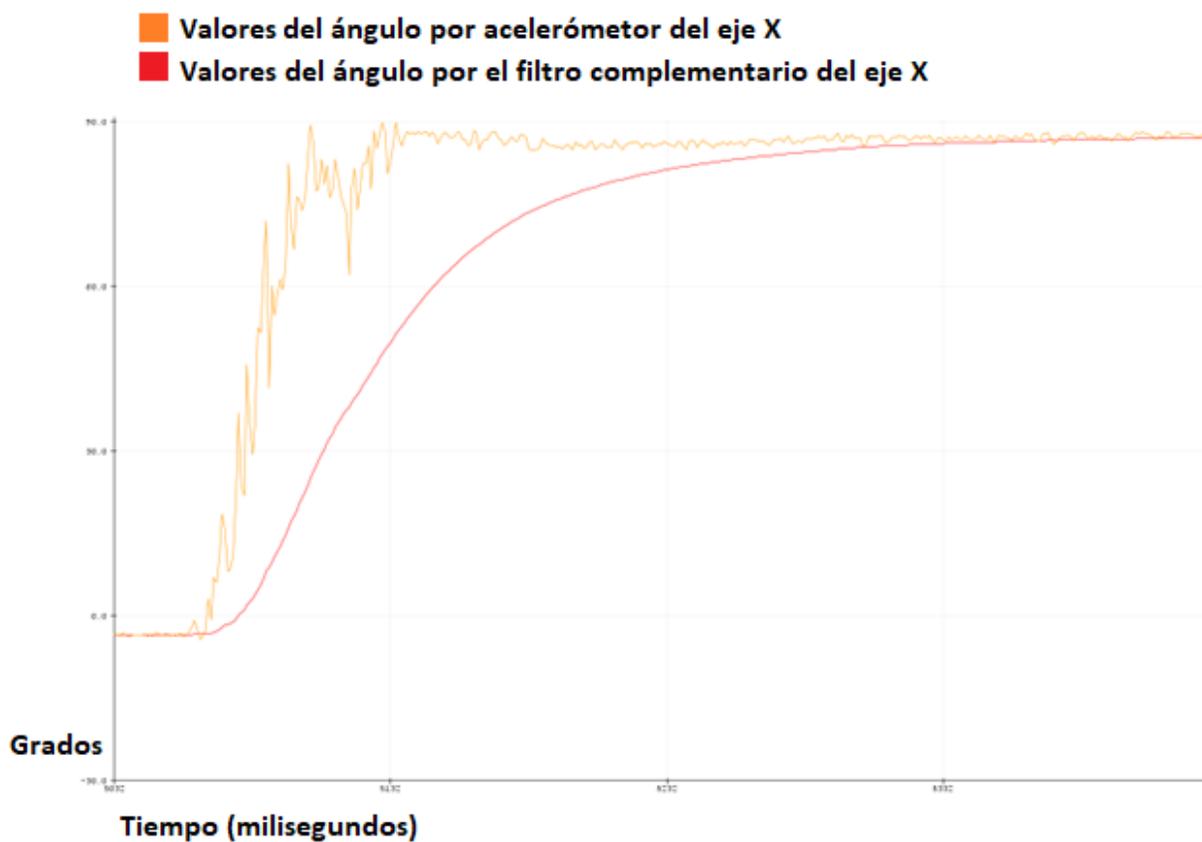


Figura 47 - Respuesta del filtro complementario a un escalón unitario

Por último, para caracterizar la respuesta del filtro frente a cambios abruptos usamos una senoidal a la entrada.

La respuesta mostrada en la figura 48 nos indica que tenemos un ligero retraso y nos podríamos beneficiar de un amplificador. También podemos concluir que ante cambios bruscos es posible que tengamos una pérdida de datos.

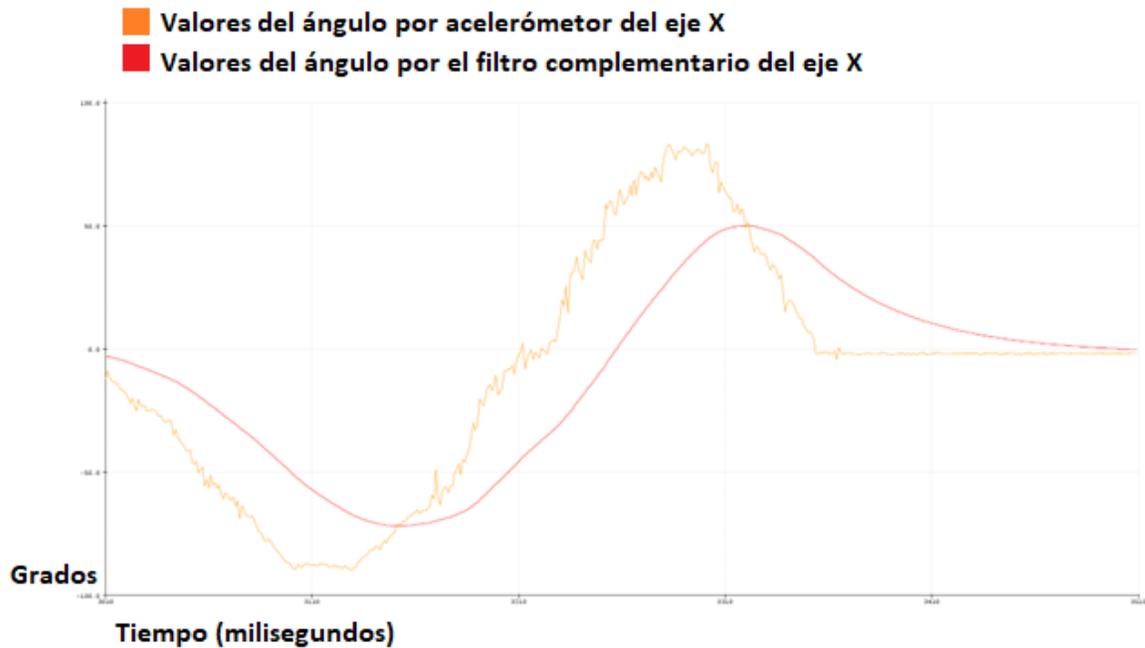


Figura 48 - Respuesta del filtro complementario a una senoidal

Validación del montaje del prototipo final sobre el cuerpo

Los nodos que contienen MPU6050 se montan en la pierna ajustando los elásticos, como muestra la figura 49. Se debe de prestar especial atención en que los sensores tengan la misma posición inicial y se encuentren sobre el costado externo de la pierna.

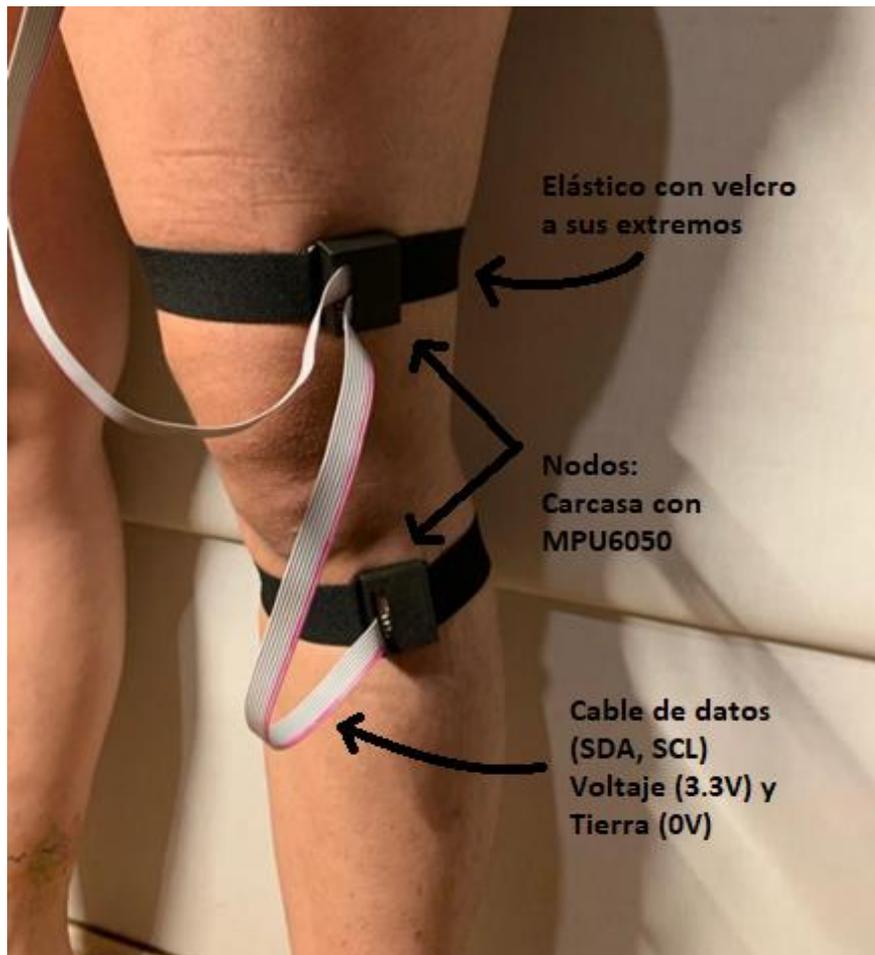


Figura 49 - Nodos posicionados correctamente en la rodilla izquierda

Del nodo superior sale un cable que contiene el bus de comunicación I2C, Voltaje (3.3 V) y tierra que se conectan al ESP8266.

El ESP8266 se coloca en la cadera usando un clip para tarjetas. No importa si se inclina o mueve ya que no tiene restricciones sobre su posición, únicamente tenemos que cuidar no tener imanes cerca de este nodo.

En la parte lateral izquierda tenemos 2 cables sin conexión, estos son TX y RX usados para hacer depuración del sistema, en la parte lateral derecha tenemos Voltaje y tierra los cuales se conectan a una fuente USB de 3.3V

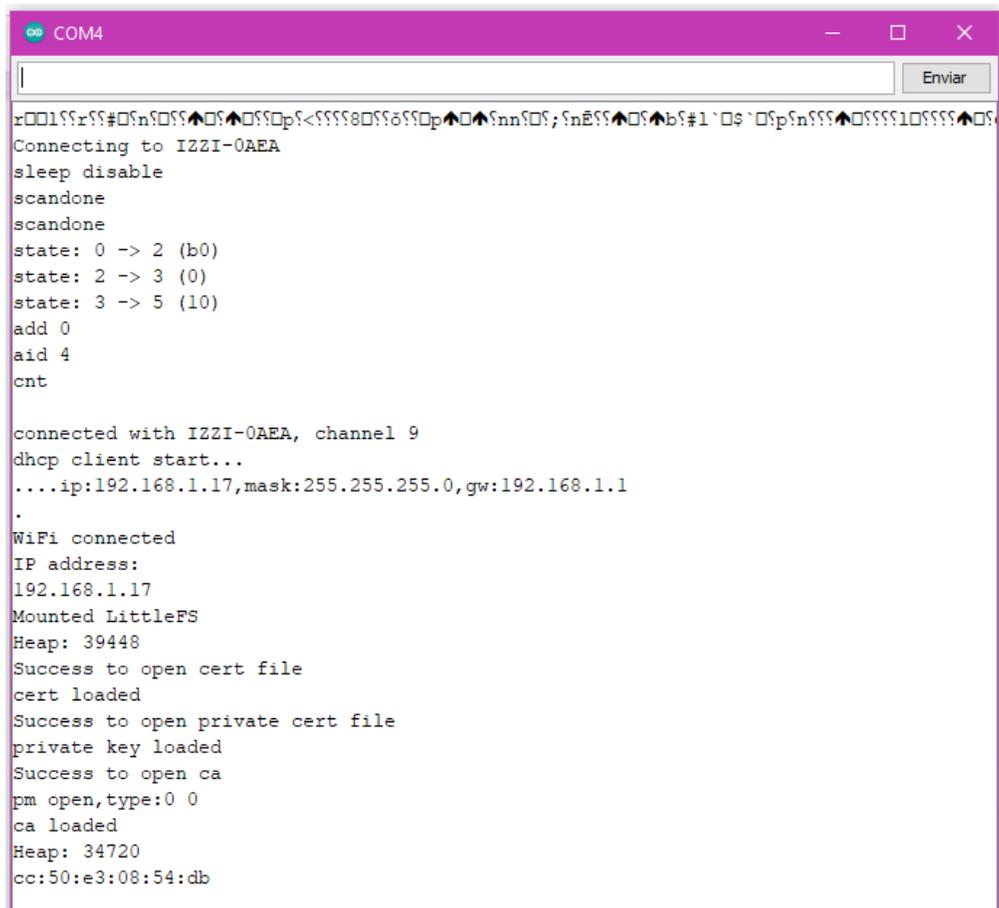


Figura 50 - Prototipo final correctamente montado sobre la extremidad inferior izquierda

Como muestra la figura 50, una vez montado el prototipo final, se tiene los nodos que contienen los MPU6050 cerca de la rodilla, y el ESP8266 en la cadera.

Validación del servidor local desde el ESP8266

Tras cargar el código y certificados al ESP8266, al encender el dispositivo nos conectamos a una red Wifi desde donde se despliega un servidor local y se conecta a los servicios de AWS, como muestra la figura 51.



```
COM4
Connecting to IZZI-0AEA
sleep disable
scandone
scandone
state: 0 -> 2 (b0)
state: 2 -> 3 (0)
state: 3 -> 5 (10)
add 0
aid 4
cnt

connected with IZZI-0AEA, channel 9
dhcp client start...
...ip:192.168.1.17,mask:255.255.255.0,gw:192.168.1.1
.
WiFi connected
IP address:
192.168.1.17
Mounted LittleFS
Heap: 39448
Success to open cert file
cert loaded
Success to open private cert file
private key loaded
Success to open ca
pm open,type:0 0
ca loaded
Heap: 34720
cc:50:e3:08:54:db
```

Figura 51 - Salida serial del sistema con la información de la conexión Wifi, servidor levantado y carga de certificados AWS

Desde la consola del servidor local mostrada en la figura 52, vemos los datos que estamos leyendo del sensor en formato JSON.

```
8 gyro_readings script.js:118
{"rotationX": "-16.56", "rotationY": "72.93", "temperature": "29"}
gyro_readings script.js:118
{"rotationX": "-16.56", "rotationY": "72.92", "temperature": "29"}
6 gyro_readings script.js:118
{"rotationX": "-16.57", "rotationY": "72.92", "temperature": "29"}
gyro_readings script.js:118
{"rotationX": "-16.56", "rotationY": "72.92", "temperature": "29"}
```

Figura 52 - Datos obtenidos de un solo nodo MPU6050 en formato JSON al llegar al servidor y desplegarse en la consola

Como vemos en la figura 53, el servidor se desplegó en la dirección ip 192.168.1.17 por lo que nos conectamos a esta desde cualquier browser.

Aquí podemos ver como se despliegan los datos de un solo nodo, se mandan a una simulación y, al mismo tiempo, a MQTT.

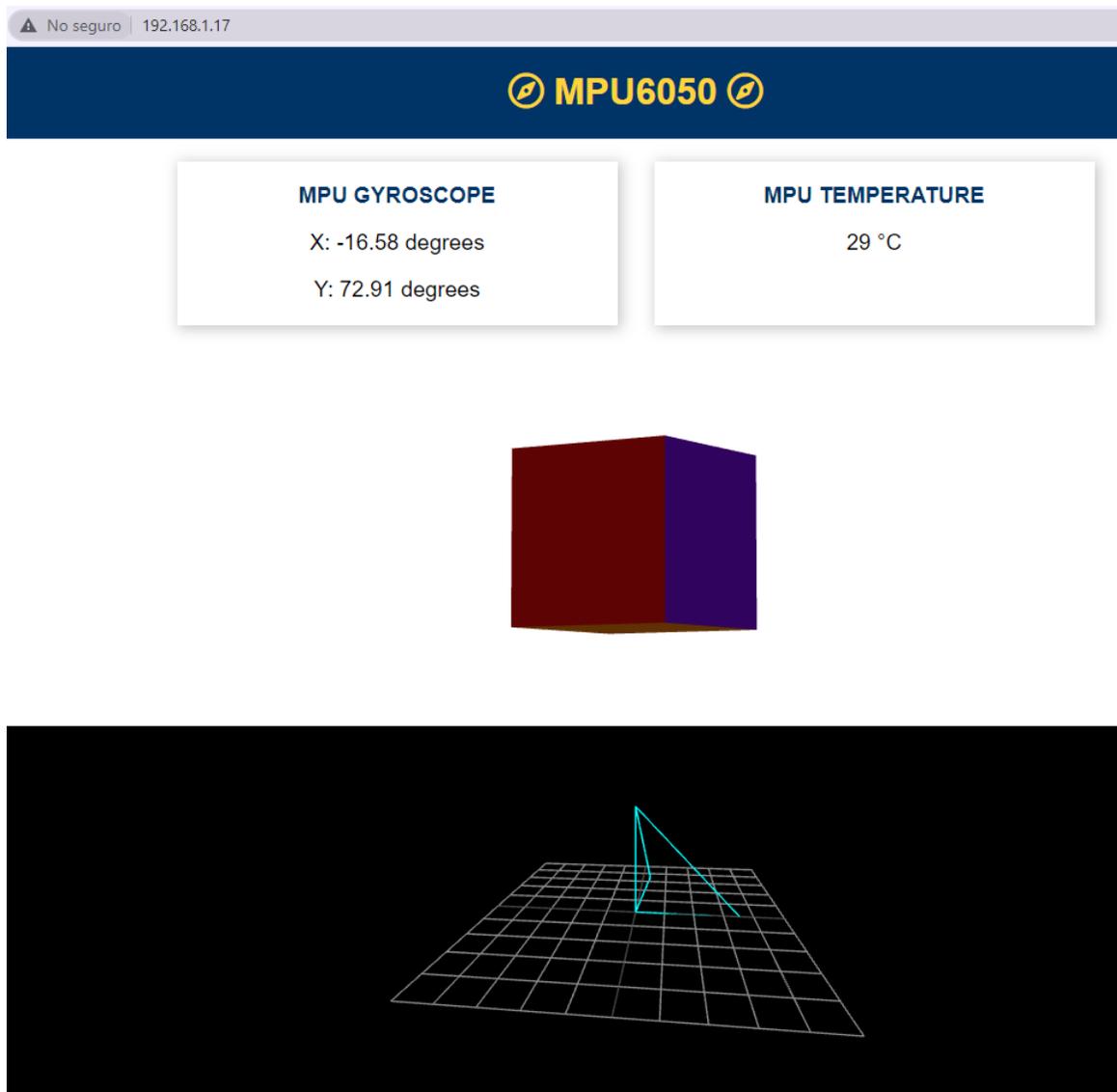


Figura 53 - Interfaz desplegando datos de un solo nodo desde la dirección 192.168.1.17

Validación de MQTT suscribiéndonos al tópico "outTopic" en AWS IoT Core

En AWS IoT nos suscribimos al tema "outTopic". Es importante señalar que el dispositivo debe estar conectado a la red WiFi, pasar la autenticación de AWS y contar con los permisos para conectarse y publicar en un tema IoT.

Si cumplimos todos estos requisitos entonces podemos ver los mensajes que mandó el servidor en el cliente MQTT mostrados en la figura 54.

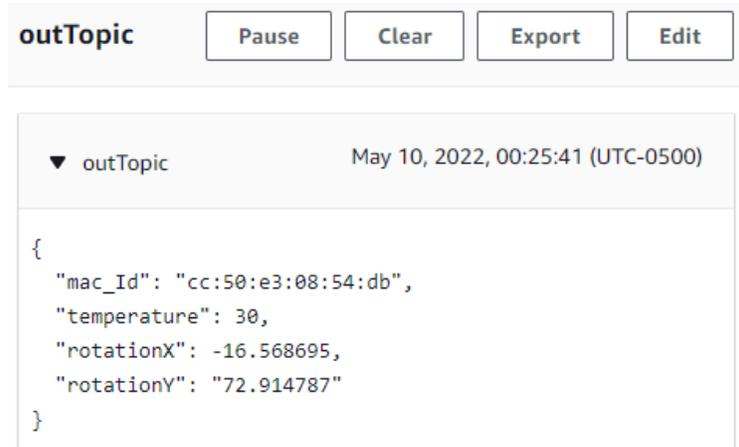


Figura 54 - Mensajes desplegados como JSON del tópico "outTopic" en la consola AWS IoT Core

Validación del servicio Lambda para mandar los datos a una tabla de DynamoDB

Al momento de recibir un mensaje en el tópico "outTopic", este es publicado en la tabla de DynamoDB, mostrada en la figura 55, como un dato agregado en las columnas particulares.

<input type="checkbox"/>	mac_Id	ts	rotationX	rotationY	temperature
<input type="checkbox"/>	cc:50:e3:08:...	165213824...	-16.536379	73.025482	32
<input type="checkbox"/>	cc:50:e3:08:...	165213824...	-16.541588	73.025024	32
<input type="checkbox"/>	cc:50:e3:08:...	165213824...	-16.544939	73.019714	32
<input type="checkbox"/>	cc:50:e3:08:...	165213825...	-16.553488	73.013565	32
<input type="checkbox"/>	cc:50:e3:08:...	165213825...	-16.562443	73.004906	32
<input type="checkbox"/>	cc:50:e3:08:...	165213825...	-16.560247	73.004356	32
<input type="checkbox"/>	cc:50:e3:08:...	165213825...	-16.56209	73.005699	32

Figura 55 - Datos guardados en la tabla DynamoDB

Validación del prototipo final para obtener los ángulos de una rodilla extendida

Con el sistema conectado, colocamos los nodos en su posición correcta sobre la rodilla izquierda y mantenemos una postura de pie procurando mantener la extremidad extendida, como muestra la figura 56.



Figura 56 - Posición de la pierna extendida con los nodos colocados

En la interfaz vamos a ver que del nodo superior nos retorna una inclinación de -63.3 grados en X y 23.67 grados en Y. En el nodo inferior vemos una inclinación de -65.84 grados en X y 23.3 grados en Y.

Con esta posición obtenemos la simulación de la figura 57.

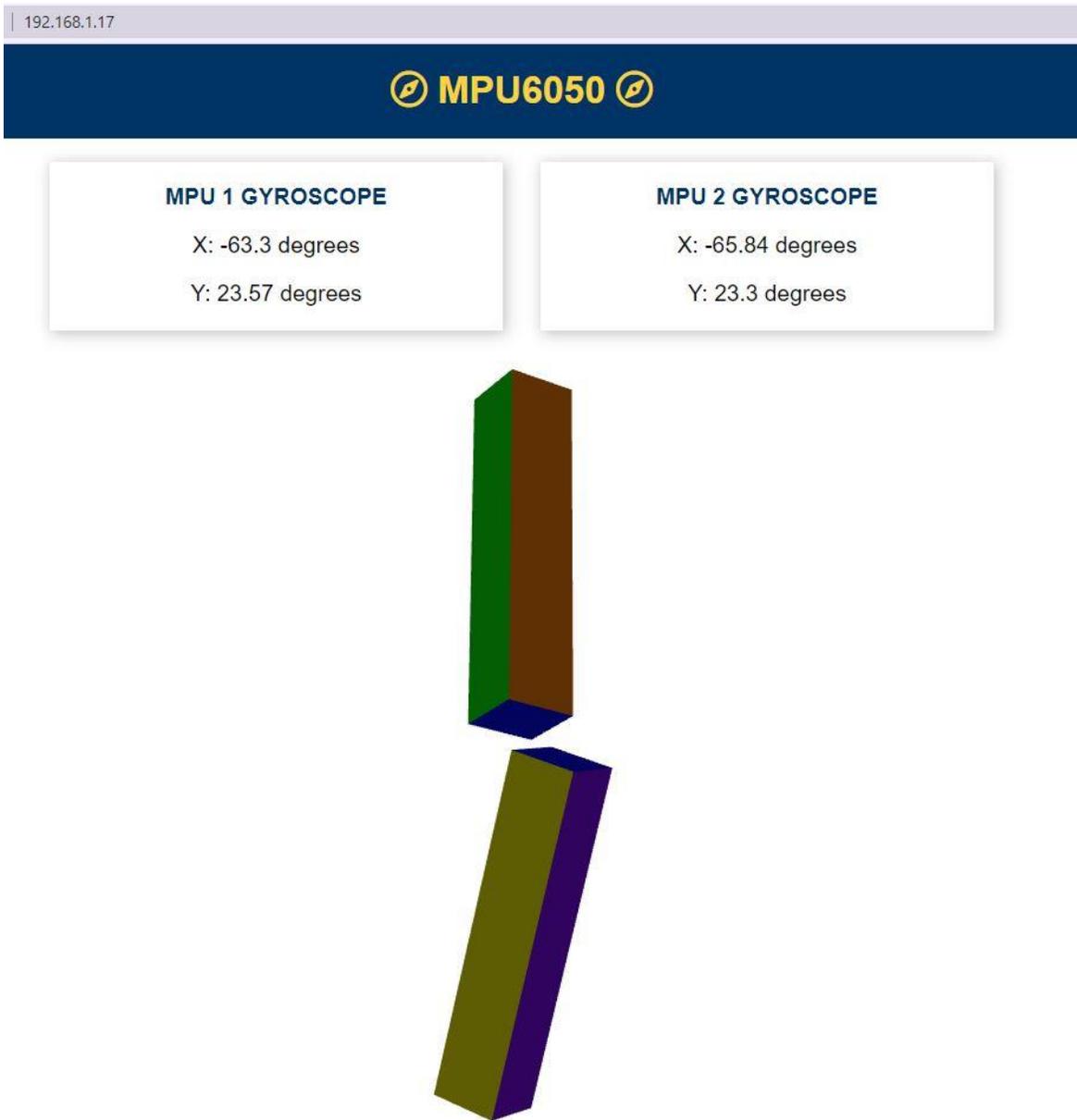


Figura 57 - Simulación y despliegue de datos de una pierna extendida

Validación del prototipo final para obtener los ángulos de una rodilla en sentadilla

Con el sistema conectado, colocamos los nodos en su posición correcta sobre la rodilla izquierda y mantenemos una postura de sentadilla procurando mantener la rodilla sin sobrepasar la punta del pie, demostrado por la figura 58.



Figura 58 - Posición de la pierna en sentadilla con los nodos colocados

En la interfaz vamos a ver que del nodo superior nos retorna una inclinación de -67.58 grados en X y 21.29 grados en Y. En el nodo inferior vemos una inclinación de -39.46 grados en X y 46.76 grados en Y.

Con esta posición obtenemos la simulación de la figura 59.

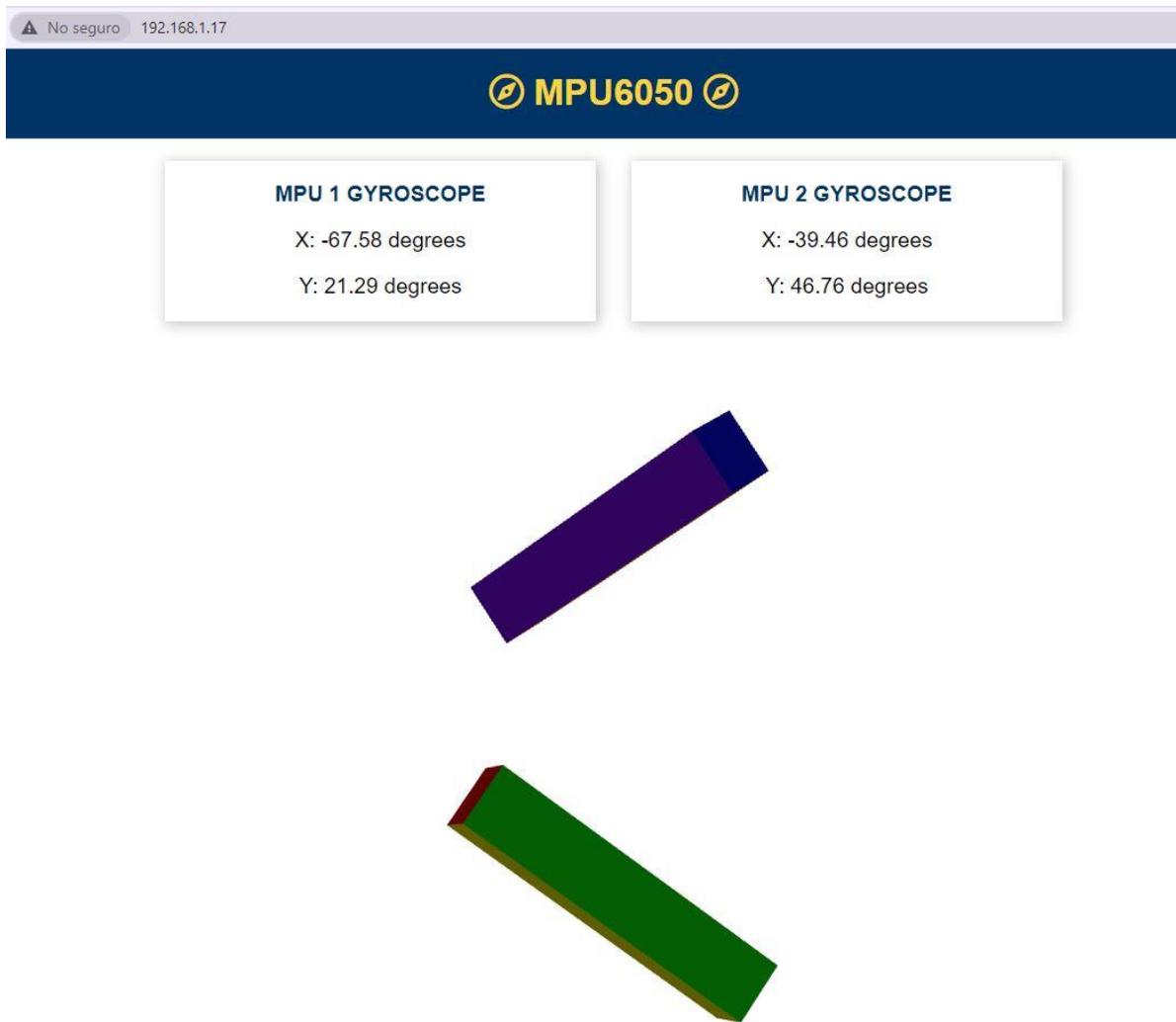


Figura 59 - Simulación y despliegue de datos de una pierna en sentadilla

Los resultados obtenidos con el prototipo final son satisfactorios ya que cumplen con los objetivos planteados, responde a la problemática planteada en el capítulo 1 y, lo más importante, permiten ver con claridad el movimiento en la simulación. A pesar de eso, tenemos las siguientes áreas de mejora que podrán ser abordadas en una segunda etapa del proyecto.

Movimiento en miembro inferior basado en patrones de marcha

Ya que nuestro objetivo se centra en el análisis del movimiento de rodilla, es importante conocer la inercia natural que esta tiene para usarla como modelo inicial de validación y como base teórica. Esta es uno de los temas no explorados pero que son necesarios si se quiere usar este sistema en el ámbito de la rehabilitación.

Fuente de poder del prototipo

Actualmente el prototipo utiliza una fuente para tablero de circuito, el cual convierte una entrada USB a voltaje 3.3 V. El problema es que esto nos limita en cuestión de movilidad ya que debemos tener un cable USB largo o estar cerca de un enchufe. La tarjeta de poder usada es mostrada en la figura 60.



Figura 60 - Tarjeta de poder MB102 DC3.3V/5V [22]

La propuesta para siguientes periodos PAP es conseguir una batería recargable de litio, esto implicaría modificar el PCB y caja 3D.

Velocidad de simulación

Este es un problema que el sistema actual tiene, como se están mandando constantemente valores al servidor y al MQTT, se tienen que mandar con cierto delay para evitar sobrecargar el servidor y tumbarlo.

Calibración de los sensores MPU6050

El proceso actual de calibración es un paso que se lleva a cabo totalmente aparte del sistema final ya que toma aproximadamente 5min por sensor y el script es bastante pesado. Se debe de implementar un proceso de calibración que nos permita recalibrar ya con el sistema integrado en la pierna y que sea menos tardado.

Largo del bus físico para el protocolo I2C

El protocolo I2C nos permite tener hasta 128 dispositivos sobre el mismo bus, el problema está en la integridad de la señal tras ser transmitida por cables de tamaño largo. Se puede corregir este error colocando cables de par trenzado para conectar el bus y revisando que la capacitancia que estos tengan sea baja.

1.7. Bibliografía y otros recursos

- [1] L. Kong, «CyberTherapy & Rehabilitation,» 5 Junio 2014. [En línea]. Available: <http://www.cybertherapyandrehabilitation.com/2014/06/japan-leading-way-elderly-care-rehabilitation-robots/>.
- [2] N. Maclean, P. Pound, C. Wolfe y A. Rudd, «The concept of patient motivation: a qualitative analysis of stroke professionals' attitudes,» Department of Public Health Sciences, Guy's, King's and St Thomas' School of Medicine, London, UK, 2002.
- [3] J. A. Acevedo Londoño, E. Caicedo Bravo y J. F. Castillo García, «Application of robotics rehabilitation technologies in children with upper limb disabilities,» *Revista de la Universidad Industrial de Santander*, vol. 49, n° 1, pp. 103-114, 2017.
- [4] D. Bravo, C. Rengifo y W. Agredo, «Comparison of two Motion Capture Systems by means of joint Trajectories of human gait,» *Revista mexicana de Ingeniería biomédica*, vol. 37, n° 2, 2016.
- [5] L. L. G. Echeverry, A. M. Jaramillo Henao, M. A. Ruiz Molina, S. M. Velásquez Restrepo, C. A. Páramo Velásquez y G. J. Silva Bolívar, «Human motion capture and analysis systems: a systematic review,» *Prospect*, vol. 16, n° 2, pp. 24-34, Diciembre 2018.
- [6] M. Gómez, J. N. Anzola y D. Barrero, «Sistemas de Captura de Movimiento - EXOCAP,» *Loop de Animación*, pp. 2-3, 2003.
- [7] N. Requena, «Captura de movimiento: historia y análisis en los videojuegos,» *MeriStation*, p. 1, 2017.
- [8] V. Venet, H. Pinard, J. McIntyre y A. Berthoz, «The Kinelite Project: a new powerful motion analyser for Spacelab and space station.,» *Acta Astronautica*, vol. 43, n° 3, pp. 277-289, 1998.
- [9] R. Rohr, «Towards Model-Based Recognition of Human Movements in Image Sequences,» *CVGIP: Image Understanding*, vol. 59, n° 1, pp. 94-115, 1994.
- [10] Organización Mundial de la Salud, «Discapacidad y salud,» 24 11 2021. [En línea]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/disability-and-health>.
- [11] A. Rodriguez-Angeles, J. L. Guzman-Gutierrez y C. Cruz-Villar, «User wearable interface based on inertial sensors for unilateral master-slave robot teleoperation,» de *7th International Conference on Electrical Engineering, Computing Science and Automatic Control*, Tuxtla Gutiérrez, Chiapas, México, 2017.
- [12] InvenSense, MPU-6000 and MPU-6050 Register Map and Descriptions Revision 4.2, California, USA: InvenSense, 2013.
- [13] M. Pérez, Diseño, Análisis y Modelado Cinemático de un Exoesqueleto Pasivo de Extremidad Inferior con Propósito de Rehabilitación., Puebla, México: Escuela de Ingeniería, Universidad de las Américas Puebla, 2011.
- [14] Naylamp Mechatronics, «TUTORIAL MPU6050, ACELERÓMETRO Y GIROSCOPIO,» 25 03 2022. [En línea]. Available: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html.

- [15] L. Llamas, «DETERMINAR LA ORIENTACIÓN CON ARDUINO Y EL IMU MPU-6050,» 22 Septiembre 2016. [En línea]. Available: <https://www.luisllamas.es/arduino-orientacion-imu-mpu-6050/>. [Último acceso: 25 Marzo 2022].
- [16] Random Nerd Tutorials, «ESP8266 NodeMCU with MPU-6050 Accelerometer, Gyroscope and Temperature Sensor (Arduino),» [En línea]. Available: <https://randomnerdtutorials.com/esp8266-nodemcu-mpu-6050-accelerometer-gyroscope-arduino/>. [Último acceso: 25 Marzo 2022].
- [17] M. Ford, «ESP-01 and ESP-01S, How program and use the Pins and Leds,» Forward Computing and Control Pty. LTd., 28 Julio 2015. [En línea]. Available: http://www.forward.com.au/pfod/ESP8266/GPIOpins/ESP8266_01_pin_magic.html#I2C.
- [18] Microcontrollers Lab, «LittleFS Introduction & Install ESP8266 NodeMCU Filesystem Uploader in Arduino IDE,» [En línea]. Available: <https://microcontrollerslab.com/littlefs-introduction-install-esp8266-nodemcu-filesystem-uploader-arduino/>.
- [19] V. S. Ami, «What is AWS IoT? How to create a thing in AWS IoT Core, its Certificates & policies,» Electronics Innovation, 10 Octubre 2019. [En línea]. Available: <https://electronicsinnovation.com/what-is-aws-iot-how-to-creat-a-thing-in-aws-iot-core-its-certificates-policies/>.
- [20] V. S. Ami, «How to connect NodeMCU ESP8266 with AWS IoT Core using Arduino IDE & MQTT,» Electronics Innovation, 14 Octubre 2019. [En línea]. Available: <https://electronicsinnovation.com/how-to-connect-nodemcu-esp8266-with-aws-iot-core-using-arduino-ide-mqtt/>.
- [21] V. S. Ami, «Storing ESP8266 data into Amazon DynamoDB using AWS IoT Core(MQTT) & Arduino,» Electronics Innovation, 7 Noviembre 2019. [En línea]. Available: <https://electronicsinnovation.com/storing-esp8266-data-into-amazon-dynamodb-using-aws-iot-coremqtt-arduino/>.
- [22] Amazon México, «Amazon,» [En línea]. Available: <https://www.amazon.com/-/es/FTCBlock-DC3-3V-Breadboard-Arduino-Solderless/dp/B07G31V67X>. [Último acceso: 5 Mayo 2022].
- [23] «Programación en Arduino con MATLAB y Simulink,» [En línea]. Available: <https://la.mathworks.com/discovery/arduino-programming-matlab-simulink.html>.
- [24] MathWorks, «Programación en Arduino con MATLAB y Simulink,» [En línea]. Available: <https://la.mathworks.com/discovery/arduino-programming-matlab-simulink.html>.

1.8. Anexos generales

Los anexos necesarios para entender la documentación del proyecto se refieren al software referenciado a lo largo del documento.

Debido a la cantidad de códigos independientes generados y su longitud, se optó por subirlos en un repositorio público de Github para futuras referencias.

https://github.com/LiliaLobato/P2022_PAP4D08A

2. Productos

A continuación, se enlistan los productos producidos a lo largo del presente periodo PAP que se entregarán como resultado del proyecto. Se enlista el producto, descripción y ubicación del archivo donde se puede consultar de forma que sirvan como punto de arranque en posteriores periodos PAP o puedan ser reutilizada en diferentes proyectos.

BOM y Gerber – ESP8266

Nombre y código del PAP	PAP Programa de Desarrollo Tecnológico para la Sustentabilidad Ambiental, Energética y Alimentaria I. (P2022_PAP4D08A)
Proyecto	Sistema de adquisición de datos biomecánicos
Descripción	Se generó un PCB con la circuitería necesaria para cargar o ejecutar un programa desde un ESP8266. Cuenta con salidas de debug por protocolo Serial y un bus I2C para conectar hacia diversos sensores y actuadores.
Autores	Lilia Arceli Lobato Martínez
Ubicación del producto	https://github.com/LiliaLobato/P2022_PAP4D08A/tree/main/PCB_BOM https://oshwlab.com/lilia.arceli/pap01

Prototipo 3D para montar un MPU6050 a una pierna

Nombre y código del PAP	PAP Programa de Desarrollo Tecnológico para la Sustentabilidad Ambiental, Energética y Alimentaria I. (P2022_PAP4D08A)
Proyecto	Sistema de adquisición de datos biomecánicos
Descripción	Archivos de impresión 3D para una caja con las medidas exactas de una MPU6050. Cuenta con agarraderas a los costados que permiten colocarla sobre el cuerpo, así como una abertura para conectarse con dispositivos externos
Autores	Lilia Arceli Lobato Martínez y Adrián Carrillo Cordero
Ubicación del producto	https://github.com/LiliaLobato/P2022_PAP4D08A/blob/main/3D_MOD_EL/MPU%206050.CATPart

Prototipo 3D para guardar la circuitería del ESP8266

Nombre y código del PAP	PAP Programa de Desarrollo Tecnológico para la Sustentabilidad Ambiental, Energética y Alimentaria I. (P2022_PAP4D08A)
Proyecto	Sistema de adquisición de datos biomecánicos
Descripción	Archivo de impresión 3D para una caja con las medidas exactas para colocar el PCB del ESP8266. Cuenta con un gancho en la parte superior para colocar una porta tarjetero y con aberturas que permiten conectarse con dispositivos externos.
Autores	Lilia Arceli Lobato Martínez y Adrián Carrillo Cordero
Ubicación del producto	https://github.com/LiliaLobato/P2022_PAP4D08A/blob/main/3D_MOD_EL/Caja.CATPart

Reporte detallado abordando el despliegue en la nube

Nombre y código del PAP	PAP Programa de Desarrollo Tecnológico para la Sustentabilidad Ambiental, Energética y Alimentaria I. (P2022_PAP4D08A)
Proyecto	“AWS IoT” creado para la materia “Cómputo en la Nube”
Descripción	Reporte detallado con el proceso necesario para conectar un dispositivo a la nube utilizando una extensa gama de servicios de AWS IoT.
Autores	Lilia Arceli Lobato Martínez, Othón Alejandro Escandón Azpeitia y Jorge Rodríguez Guatemala.
Ubicación del producto	https://drive.google.com/file/d/1ZZyCDUghAZm3amW1iJiRKzDu8foKCn6M/view?usp=sharing

Prototipo Final de hardware

Nombre y código del PAP	PAP Programa de Desarrollo Tecnológico para la Sustentabilidad Ambiental, Energética y Alimentaria I. (P2022_PAP4D08A)
Proyecto	Sistema de adquisición de datos biomecánicos
Descripción	Sistema de adquisición de datos montado al cuerpo utilizando 2 impresiones 3D personalizadas que se montan sobre el cuerpo usando un sistema de elásticos. Dentro de cada impresión se encuentra una tarjeta perforada con la circuitería necesaria para el proyecto, sensores y tarjetas de desarrollo.
Autores	Lilia Arceli Lobato Martínez
Ubicación del producto	Al ser un producto físico está en propiedad del Autor.

3. Reflexión crítica y ética de la experiencia

El RPAP tiene también como propósito documentar la reflexión sobre los aprendizajes en sus múltiples dimensiones, las implicaciones éticas y los aportes sociales del proyecto para compartir una comprensión crítica y amplia de las problemáticas en las que se intervino.

3.1 Sensibilización ante las realidades

Este proyecto se orienta como una ayuda en las terapias de rehabilitación para personas que sufren discapacidades motrices, al tener un usuario objetivo fue de suma importancia estar conscientes de la realidad que este grupo vulnerable. Fue necesario plantear el prototipo final tomando en consideración las necesidades, capacidades físicas y socioeconómicas de los futuros usuarios, por lo tanto, nos llegamos a enfrentar con tener que rediseñar o rechazar opciones que no generaban una propuesta que permitía realizar movimientos de miembros inferiores de forma ergonómica, segura y funcional, no eran entendible o fácil de usar, no eran escalables ni contaban con las características necesarias para su uso en el hogar o eran poco accesibles económicamente.

Por otro lado, se observó que la rehabilitación robótica posee su propio enfoque ante los problemas técnicos que enfrentan ingenieros y diseñadores; esto significa que, para desarrollar nuevos sistemas de rehabilitación robótica, se requiere tener en cuenta la adaptabilidad al usuario final, el consumo energético y la aceptabilidad de la tecnología por terapeutas e instituciones especializadas en rehabilitación.

No obstante, se debe entender que el propósito de implementar nuevas tecnologías en terapias es simplemente una herramienta, en ningún momento puede sustituir el papel del fisioterapeuta o médico especialista para orientar la terapia e interpretar los resultados. En este orden de ideas, se debe garantizar primero el acceso al recurso humano necesario para recibir sesiones de rehabilitación con una frecuencia deseable y, partiendo de este mínimo, adicionar tecnologías que mejoren los resultados.

3.2 Aprendizajes logrados

Este proyecto representa el mayor esfuerzo hasta la fecha que he realizado para aplicar mis conocimientos de la carrera de electrónica mientras los armonizo con mis recién adquiridos conocimientos de sistemas computacionales, tanto los conocimientos enfocados a la codificación y acceso a servicios como los conocimientos de administración de proyectos. Fue un proyecto sumamente ambicioso desde el inicio ya que es una nueva línea de investigación y no contamos con información previa o algún tipo de prototipo previo.

Al inicio del PAP enfrenté el reto de aprender a programar un ESP8266, esta tarjeta es bastante poderosa y tiene un tamaño perfecto para aplicaciones de IoT, pero cuenta con puertos reducidos por lo que para programar se tiene que realizar un proceso específico y delicado.

Una de los puntos en los que estoy orgullosa fue el poder implementar una metodología ágil para la creación de este proyecto. Al ser una arquitectura modular pude ir asignando cada módulo o tecnología a un sprint el cual tuvo las iteraciones necesarias hasta que generé un prototipo con la calidad necesaria. Tras concluir los Sprints enfocados en módulos, comencé a conectar poco a poco los módulos hasta terminar con un prototipo final.

A pesar de tener un buen proceso de desarrollo, la documentación la fui dejando de lado y en la semana 16 me tuve que dedicar prácticamente a terminar el reporte. En próximos proyectos de investigación voy a enfocarme a documentar conforme trabajo e implemento.

Termino mi primer periodo de investigación con la certeza de haber creado un proyecto de calidad, con el cual se sientan las bases para futuros proyectos.

3.3 Inventario de competencias Inicial e Inventario de competencias Final

Tabla 11 - Inventario de Competencias enfocada en Conocimientos

Competencia	Evidencia	Relevancia Fortaleza	Competencia al final del PAP	Competencias Potencializadas	
Conocimientos	Prototipado electrónico	Proyectos pasados.	Permite crear sistemas a pequeña escala con el fin de probar su viabilidad y analizar su funcionalidad.	Prototipado electrónico llevado a la creación de PCB, prototipos 3D modulares y escalable. Armado cuidadoso de los sistemas incluyendo soldado, diseño y modificación de materiales.	Reforcé mis conocimientos de prototipado rápido, aprendizaje de modelado e impresión en 3D enfocado a electrónicos.
	Programación	Proyectos pasados.	Permite obtener datos de diversas fuentes y poder procesarlos para convertirlos en información legible y pertinente.	Integración de distintas áreas de programación como cómputo en la nube y procesamiento de señales	Logré tener nuevos conocimientos dentro de las herramientas de cómputo en la nube por lo que, a diferencia de proyectos pasados, ahora cuento con conocimientos sobre la correcta implementación y protocolos necesarios para aplicaciones de IoT
	Instrumentación electrónica	Haber cursado satisfactoriamente el curso equivalente en ITESO.	Permite acondicionar sensores y actuadores bajo diversas condiciones para obtener datos procesables.	Instrumentación electrónica enfocada en el filtrado de señales digitales	Ahora soy capaz de combinar diferentes técnicas de instrumentación tanto físicas como digitales para obtener una señal con la mayor integridad posible.
	Sistemas electrónicos analógicos y	Haber cursado satisfactoriamente el curso equivalente en ITESO.	Permite tener un amplio conjunto de herramientas para el prototipado y acondicionamiento.	Rápido prototipado de sistemas analógicos y resolución de problemas con diversas técnicas	Cuento con experiencia integrando diversos circuitos para obtener información de sensores, para generar ambientes de programación de tarjetas. Adicionalmente, soy más hábil reconociendo los problemas que se originan en hardware y como resolverlos.
	Desarrollo de aplicaciones y servicios web	Proyectos pasados y conocimientos adquiridos en varias materias.	Permite desplegar la información de un proyecto de forma simple y entendible para el usuario final o para su futuro análisis.	Desarrollo de aplicaciones web con backend y frontend. Conexión a servicios de cómputo en la nube.	Me fue muy útil realizar el backend y frontend del proyecto pues me ayudó a cementar los conocimientos de desarrollo de aplicaciones y cómputo en la nube

Tabla 12 - Inventario de Competencias enfocada en Actitudes

Competencia	Evidencia	Relevancia Fortaleza	Competencia al final del PAP	Competencias Potencializadas
Actitudes	Comunicación efectiva y asertiva	<p>Genera un ambiente de colaboración entre los integrantes de un equipo.</p> <p>Permite llegar a conclusiones y soluciones tras discusiones técnicas.</p> <p>Permite plantear objetivos, procesos y conclusiones en palabras sencillas y con redacción entendible.</p>	Capacidad de explicar procesos, resultados y áreas de mejora de una forma clara y concisa	Mejoré mi capacidad de comunicación mediante el escrito de un reporte formal: y mediante la definición clara de tareas y tiempos de entrega entre los integrantes de mi equipo
	Delegación de tareas	<p>Poder plantear un proyecto como la colaboración de varios esfuerzos da un plan de acción conciso donde no se dupliquen esfuerzos y se trabaje hacia el mismo objetivo.</p>	Liderazgo, planeación, manejo de recursos	Esta competencia tiene grandes áreas de mejora, lamentablemente nos quedamos sin un integrante activo en el proyecto por lo que no pudimos delegar tareas. Lo que si se pudo potencializar fue el manejo de recursos ya que pudimos presentar una pequeña parte del producto final como proyecto de otra materia.
	Autodidacta	<p>Poder realizar trabajo de forma autodidacta nos permite gestionar nuestras destrezas y conocimientos.</p>	Búsqueda de documentación para la resolución de obstáculos, aprendizaje por medio de recursos gratuitos	Al ser un proyecto de investigación fue sumamente importante ser autodidacta respecto a las tecnologías que se fueron integrando.
	Iniciativa	<p>Tomar iniciativa en un proyecto nos permite idear, inventar y emprender diferentes soluciones de manera que nunca nos quedamos parados en seco.</p> <p>Esta forma de trabajo implica saber pedir ayuda y hacer las preguntas necesarias sin miedo.</p>	Resolución de problemas reales, de manera proactiva, combinando áreas de software y hardware	Cuento con mayor iniciativa para trabajar en proyecto con una complejidad mayor y que involucren varias áreas.
	Calendarización de tareas	<p>Tener un plan cronológico me permite priorizar tareas, tener un panorama real del avance del proyecto y estar consiente de los objetivos para un periodo.</p>	Uso de calendario de actividades, manejo de recursos, manejo detallado del tiempo	Esta competencia es la que mayores áreas de mejora tiene, a pesar de haber creado calendarios me terminé atrasando con los entregables y le tuve que dedicar más tiempo conforme se acababa el semestre.

Tabla 13 - Inventario de Competencias enfocada en Habilidades

Competencia	Evidencia	Relevancia Fortaleza	Competencia al final del PAP	Competencias Potencializadas	
Habilidades	Resolución de problemas	Varios proyectos que involucran software y hardware, competencias y hackáthones.	Al trabajar en un proyecto es natural encontrar problemas y obstáculos en el proceso. Es necesario poder identificar la raíz del problema y partir de ella para llegar a soluciones o crear alternativas viables.	Resolución de problemas basado en la búsqueda de información en documentación pertinente	Al estar familiarizados con el contexto de la problemática es sencillo identificar rápidamente la información que nos puede ayudar a resolver problemas teóricos. Para los problemas prácticos, ahora soy capaz de identificar la raíz del problema y, basándome en ella, filtrar la información que encontraba en internet.
	Uso y programación de tarjetas de desarrollo	Proyectos pasados y conocimientos adquiridos en varias materias.	Al conocer diferentes tarjetas, sus características y diferencias se puede elegir la opción que cubra las necesidades de un proyecto en específico. Poder usar diferentes plataformas puede llegar a facilitar algunas operaciones ya que no estamos limitados a puertos, memoria, sistema operativo, tamaño, batería o conectividad.	Aprendizaje de nuevas tarjetas de desarrollo y nuevos métodos de programación	Este proyecto me impulsó a realizar comparativas más completas para saber qué tarjeta de desarrollo, sensores, arquitectura, servicios y materiales serían la mejor opción
	Uso y programación en herramientas de simulación y gráficas	Proyectos pasados y conocimientos adquiridos en varias materias.	Contar con la habilidad de usar herramientas de simulación nos permite trabajar con datos en formato gráfico, modelar soluciones antes de construirlas y validar prototipos con mayor facilidad. Poder graficar los datos me permite encontrar problemas con los sensores y diagnosticar si provienen de la etapa de acondicionamiento o de nuestros algoritmos.	Aprendizaje de nuevas herramientas de visualización	Es una de las habilidades con mayor avance, pasé de no haber utilizado ninguna librería en JavaScript de simulación a tener un modelo tridimensional. Fue una parte integral del proyecto poder hacer uso de estas herramientas ya que siempre es más sencillo ver una gráfica a ver un flujo infinito de datos, nos sirvieron especialmente para caracterizar los filtros y como principal puerto de depuración de problemas.
	Análisis teórico complementado con simulaciones	Proyectos pasados y conocimientos adquiridos en varias materias.	El trabajar con sensores implica usar diferentes estrategias de procesamiento y acondicionamiento, algunas de ellas tienen que validarse y compararse antes de poder aplicarse en el proyecto. Podemos validar por medio de simulación si los cálculos son correctos.	Simulación basada en resolución de distintas ecuaciones para validar su resultado de manera gráfica	

Escalamiento progresivo de prototipos electrónicos funcionales	Proyectos pasados de robótica y wearables.	Al trabajar con proyectos que involucran electrónica es importante prototipar de manera gradual hasta llegar a un producto mínimo viable que cumpla con ciertas características. Para un proyecto vestible, se debe de cuidar que el prototipo sea cómodo sin perder ninguna funcionalidad.	Inclusión de ergonomía al realizar prototipos rápidos	Logré aprender a usar un programa de modelado 3D que me ayuda a realizar un prototipo mínimo funcional mediante la impresión 3D
Acondicionamiento de señales	Proyectos pasados y conocimientos adquiridos en varias materias.	Este es el proceso de adquisición de datos mediante un sensor o conjunto de sensores. La señal de un sensor no es perfecta, debe ser linealizada, filtrada y/o amplificada.	Acondicionamiento de señales en nuevos dispositivos utilizando nuevas herramientas	Durante el proyecto, identifiqué potenciales problemas relacionados al acondicionamiento de señales, como la posible pérdida de datos en movimientos rápidos y problemas con el bus de I2C
Procesamiento de datos	Proyectos pasados y conocimientos adquiridos en varias materias.	Tras obtener los datos, debemos tener algún método de análisis para convertirlo en información. Estos métodos pueden ir desde minería de datos hasta AI.	Refuerzo de conocimiento de procesamiento de datos en el área de wearables	Obtuve nuevos conocimientos que me ayudarán a convertir los datos en información útil de manera efectiva

Síntesis del inventario de competencias

Cuando se está constantemente trabajando sobre el mismo proyecto, cumpliendo con los entregables, generando avance e ir resolviendo problemas, se puede llegar a tener una especie de visión de túnel. Tras llenar las tablas 10, 11 y 12, me di cuenta de lo lejos que el proyecto ha llegado, los conocimientos que fui adquiriendo y los que reforcé. Me sorprende y enorgullece poder tomar un respiro y ver todos los detalles de los que no me había percatado respecto a mi desarrollo como profesionista y como la trayectoria de este proyecto me ha dado habilidades únicas.

Me da mucho gusto haber participado en la resolución de este problema, y espero haber aportado tanto avance y conocimiento al proyecto como el proyecto me aportó.