



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE BIOMEDICINA**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
TELE-ASISTENCIA PARA LA ATENCIÓN DE EMERGENCIAS A
PERSONAS CON MOVILIDAD REDUCIDA EN LA UNIVERSIDAD
POLITÉCNICA SALESIANA DE GUAYAQUIL.**

Trabajo de titulación previo a la obtención del
Título de Ingeniero Biomédico

AUTORES: Euclides Leonel Moran Alvarez
Anna Paula Sanchez Montiel
TUTOR: Ing. Roberto Gerardo Bayas Toro, Mgs.

Guayaquil - Ecuador
2025

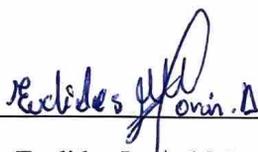
CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, **Euclides Leonel Moran Alvarez** con documento de identificación N° **0940455744** y **Anna Paula Sanchez Montiel** con documento de identificación N° **0940794027**; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo.

Guayaquil, 14 de febrero del año 2025

Atentamente,



Euclides Leonel Moran Alvarez
0940455744



Anna Paula Sanchez Montiel
0940794027

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA
UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, **Euclides Leonel Moran Alvarez** con documento de identificación N° **0940455744** y **Anna Paula Sanchez Montiel** con documento de identificación N° **0940794027**, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del **Proyecto Técnico: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE TELE-ASISTENCIA PARA LA ATENCIÓN DE EMERGENCIAS A PERSONAS CON MOVILIDAD REDUCIDA EN LA UNIVERSIDAD POLITÉCNICA SALESIANA DE GUAYAQUIL**, el cual ha sido desarrollado para optar por el título de: Ingeniero Biomedico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 14 de febrero del año 2025

Atentamente,



Euclides Leonel Moran Alvarez
0940455744



Anna Paula Sanchez Montiel
0940794027

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, **Roberto Gerardo Bayas Toro**, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE TELE-ASISTENCIA PARA LA ATENCIÓN DE EMERGENCIAS A PERSONAS CON MOVILIDAD REDUCIDA EN LA UNIVERSIDAD POLITÉCNICA SALESIANA DE GUAYAQUIL**, realizado por **Euclides Leonel Moran Alvarez** con documento de identificación N° **0940455744** y por **Anna Paula Sanchez Montiel** con documento de identificación N° **0940794027**, obteniendo como resultado final el trabajo de titulación bajo la opción **Proyecto técnico** que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 14 de febrero del año 2025

Atentamente,



Ing. Roberto Gerardo Bayas Toro, Mgs.
0940622608

DEDICATORIA

Este trabajo de titulación está dedicado con todo mi amor y gratitud a mi madre Aracelly de los Ángeles Alvarez Castro, por inculcarme valores tan importantes como la disciplina, la responsabilidad y la ética. Su amor inmenso, su inmenso amor, cuidado y apoyo incondicional han sido fundamentales para mi bienestar y para superar todos los desafíos a lo largo de estos años de estudios universitarios. También, dedico este logro a la memoria de mi padre Euclides Moran Torres, quien, aunque ya no está en este mundo, me enseñó a luchar por mis metas y a perseguir con determinación los sueños que anhelamos alcanzar en la vida .

A mis amigos, por su confianza inquebrantable, su compañía y palabras de aliento que hicieron este camino más llevadero. A mis profesores, cuya guía amplió mis conocimientos y despertó en mí el deseo de aportar algo significativo a mi campo. Este trabajo es el resultado del apoyo y la influencia de quienes, con su esfuerzo y ejemplo, moldearon al profesional y ser humano que soy hoy.

Finalmente agradezco a mi hermana Edith Moran y mi hermano Francisco Moran a mi abuela paterna Estela Torres Cox y también a mi tío por su confianza absoluta que me ha impulsado a seguir adelante y quienes siempre me apoyado en todo momento.

Euclides Leonel Moran Alvarez

Este trabajo de titulación esta dedicado a mis padres Guillermo Sanchez y Katuska Montiel, cuyo amor incondicional, apoyo constante y ejemplo de integridad han sido mi mayor inspiración por brindarme las herramientas necesarias para alcanzar mis metas son el motivo por el cual este logro es también suyo. Gracias por enseñarme a enfrentar los desafíos con valentía y perseverancia.

A mis amigos, por su inquebrantable confianza en mí, incluso en los momentos de duda. Su compañía, palabras de aliento y la alegría que aportaron a esta etapa hicieron que este camino fuese mucho más llevadero. Cada uno de ustedes ha dejado una huella imborrable en esta experiencia. A mis profesores, cuya guía ha sido fundamental en mi formación ayudando a ampliar mis conocimientos, sino que también despertaron en mí el deseo de contribuir con algo significativo a mi campo. Este trabajo no habría sido posible sin el apoyo y la influencia de cada una de las personas que han formado parte de mi vida, quienes, con su presencia, esfuerzo y ejemplo, han moldeado el profesional y ser humano que soy hoy.

A mi hermana Victoria Sánchez quien siempre estuvo en todos los momentos duros de mi vida con su presencia, esfuerzo y apoyo en todo este proceso.

Anna Paula Sanchez Montiel

AGRADECIMIENTO

Agradezco a Dios, a mi familia por su amor incondicional y siempre esta apoyándome en todos momento, por inculcarme buenos valores y la perseverancia para afrontar los retos y dificultades a lo largo de la vida. Este logro es tanto mío como de ellos, ya que su ejemplo de esfuerzo y dedicación ha sido la base sobre la que he construido este camino.

Agradezco además de manera especial al Ing. Roberto Bayas por su guía ante la realización de este proyecto con sugerencias y la confianza brindada para culminarlo. Asimismo, extendo mi gratitud a mis compañeros y amigos, quienes con su apoyo, compañía y palabras de ánimo han hecho de esta experiencia un camino más llevadero y significativo.

Euclides Leonel Moran Alvarez

Agradezco a Dios por darme las fuerzas de seguir adelante siempre, a mis padres, quienes con su amor incondicional, su apoyo constante y sus palabras de aliento han sido mi mayor motivación durante todo este proceso. Su confianza en mis capacidades me ha impulsado a superar cada obstáculo y a dar lo mejor de mí en cada etapa de este proyecto. Este logro es tanto mío como de ellos, ya que su ejemplo de esfuerzo y dedicación ha sido la base sobre la que he construido este camino.

Agradezco también al Ing. Roberto Bayas, quien ha sido fundamental para alcanzar este objetivo con su paciencia, disposición y compromiso para culminar el mismo. Asimismo, extendo mi gratitud a mis compañeros y amigos, quienes con su apoyo, compañía y palabras de ánimo han hecho de esta experiencia un camino más llevadero y significativo.

Anna Paula Sanchez Montiel

RESUMEN

El presente proyecto comprende el desarrollo y la implementación de un prototipo de sistema de tele-asistencia integral para responder de manera oportuna a los requerimientos de emergencia de personas con movilidad reducida que forman parte de la Universidad Politécnica Salesiana de Guayaquil. El diseño del prototipo comprende de varias etapas donde se abarcan aspectos como la seguridad, asistencia rápida y atención de calidad, promoviendo así la inclusión mediante una solución basada en tecnologías IoT que permita la asistencia remota de manera eficiente.

El diseño y la simulación del prototipo se realizó en la plataforma Wokwi. Para el hardware se utilizó como dispositivo principal Arduino Uno que sirve para la adquisición de datos y Android Studio para el desarrollo de una aplicación móvil que facilite la comunicación y la gestión de alertas del dispositivo.

El sistema cuenta con un sensor que permite medir la aceleración, el cambio en la orientación y la velocidad angular del usuario, y en consecuencia enviará notificaciones en tiempo real indicando posibles problemas de movilidad como caídas, afecciones inesperadas, entre otros.

Para la programación y simulación de los componentes electrónicos se utilizó la plataforma Wokwi. Además, el diseño y la lógica de la interfaz de usuario se desarrollaron en Android Studio, para garantizar una experiencia intuitiva y funcional para los usuarios finales. Finalmente, se evaluó el desempeño del sistema mediante pruebas de movilidad, con el objetivo de determinar la eficiencia del sensor y la rapidez de respuesta del sistema ante la identificación de problemas de movilidad.

Palabras claves: tele-asistencia, movilidad reducida, detección de caída, IoT, Wokwi, Arduino Uno, Android Studio

ABSTRACT

This project includes the development and implementation of a prototype of a comprehensive tele-assistance system to respond in a timely manner to the emergency requirements of people with reduced mobility who are part of the Salesiana Polytechnic University of Guayaquil. The design of the prototype includes several stages that cover aspects such as security, quick assistance and quality care, thus promoting inclusion through a solution based on IoT technologies that allows remote assistance efficiently.

The design and simulation of the prototype was carried out on the Wokwi platform. For the hardware, Arduino Uno was used as the main device, which serves for data acquisition, and Android Studio for the development of a mobile application that facilitates communication and management of device alerts.

The system has a sensor that allows it to detect the change in the orientation and angular speed of the user, and consequently will send notifications in real time indicating possible mobility problems such as falls, unexpected conditions, among others.

The Wokwi platform was used for programming and simulation of the electronic components. Additionally, the UI design and logic were developed in Android Studio, to ensure an intuitive and functional experience for end users. Finally, the performance of the system was evaluated through mobility tests, with the objective of determining the efficiency of the sensor and the speed of response of the system when identifying mobility problems.

Keywords: tele-assistance, reduced mobility, fall detection, IoT, Wokwi, Arduino Uno, Android Studio.

ÍNDICE

I.	Introducción	1
II.	Problema	2
III.	Justificación	3
IV.	Objetivos	4
IV-A.	Objetivo general	4
IV-B.	Objetivos específicos	4
V.	Marco Teórico	5
V-A.	Tele-Asistencia	5
V-A1.	Definición	5
V-A2.	Como funciona la tele-asistencia	5
V-A3.	Diferente tipo de modalidad de tele-asistencia	6
V-A4.	Como se implementa la Tele-asistencia	7
V-B.	ESP32	9
V-B1.	Introducción ESP32	9
V-B2.	Descripción del ESP32	10
V-B3.	ESP32 Wifi y Bluetooth en un solo chip	10
V-C.	Sensor MPU6050	11
V-C1.	MPU6050 Arduino, Acelerómetro y Giroscopio	11
V-C2.	Sensor Giroscópico	12
V-C3.	Sensor Acelerómetro	12
V-C4.	Especificaciones del MPU6050	13
V-D.	Iot o Internet de las cosas	13
V-D1.	Introducción	13
V-D2.	Concepto de Iot	13
V-E.	Software Android Studio	14
V-E1.	Estructura de un proyecto en Android Studio	15
V-E2.	Sistema de compilación de Gradle	16
V-F.	Variantes de compilación y APKs múltiples	16
V-G.	Administración de dependencias	16
V-G1.	Herramientas integradas de depuración y análisis	17
V-G2.	Inspecciones de código	17
V-H.	Software Arduino studio	17
V-H1.	Introducción Arduino Studio	17
V-H2.	Funcionamiento de Arduino Studio	17
V-H3.	La bibliotecas de Arduino	18
V-H4.	Instalación de bibliotecas	18
V-H5.	Arduino IDE 2	20
V-H6.	Características de Arduino IDE2	20
V-I.	Software Wokwi	20
V-J.	Postman	21
V-J1.	Para que sirve el Postman	21
V-J2.	Métodos más utilizados	21

VI. Marco Metodológico	23
VI-A. Fase de encuesta	23
VI-B. Fase de investigación	24
VI-B1. Análisis de la necesidad del proyecto y la solución tecnológica innovadora a aplicar	24
VI-B2. Normas ISO	24
VI-C. Fase del diseño del dispositivo de asistencia remota	25
VI-C1. Selección de componentes mediante comparación de las características	25
VI-C2. Diseño del circuito	26
VI-C3. Placa en pcb	27
VI-C4. Simulación del circuito Arduino y post man	28
VI-C5. Fabricación del modulo externo del dispositivo en 3D	31
VI-C6. Verificación del funcionamiento exitoso del circuito	31
VI-D. Fase de diseño y desarrollo de la app móvil	31
VI-D1. Funcionalidades principales de la aplicación	32
VI-D2. Proceso de diseño y desarrollo	32
VI-E. Analisis de batería	37
VI-F. Modelo matemático para el análisis de aceleración y movimientos en el sensor MPU6050 .	37
VI-F1. Rango del sensor	37
VI-F2. Cálculos de la aceleración en “g”	37
VI-G. Fase de integración del sistema	38
VI-H. Fase de evaluación y validación del sistema completo	38
VII. Resultados	40
VII-A. Encuestas realizadas	40
VII-B. Dispositivo final implementado	42
VII-C. Modulo externo del dispositivo 3D	43
VII-C1. Dispositivo de prueba	43
VII-C2. Dispositivo final en 3D	44
VII-C3. Pruebas del sistema	45
VII-D. Aplicación móvil	46
VII-D1. Pantalla principal	47
VII-D2. Alertas en la aplicación	48
VII-D3. Ficha medica	49
VII-E. Analisis de batería del dispositivo	50
VII-F. Modelo matemático	50
VII-F1. Cálculo de ejes de movilidad en el diseño del dispositivo de asistencia remota en emergencias.	50
VII-F2. Tiempo de promedio de respuesta desde la detección de la emergencia hasta su notificación al administrador	50
VII-F3. Respuesta de las pruebas	52
VIII. Cronograma	54
IX. Presupuesto	55
X. Conclusiones	56
XI. Recomendaciones	57
Referencias	58
XII. Anexos	60

ÍNDICE DE FIGURAS

1.	Como funciona telemonitoreo [10]	6
2.	Funcionamiento tele-asistencia [13]	8
3.	Mecanismo tele-asistencia [13]	9
4.	ESP-32 [16]	10
5.	ESP-WROOM-3 datasheet [16]	10
6.	Diagrama de bloques del ESP32 [19]	11
7.	Sensor MPU6050 [22]	12
8.	Evolución de Internet [23]	14
9.	Software Android estudio [24]	14
10.	Archivos de proyecto de Android [24]	16
11.	Inspección de lint [24]	17
12.	Editor de texto [25]	18
13.	Liberia de Arduino [25]	19
14.	Lista de bibliotecas [25]	19
15.	Interfazar de IDE 2 de Arduino [25]	20
16.	Metodos utilizado en el postman [28]	22
17.	Circuito en wokwi para la implementación, Autor propio.	26
18.	Diagrama de flujo de dispositivos, Autor propio	27
19.	Placa PCB en 2D, Autor propio.	28
20.	Visualización del código implementado en arduino para el proyecto, Autor propio.	28
21.	Visualización del código para la obtención de los ejes x,y, z, Autor propio.	29
22.	Visualización de la obtención de la IP del dispositivo, Autor propio.	29
23.	Visualización de la obtención de las coordenadas mediante el simulador Post Man, Autor propio.	30
24.	Visualización de la obtención de las coordenadas mediante el simulador Post Man, Autor propio.	30
25.	Funciones principales de la app, Autor propio.	32
26.	Diagrama de flujo de la aplicación, Autor propio.	33
27.	Visualización del código de alertas, Autor propio.	34
28.	Visualización de actividad de la alerta con sus parámetros a notificar, Autor propio.	34
29.	Ayudante de base de datos para almacenar la información obtenida, Autor propio.	35
30.	Parámetros que se obtienen en la ficha medica, Autor propio.	35
31.	Buscador de ficha de los usuarios para obtención de datos de forma mas rápida, Autor propio.	36
32.	Gráficas sobre la importancia de desarrollar e implementar tecnologías en la universidad , Autor propio.	40
33.	Gráficas sobre la importancia de tecnología de tele-asistencia remota, Autor propio.	40
34.	Gráficas sobre la importancia de tener un sistema de tele-asistencia, Autor propio.	41
35.	Gráficas sobre el impacto de agregar un sistema de tele-asistencia en la universidad, Autor propio.	41
36.	Gráficas sobre las diferentes características del desarrollo de un proyecto como el mencionado en esta encuesta, Autor propio.	42
37.	Gráficas sobre el apoyo que creen que tiene la universidad sobre este tipos de temas, Autor propio.	42
38.	Modulo externo vista frontal , Autor propio.	43
39.	Modulo externo vista planta, Autor propio.	43
40.	Modulo externo vista perfil, Autor propio.	43
41.	Visualización del dispositivo de prueba de la vista lateral, Autor propio.	44
42.	Visualización del dispositivo de prueba de la vista frontal, Autor propio.	44
43.	Visualización del dispositivo final, Autor propio.	44
44.	Visualización del dispositivo final 2, Autor propio.	45
45.	Simulación de escenario para prueba de dispositivo, Autor propio.	45
46.	Enlaces de conexión del sistema, Autor propio.	46
47.	Ingreso a la app, Autor propio.	46
48.	Visualización de la pantalla principal, Autor propio.	47

49.	Botón de emergencia, Autor propio	48
50.	Visualización de las alertas,Autor propio.	49
51.	Visualización de la ficha medica del usuario, Autor propio.	49
52.	La matriz de confusión de la prueba realizadas, Autor propio.	53
53.	Ficha tecnica del componente ESP32 [29].	60
54.	Ficha tecnica del Arduino UNO [30].	60
55.	Ficha técnica del MPU-6050 [31].	61
56.	Características de cámara boykeep [32].	61
57.	Características de soporte de batería de litio [33].	62
58.	Simulación de las conexiones de prueba, Autor propio.	62
59.	Simulación de la placa fallida, Autor propio.	63
60.	Simulación de la placa PCB fallida, Autor propio	63

ÍNDICE DE TABLAS

I.	Tabla de dato de movimiento del el sensor MPU6050	13
II.	Comparación de Microcontroladores para la elaboración del dispositivo	25
III.	Comparación de Sensores de ejes de movimiento	25
IV.	Comparación de Cámaras para el sistema implementado	26
V.	Tabla de capacidad y consumo de batería	50
VI.	Cronograma	54
VII.	Presupuesto	55

I. INTRODUCCIÓN

Las personas con movilidad reducida enfrentan múltiples desafíos en su vida cotidiana, especialmente al trasladarse. Durante estos desplazamientos, están expuestas a sufrir accidentes que pueden provocar lesiones o agravar sus dificultades de movilidad. Además, en caso de un incidente, a menudo no cuentan con asistencia inmediata y especializadas al alcance de sus necesidades.

En este contexto, impulsar la gestión tecnológica en beneficio de las personas con limitaciones de movilidad es una prioridad para la comunidad universitaria. Por ello, se plantea un trabajo de titulación que aborde temas claves como la atención rápida, mejora de la seguridad y el fomento de la autonomía, a través de la implementación de un prototipo de sistema de tele-asistencia.

El presente trabajo de investigación plantea un sistema que para la atención de emergencias dirigido a personas con movilidad reducida dentro de la Universidad Politécnica Salesiana de Guayaquil. La propuesta se basa en el desarrollo de un sistema tecnológico que integre sensor, dispositivos electrónicos y una aplicación móvil, permitiendo la emisión de alertas automáticas y la atención inmediata en caso de incidentes.

Para llevar a cabo la propuesta, se realizará un análisis de las necesidades generales de las personas con movilidad reducida dentro de la universidad, seguido del diseño y desarrollo del sistema de teleasistencia. Posteriormente, se evaluará la efectividad de la implementación para determinar su alcance potencial y los beneficios que podría ofrecer.

II. PROBLEMA

En la actualidad, más del 15 % de la población mundial sufre de algún tipo de discapacidad [1], mientras que en América Latina y el Caribe se estima que casi el 12 % de la población vive con al menos una discapacidad, casi un total de 66 millones de personas [2]. En Ecuador, el porcentaje de personas con discapacidad es del 2.6 %, lo que equivale a aproximadamente 480,776 personas que presentan algún tipo de discapacidad, ya sea visual, motriz, auditiva u otras [3].

Por otra parte, en el ámbito educativo, la inclusión de estudiantes con discapacidad en la educación superior en Ecuador es alarmante; solo el 1.29 % de esta población participa en este nivel académico según el Consejo Nacional para la Igualdad de Discapacidades [4]. Un estudio realizado por la Universidad Nacional Autónoma de México, reveló que los estudiantes con discapacidad motriz enfrentan significativas barreras de acceso a instalaciones como salones, bibliotecas y laboratorios. Muchos de los accesos, aunque existen, no están adecuadamente diseñados, lo que dificulta su movilidad [5].

La Universidad de Agder ha identificado que la incorporación de tecnologías, como dispositivos móviles y sistemas de teleasistencia, aporta beneficios significativos para los estudiantes al optimizar la comunicación, la efectividad en la respuesta y la experiencia práctica en la gestión de emergencias. Estos beneficios se han demostrado mediante simulaciones grupales, el empleo de dispositivos móviles y el desarrollo de escenarios simulados que facilitan la representación de roles en situaciones de emergencia [6] .

En la actualidad, existen diversas tecnologías que son capaces de brindar asistencia remota y/o presencial a las personas con discapacidad al momento de desplazarse de un lugar a otro. Entre estas tecnologías se tiene al Global Positioning System(GPS) que permite ubicar con cierta precisión a una persona a través de las señales emitidas vía satélite. Otros dispositivos como pulseras inteligentes utilizan sensores que permiten detectar obstáculos y emitir alertas a los usuarios. Sin embargo, el diseño de estos sistemas conlleva altos costos lo que implica la falta de acceso de usuarios con estas necesidades [7].

La Universidad Politécnica Salesiana (Sede Guayaquil, Campus Centenario), en periodo 65 está conformada por una población de aproximadamente 4,000 personas, incluyendo profesores, estudiantes y personal administrativo. Entre ellos, se encuentra un número reducido de personas con movilidad reducida, que requieren un entorno universitario inclusivo, reduciendo los riesgos del diario vivir al que pueden verse expuestos.

Aunque la universidad cuenta con elementos tecnológicos y humanos para el monitoreo y la seguridad de toda la comunidad, es importante contar con un sistema adicional de apoyo que identifique las emergencias a las que cualquier persona puede estar expuesta. Por lo tanto, explorar la idea del desarrollo de un sistema de teleasistencia que brinde una respuesta eficiente en la atención de emergencias a personas con movilidad reducida, es importante para lograr un entorno inclusivo adaptado a las necesidades para todas las personas que conforman esta sede.

III. JUSTIFICACIÓN

En la actualidad, el sistema de vigilancia de la universidad se encarga del registro de información multimedia para identificar las distintas situaciones que se viven dentro de la sede, pero no está diseñado para asistir de forma inmediata a un individuo que este pasando una situación de emergencia.

A partir de este contexto, la presente investigación se centra en el desarrollo de un sistema de atención de emergencias basado en la aplicación de conceptos de tele-asistencia que brinde soporte a personas con movilidad reducida. Para abordar esta necesidad, se propone el desarrollo de un sistema que permita la detección y notificación de cualquier incidente registrado, facilitando una intervención oportuna y reduciendo el tiempo de espera. Este sistema busca no solo mejorar la seguridad de las personas con movilidad reducida, sino también promover un entorno universitario inclusivo y adaptado a las necesidades de toda la comunidad.

La implementación de este sistema aportará beneficios significativos tanto a los estudiantes como al personal administrativo y docente con movilidad reducida, porque permitirá una respuesta más eficiente ante posibles incidentes. Este sistema también contará con un enfoque personalizado, incluyendo las fichas médicas de cada usuario que porte el dispositivo, lo cual permitirá al personal de administración acceder a información relevante sobre su estado de salud y necesidades específicas en caso de una emergencia. Al reducir el tiempo de espera y proporcionar una atención adaptada a cada persona, el sistema contribuirá a minimizar las posibles secuelas de los accidentes y mejorará la calidad de vida.

IV. OBJETIVOS

IV-A. Objetivo general

Diseñar e implementar un sistema de tele-asistencia para la atención de emergencias a personas con movilidad reducida, con el fin de mejorar su seguridad y proporcionar una respuesta rápida y eficiente ante posibles incidentes.

IV-B. Objetivos específicos

- Establecer los requisitos técnicos necesarios para el desarrollo de un dispositivo de asistencia remota mediante la investigación y el uso apropiado de normativas, conceptos de tele-asistencia e Internet de las Cosas (IoT).
- Desarrollar un dispositivo de asistencia remota aplicando los requisitos establecidos para la atención de emergencias a personas con movilidad reducida.
- Diseñar una app móvil que integre funciones de monitoreo en tiempo real y asistencia remota para la identificación y notificación de emergencias.
- Integrar el dispositivo de asistencia remota y la app móvil en un sistema Tele-asistencia que asegure el monitoreo continuo y la respuesta inmediata ante emergencias.
- Evaluar la confiabilidad, la aceptación y el tiempo de respuesta en la detección de emergencias.

V. MARCO TEÓRICO

V-A. *Tele-Asistencia*

V-A1. *Definición:* El término tele-asistencia se refiere a la prestación de servicios de apoyo social a distancia mediante el uso de tecnologías de la información y comunicación (TIC). Este concepto surgió en la década de 1990 como una solución para brindar atención en el hogar a personas mayores ante situaciones de emergencia. Hacia mediados de esa década, la teleasistencia domiciliaria evolucionó al incorporar modelos de atención centralizados, gestionados por equipos de profesionales, como trabajadores sociales y psicólogos, que operaban desde un espacio físico compartido. De esta manera, se desarrolló una modalidad de teleasistencia social basada en centros especializados, conocidos como “call-centers”, los cuales gestionaban las llamadas y almacenaban los datos médicos y sociales de los usuarios en sistemas de información [8].

V-A2. *Como funciona la tele-asistencia :* La tele-asistencia permite una comunicación inmediata desde cualquier lugar del hogar del usuario, ya sea para conversar o, en situaciones de emergencia, para activar los recursos necesarios. Este servicio utiliza principalmente la comunicación telefónica, ya sea mediante líneas fijas, móviles o llamadas a través de Internet (IoT). Se basa en un sistema compuesto por dispositivos tecnológicos y la atención de teleoperadores desde un centro especializado (CA), diseñado para responder a las necesidades y emergencias de los usuarios [9].

El dispositivo principal es una combinación de una unidad de control remoto (UCR), conocida como pulsador, y una terminal de comunicación. El pulsador es un botón portátil que puede usarse como colgante o pulsera, diseñado para actividades cotidianas, incluso en la ducha. En caso de emergencia, al presionarlo, el pulsador envía una señal a la terminal, que realiza automáticamente una llamada en modo manos libres al centro de atención. Algunos dispositivos avanzados también detectan automáticamente caídas bruscas, eliminando la necesidad de presionar el pulsador [9].

La terminal de comunicación, conectada a una línea telefónica fija o mediante tarjeta SIM, permite realizar llamadas al centro de atención sin necesidad de manipular el teléfono [9].

Los teleoperadores del centro de atención están disponibles las 24 horas del día durante todo el año y siguen protocolos específicos según el tipo de situación. Si un usuario llama por error o simplemente para conversar, los teleoperadores están capacitados para mantener la comunicación y evaluar su estado sociosanitario. En emergencias, el tiempo de respuesta suele ser inferior a 30 segundos, aunque en algunos países, como España, la normativa exige tiempos máximos de 10 segundos. Según la emergencia y las características del usuario, se movilizan recursos como familiares, ambulancias o personal médico [9].

Este servicio remoto se complementa con equipos presenciales que recopilan información sociosanitaria, realizan seguimientos en casos complejos y coordinan con los servicios sociales y otras instituciones públicas. En algunos lugares, también se dispone de vehículos de respuesta rápida tripulados por personal capacitado en primeros auxilios, que brindan asistencia inicial hasta que llegan los recursos médicos necesarios[9].

Actualmente, existen otros dispositivos de tele-asistencia cuyo uso es menos frecuente: los sensores o dispositivos periféricos. Estos equipos están diseñados para detectar automáticamente posibles riesgos, supervisar las rutinas diarias del usuario y responder ante cambios inesperados en esas actividades. Entre los sensores periféricos más comunes se encuentran los detectores de humo, gas, apertura de puertas, movimiento, ocupación de asientos y caídas, entre otros [9].

Además de la tele-asistencia fija, en los últimos años ha surgido la tele-asistencia móvil (TAM), una modalidad dirigida a personas mayores con un buen nivel de autonomía y calidad de vida, que pasan gran parte del día fuera de sus hogares. Este sistema utiliza un dispositivo móvil conectado a redes de telefonía 3G, 4G, LTE o 5G, que

permite la comunicación inmediata por voz con el centro de atención y la localización geográfica del usuario, facilitando la asistencia oportuna en caso de necesidad [9].

El servicio de tele-asistencia facilita y garantiza condiciones de mayor seguridad y tranquilidad para las personas, al asegurar una atención profesional adecuada situaciones de emergencia. Este sistema está diseñado para responder ante casos en los que se perciba inseguridad, malestar o síntoma inesperados en cualquier momento. La atención en emergencia se realiza a través dispositivos que suele ser pulsera, colgador o App conectado a una red Iot. Estos dispositivos permiten al usuario emitir una alerta con solo pulsar un botón cuando se encuentra en una situación riesgo o emergencia. Mediante alerta que se envía activa una respuesta inmediata, lo que posibilita la asistencia profesional necesaria y la movilización de los recursos específicos para atender la demanda [9].

Como lo puede ser a través de un sistema de tele-asistencia medica tipo Supervisión, Control y adquisición de Datos(SCADA) es una red inalámbrica que permite el monitoreo de signos vitales. El sistema de telemonitoreo son gran aporte para desarrollo que permitan tener una transmisión a través de una red móvil GSM permite la transmisiones inalámbricas es una área especifica de cobertura como se muestra en la figura 1 [10].

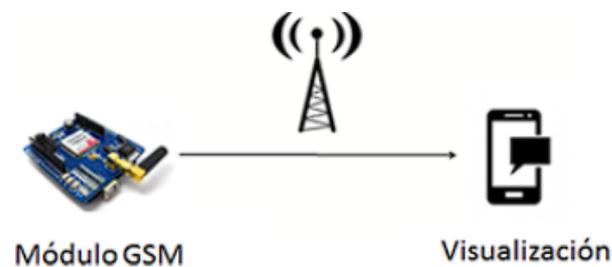


Figura 1. Como funciona telemonitoreo [10]

Esta tecnología se implementa para brindar asistencia remota en casos de emergencia que puedan presentarse en personas con movilidad reducida o condiciones de salud especiales. Es útil no solo para adultos mayores, sino también para personas con discapacidades o enfermedades crónicas que requieren apoyo para llevar a cabo actividades cotidianas. Además, este tipo de sistema ofrecer diferentes servicios, que incluyen el uso de dispositivos, en seguridad y monitoreo. Estos dispositivos pueden integrar cámaras, sensores y botones de emergencia, permitiendo recibir asistencia a distancia o alertar al personal de salud en caso de riesgo [11].

Estos sistemas pueden ser muy útiles para mejorar la calidad de vida de las personas que necesitan apoyo y para reducir la carga sobre sus cuidadores [11].

El servicio de tele-asistencia ha tenido crecimiento significativamente durante los últimos años a nivel mundial debido a lo beneficio para sus usuarios, cuidadores en el sistema de salud. A través de un estudio en Europa, la tasa de penetración de la tele-asistencia entre las personas mayores de 65 años varía desde menos del 3 % hasta el 11 %. En América Latina y el Caribe la tele-asistencia aún no se ha desarrollado conforme a la necesidad existente de servicios de cuidado y los estudios disponibles no cuentan con una visión general del potencial impacto de este servicio en la región [12].

V-A3. Diferente tipo de modalidad de tele-asistencia: La tele-asistencia fue evolucionando a través de lo años, mediante los avances tecnológicos y principalmente de las innovaciones que desarrollaron y hoy en día se distinguir cuatro modalidades de tele-asistencia [9].

Tele-asistencia reactiva: La tele-asistencia reactiva se activa de forma inmediata ante una emergencia. Este servicio permite que el usuario se comunique directamente con el centro de atención a través de dispositivos específicos, como los pulsadores. Una vez establecida la conexión, se procede a movilizar los recursos adecuados para atender la situación, ya sea contactando a personas con acceso al domicilio del usuario o alertando a los servicios médicos o de emergencia correspondientes [9].

Tele-asistencia proactiva: La tele-asistencia proactiva complementa la modalidad reactiva mediante servicios diseñados para prevenir situaciones críticas. Este enfoque, impulsado desde el centro de atención, ofrece un soporte más integral tanto a los usuarios como a sus cuidadores. Incluye actividades como llamadas salientes programadas (diarias, semanales o mensuales, según las necesidades sociosanitarias de cada persona), visitas domiciliarias semestrales o anuales, y campañas de información, orientación y asesoramiento, lo que permite un acompañamiento más efectivo [9].

Entre las acciones más representativas de la tele-asistencia proactiva se encuentran los recordatorios, ya sea telefónicos o mediante mensajes, relacionados con el cumplimiento de tratamientos médicos, la toma de medicamentos o recomendaciones específicas, como mantenerse hidratado durante los días de altas temperaturas [9].

Tele-asistencia personalizada: La tele-asistencia personalizada intensifica los contactos proactivos con usuarios que presentan un mayor grado de dependencia, ajustando las intervenciones según sus características individuales y las condiciones crónicas que puedan padecer. Por ejemplo, esta modalidad permite implementar sistemas de monitoreo continuo de signos vitales, así como ofrecer orientación y apoyo disponible en todo momento para pacientes con enfermedades crónicas, garantizando una atención más adaptada y específica a sus necesidades [9].

Tele-asistencia predictiva: La modalidad de tele-asistencia todavía se encuentra en desarrollo y tiene la menor cantidad de usuarios comparada con las otras modalidades. La tele-asistencia predictiva tiene como principal objetivo detectar los riesgos posibles, mediante la utilización de sensores y otros dispositivos que recopila información sobre los procesos rutinarios de los usuarios. Además, permite actuar para evitar situaciones de emergencia. Se recoge la información diaria de los procesos y, a través del análisis de estos datos, la inteligencia artificial percibe cambios de conductas que resultan sospechosos [9].

V-A4. *Como se implementa la Tele-asistencia:* La elección de tele-asistencia dependiendo del tipo de accionamiento también está relacionada con el grado de dependencia en la emergencia. Por ejemplo, hay casos en los que no son capaces de activar por ellos mismos el botón SOS, por ello es necesario instalar un sistema capaz de detectar que ciertas actividades cotidianas se han dejado de hacer y activar la emergencia [13], como se observa en la figura 2.

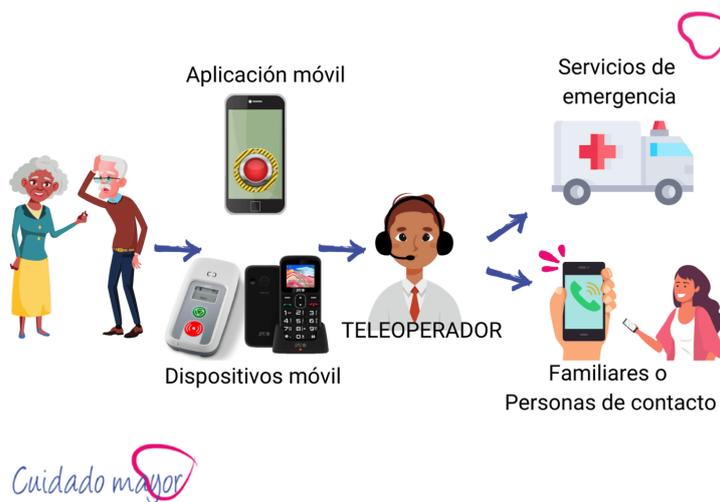


Figura 2. Funcionamiento tele-asistencia [13]

La tele-asistencia se utiliza de diferente manera como las siguiente:

- **Asistencia en caso de emergencia:** Es el objetivo principal del servicio. Los usuarios cuentan con una atención inmediata las 24 horas del día, los 365 días del año. Ante cualquier emergencia y con solo pulsar un botón, el personal cualificado atenderá la llamada y pondrá en marcha los recursos necesarios que requiera la situación, avisando a familiares y servicios de emergencia, en caso de ser necesario [14], como se muestra en la figura 3.
- **Llamadas de tele-asistencia:** Además, los usuarios de tele-asistencia recibirán periódicamente llamadas de seguimiento de los profesionales del Centro de Atención para revisar su estado, acompañarle en situaciones de soledad no deseada e informar sobre cualquier novedad. Esta actuación permite identificar cualquier anomalía y realizar un correcto seguimiento y acompañamiento de cada uno de los usuarios [14].
- **Seguimiento de la agenda personal:** El servicio quiere garantizar la calidad de vida del usuario y en este aspecto la salud es algo esencial. Para ello, se incluye un servicio recordatorio de citas médicas y toma de medicamentos [14].
- **Asesoramiento:** Existen a disposición de usuarios y familiares diferentes vías de comunicación con el servicio de tele-asistencia para conocer todas las novedades, informar sobre incidencias y otro tipo de cuestiones relacionadas con el cuidado del mayor. Además, como servicio extra puedes incluir orientación médica telefónica para recibir asesoramiento médico personalizado [14].
- **Localización GPS:** Para aquellos usuarios que quieran contar con atención inmediata también fuera de casa, existe la tele-asistencia móvil, con dispositivos que incluyen geolocalización para que en caso de desorientación, sean fácilmente localizables [14].
- **Servicios adicionales:** Además de los servicios mencionados, existen otros muchos con el objetivo de adaptarse al máximo a cada usuario como es el caso de la unidad móvil y custodia de llaves o los talleres y actividades realizadas para usuarias con el objetivo de favorecer el envejecimiento activo [14].

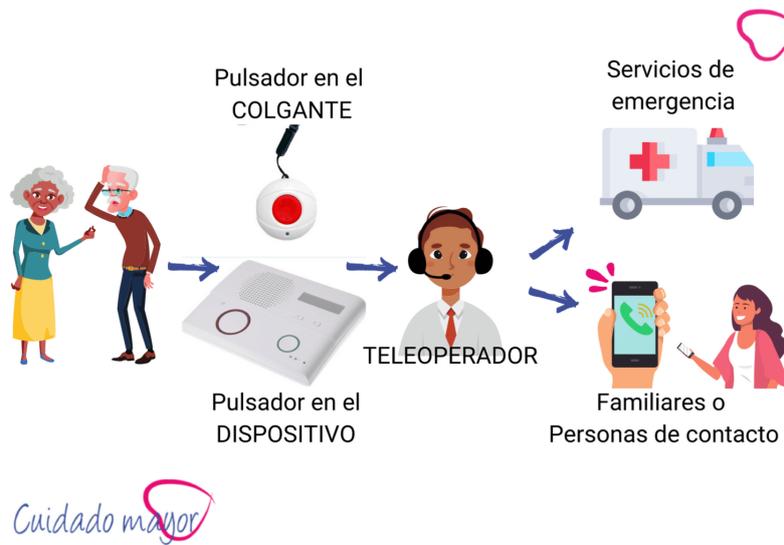


Figura 3. Mecanismo tele-asistencia [13]

V-B. ESP32

V-B1. Introducción ESP32: El microcontrolador ESP32, diseñado por “Espressif Systems”, una empresa tecnológica china, es una solución autónoma para redes WiFi. Este dispositivo actúa como un puente entre microcontroladores tradicionales y redes WiFi, además de ser capaz de ejecutar aplicaciones de forma independiente. Aunque fue presentado como una novedad tecnológica, su producción en masa comenzó a finales de 2016, lo que lo convierte en un procesador relativamente reciente [15].

Existen diversos modelos de placas con ESP32, pero debido a su tamaño compacto, conectar cables directamente al microcontrolador para usarlo en placas de pruebas es complicado para los aficionados. Para resolver este problema, los OEM adquieren los circuitos integrados en grandes cantidades, diseñan circuitos básicos, producen placas de circuito impreso y ensamblan placas listas para su uso inmediato, con los microcontroladores preinstalados [15].

El ESP32 es un microcontrolador que permite realizar una variedad de tareas en el mundo de la electrónica. Se podría decir que es una pequeña computadora por que cuenta con múltiples funcionalidades integradas [16] como se muestra en la figura 4.

Un dispositivo microcontrolador nuevo ha ingresado al mundo de la electrónica, bajando los costos de sistemas que antes eran complejos en su desarrollo, como el llamado IoT (Internet of Things) y la automatización de casas (Home Automation). Se trata del nuevo NodeMCU en su versión ESP32 [16].

Sus aplicaciones típicas se encuentran en el llamado ‘Internet of Things’ es decir, la supervisión y el control remoto de dispositivos ya sea a través de una red local LAN ó remotamente operado como servidores de internet (web server), para funciones de robótica y domótica [17].

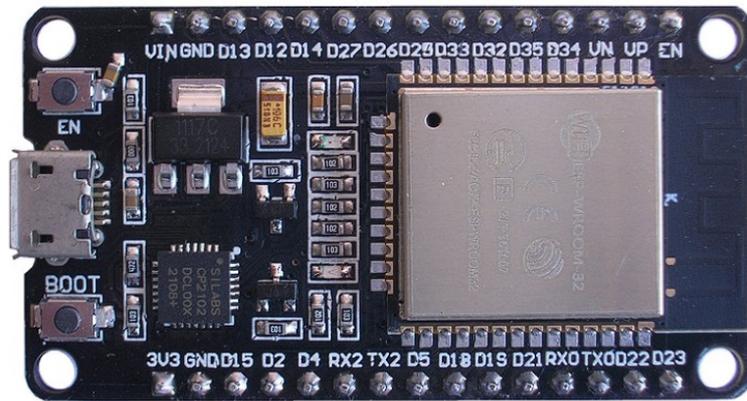


Figura 4. ESP-32 [16]

V-B2. Descripción del ESP32: El ESP32 es un dispositivo altamente integrado que incluye un switch de antena, balun para RF, amplificador de potencia, amplificador de recepción de bajo ruido, filtros y módulos de gestión de energía, todos completamente incorporados en el mismo chip. Diseñado específicamente para dispositivos móviles, así como para aplicaciones en electrónica e IoT (Internet de las Cosas), el ESP32 destaca por su bajo consumo energético gracias a sus funciones avanzadas de ahorro de energía. Estas incluyen la sintonización de reloj con alta resolución, modos de potencia variados y escalado dinámico de energía [18].

El NodeMCU ESP32 es una plataforma completa de desarrollo, módulo SoC (system on chip), basado en el microcontrolador Tensílica de 32 bits, funcionando a una frecuencia de reloj de 240 MHz, con una memoria Flash de 4MB y 500 KB de SRAM. Tiene conectividad inalámbrica Wifi, estándar 802.11 b/g/n y Bluetooth integrado BLE, en la figura 5 muestra IOS pines de conexión del ESP32. [16].

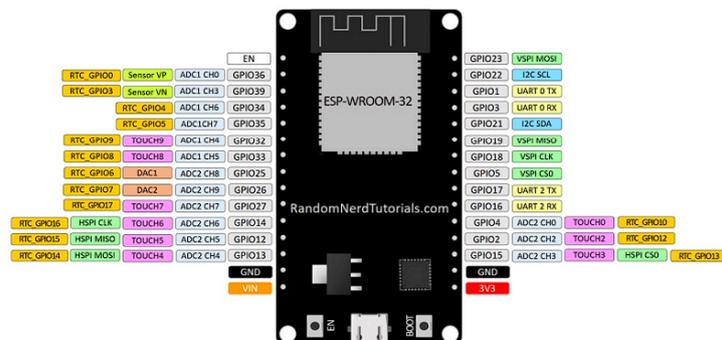


Figura 5. ESP-WROOM-3 datasheet [16]

Está diseñado para lograr el mejor rendimiento de potencia y RF, mostrando robustez, versatilidad y confiabilidad en una amplia variedad de aplicaciones y escenarios de energía [19]. ESP32 es un único chip combinado de Wi-Fi y Bluetooth de 2,4 GHz diseñado con el chip de bajo consumo de 40 nm de TSMC [19].

V-B3. ESP32 Wifi y Bluetooth en un solo chip: La evaluación de la comunicación entre microcontroladores ESP32 busca determinar la eficacia con la que estos dispositivos pueden interactuar entre sí. Este análisis se lleva a cabo mediante pruebas prácticas en entornos controlados, donde los microcontroladores emplean protocolos como

Wi-Fi, Bluetooth o incluso redes de área local (LAN) para establecer comunicación [20].

ESP32 es una familia de microcontroladores de la empresa Espressif Systems. Su analogía más clara es la de un ESP8266 que incluye Wifi, Bluetooth y otras potentes características extra [17].

La cantidad de fabricantes que trabajan sobre esta plataforma y la información dispersa puede ser abrumadora. Por ello, se debe de conocer todas sus cualidades y elegir la placa de desarrollo que más convenga al trabajo a realizar [17].

A continuación se observa como se diferencian las diversas presentaciones del ESP32, como pueden ser [17]:

- Chips: SoC,Sip como se observa en la figura 6.
- Módulos
- Placas de desarrollo

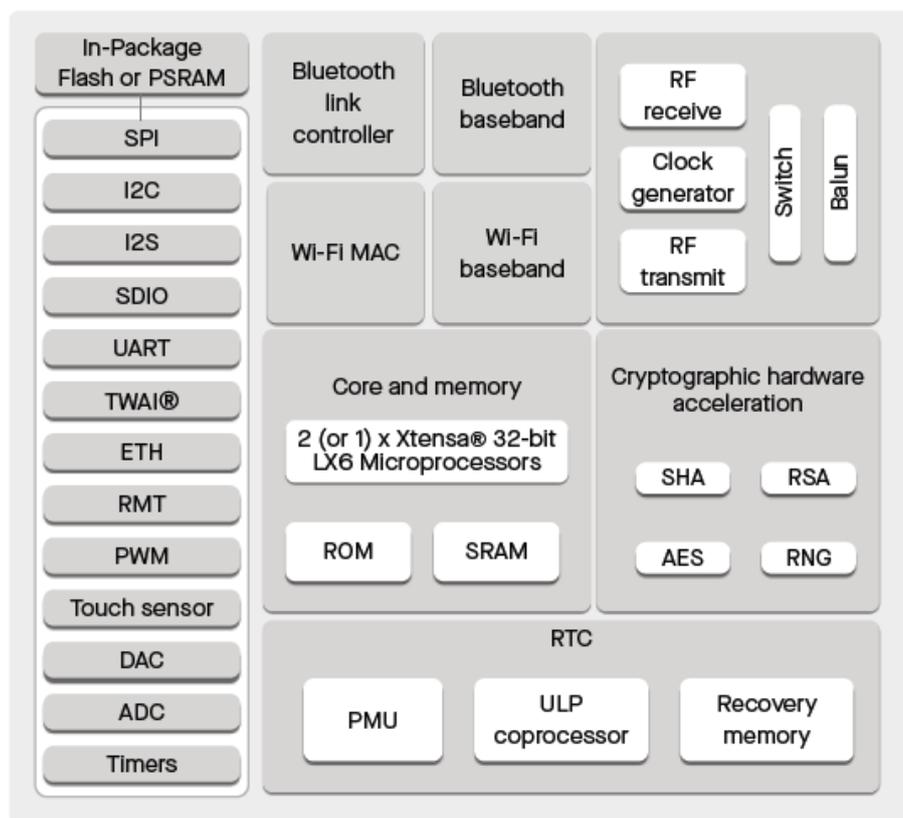


Figura 6. Diagrama de bloques del ESP32 [19]

V-C. Sensor MPU6050

V-C1. *MPU6050 Arduino, Acelerómetro y Giroscopio:* El MPU6050 es un sensor y acelerómetro para Arduino con 6 grados de libertad. El propósito del sensor consiste de dos partes, la primera será mostrar lo que es un acelerómetro y giroscopio y la lectura de valores del MPU-6050 utilizando un Arduino Uno. También se mostrarán

las mediciones en el monitor serial [21].

Como se menciona y se observa en la figura 7 el sensor MPU6050 posee 6 grados de libertad: Acelerómetro de 3 ejes y Giroscopio de 3 ejes, su alimentación es de 3.3v pero gracias a que tiene un regulador se puede alimentar directamente con 5v. Su comunicación es a través del protocolo I2C y a una velocidad máxima de 400Khz, la version MPU-600 posee el protocolo SPI [22].



Figura 7. Sensor MPU6050 [22]

V-C2. *Sensor Giroscópico:* Un giroscopio es un dispositivo que funciona para medir velocidades angulares basándose en el mantenimiento del impulso de rotación como se muestra en la siguiente tabla I. Si se intenta hacer girar un objeto que este en movimiento sobre un eje que no es el eje sobre el que está rotando, el objeto ejercerá un momento contrario al movimiento con el fin de preservar el impulso de rotación total. El giroscopio muestra el cambio de rango en rotación en sus ejes X, Y y Z [21].

V-C3. *Sensor Acelerómetro:* Mide la aceleración, inclinación y transforma la magnitud física de aceleración en otra magnitud eléctrica que sera la que se emplea en los equipos de adquisición estándar. Los rangos de medida van desde las décimas de grados hasta los miles de grados [21] como se muestra en la tabla I.

El circuito integrado MPU-6050 contiene un acelerómetro y giroscopio MEMS en un solo empaque. Cuenta con una resolución de 16-bits, lo cual significa que divide el rango dinámico en 65536 fracciones, estos aplican para cada eje X, Y y Z al igual que en la velocidad angular. El sensor es ideal para diseñar control de robótica, medición de vibración, sistemas de medición inercial (IMU), detector de caídas, sensor de distancia y velocidad. El MPU-6050 contiene un giroscópico, un acelerómetro, además de un sensor de temperatura, mediante I2C regresa unos valores conocidos como raw o “crudos” según el registro seleccionado [21].

Tabla I
 TABLA DE DATO DE MOVIMIENTO DEL EL SENSOR MPU6050

Rango escala completa Giroscopio	Sensibilidad de Giroscopio	Rango escala completa Acelerometro	Sensibilidad del Acelerómetro
±250	131	±2	16384
±500	65.5	±4	8192
±1000	32.8	±	4096
±2000	16.4	±16	2048

V-C4. *Especificaciones del MPU6050:*

- Salida digital de 6 ejes[21]
- Giroscopio con sensibilidad de ±250, ±500, ±1000 y ± 2000dps [21].
- Acelerómetro con sensibilidad de ±2g, ±4g ,±8g y ±16g [21].
- Algoritmos embebidos para calibración [21].
- Sensor de Temperatura digital [21].
- Entrada digital de video FSYNC [21].
- Interrupciones programables [21].
- Voltaje lógico:1.8±5 % o VDD [21].
- Voltaje de alimentación: 2.37 a 3.46V [21].
- 10000g tolerancia de aceleración máxima [21].

V-D. *Iot o Internet de las cosas*

V-D1. *Introducción:* El Internet de las cosa se define como una red de objeto físico, incluyendo dispositivos , instrumentos, vehículos ademas de otros elementos equipados con componentes electrónicos,, circuitos, software, sensores y conectividad de red. Esta tecnología permite la recopilación e intercambio de datos de manera autónoma. Donde IoT facilita la detección y el control remoto de objetos mediante la infraestructura de red existente, lo cual genera nuevas oportunidades para integrar el mundo físico con los sistemas informáticos. Esto resulta en una mejora significativa en términos de eficiencia y precisión en diversos procesos [23].

V-D2. *Concepto de Iot:* El IoT es capaz de interactuar sin intervención humana. Las tecnologías del IoT están en sus etapas iniciales sin embargo, se han producido muchos avances nuevos en la integración de objetos con sensores en Internet. El desarrollo del IoT implica muchas cuestiones, como infraestructura, comunicaciones, interfaces, protocolos y estándares [23].

Kevin Ashton fue el primero en proponer el concepto de IoT en 1999, y se refirió a IoT como objetos conectados que se pueden identificar de forma única mediante tecnología de identificación por radiofrecuencia (RFID). Sin embargo, la definición exacta de IoT todavía está en proceso de elaboración y depende de las perspectivas que se adopten como se muestra en la figura 8 hasta la actualidad se definió generalmente como “infraestructura de red

global dinámica con capacidades de auto-configuración basadas en estándares y protocolos de comunicación” [23].

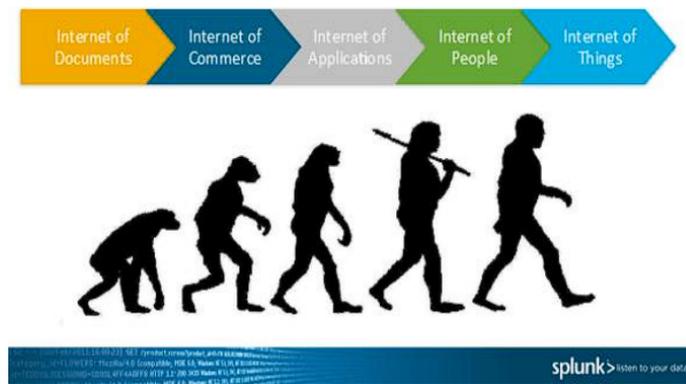


Figura 8. Evolución de Internet [23]

V-E. Software Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial que se usa en el desarrollo de apps para Android. Basado en el potente editor de código y las herramientas para desarrolladores de IntelliJ IDEA, como se muestra en la figura 9 Android Studio ofrece aún más funciones que mejoran tu productividad cuando compilas apps para Android, como las siguientes [24]:

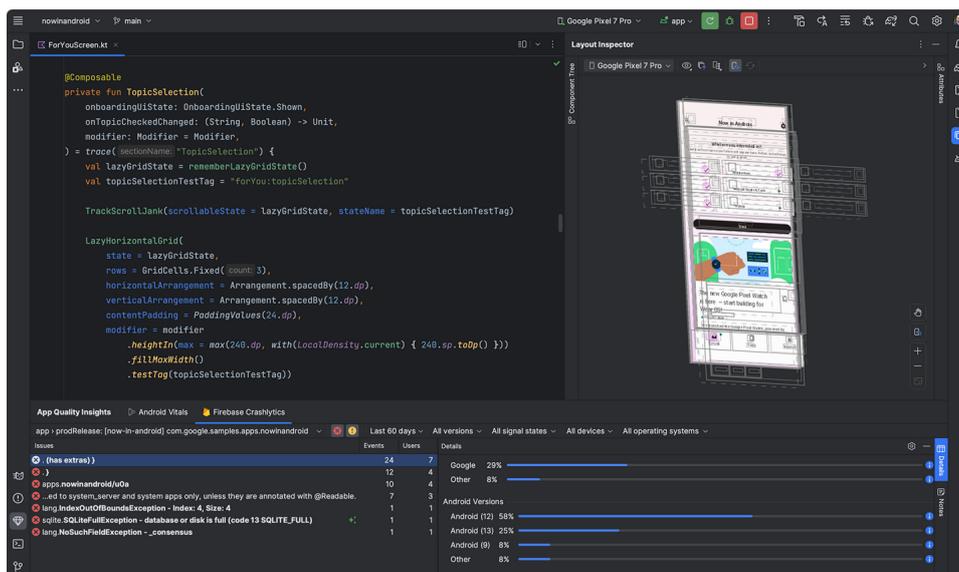


Figura 9. Software Android estudio [24]

- Sistema de compilación flexible basado en Gradle [24].
- Emulador rápido y cargado de funciones [24].
- Entorno unificado donde puedes desarrollar para todos los dispositivos Android [24].

- Ediciones en vivo para actualizar elementos compuestos en emuladores y dispositivos físicos, en tiempo real [24].
- Integración con GitHub y plantillas de código para ayudar a compilar funciones de apps comunes y también importar código de muestra [24].
- Variedad de marcos de trabajo y herramientas de prueba [24].
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones [24].
- Compatibilidad con C++ y NDK [24].
- Compatibilidad integrada con Google Cloud Platform, que facilita la integración con Google Cloud Messaging y App Engine [24].

V-E1. Estructura de un proyecto en Android Studio: Un proyecto en Android Studio se compone de uno o más módulos que contienen archivos de código fuente y recursos. Entre los módulos más comunes están [24]:

- Módulos de aplicaciones Android [24].
- Módulos de bibliotecas [24].
- Módulos de Google App Engine[24].

Por defecto, los archivos del proyecto se presentan en la vista Android, organizada en módulos para facilitar el acceso a los archivos principales. Los scripts de compilación se encuentran en el nivel superior, dentro de la sección Gradle Scripts [24].

Cada módulo incluye las siguientes carpetas:

- manifests: Contiene el archivo AndroidManifest.xml [24].
- java: Incluye los archivos fuente en Kotlin o Java, además de las pruebas JUnit [24].
- res: Almacena recursos no relacionados con código, como cadenas de texto e imágenes [24].

Si necesitas visualizar la estructura real de los archivos, selecciona Project en el menú en lugar de Android [24] como lo muestra la figura10.

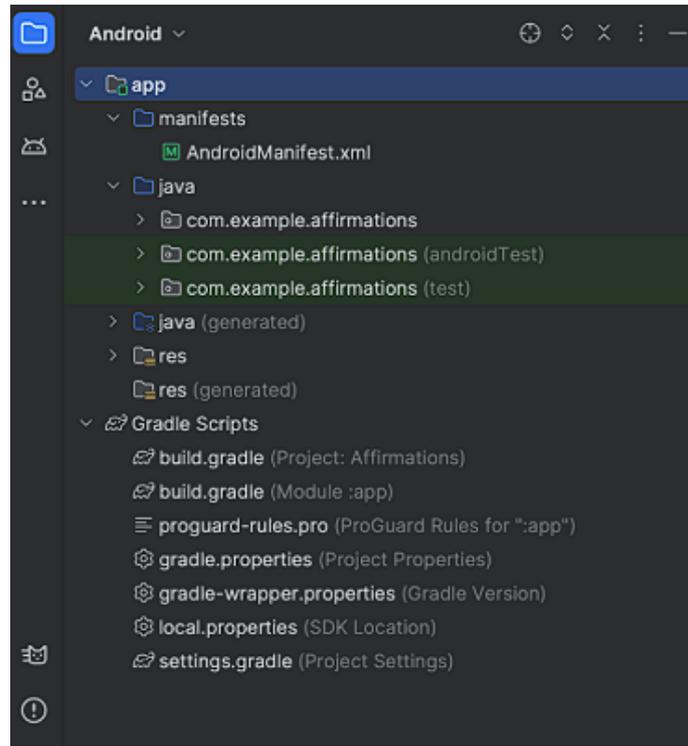


Figura 10. Archivos de proyecto de Android [24]

V-E2. Sistema de compilación de Gradle: Android Studio utiliza Gradle como su sistema de compilación base, complementado por el plugin de Android para Gradle, que ofrece funciones específicas para aplicaciones Android. Este sistema puede ejecutarse desde la interfaz del IDE o mediante comandos en la línea de comandos. Con Gradle, es posible [24]:

- Personalizar y extender el proceso de compilación [24].
- Generar múltiples APKs para una app, manteniendo el mismo proyecto y módulos [24].
- Reutilizar código y recursos entre diferentes conjuntos de orígenes [24].

Los archivos de configuración se denominan `build.gradle.kts` al usar Kotlin o `build.gradle` si se utiliza Groovy. Cada proyecto tiene un archivo de nivel superior y archivos individuales para cada módulo. Android Studio genera automáticamente estos archivos al importar un proyecto existente [24].

V-F. Variantes de compilación y APKs múltiples

El sistema de Gradle permite crear distintas versiones de una misma app desde un solo proyecto, lo que es útil para ofrecer versiones gratuitas y de pago o para generar APKs específicos para diferentes configuraciones de dispositivos. También se pueden crear APKs múltiples basados en densidades de pantalla o arquitecturas ABI, optimizando recursos y compartiendo configuraciones comunes como pruebas y reglas de ProGuard [24].

V-G. Administración de dependencias

Las dependencias de un proyecto se declaran en los scripts de compilación de cada módulo. Gradle las localiza y las integra automáticamente en la compilación [24]. Pueden ser:

- Dependencias entre módulos [24].
- Dependencias binarias remotas [24].
- Dependencias binarias locales [24].

V-G1. Herramientas integradas de depuración y análisis: Android Studio ofrece herramientas avanzadas para depurar y optimizar el rendimiento del código :

- Depuración integrada: Permite verificar variables, referencias y expresiones directamente en el editor, mostrando valores de retorno y expresiones lambda [24].
- Generador de perfiles de memoria: Ayuda a rastrear asignaciones de memoria y optimizar el uso de recursos al analizar las acciones específicas que las generan [24].

V-G2. Inspecciones de código: Automatiza análisis de calidad estructural con herramientas como lint, que detecta problemas relacionados con precisión, rendimiento, seguridad, accesibilidad e internacionalización [24] como se muestra en la figura 11.

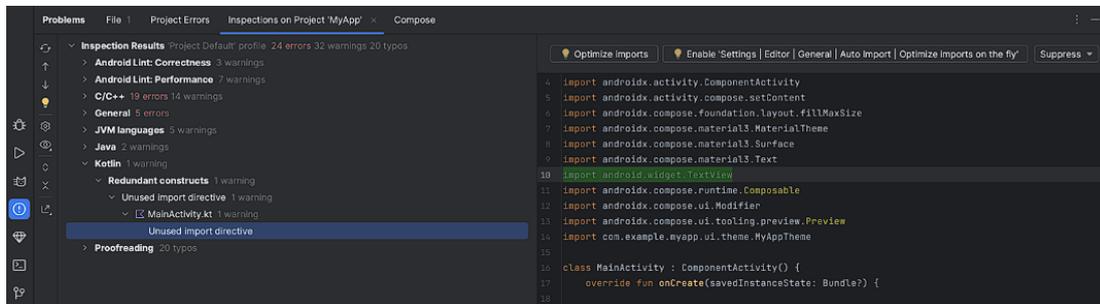


Figura 11. Inspección de lint [24]

V-H. Software Arduino studio

V-H1. Introducción Arduino Studio: Arduino es una plataforma electrónica de código abierto basada en hardware y software fáciles de usar. Las placas Arduino pueden leer entradas y convertirlas en una salida indicando en la placa qué hacer enviando un conjunto de instrucciones al microcontrolador de la placa. Para ello, utiliza el lenguaje de programación Arduino (basado en Wiring) y el software Arduino (IDE) , basado en Processing [25].

Arduino nació en el Ivrea Interaction Design Institute como una herramienta sencilla para prototipado rápido, dirigida a estudiantes sin conocimientos previos de electrónica y programación. En cuanto llegó a una comunidad más amplia, la placa Arduino empezó a cambiar para adaptarse a nuevas necesidades y desafíos, diferenciando su oferta desde simples placas de 8 bits hasta productos para aplicaciones IoT, wearables, impresión 3D y entornos embebidos [25].

V-H2. Funcionamiento de Arduino Studio: El Entorno de Desarrollo Integrado (IDE) de Arduino, también conocido como Arduino Software, ofrece un editor de texto para escribir código, una consola para mensajes y texto, una barra de herramientas con accesos directos a funciones frecuentes y varios menús como se muestra en la figura 12. Este entorno permite interactuar con el hardware de Arduino para cargar programas y establecer comunicación con el dispositivo. [25].

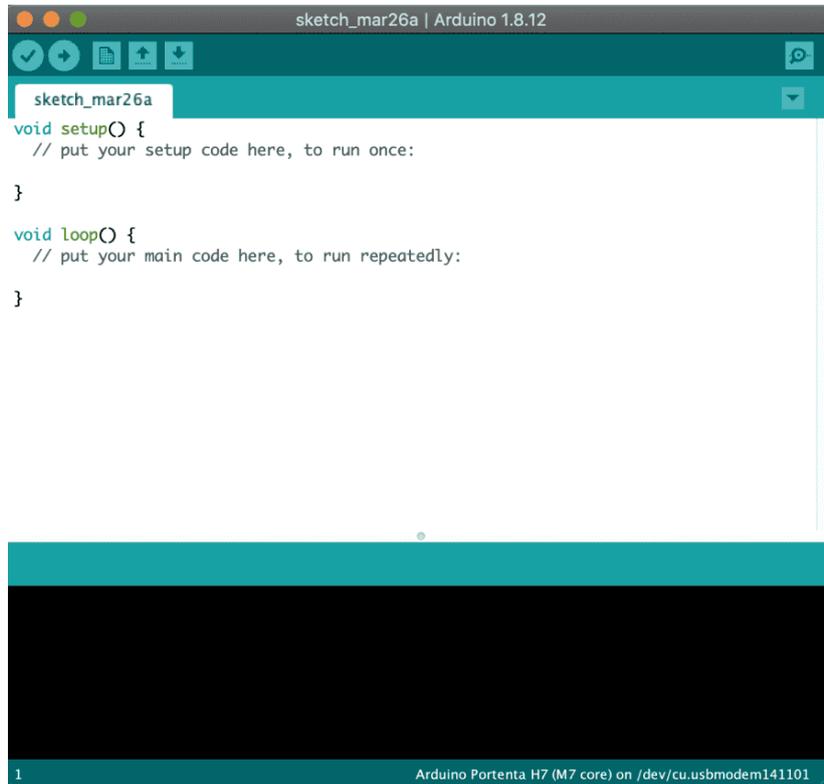


Figura 12. Editor de texto [25]

El editor del IDE de Arduino incluye herramientas para cortar, pegar, buscar y reemplazar texto. El área de mensajes muestra información sobre los procesos de guardado y exportación, además de notificar errores. Por su parte, la consola presenta la salida de texto generada por el software, incluyendo mensajes de error detallados y otra información relevante. En la esquina inferior derecha de la ventana se visualizan la placa configurada y el puerto serie. Los botones de la barra de herramientas facilitan tareas como verificar y cargar programas, así como crear, abrir, guardar bocetos y acceder al monitor serie [25].

V-H3. La bibliotecas de Arduino: Las bibliotecas ofrecen funciones adicionales para los bocetos, como herramientas para manejar hardware o procesar datos. Para utilizarlas, se deben seleccionar desde el menú Boceto ¿Importar biblioteca, lo que agregará automáticamente declaraciones '#include' en la parte superior del código y vinculará la biblioteca al boceto durante la compilación. Sin embargo, al ser cargadas junto con el boceto en la placa, incrementan el espacio que ocupa el programa. Si una biblioteca deja de ser necesaria, basta con eliminar las declaraciones '#include' correspondientes del código [25].

En la referencia se encuentra una lista de bibliotecas disponibles. Algunas vienen integradas con el software de Arduino, mientras que otras pueden descargarse desde diversas fuentes o gestionarse mediante el Administrador de bibliotecas. Desde la versión 1.0.5 del IDE, también es posible importar bibliotecas directamente desde archivos ZIP para usarlas en los bocetos [25].

V-H4. Instalación de bibliotecas: Para instalar una nueva biblioteca en el IDE de Arduino, puede utilizar el Administrador de bibliotecas (disponible a partir de la versión IDE 1.6.2). Abra el IDE y haga clic en el menú "Sketch" y luego en Include Library Manage Libraries [25] como los muestra la figura 13.

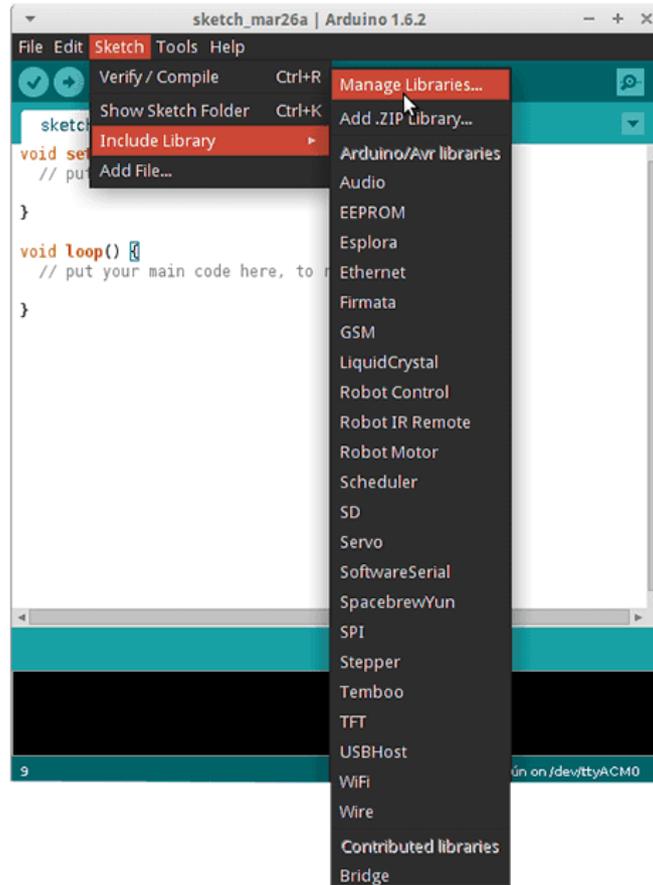


Figura 13. Librería de Arduino [25]

Al abrir el Administrador de bibliotecas, se mostrará una lista de librerías disponibles para instalación. Por ejemplo, para instalar la biblioteca **Bridge** se debe buscar su nombre en la lista, haz clic en ella y selecciona la versión que deseas instalar. En algunos casos, solo hay una versión disponible, por lo que el menú de selección de versiones puede no aparecer. Esto es completamente normal y no requiere ninguna acción adicional [25] como se observa en la figura 14 .

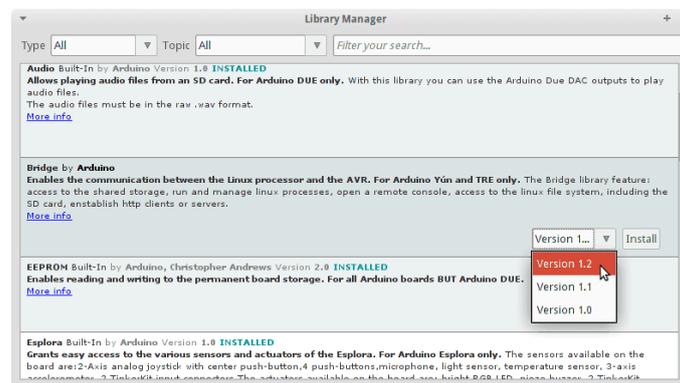


Figura 14. Lista de bibliotecas [25]

V-H5. *Arduino IDE 2* : Como se muestra en la Figura 15, Arduino IDE 2 es una versión mejorada del entorno clásico, que ofrece un rendimiento optimizado, una interfaz de usuario renovada y nuevas funcionalidades, como autocompletado, un depurador integrado y la sincronización de bocetos con Arduino Cloud [25]. Arduino IDE 2 incorpora una nueva barra lateral diseñada para facilitar el acceso rápido a las herramientas más utilizadas, mejorando la experiencia del usuario [25].

- Verificar / Cargar: compila y carga tu código en la placa de Arduino [25].
- Seleccionar placa y puerto: Las placas Arduino detectadas se muestran automáticamente en esta sección, junto con el número de puerto correspondiente [25].
- Administrador de placas: Facilita la búsqueda e instalación de paquetes tanto de Arduino como de terceros. Por ejemplo, para usar una placa MKR WiFi 1010, se necesita instalar el paquete Arduino SAMD Boards [25].
- Administrador de bibliotecas: Ofrece acceso a miles de bibliotecas desarrolladas por Arduino y su comunidad [25].
- Depurador: Permite probar y depurar programas en tiempo real [25].
- Buscar: Herramienta para localizar palabras clave dentro del código [25].
- Abrir Serial Monitor: Inicia el Monitor Serial como una nueva pestaña en la consola [25].

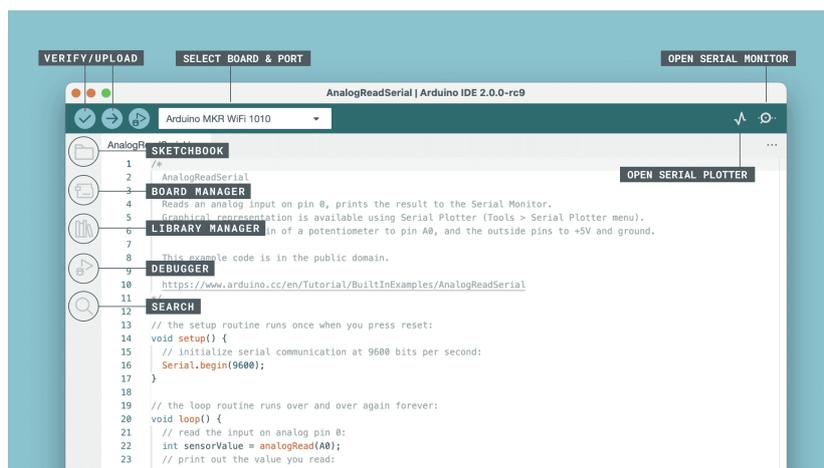


Figura 15. Interfaz de IDE 2 de Arduino [25]

Esta guía aborda los aspectos fundamentales de Arduino IDE 2 e incluye enlaces a recursos más detallados para aprender a utilizar sus funciones específicas [25].

V-H6. *Características de Arduino IDE2*: El cuaderno de bocetos es donde se almacenan los archivos de código. Los bocetos de Arduino se guardan como archivos en blanco y deben almacenarse en una carpeta con el mismo nombre. Por ejemplo, un boceto llamado my-sketch.ino debe almacenarse en una carpeta llamada my-sketch. Normalmente, sus bocetos se guardan en una carpeta llamada Arduino en la carpeta de documentos. Para acceder al cuaderno de bocetos, haga clic en el ícono de carpeta ubicado en la barra lateral [25].

V-I. *Software Wokwi*

Wokwi es una fascinante herramienta consistente en un simulador de proyectos Arduino, que también sirve para microcontroladores ESP32 y STM32. Puede simular visualmente mediante objetos y conexiones móviles las placas,

cables, componentes, ejecutar el código y mostrar resultados. Es como utilizar Arduino pero sin tener una placa, simplemente visualizando qué es lo que sucedería [26].

Además, resulta ideal para el mundo educativo, porque permite hacer cosas con los componentes electrónicos sin disponer de ellos físicamente, por consiguiente, puede ayudar ahorrar tiempo y dinero[26].

V-J. Postman

Postman es una herramienta que nos permite diseñar y documentar API (Interfaz de programación de aplicaciones), donde simplifica cada paso del ciclo de vida de las API y optimiza la colaboración para que permita que las aplicaciones intercambien datos, funciones y características como la visualización de coordenadas de una IP[27]. Postman en sus inicios nace como una extensión que podía ser utilizada en el navegador Chrome de Google y básicamente nos permite realizar peticiones de una manera simple para testear APIs de tipo REST propias o de terceros [28].

V-J1. Para que sirve el Postman: Postman sirve para múltiples tareas, dentro de las cuales se destacan en esta oportunidad las siguientes [28].

- Testear colecciones o catálogos de APIs tanto para Frontend como para Backend [28].
- Organizar en carpetas, funcionalidades y módulos los servicios web [28].
- Permite gestionar el ciclo de vida (conceptualización y definición, desarrollo, monitoreo y mantenimiento) de nuestra API [28].
- Generar documentación de nuestras APIs [28].

V-J2. Métodos más utilizados: Postman cuenta con una serie de métodos que permiten tomar acción ante peticiones. A continuación se muestra en la figura 16, te dejamos los más utilizados [28] :

- GET: Obtener información [28].
- POST: Agregar información [28].
- PUT: Reemplazar la información [28].
- PATCH: Actualizar alguna información [28].
- DELETE: Borrar información [28].

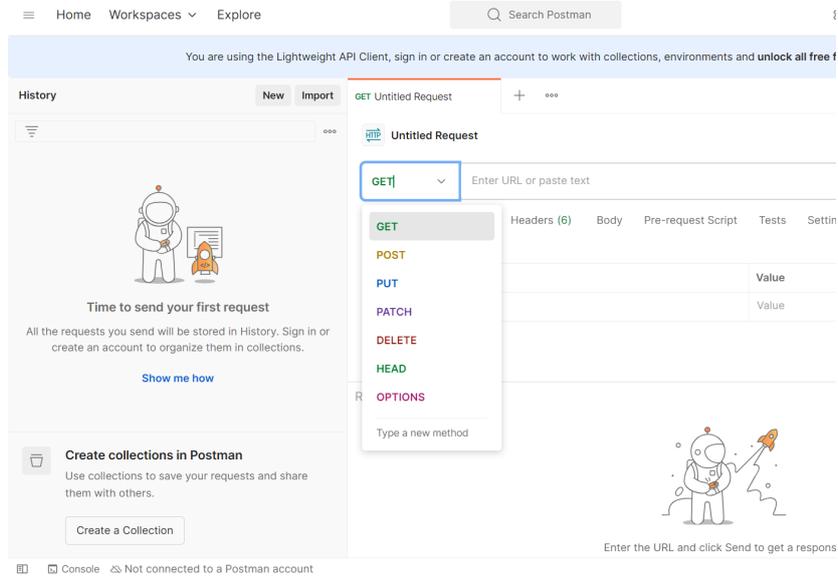


Figura 16. Metodos utilizado en el postman [28]

VI. MARCO METODOLÓGICO

La base de este proyecto es de diseño e investigación, mediante la recopilación y análisis de datos medibles y parametrizables para evaluar la eficacia de un sistema de emergencias basado en tele-asistencia. Este sistema no solo detecta accidentes como caídas en tiempo real, sino que también permite una notificación eficiente de cualquier incidente de emergencia al personal encargado. Este enfoque es crucial para obtener resultados objetivos que respalden la implementación de este sistema en el entorno universitario, asegurando que las pruebas empíricas constituyan el núcleo de la validación del proyecto.

Además, se adopta un enfoque exploratorio-descriptivo con el fin de identificar necesidades del grupo social del estudio. Esto incluye también la integración de herramientas tecnológicas que permitan mejorar la seguridad y respuesta ante diversos tipos de incidentes dentro de la comunidad universitaria.

VI-A. Fase de encuesta

El objetivo de las preguntas realizadas es recopilar información directa de la comunidad salesiana para entender sus necesidades, perspectivas y expectativas respecto a las tecnologías y sistemas que podrían mejorar su experiencia universitaria ante las adversidades cotidianas.

A continuación, se explican las preguntas planteadas junto con su relación e importancia dentro del proyecto:

- ¿Consideras necesario la implementación de tecnologías en la universidad que abarquen aspectos como seguridad, asistencia rápida, atención de calidad e inclusión para personas con movilidad reducida? Esta pregunta busca evaluar la percepción de los encuestados sobre la necesidad de implementar tecnologías que mejoren la seguridad y asistencia de personas con movilidad reducida dentro de la universidad.
- ¿Cuál de estas opciones consideras es de mayor importancia a tratar al momento de desarrollar tecnologías de asistencia remota (tele-asistencia)? Esta información permitirá priorizar los recursos y funcionalidades del sistema según las necesidades más relevantes para los usuarios, asegurando que se desarrolle una solución efectiva y alineada con las expectativas de la comunidad universitaria.
- ¿Qué tan importante consideras contar con un sistema de tele-asistencia que se encargue de monitorear en caso de emergencias a personas de movilidad reducida y a la comunidad salesiana en general a futuro? Esta pregunta mide el grado de importancia que los encuestados otorgan al sistema de tele-asistencia, lo que ayuda a cuantificar la percepción de valor del proyecto dentro de la universidad.
- ¿Crees que la implementación de un sistema de tele-asistencia para el monitoreo de emergencias tendría un impacto considerable en la comunidad universitaria? Si la mayoría responde afirmativamente, se valida el impacto positivo del proyecto. En caso contrario, las respuestas pueden aportar ideas sobre posibles mejoras o factores que podrían limitar su efectividad.
- ¿Qué características consideras necesarias para el desarrollo de un proyecto como el mencionado en esta encuesta? Esta pregunta busca recopilar información sobre las expectativas y necesidades específicas de la comunidad universitaria respecto a la tele-asistencia.
- ¿Consideras que la universidad muestra un apoyo completo en este tipo de temas? Esta pregunta evalúa la percepción de la comunidad universitaria sobre el nivel de compromiso institucional en la implementación de tecnologías que mejoren la seguridad, inclusión y asistencia para personas con movilidad reducida.

VI-B. Fase de investigación

VI-B1. Análisis de la necesidad del proyecto y la solución tecnológica innovadora a aplicar: La implementación busca dar respuesta a una necesidad dentro del entorno, ofreciendo una solución práctica y eficaz que prioriza el bienestar de quienes enfrentan desafíos en su movilidad. Esta iniciativa surge tras analizar estudios que evidencian que las personas con movilidad reducida son más propensas a enfrentar situaciones de riesgo. Además, en muchas ocasiones, su capacidad para comunicarse de manera rápida y efectiva durante una emergencia puede verse limitada, lo que incrementa la vulnerabilidad en este tipo de circunstancias.

Se eligió el entorno universitario por ser un entorno diverso y dinámico, donde cada interacción ofrece valiosas oportunidades para perfeccionar el proyecto y entender mejor las necesidades reales de la comunidad. Este esfuerzo no solo busca resolver un problema, sino también transformar el lugar en un espacio más inclusivo y preparado para atender las diferencias individuales.

La solución propuesta es el uso de dispositivos portátiles que se conectan a sensores ubicados en el entorno, permitiendo detectar cualquier incidente y activar alertas automáticamente. Estas alertas se gestionan a través de una plataforma centralizada que coordina la respuesta y asegura una intervención adecuada y eficiente. El objetivo es no solo garantizar una respuesta rápida, sino también crear un entorno más inclusivo, proporcionando mayor confianza y tranquilidad a quienes se encuentran en situaciones vulnerables.

VI-B2. Normas ISO: Se utilizan para garantizar que el proyecto cumpla con estándares internacionales de calidad, seguridad y accesibilidad en todas sus etapas de desarrollo. Estas directrices permiten establecer procesos bien definidos, asegurar la conexión entre los componentes del sistema, gestionar los riesgos de manera efectiva y desarrollar soluciones que sean funcionales, accesibles y seguras para los usuarios. Además, aseguran que el diseño y la implementación se realicen de forma estructurada y cumplan con los requisitos específicos, optimizando la experiencia del usuario y promoviendo la confiabilidad del sistema final.

Las normas con las que se diseñó el sistema son:

- ISO 9001:2015: asegura la gestión de calidad en cada fase, desde el diseño hasta la implementación y validación de este.
- ISO/IEC 25010:2011: se implementa para la definición de los requisitos de calidad del software desde la etapa de diseño.
- ISO/IEC 30141:2018: Proporciona la referencia de una estructura para el sistema de teleasistencia, asegurando interoperabilidad entre el dispositivo, la cámara y la app móvil.
- ISO 9241-171: Aplica en el diseño de la app móvil garantizando accesibilidad para personas con movilidad reducida.
- ISO 13485:2016: Esta norma regula los requisitos del sistema de gestión de calidad de cualquier dispositivo médico.
- ISO 14971:2019: Ayuda a la gestión de riesgo, esencial para identificar y mitigar los riesgos asociados con el diseño, los materiales y la funcionalidad del dispositivo de emergencia.
- ISO 9999:2022: Ayuda a la clasificación de productos de apoyo para personas con discapacidad, relevante para asegurar que el diseño del dispositivo sea apropiado y funcional.

VI-C. Fase del diseño del dispositivo de asistencia remota

VI-C1. Selección de componentes mediante comparación de las características: Durante el desarrollo del prototipo, fue necesario realizar una revisión exhaustiva de cada componente, incluyendo microcontroladores y sensores. Todo esto para seleccionar aquellos que cumplieran con los requisitos funcionales necesarios para la implementación del sistema IoT. Este proceso permitió asegurar que los componentes elegidos no solo respondieran a los objetivos del proyecto, sino que también optimizaran la eficiencia y la precisión del sistema en condiciones de uso real, como se muestra en la tabla II

Microcontroladores	Precio	Alimentación	Conectividad	Memoria	Temperatura de operación	Número de entradas
Raspberry Pi3 modelo B	60	5V via micro-USB o GPIO	Wi-Fi, Bluetooth, Ethernet	1 GB RAM	0°C a 50°C	40 GPIO
ESP32 WROOM-32E	22	3.3V	Wi-Fi, Bluetooth	520 KB SRAM + 4 MB Flash	-40°C a 85°C	34 GPIO
Arduino Uno	15	5V via USB o 7-12V via pin	Requiere módulo	2 KB SRAM, 32 KB Flash	-40°C a 85°C	14 digitales, 6 analógicos

Tabla II
COMPARACIÓN DE MICROCONTROLADORES PARA LA ELABORACIÓN DEL DISPOSITIVO

Tras analizar las comparaciones en la tabla anterior, se determina que el ESP32-WROOM es la opción más adecuada para el desarrollo de este proyecto. El microcontrolador de doble núcleo, con velocidades de hasta 240 MHz ofrece conectividad Wi-Fi y Bluetooth integrada, lo que garantiza una comunicación robusta y eficiente. Su bajo consumo energético lo hace ideal para sistemas de monitoreo continuo, mientras que su amplia capacidad de memoria y compatibilidad con sensores y periféricos permiten gestionar múltiples tareas de forma simultánea.

Sensor	Precio	Alimentación	Conexión	Rango seleccionable	Detección
MPU6500	12	2.4V - 3.6V	I2C, SPI	Acelerómetro: $\pm 2, \pm 4, \pm 8, \pm 16$ g; Giroscopio: $\pm 250, \pm 500, \pm 1000, \pm 2000$ dps	Detección de movimiento y orientación en 6 ejes
GY-521	4	5V	I2C	Acelerómetro: ± 2 g; Giroscopio: $\pm 250, \pm 500, \pm 1000, \pm 2000$ dps	Baja precisión; detección limitada
MPU6050	15	3.3V - 5V	I2C	Acelerómetro: $\pm 2, \pm 4, \pm 8, \pm 16$ g; Giroscopio: $\pm 250, \pm 500, \pm 1000, \pm 2000$ dps	Detección de movimiento y orientación en 6 ejes

Tabla III
COMPARACIÓN DE SENSORES DE EJES DE MOVIMIENTO

La evaluación de los distintos sensores como se muestra en la tabla III, permitió identificar el más adecuado para el proyecto: el sensor MPU6050. Este dispositivo destaca por sus características óptimas, entre las que se incluye una alimentación de hasta 5V, un requisito indispensable para el funcionamiento del prototipo. Además, la integración de un acelerómetro y un giroscopio permite una captura rápida y precisa de datos, lo que favorece una respuesta ágil al conectarlo con la aplicación.

Como se muestra en la tabla IV, se detallan las características técnicas evaluadas para seleccionar el modelo de cámara WIFI, más apropiada.

Cámara	Precio	Alimentación	Interfaz	Resolución	Calidad de imagen
Boykeep	25	5V USB o batería re-cargable	Wi-Fi, USB	1080p Full HD	Buena, clara en condiciones de luz media
V380 Pro Mini	15	5V USB	Wi-Fi, USB	480p	Baja, granulada y de baja nitidez
Mini Cámara Wi-Fi HD 360p	10	5V USB	Wi-Fi	360p	Muy baja, borrosa y sin detalles finos

Tabla IV
COMPARACIÓN DE CÁMARAS PARA EL SISTEMA IMPLEMENTADO

De las opciones disponibles, la cámara Boykeep se destaca como una de las alternativas más adecuadas para este proyecto, gracias a su capacidad de capturar imágenes en alta resolución, fundamental para lograr precisión en la detección y monitoreo de eventos críticos. Además, su compatibilidad con el módulo ESP32 simplifica la integración en el sistema IoT, garantizando una comunicación eficiente y estable con la aplicación. Su bajo consumo energético y rápida velocidad de procesamiento la convierten en una opción ideal para aplicaciones en tiempo real, cumpliendo con los requisitos de rendimiento y autonomía necesarios para el prototipo.

VI-C2. Diseño del circuito: El diseño del circuito como se observa en la figura 17, está basado en la integración del microcontrolador ESP32 WROOM-32 con el sensor MPU6050, permitiendo la detección de movimientos y caídas.

El ESP32 WROOM es el núcleo del dispositivo, que se utiliza para el procesamiento y transmisión de datos del sensor hacia la aplicación móvil que se realizó, mientras que el sensor MPU-6050 dispone de 6 grados de libertad, el cual tiene integrado un acelerómetro y giroscopio para la detección de cualquier cambio de movimiento del mismo. Adicional, el microcontrolador tiene la capacidad de conectarse fácilmente al WIFI.

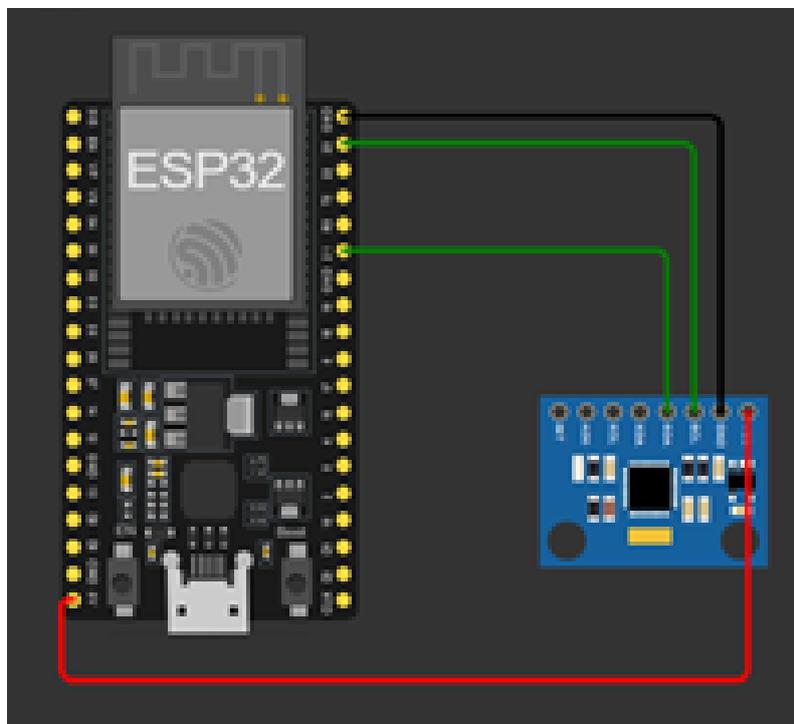


Figura 17. Circuito en wokwi para la implementación, Autor propio.

En la configuración del circuito se implementó un protocolo de comunicación I2C para la transferencia de datos entre el sensor y el ESP32, donde las conexiones se establecieron de la siguiente manera: El pin VCC del sensor MPU-6050 se conectó al pin 3.3V del ESP32, proporcionando un suministro estable. Por otro lado, el pin GND del MPU-6050 se conectó al pin GND del ESP32 como referencia a tierra.

Además, en la comunicación I2C se conectaron los pines SCL Y SDA, los cuales están conectados desde el sensor hacia el ESP32 en los pines D22 Y D21.

Una vez adquirida la instrumentación y los dispositivos necesarios para el sistema, se lleva a cabo la implementación del dispositivo IoT, verificado su correcto funcionamiento, en la fase relacionada con el hardware y software como se muestra en la figura 18.

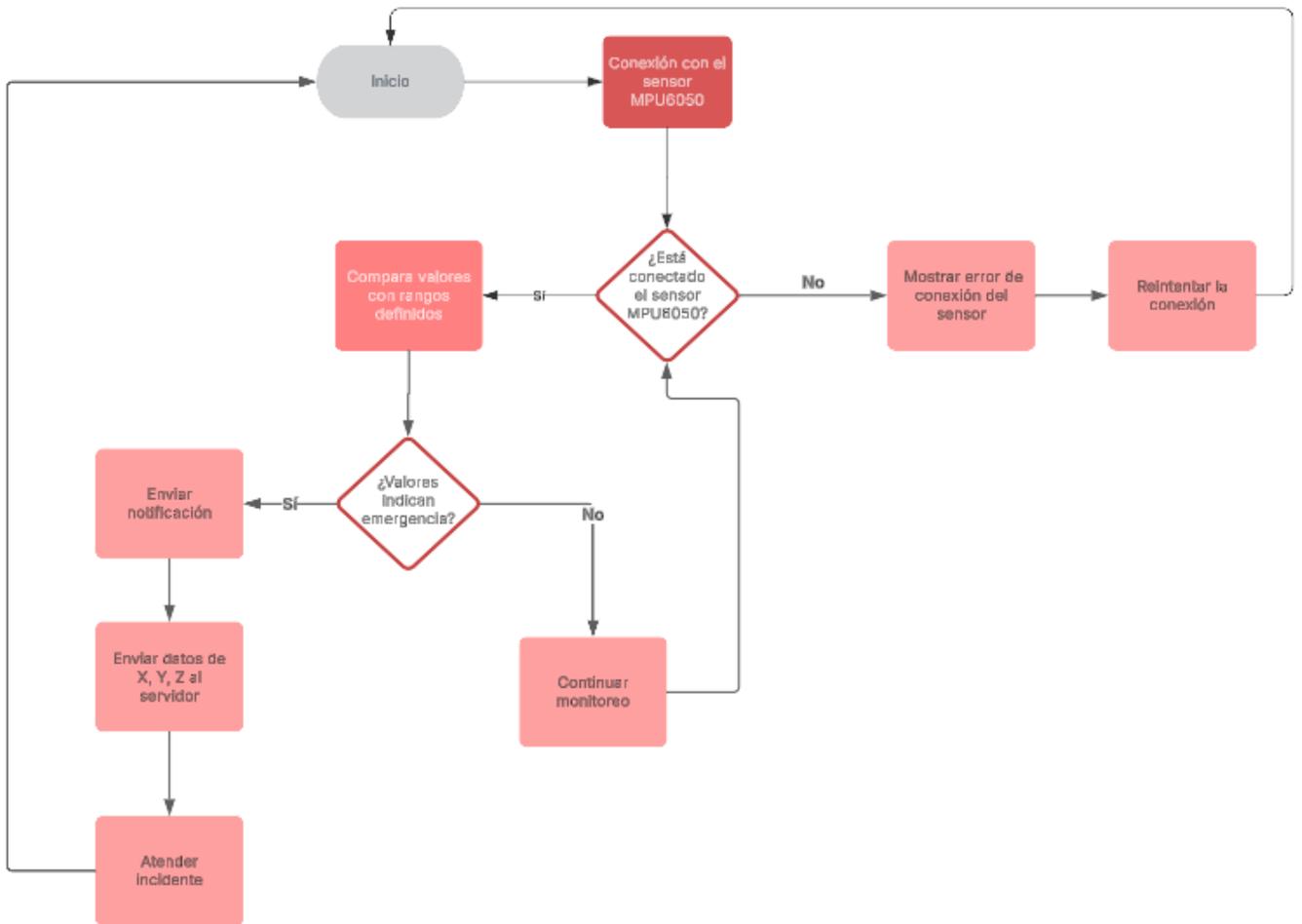


Figura 18. Diagrama de flujo de dispositivos, Autor propio

VI-C3. *Placa en pcb:* En la siguiente imagen se muestra la placa en el software se observa en siguiente figura 19:

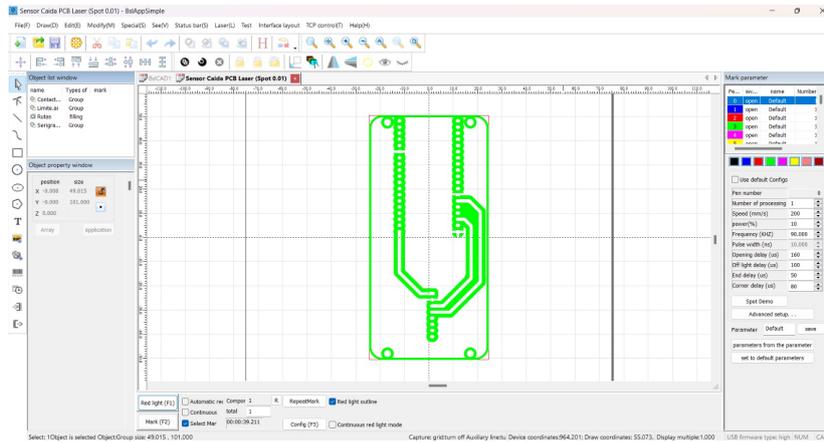


Figura 19. Placa PCB en 2D, Autor propio.

VI-C4. Simulación del circuito Arduino y post man: En esta sección se describe el funcionamiento del código implementado para la simulación del dispositivo que se observa en la figura 20 y 21, utilizando las aplicaciones de Arduino y Postman. El objetivo principal es establecer una conexión eficiente entre los componentes del sistema y visualizar los datos obtenidos del sensor MPU6050 en tiempo real.

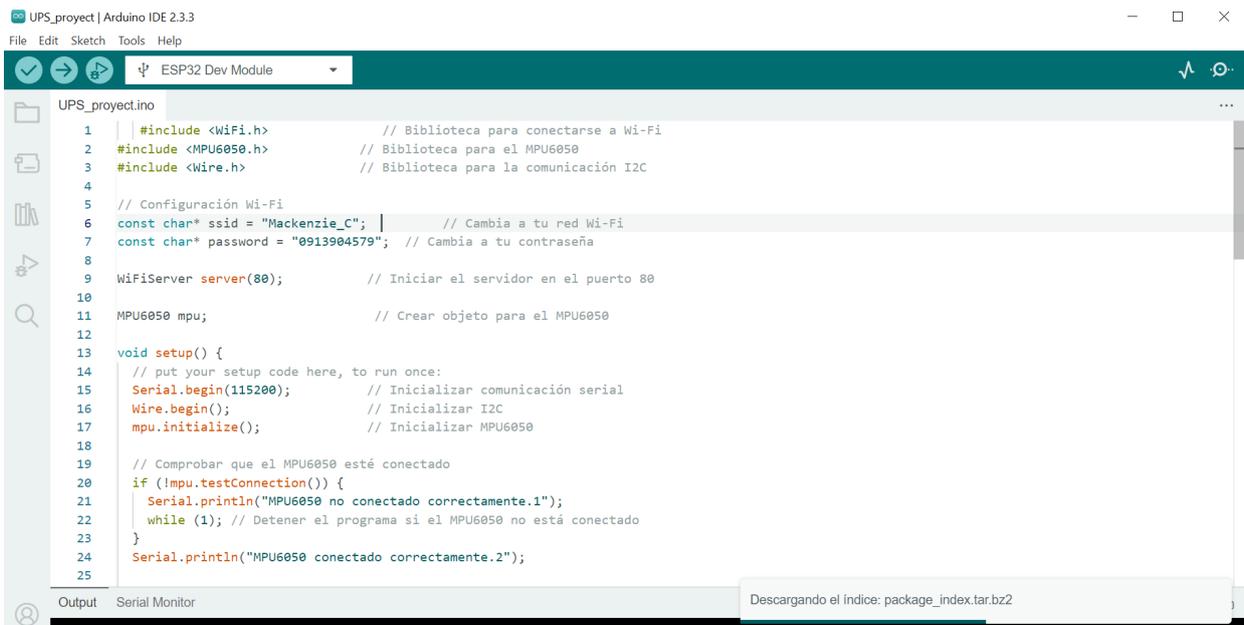


Figura 20. Visualización del código implementado en arduino para el proyecto, Autor propio.

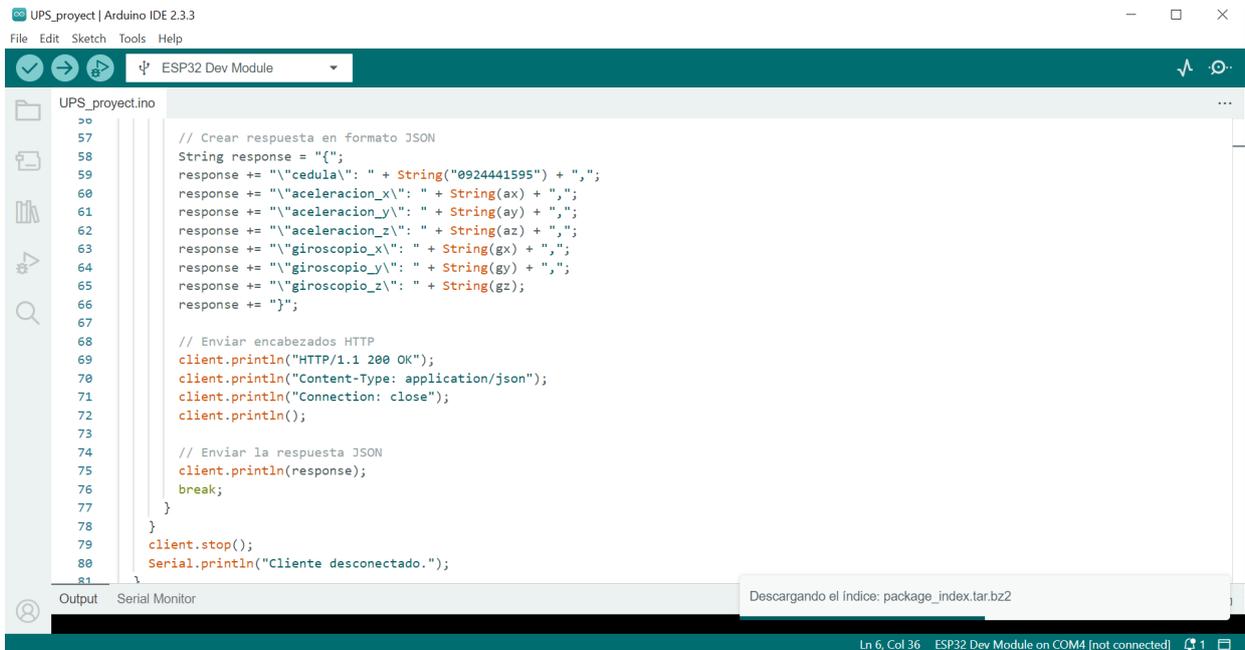


Figura 21. Visualización del código para la obtención de los ejes x,y, z. Autor propio.

El código comienza incluyendo las bibliotecas necesarias para la conexión a la red WiFi, como la biblioteca WiFi.h. Estas bibliotecas son esenciales para que el dispositivo pueda conectarse a una red inalámbrica y, posteriormente, generar una dirección IP, como se muestra en la figura 22, la cual es crucial, por que permite la comunicación entre el dispositivo y el simulador Postman, facilitando la simulación de los datos generados por el acelerómetro y el giroscopio.

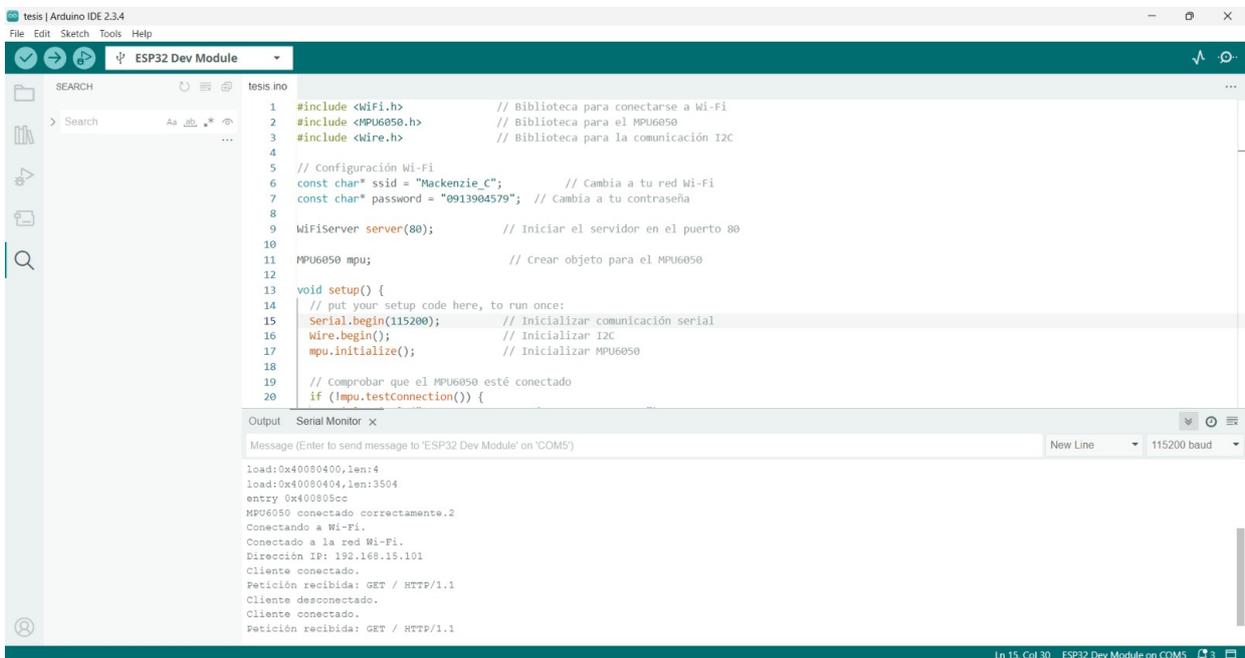


Figura 22. Visualización de la obtención de la IP del dispositivo, Autor propio.

Una vez establecida la conexión WiFi y obtenida la dirección IP, el código inicializa la comunicación serial entre el microcontrolador y el sensor MPU6050. Este proceso es llevado a cabo mediante funciones que configuran los parámetros del sensor, asegurando que esté correctamente conectado y listo para transmitir datos. La verificación de la conexión del sensor es una etapa crítica, por que garantiza la integridad de la información recopilada.

En el código se reflejan los valores correspondientes a los ejes X, Y y Z tanto del acelerómetro como del giroscopio. Estos datos se transmiten en tiempo real y pueden visualizarse mediante Postman, como se observa en la figura 23 y 24, permitiendo evaluar el comportamiento del dispositivo bajo diferentes condiciones.

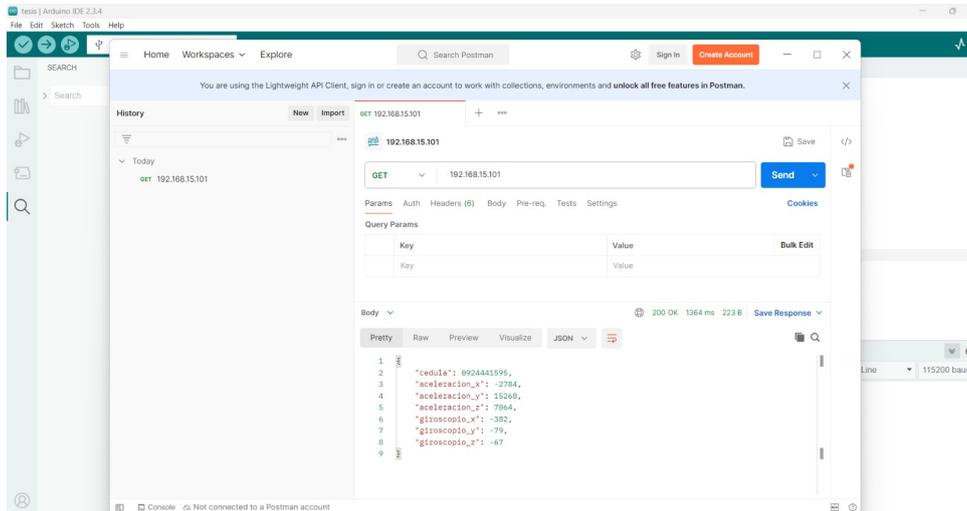


Figura 23. Visualización de la obtención de las coordenadas mediante el simulador Post Man, Autor propio.

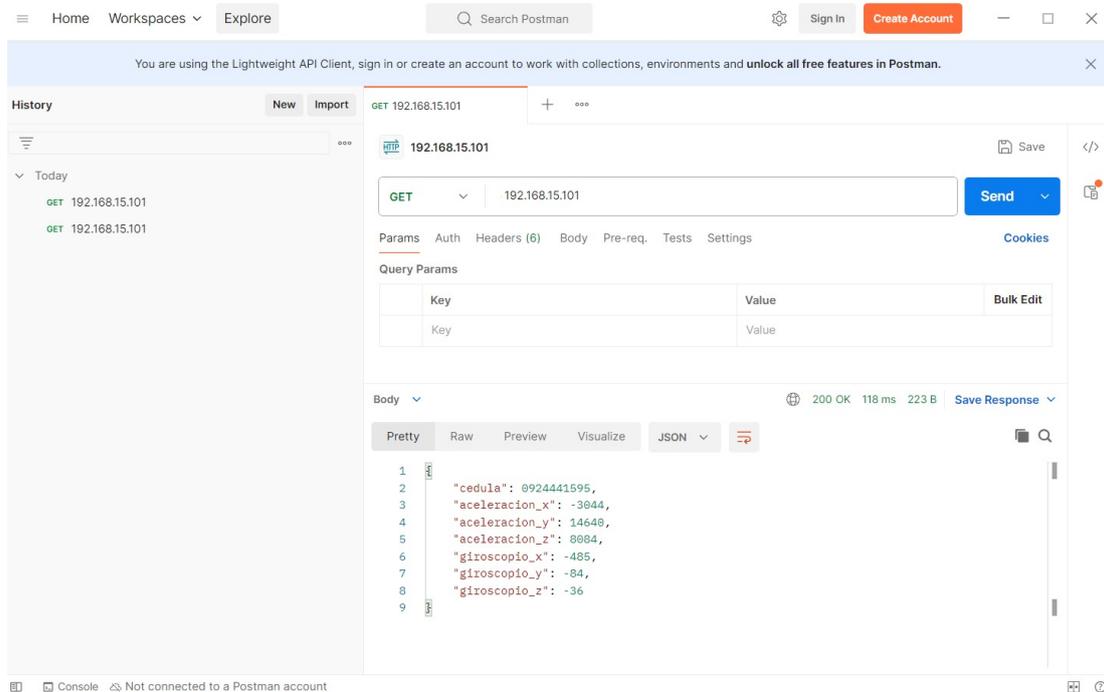


Figura 24. Visualización de la obtención de las coordenadas mediante el simulador Post Man, Autor propio.

VI-C5. Fabricación del modulo externo del dispositivo en 3D : Se realizó el diseño del modulo externo en 3D para la protección y soporte de los componentes electrónicos empleados en el dispositivo, garantizando la funcionalidad y durabilidad del dispositivo considerando los diferentes aspectos ergonómicos y estéticos para facilitar su uso.

El proceso de diseño se llevará a cabo mediante el software SolidWorks para la creación del modelo tridimensional detallado y personalizable con todas las características específicas. En este proceso se deben considerar diversos factores como la compatibilidad de los componentes del dispositivo, la accesibilidad a los puertos de alimentación, estructura compacta, liviana y bordes redondeados para evitar molestias al usuario, así como la resistencia ante diferentes escenarios que puedan presentarse.

Por otro lado, el modelo diseñado en SolidWorks será exportado en formato STL enviado para su impresión. Se utilizó el ácido polilactico debido a su facilidad de impresión y bajo costo. Una vez impreso el modulo externo se inicia el proceso de ensamblaje con los componentes electrónicos para la verificación de su funcionalidad.

Además, la estructura proporcionará un soporte seguro y funcional para el dispositivo, cumpliendo con los estándares de protección y comodidad para el sistema implementado. Su diseño innovador mejora la usabilidad del producto y garantizando su resistencia frente a condiciones cotidianas.

VI-C6. Verificación del funcionamiento exitoso del circuito: Las conexiones del circuito de alimentación, interfaz I2C fueron verificadas con éxito, donde el microcontrolador estableció conexión con el sensor MPU-6050, permitiendo la detección de cambios de orientación y la aceleración del dispositivo. Además, se verificó que el módulo ESP32 reciba los datos brutos del sensor y ejecute el algoritmo de detección de patrones para la identificación de emergencia.

El circuito fue probado en condiciones simuladas, donde se evaluó la precisión en la detección de emergencias o caídas, así como la respuesta del dispositivo. También se comprobó la conectividad Wi-Fi, verificando la transmisión correcta de los datos hacia la aplicación y la captura de emergencias enviada al área encargada de gestionarlas y, por último, se realizó un análisis de la duración de la batería del dispositivo.

VI-D. Fase de diseño y desarrollo de la app móvil

Para el desarrollo de la aplicación móvil se utilizó el software Android Studio como se muestra en la figura 25 que permitirá la gestión y la notificación de emergencias de manera eficiente la misma que servirá como interfaz entre el usuario del dispositivo y el administrador del sistema.

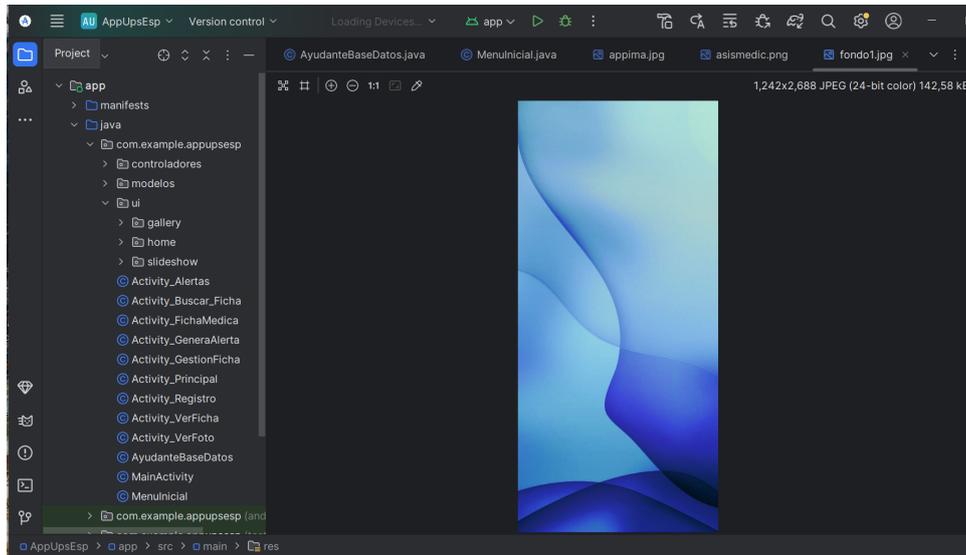


Figura 25. Funciones principales de la app, Autor propio.

VI-D1. Funcionalidades principales de la aplicación :

- **Notificaciones automáticas de la emergencia:** La aplicación recibe alertas enviadas por el dispositivo de asistencia remota al detectar un movimiento inusual, luego se activa la alerta de emergencia gestionando la geolocalización del dispositivo y el enlace directo en transmisión en vivo desde la cámara instalada para evaluar la gravedad del accidente.
- **Botón de emergencia manual:** El botón de pánico se lo incorpora en la pantalla principal diseñada para el fácil acceso del usuario en caso de una emergencia, permitiendo generar una alerta manual en el caso de que el dispositivo no detecte automáticamente el imprevisto al activarse la notificación, el administrador puede observar la ubicación geografica del usuario, el enlace de transmisión y la ficha medica.
- **Ficha medica del usuario:** La app consta con un sistema de registro y almacenamiento de información médica del usuario, el administrador puede visualizar el registro de todos los usuarios, los cuales incluyen:
 1. Nombre y apellido
 2. Fecha de nacimiento
 3. Genero
 4. Nivel de riesgo medico (bajo,medio,alto)
 5. Alergias
 6. Contacto de emergencia

VI-D2. Proceso de diseño y desarrollo:

- **Diseño de la interfaz de usuario:** Se prioriza una interfaz grafica simple, didáctica y optimizada. La pantalla de la app incluirá:
 1. Acceso al botón emergencia
 2. Notificación en tiempo real
 3. Geolocalización
 4. Trasmisión de emergencia en vivo
 5. Ficha médica.
- **Estructura del desarrollo:** Mediante la comunicación con el dispositivo de asistencia y el servidor en el backend, los datos del usuario y de emergencias se van a ir almacenando en una base de datos de firebase donde cada

dato queda registrado, cumpliendo con los estándares de protección de datos integrando así las funcionalidades como es la conexión del ESP32 para recibir las alertas generadas, implementar el servicio de transmisión en vivo y enviando los mensajes de geolocalización al momento de la integración del funcionamiento.

En esta sección ,se explica parte del código desarrollado en Android Studio para la aplicación utilizada en el sistema de asistencia remota. Este código incluye múltiples componentes que permiten la gestión eficiente de las emergencias y la administración de la información de los usuarios.

Como se muestra en la figura 26 funcionalidad de la aplicación en hardware y software diagrama de flujos.

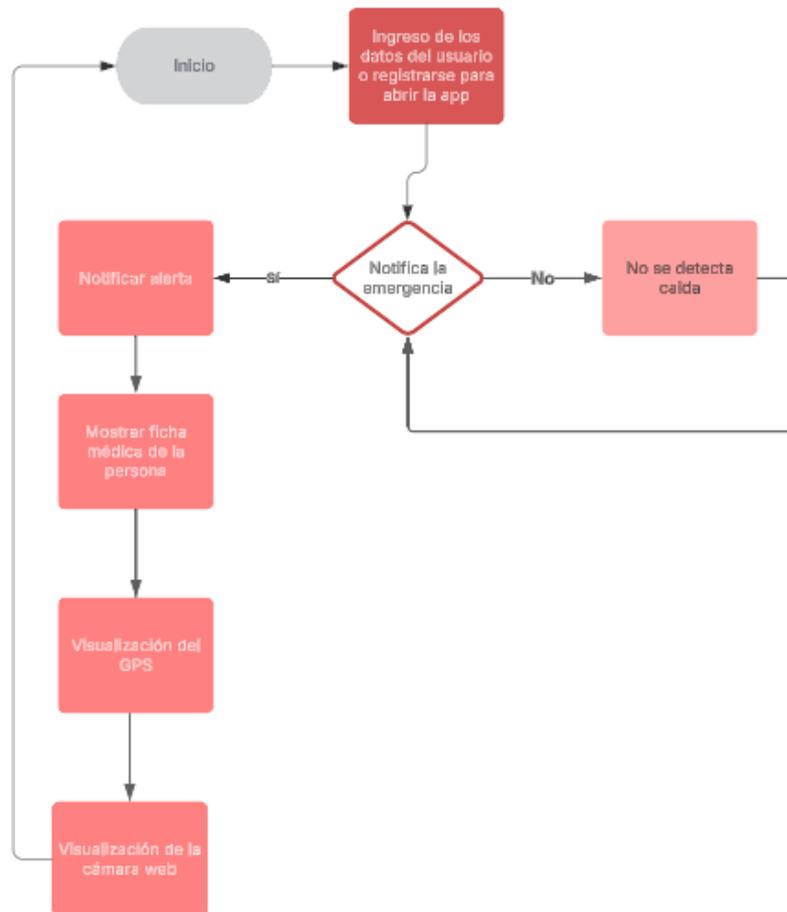


Figura 26. Diagrama de flujo de la aplicación, Autor propio.

Los códigos de alerta se muestran en la figura 27 , permitiendo generar y enviar notificaciones automáticas al administrador cuando se detecta una situación de emergencia como se observa en la figura 28 . Además, se presenta el código del ayudante de base de datos como se muestra en la figura 29 , una clase encargada de gestionar el almacenamiento de toda la información recibida desde el dispositivo de asistencia remota. Esta estructura asegura que los datos se almacenen de manera ordenada y accesible.

```

1 package com.example.appupspesp;
2
3 import ...
42
43 public class Activity_Alertas extends AppCompatActivity implements SelectListener {
44
45     5 usages
46     ListView listView;
47     5 usages
48     ArrayList<String> alertas;
49     4 usages
50     ArrayList<Integer> idUsuario;
51     2 usages
52     AlertaBD alertaBD;
53     2 usages
54     FichaMedicaBD fichaMedicaBD;
55     11 usages
56     Context context;
57
58     4 usages
59     private Handler handler;
60     3 usages
61     private Runnable runnable;
62     1 usage
63     private static final int INTERVALO_TIEMPO = 5000;
64
65

```

Figura 27. Visualización del código de alertas, Autor propio.

```

43 public class Activity_Alertas extends AppCompatActivity implements SelectListene
194 private void llenarListView(){
124     listView.refreshDrawableState();
125     listView.setAdapter(adapter);
126     listView.setOnItemClickListener(new AdapterView.OnItemClickListener(){
127         @Override
128         public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
129             String id = idUsuario.get(i).toString();
130             FichaMedica ficha = fichaMedicaBD.element(id);
131             if(ficha != null){
132                 Bundle bolsa = new Bundle();
133                 bolsa.putInt("id", ficha.getId());
134                 bolsa.putString("cedula", ficha.getCedula());
135                 bolsa.putString("nombres", ficha.getNombres());
136                 bolsa.putString("apellidos", ficha.getApellidos());
137                 bolsa.putString("fecha_nacimiento", ficha.getFecha_nacimiento());
138                 bolsa.putString("genero", ficha.getGenero());
139                 bolsa.putString("telefono", ficha.getTelefono());
140                 bolsa.putString("direccion", ficha.getDireccion());
141                 bolsa.putString("patologia", ficha.getPatologia());
142                 bolsa.putString("estado", ficha.getEstado());
143
144                 String lsFoto = alertas.get(i);
145                 lsFoto = lsFoto.split(regex: " ")[3];
146                 bolsa.putString("foto", lsFoto);
147

```

Figura 28. Visualización de actividad de la alerta con sus parámetros a notificar, Autor propio.

```

package com.example.appupsp;

import ...

public class AyudanteBaseDatos extends SQLiteOpenHelper {

    private static final String
        NOMBRES_BASE_DATOS = "registros",
        TABLA_USUARIO = "usuario",
        TABLA_FICHA = "ficha",
        TABLA_ALERTA = "alerta";

    private static final int VERSION_DB = 1;

    public AyudanteBaseDatos(@Nullable Context context) {
        super(context, NOMBRES_BASE_DATOS, factory: null, VERSION_DB);
    }
}

```

Figura 29. Ayudante de base de datos para almacenar la información obtenida, Autor propio.

Otro componente importante es el código de la ficha médica, que incluye diversos parámetros relacionados con la información completa del usuario como se observa en la figura 30 . Esta ficha es esencial para que el administrador pueda acceder en el momento de una emergencia, a datos clave sobre la persona en riesgo, como su historial médico y contactos de emergencia. En el caso que se necesite los datos de algún usuario también puede buscarlos por su ID como se muestra en la figura 31 .

```

public class FichaMedica {

    private int id;

    private String cedula, nombres, apellidos, fecha_nacimiento, genero, telefono, direccion, pato

    public FichaMedica(){

    }

    public FichaMedica(int id, String cedula, String nombres, String apellidos, String fecha_nacim

        this.id = id;
        this.cedula = cedula;
        this.nombres = nombres;
        this.apellidos = apellidos;
        this.fecha_nacimiento = fecha_nacimiento;
        this.genero = genero;
        this.telefono = telefono;
        this.direccion = direccion;
        this.patologia = patologia;
        this.estado = estado;
    }
}

```

Figura 30. Parámetros que se obtienen en la ficha medica, Autor propio.

```

1 package com.example.appupsesp;
2
3 > import ...
4
19
20 public class Activity_Buscar_Ficha extends AppCompatActivity {
21
22     Context contexto;
23     EditText txtApellido;
24     Button cmdBuscar;
25     FichaMedicaBD fichaMedicaBD;
26
27     @Override
28     protected void onCreate(Bundle savedInstanceState) {
29         super.onCreate(savedInstanceState);
30         EdgeToEdge.enable(this);
31         setContentView(R.layout.activity_buscar_ficha);
32         ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
33             Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
34             v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
35             return insets;
36         });
37     }
38 }

```

Figura 31. Buscador de ficha de los usuarios para obtención de datos de forma más rápida, Autor propio.

El botón de emergencia, por otro lado, se encuentra ubicado en la pantalla principal de la aplicación. Este botón, al ser activado, genera una notificación inmediata para alertar al administrador sobre la situación.

- Pruebas y validación de la app: Las pruebas para garantizar el correcto funcionamiento de la app constan de la detección y envío de alertas automáticas del dispositivo, generar alertas manuales con el botón, precisión de la geolocalización, fluidez en la transmisión y acceso a la ficha medica durante la emergencia, cumpliendo las normativas.
- Cumplimiento de normativas:
 1. La ISO 25010 es una norma internacional que define un modelo de calidad para evaluar las características esenciales del software. Este estándar se centra en ocho dimensiones clave: funcionalidad, confiabilidad, usabilidad, eficiencia en el rendimiento, compatibilidad, seguridad, mantenibilidad y portabilidad. Su propósito principal es garantizar que los sistemas de software cumplan con las expectativas de los usuarios y las necesidades específicas del proyecto. Por ejemplo, en el contexto de la aplicación móvil de teleasistencia, la funcionalidad y confiabilidad son esenciales para detectar emergencias de manera precisa y notificar al administrador sin fallos técnicos. Además, la seguridad y usabilidad deben estar garantizadas para que el usuario pueda operar la app sin complicaciones, incluso en situaciones de estrés.
 2. La ISO 27001 es el estándar internacional que establece los requisitos para la implementación de un Sistema de Gestión de Seguridad de la Información (SGSI). Su objetivo principal es garantizar la confidencialidad, integridad y disponibilidad de los datos, lo que resulta fundamental en aplicaciones que manejan información sensible como datos médicos. En el contexto de este proyecto, esta norma es crucial, por que la aplicación móvil almacenará datos médicos del usuario (nombre, alergias, nivel de riesgo, etc.) y enviará notificaciones que incluyen geolocalización. La adopción de medidas como la encriptación de datos, autenticación segura y gestión adecuada de permisos cumple con los lineamientos de la ISO 27001, reduciendo riesgos como accesos no autorizados o pérdida de información.

VI-E. Analisis de bateria

Para el análisis de la duración de la batería en el dispositivo, es fundamental considerar que opera con dos baterías de 8800 mAh, lo que permite calcular el consumo por hora utilizado para todo el sistema. También, se debe tomar en cuenta los siguientes parámetros:

- Consumo Acelerómetro:
Corriente del giroscopio = 3.6 mA
Corriente del acelerómetro = 5 μ A
- Consumo amplificador de voltaje: 600 mA
- Consumo ESP32:
Modo normal = 50 mA
Modo WIFI = 100 mA

Para el análisis total de los componentes se necesita usar la siguiente ecuación (1)

$$\text{Consumo de los componentes} = \text{Consumo acelerómetro} + \text{Consumo amplificador} + \text{Consumo ESP32} \quad (1)$$

Por otro lado, se va a utilizar la ecuación (2), para la obtención de carga de las baterías a emplear en horas.

$$\text{Duración batería} = \frac{\text{Capacidad disponible} \cdot \text{mA} \cdot \text{h}}{\text{Consumo de los componentes} \cdot \text{mA} \cdot \text{h}} = \text{Horas} \quad (2)$$

VI-F. Modelo matemático para el análisis de aceleración y movimientos en el sensor MPU6050

La magnitud de la aceleración ($|\mathbf{a}|$) se calcula como un valor escalar derivado de las componentes de la aceleración en los ejes X, Y y Z. Este cálculo se realiza mediante la aplicación de la fórmula correspondiente (3), que integra dichos componentes.

$$|\mathbf{a}| = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (3)$$

En donde a_x , a_y , a_z son las componentes de aceleración en los ejes X, Y, Z, medidas en unidades proporcionales a la aceleración debida a la gravedad terrestre. Mientras $|\mathbf{a}|$ es la magnitud total de la aceleración.

VI-F1. Rango del sensor: El sensor MPU6050 es un dispositivo MEMS (Microelectromechanical Systems) que puede medir aceleraciones en un rango que varía de 0 a 16500 unidades digitales. Este rango corresponde a un límite físico de $\pm 2g$, donde g representa la aceleración debida a la gravedad terrestre (aproximadamente $9,81 \text{ m/s}^2$). El valor máximo de 16500 equivale a una aceleración de $2g$, mientras que un valor de 0 indica reposo total o ausencia de movimiento detectado en el eje correspondiente.

VI-F2. Cálculos de la aceleración en "g": Para interpretar los datos proporcionados por el sensor en unidades físicas, es necesario convertir el valor magnitud a unidades de gravedad terrestre (g). Este proceso de conversión permite relacionar los datos digitales del sensor con un contexto físico tangible.

La relación entre los valores digitales y las unidades de gravedad terrestre se define mediante la siguiente expresión (4):

$$a_g = \frac{|a| \text{ valor magnitud}}{16500} \times 2 \quad (4)$$

Donde $|a|$ representa la magnitud de la aceleración calculada a partir de los datos del sensor. Esta ecuación asegura que las unidades digitales del sensor se normalicen dentro del rango de $\pm 2g$, permitiendo interpretar los resultados en un contexto práctico y relevante para aplicaciones como el análisis de movimiento, vibraciones o inclinaciones.

Además, es importante considerar el impacto del ruido en las mediciones del sensor. Los datos de aceleración pueden estar sujetos a variaciones debido a factores externos, como vibraciones o interferencias ambientales. Por ello, se recomienda aplicar técnicas de filtrado, como el uso de un filtro pasa-bajas, para obtener mediciones más precisas y estables.

VI-G. Fase de integración del sistema

Esta fase representa la finalización del proyecto donde se integran todos los componentes del sistema como el dispositivo de asistencia remota, la cámara IP y la aplicación móvil. Este sistema está diseñado para proporcionar una solución integral que garantice la seguridad y asistencia inmediata a personas con movilidad reducida en situaciones de emergencia dentro del entorno universitario.

Estructura del sistema

1. **Camara IP:** La cámara se integra al sistema para ayudar a visualizar la emergencia en tiempo real al momento que se genere la alerta desde el dispositivo de asistencia remota. La cámara mantiene comunicación con la aplicación móvil para notificar al administrador, esto ocurre gracias a un enlace directo que habilita el acceso inmediato a la misma.
2. **Aplicación móvil:** La aplicación desarrollada en Android Studio actúa como el núcleo de comunicación del sistema y está diseñada para recibir y gestionar las notificaciones de emergencia enviadas desde el dispositivo de asistencia integrado de diversas funcionalidades como :
 - La gestión de alertas al momento de recibir una notificación automática o manual del usuario, donde también se incluye el mensaje de geolocalización del usuario.
 - Registro médico donde se almacena la información relevante del usuario facilitando la asistencia personalizada en situaciones de emergencias.
 - Visualización en tiempo real mediante la cámara IP instalada en el área.
3. **Proceso de funcionamiento:** La integración de estos tres elementos garantiza un sistema remoto y eficiente de teleasistencia, donde la combinación de monitoreo en tiempo real, localización precisa y acceso inmediato a datos médicos contribuye a mejorar la seguridad y el tiempo de respuesta ante emergencias.

VI-H. Fase de evaluación y validación del sistema completo

La evaluación del sistema de tele-asistencia se lleva a cabo mediante un enfoque estructurado que combine entrevistas, simulaciones controladas y recolección de datos para garantizar que el sistema cumpla con los objetivos de seguridad y usabilidad establecidos. Además, se utilizará un diseño de muestreo para seleccionar participantes representativos que permitan obtener resultados relevantes y aplicables. La validación del sistema se realizará mediante pruebas de campo y simulaciones controladas, permitiendo analizar el rendimiento del dispositivo de asistencia remota, la aplicación móvil y la cámara IP.

1. **Encuestas:** Este método ayuda a identificar la percepción y satisfacción de los usuarios y administradores con respecto al funcionamiento del sistema.

2. Simulación del sistema: Consiste en evaluar el rendimiento en condiciones controladas que simulen emergencias reales, analizar el tiempo promedio de respuesta desde la detección de la emergencia hasta la recepción de la notificación en la app, la precisión en la geolocalización del usuario y la calidad de la transmisión de video en tiempo real desde la cámara IP.
3. Recolección de datos del sistema: Obtener métricas sobre el funcionamiento del sistema donde se utiliza la técnica del registro de la aplicación, monitoreo de la cámara.
4. Análisis de datos: El análisis de datos se realizará métodos cuantitativos y cualitativos para evaluar el rendimiento del sistema de tele-asistencia. En el aspecto cuantitativo, se analizarán métricas como el tiempo promedio de respuesta desde la detección de una emergencia hasta su notificación al administrador, la precisión del sistema en la detección de caídas y la tasa de falsos positivos y negativos. Para esto, se emplearán herramientas estadísticas que permitan describir y comparar los resultados obtenidos.

VII. RESULTADOS

VII-A. Encuestas realizadas

La encuesta realizada tiene como finalidad conocer la opinión de la comunidad salesiana acerca de la implementación de un prototipo de sistema de tele-asistencia que brinde soporte a las personas con movilidad reducida en el campus. La encuesta tiene un total de 70 respuestas correspondiente a estudiantes, graduados y docentes.

En la figura 32 se observa el resultado de la pregunta: ¿Consideras necesario la implementación de tecnologías en la universidad que abarquen aspectos como seguridad, asistencia rápida, atención de calidad e inclusión para personas con movilidad reducida?.

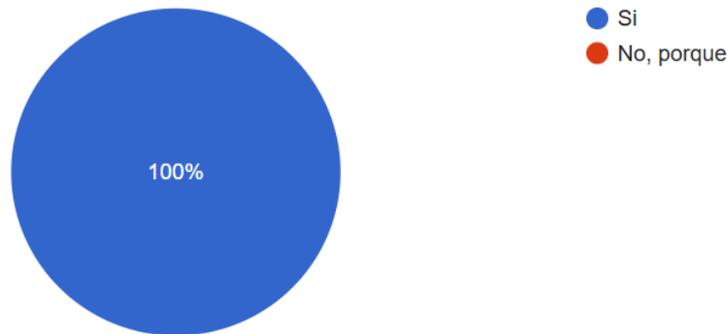


Figura 32. Gráficas sobre la importancia de desarrollar e implementar tecnologías en la universidad , Autor propio.

El 100 % de los encuestados respondieron "Sí", indicando que consideran necesaria la implementación de este tipo de tecnologías. Mientras que , el 0 % respondió "No", lo que significa que no hubo opiniones en contra de esta implementación.

Según los resultados de la figura 33 de la pregunta: ¿Cuál de estas opciones consideras es de mayor importancia a tratar al momento de desarrollar tecnologías de asistencia remota (tele-asistencia)?.

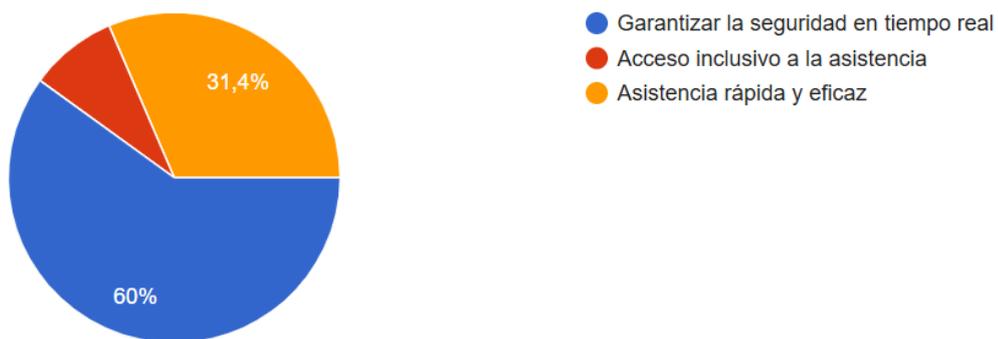


Figura 33. Gráficas sobre la importancia de tecnología de tele-asistencia remota, Autor propio.

Se obtuvo que el 60 % de los encuestados consideran mejor opción "Garantizar la seguridad en tiempo real". Mientras que el 8.6 % de los encuestados eligieron como opción el "Acceso inclusivo a la asistencia"

En la figura 34 se observa el resultado de la pregunta: ¿Qué tan importante consideras contar con un sistema de tele-asistencia que se encargue de monitorear en caso de emergencias a personas de movilidad reducida y a la comunidad salesiana en general a futuro?.

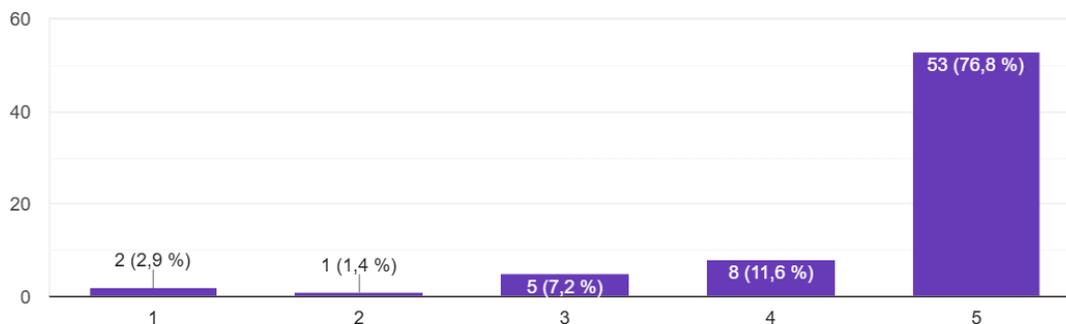


Figura 34. Gráficas sobre la importancia de tener un sistema de tele-asistencia, Autor propio.

Segun el grafico, 76.8% de los encuestados consideran que es muy importante contar con un sistema de tele-asistencia. Mientras el 3.3% considera que no tiene tanta relevancia.

La figura 35 nos muestra los resultados de la pregunta: ¿Crees que la implementación de un sistema de tele-asistencia para el monitoreo de emergencias tendría un impacto considerable en la comunidad universitaria?.

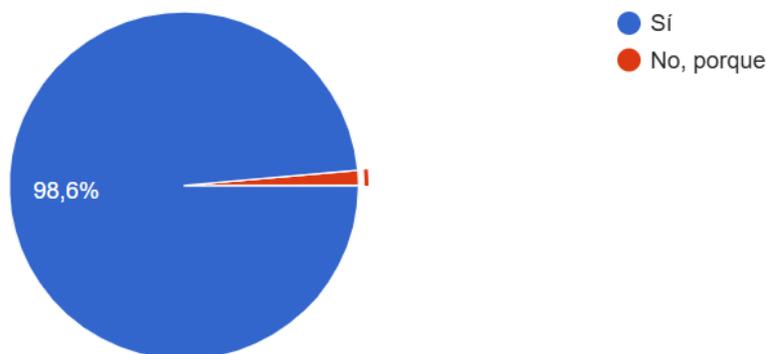


Figura 35. Gráficas sobre el impacto de agregar un sistema de tele-asistencia en la universidad, Autor propio.

La diferencia entre ambas respuestas es muy marcada, lo que indica un consenso casi unánime en la importancia y relevancia de este tipo de tecnología en el ámbito universitario. Mientras que, el 1.4% que no considera un impacto considerable, podría estar basado en razones particulares, como desconocimiento del sistema dentro del entorno universitario.

En la figura 36 se observa los resultados de la pregunta: ¿Qué características consideras necesarias para el desarrollo de un proyecto como el mencionado en esta encuesta?.

Apoyo sostenible
Apoyo sostenible para la realización del proyecto
Eficaz
Accesibilidad y usabilidad
Que sea mas accesible por ejemplo se pueda aplicar desde un celular
Nada
La seguridad
Efectivo
Que sea fácil de mantener

Figura 36. Gráficas sobre las diferentes características del desarrollo de un proyecto como el mencionado en esta encuesta, Autor propio.

La mayoría de los encuestados concordaron en que el apoyo sostenible es un aspecto clave para la realización del proyecto. Además, se destacó la importancia de la eficacia, accesibilidad y facilidad de uso, asegurando que el sistema sea práctico y eficiente para los usuarios. Otros aspectos relevantes mencionados fueron la seguridad, la rapidez en la asistencia, la conectividad y la integración con dispositivos.

En la figura 37 se observa el resultado de la pregunta: ¿Consideras que la universidad muestra un apoyo completo en este tipo de temas?.

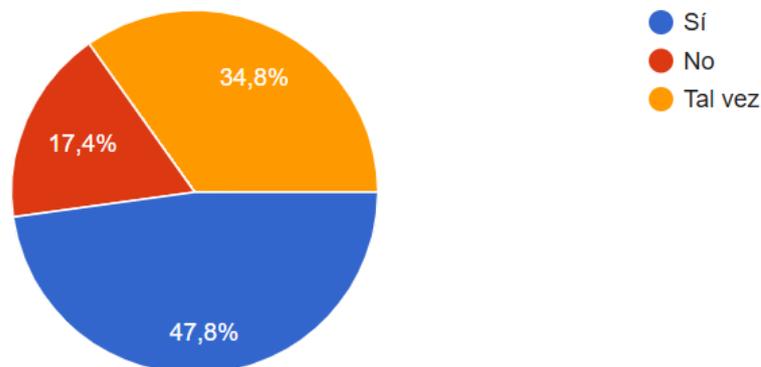


Figura 37. Gráficas sobre el apoyo que creen que tiene la universidad sobre este tipos de temas, Autor propio.

Según el gráfico el 47,8% de los encuestados consideran que la universidad sí brinda un apoyo completo en temas relacionados con la implementación de tecnologías de tele-asistencia. Mientras que el 17,4% considera que la universidad no ofrece un apoyo completo, lo que refleja una percepción de falta de respaldo en este tipo de iniciativas.

VII-B. Dispositivo final implementado

En la figura se presenta el diseño final del dispositivo, donde esta implementada la placa PCB que constituye el núcleo del sistema. Este diseño ha sido optimizado para garantizar la integración eficiente de los componentes electrónicos, asegurando un funcionamiento estable y confiable del dispositivo.

VII-C. Modulo externo del dispositivo 3D

En las figuras 38, 39 y 40 se visualiza el modulo en diferentes vistas.

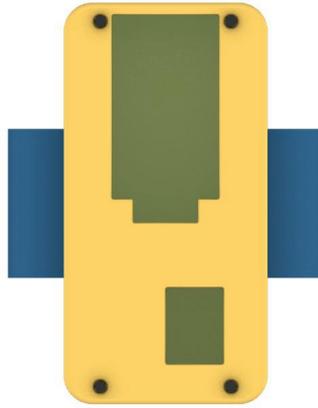


Figura 38. Modulo externo vista frontal , Autor propio.



Figura 39. Modulo externo vista planta, Autor propio.

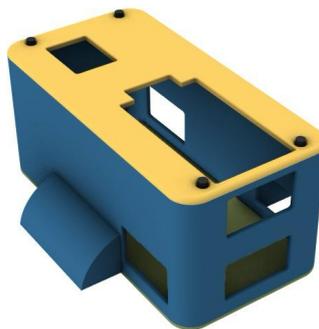


Figura 40. Modulo externo vista perfil, Autor propio.

VII-C1. Dispositivo de prueba: En las figuras 41 y 42 se puede observar el dispositivo en fase de prueba en diferentes angulos.



Figura 41. Visualización del dispositivo de prueba de la vista lateral, Autor propio.



Figura 42. Visualización del dispositivo de prueba de la vista frontal, Autor propio.

VII-C2. Dispositivo final en 3D: En las figuras 43 y 44 se observa la impresión en 3D final del dispositivo realizado.



Figura 43. Visualización del dispositivo final, Autor propio.



Figura 44. Visualización del dispositivo final 2, Autor propio.

VII-C3. Pruebas del sistema: Con el objetivo de verificar la funcionalidad y la eficacia del prototipo del sistema de tele-asistencia diseñado, se llevaron a cabo pruebas controladas en escenarios simulados. Estas pruebas consistieron en evaluar la capacidad del dispositivo para detectar eventos de emergencia, como la caída de una persona, y generar alertas en tiempo real para su monitoreo y atención.

Asimismo, las pruebas incluyeron la evaluación de la precisión en la detección de caídas, la rapidez en la generación de notificaciones y la correcta visualización de la geolocalización en tiempo real mediante Postman. Los resultados demostraron que el sistema es capaz de detectar caídas, mientras se asegura que la información clave, como la ubicación del usuario, se transmita correctamente al momento de la emergencia.

Estas pruebas no solo validan la funcionalidad técnica del sistema, sino que también permiten la identificación de posibles mejoras para optimizar la experiencia del usuario y la robustez del sistema en situaciones reales, Como se muestra en la figura 45 en la misma que se observa la conexión directa a la cámara mas cerca a la emergencia según el GPS del dispositivo el mismo que se conecta por la IP mas cerca de las cámaras.



Figura 45. Simulación de escenario para prueba de dispositivo, Autor propio.

A continuación en la figura 46 se muestran las conexiones del sistema

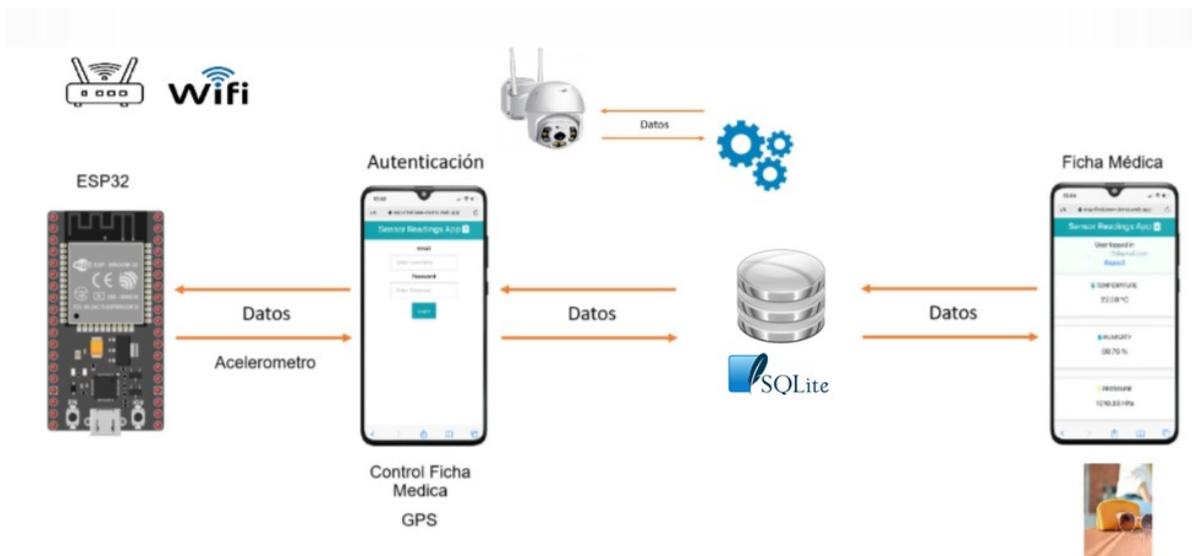


Figura 46. Enlaces de conexión del sistema, Autor propio.

VII-D. Aplicación móvil

En la figura 47 se muestra la interfaz de la pantalla de inicio de sesión del sistema de tele-asistencia. En esta pantalla, se solicita al usuario ingresar sus credenciales, como el nombre de usuario y la contraseña, para garantizar un acceso seguro al sistema.

Adicionalmente, se incluye un botón de registro que permite crear nuevas cuentas, facilitando la incorporación de otros usuarios. Este diseño busca ofrecer una experiencia de usuario sencilla y accesible, alineados a los objetivos brindando seguridad y una respuesta rápida ante emergencias.

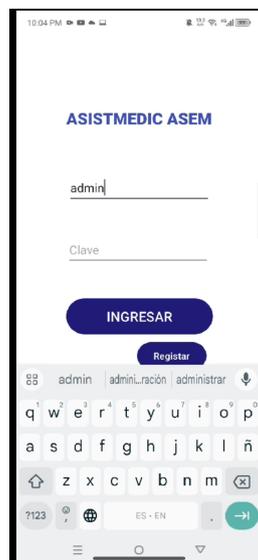


Figura 47. Ingreso a la app, Autor propio.

VII-D1. *Pantalla principal:* La Figura 48 corresponde a la pantalla principal de la aplicación desarrollada. Esta interfaz constituye el núcleo operativo del sistema, integrando de manera organizada y accesible los siguientes comandos principales:

- Inicio: Permite al usuario retornar a la página inicial del sistema desde cualquier sección, asegurando una navegación intuitiva.
- Registro de Fichas: Facilita la creación de nuevos registros para usuarios, detallando información relevante como datos personales, condiciones de salud y contactos de emergencia.
- Listado de Fichas: Presenta una vista estructurada y jerarquizada de todas las fichas registradas, lo que permite un acceso rápido y eficiente a la información almacenada.
- Búsqueda de Fichas: Incluye un sistema de búsqueda optimizado que permite localizar fichas específicas mediante filtros como nombre, identificación o fecha de registro.
- Cámara de monitoreo en tiempo real: Incluye un comando que permite visualizar la cámara más cerca a la emergencia mediante la IP más cercana al dispositivo.
- GPS del dispositivo de asistencia: Incluye un GPS de coordenadas en las cuales se verán reflejadas al momento que se envió la alerta.
- Alertas de Emergencia: Proporciona un panel donde se visualizan notificaciones en tiempo real relacionadas con situaciones de emergencia detectadas por el sistema, permitiendo un monitoreo constante.

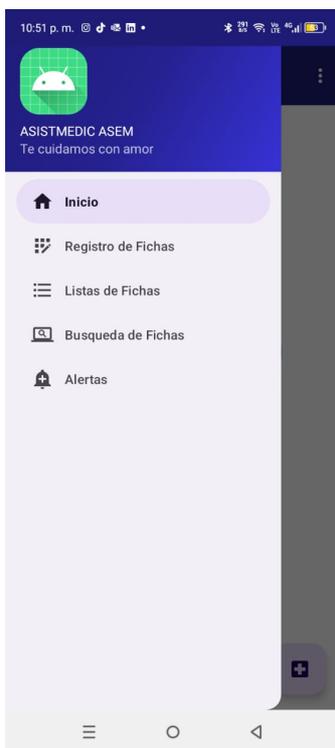


Figura 48. Visualización de la pantalla principal, Autor propio.

Adicionalmente, en la esquina inferior derecha de la pantalla se encuentra un Botón de Emergencia como se muestra en la figura 49 , diseñado para ser utilizado en cualquier momento. Este botón activa de forma inmediata una alerta dentro del sistema, enviando notificaciones a los responsables designados y priorizando la asistencia rápida al usuario en situación crítica.



Figura 49. Botón de emergencia, Autor propio

El diseño de esta pantalla principal no solo optimiza la interacción usuario-sistema, sino que también garantiza accesibilidad y eficiencia en el manejo de emergencias.

VII-D2. Alertas en la aplicación: En la figura 50 se muestra el flujo de actividades relacionadas con la generación de alertas y la captura de eventos de emergencia, específicamente la interacción entre el dispositivo que detecta la emergencia y el administrador encargado de recibir las notificaciones. En este proceso, el sistema genera alertas que incluyen información clave como la ubicación geolocalizada del incidente, permitiendo a los responsables actuar de manera rápida y precisa.

El código subyacente maneja tanto el registro como la visualización de estas notificaciones, asegurando un funcionamiento intuitivo y eficiente del sistema. Además, la interfaz ha sido diseñada para ser didáctica y accesible, facilitando su uso tanto para los administradores encargados de gestionar las emergencias como para los usuarios finales, garantizando así una experiencia fluida y confiable en situaciones críticas.

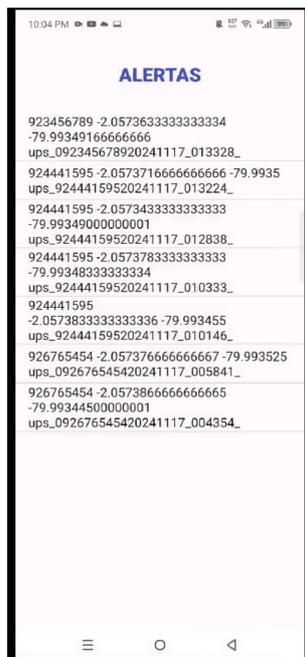


Figura 50. Visualización de las alertas,Autor propio.

VII-D3. Ficha medica: La ficha médica del sistema representa un componente esencial para el registro y la gestión de información personal y clínica de los usuarios. En esta sección, se recopilan datos clave que permiten identificar al usuario y evaluar su estado de salud de manera eficiente. Entre los parámetros registrados se incluyen información personal básica, como datos de identidad, contacto, así como detalles médicos relevantes como patologías existentes y el estado de atención asignado, el cual puede clasificarse como alto, medio o bajo, según la necesidad de supervisión o intervención, como se observa en la figura 51.



Figura 51. Visualización de la ficha medica del usuario, Autor propio.

Esta ficha médica, además de centralizar la información, proporciona una visión integral del usuario que facilita la toma de decisiones rápidas y precisas en situaciones de emergencia. Su diseño se ha orientado a garantizar una interfaz clara y funcional, permitiendo a los administradores acceder de forma ágil a los datos necesarios para brindar una atención oportuna y efectiva.

VII-E. Analisis de batería del dispositivo

A continuación, se observa en la ecuación (5) sobre el consumo de los componentes.

$$\text{Consumo total de los componentes} = 753,65 \text{ mA}\cdot\text{h} \quad (5)$$

También se pudo obtener en la ecuación (6) la duración de la batería en horas.

$$\text{Duración batería} = \frac{17600 \text{ mA} \cdot \text{h}}{753,65 \text{ mA} \cdot \text{h}} = 23,35 \text{ Horas} \approx 23 \text{ horas} \quad (6)$$

El cálculo se comprobó con ayuda de una herramienta digital como se muestra en la tabla V, que el total de horas de funcionamiento de funcionamiento es de 23 horas aproximadamente.

Parámetro	Valor	Unidad
Capacidad Disponible	17600	mA·h
Consumo	753.65	mA
Duración	23,3530	Horas

Tabla V
TABLA DE CAPACIDAD Y CONSUMO DE BATERÍA

VII-F. Modelo matemático

VII-F1. *Cálculo de ejes de movilidad en el diseño del dispositivo de asistencia remota en emergencias.*: El rango va de 0 a 16500. Si el rango llega a más de 5000, se determina que ha ocurrido una caída.

El umbral para detectar una caída se establece en 5000 para el valor de la magnitud porque tiene un factor relacionado con el comportamiento físico esperado durante la caída, dado a que varios estudios han llegado a la conclusión de que el cuerpo experimenta una aceleración repentina debida al impacto con el suelo. Esto corresponde aproximadamente a una aceleración de $0,61g$ como se observa en la ecuación (7).

$$a_g = \frac{5000}{16500} \times 2 = 0,61g \quad (7)$$

Por otro lado, la condición para la detección de caída va a ser de la siguiente manera, Si $|a|$, la magnitud, ≥ 5000 , se detecta una caída. Esto puede indicar una aceleración repentina o un cambio brusco en la velocidad.

VII-F2. *Tiempo de promedio de respuesta desde la detección de la emergencia hasta su notificación al administrador*: Es un sistema de tele-asistencia que involucra dispositivo de asistencia, cámara y aplicación móvil, el tiempo de respuesta es una métrica clave para evaluar el rendimiento del sistema, donde el tiempo de respuesta se refiere al intervalo que transcurre desde que el usuario realiza un movimiento repentino hasta que el sistema responda con una acción apropiada, enviando una alerta al administrador encargado.

El tiempo de respuesta $T_{\text{respuesta}}$ del sistema de tele-asistencia se puede expresar mediante la siguiente fórmula general (8):

$$T_{\text{respuesta}} = T_{\text{dispositivo}} + T_{\text{comunicación}} + T_{\text{procesamiento}} + T_{\text{interfaz}} + T_{\text{acción}} \quad (8)$$

Donde:

- $T_{\text{dispositivo}}$: Tiempo que tarda en activarse el dispositivo (por ejemplo, un botón de emergencia o sensor). Esto incluye la detección de la activación y la transmisión de la señal.
- $T_{\text{comunicación}}$: Tiempo de transmisión de la señal desde el dispositivo hasta el servidor o la central de monitoreo. Este tiempo depende de la red utilizada (como Wi-Fi, LTE, 5G) y su latencia.
- $T_{\text{procesamiento}}$: Tiempo necesario para que el sistema procese la solicitud, lo que incluye la recolección de datos de los dispositivos (por ejemplo, ubicación, estado del usuario) y el análisis de los mismos (por ejemplo, detección de eventos o caídas).
- T_{interfaz} : Tiempo requerido para mostrar la información relevante en la interfaz del operador o en la aplicación del usuario. Esto incluye la presentación de la alerta visual o el acceso a la cámara.
- $T_{\text{acción}}$: Tiempo que tarda el sistema en tomar una acción en respuesta, como notificar a un operador, contactar a los servicios de emergencia o iniciar una videollamada a través de la cámara.

Para un sistema de tele-asistencia que incluye una aplicación, dispositivos y cámaras, el tiempo de respuesta se puede detallar aún más, considerando los aspectos específicos de cada componente.

La fórmula es la siguiente:(9)

$$T_{\text{respuesta}} = T_{\text{dispositivo}} + T_{\text{red}} + T_{\text{procesamiento}} + T_{\text{visualización}} \quad (9)$$

Donde:

- $T_{\text{dispositivo}}$: Tiempo que tarda al activarse el dispositivo, como un sensor.
- T_{red} : Tiempo de latencia de la red, el tiempo necesario que la señal viaja del dispositivo hasta el servidor o central de monitoreo. La latencia puede depender de la infraestructura de red utilizada (Wi-Fi, 4G, 5G, etc.).
- $T_{\text{procesamiento}}$: Tiempo necesario para procesar los datos recibidos desde los dispositivos que puede incluir el análisis detectar eventos críticos (por ejemplo, caídas).
- $T_{\text{visualización}}$: Tiempo requerido donde la información es procesada se visualice correctamente en la interfaz del operador o en la aplicación del usuario.

Al calcular el tiempo de respuesta, es importante medir o estimar cada uno de los componentes mencionados para obtener un valor realista con la siguiente ecuación (10):

$$T_{\text{respuesta}} = T_{\text{dispositivo}} + T_{\text{red}} + T_{\text{procesamiento}} + T_{\text{visualización}} \quad (10)$$

$$T_{\text{respuesta}} = 13_s + 8,9_{\text{ms}} + 10_{\text{ms}} + 16_s \quad (11)$$

$$T_{\text{respuesta}} = 29,0189_s \quad (12)$$

VII-F3. *Respuesta de las pruebas:* De las pruebas realizadas, se obtuvieron los siguientes resultados:

- **Total de pruebas realizadas:** 55
- **Verdaderos positivos (caídas reales):** 34
- **Detecciones correctas:** 33
- **Falsos negativos:** 1
- **Falsos positivos:** 3

Métricas del sistema

1. **Tasa de falso negativo (FN):**

$$\text{FN} = \frac{\text{Falsos negativos}}{\text{Caídas reales}} \times 100 \quad (13)$$

$$\text{FN} = \frac{1}{34} \times 100 \quad (14)$$

Resultado: 2,94 %

Esto significa que el sistema detectó el 2.94 % de las caídas reales.

2. **Tasa de falso positivo (FP):**

$$\text{FP} = \frac{\text{Falsos positivos}}{\text{Total de pruebas}} \times 100 \quad (15)$$

$$\text{FP} = \frac{3}{55} \times 100 \quad (16)$$

Resultado: 5,45 %

Esto indica que el sistema generó alertas falsas en solo el 5.45 % de las pruebas realizadas.

3. **Sensibilidad (Recall):**

$$\text{Sensibilidad} = \frac{\text{Detecciones correctas}}{\text{Caídas reales}} \times 100 \quad (17)$$

$$\text{Sensibilidad} = \frac{33}{34} \times 100 \quad (18)$$

Resultado: 97,05 %

El sistema detectó correctamente el 97.05 % de las caídas reales.

4. **Precisión:**

$$\text{Precisión} = \frac{\text{Detecciones correctas}}{\text{Detecciones correctas} + \text{Falsos positivos}} \times 100 \quad (19)$$

$$\text{Precisión} = \frac{33}{33 + 3} \times 100 \quad (20)$$

Resultado: 91,66 %

De todas las alertas generadas, el 91.66 % correspondieron a caídas reales.

Interpretación

- El sistema muestra un buen desempeño al momento de la prueba, con una sensibilidad del 97.05 %, lo que indica que detecta la totalidad de las caídas reales.
- La tasa de falsos positivos es de 5.45 %, lo que nos da como resultado que el sistema genera muy pocas alertas falsas.
- La precisión de la notificación del dispositivo es de 91.66 % , la misma que asegura que la mayoría de las alertas corresponden a eventos reales, consolidando la confiabilidad del sistema.

En la figura 52 se muestra la matriz de confusión que representa los valores obtenidos de las métricas evaluadas. Indicando que el dispositivo tiene un alto desempeño en la detección de incidentes.

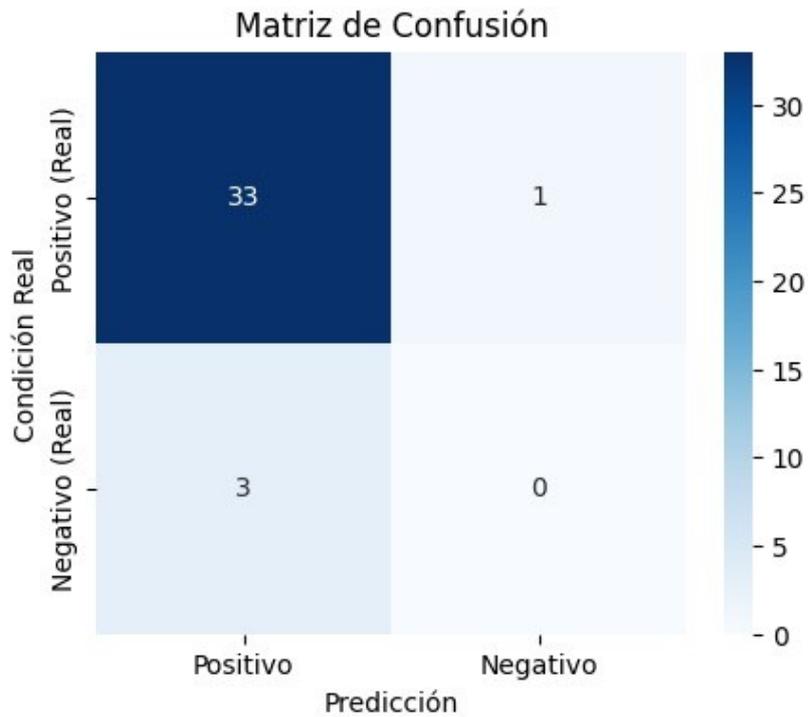


Figura 52. La matriz de confusión de la prueba realizadas, Autor propio.

VIII. CRONOGRAMA

A continuación se muestra el cronograma de trabajo en la siguiente tabla VI.

Tabla VI
CRONOGRAMA

PROYECTO DE GRADO ALUMNOS OBJETIVO GENERAL OBJETIVO ESPECIFICO	TEMA EUCLIDES LEONEL MORAN ALVAREZ , ANNA PAULA SANCHEZ MONTIEL	AMBITO DE APLICACION: MESES											
		10	11	12	1	2							
ACTIVIDAD	RESPONSABLE												
DEFINICION DEL TITULO DEL ANTEPROYECTO DE INVESTIGACION	MORAN-SANCHEZ												
PROBLEMA DE INVESTIGACION Y DELIMITACION DE ESTE	MORAN-SANCHEZ												
JUSTIFICACION	MORAN-SANCHEZ												
OBJETIVOS GENERALES Y ESPECIFICOS	MORAN-SANCHEZ												
MARCO TEORICO	MORAN-SANCHEZ												
METODOLOGIA	MORAN-SANCHEZ												
PRESUPUESTO Y DE ACUERDO AL PROYECTO A IMPLEMENTAR	MORAN-SANCHEZ												
CREACION Y DISEÑO DEL PROTOTIPO	MORAN-SANCHEZ												
DESARROLLO DE LA APLICACION A IMPLEMENTAR	MORAN-SANCHEZ												
SIMULACION DEL SOFTWARE	MORAN-SANCHEZ												
IMPLEMENTACION DEL PROTOTIPO FINAL	MORAN-SANCHEZ												
PRUEBAS DE FUNCIONAMIENTO DEL PROTOTIPO CON LA APLICACION	MORAN-SANCHEZ												
TOMA DE DATOS EXPERIMENTALES PARA COMPROBAR EL ADECUADO FUNCIONAMIENTO DEL PROTOTIPO	MORAN-SANCHEZ												

IX. PRESUPUESTO

Tabla VII
PRESUPUESTO

Descripcion	Cantidad	Ingresos	Egresos
Recursos estimados	1	3.000\$	
Horas de trabajo	350 horas		2.300\$
Instrumentación electrónicos (Multímetro)	1		40 \$
Papelería	1		60 \$
Modulo externo en 3D	1		25\$
Impresión de PCB	1		46 \$
Baterías de litio	2		7 \$
Modulo cargador de baterías de litio	1		19 \$
ESP32-WROOM-32E	1		22\$
Modulo de 6 ejes MPU-6050	1		16 \$
Camara inalámbrica	1		25 \$
TOTAL			2.659\$

X. CONCLUSIONES

El prototipo incorpora un microcontrolador ESP32-WROOM destacable por su versatilidad (Wifi y Bluetooth), junto con un sensor MPU6050, que integra un acelerómetro y giroscopio de 3 ejes que lo vuelve importante para la detección de movimientos, inclinación, rotación entre otras funciones. Estos componentes, en conjunto con una cámara Boykeep y el uso de protocolos I2C, permiten el desarrollo de un sistema robusto y multifuncional.

El dispositivo móvil para el usuario fue diseñado en Wokwi debido a su interfaz amigable y de fácil interpretación. El mismo tiene integrado el microcontrolador ESP32-WROOM y sensor MPU6050 para garantizar un funcionamiento eficiente. Por consiguiente, se obtuvo un diseño funcional y de bajo costo, alineado con las normativas de los objetivos establecidos.

El uso de Android Studio para el desarrollo de la app móvil centraliza las operaciones del sistema, permitiendo la gestión de alertas automáticas, geolocalización precisa y transmisión de video en tiempo real. También, almacena información médica del usuario para personalizar la asistencia en caso de emergencias, asegurando que los datos se gestionen de manera segura y confidencial siguiendo normativas como la ISO 27001.

Se utilizó un módulo externo impreso en 3D para integrar el dispositivo y la app móvil, que protege y soporta los componentes electrónicos. Esta estructura fue diseñada con enfoque ergonómico y estético para garantizar la comodidad del usuario y la resistencia del dispositivo en diferentes escenarios. La carcasa también incluye accesibilidad a los puertos y conexiones, lo que facilita el ensamblaje y mantenimiento, cumpliendo con los estándares de calidad y funcionalidad requeridos.

El diseño ha demostrado resultados satisfactorios en las métricas evaluadas. Con una sensibilidad del 97.05 %, el sistema es capaz de detectar de manera efectiva los incidentes, minimizando la omisión de casos importantes. Además, su precisión del 91.66 % indica que la mayoría de las detecciones realizadas son correctas, reduciendo los falsos positivos y mejorando la confiabilidad del sistema. También, la tasa de falsos negativos del 2.94 % y la tasa de falsos positivos del 5.45 % reflejan un desempeño apropiado en la identificación de eventos, garantizando un nivel de seguridad adecuado para aplicaciones en entornos reales. Estos resultados respaldan la viabilidad del sistema y sugieren que puede seguir perfeccionándose mediante la optimización de algoritmos y la ampliación del área y población de estudio.

XI. RECOMENDACIONES

Es esencial garantizar la precisión de los sensores en un sistema de tele-asistencia basado en IoT. Es importante calibrar adecuadamente sensores como los acelerómetros y giroscopios. Así mismo, el entorno donde se utilice el sistema debe ofrecer condiciones favorables, como una conexión estable y mínimas interferencias, para asegurar una transmisión de datos efectiva.

Los dispositivos portátiles que forman parte de un sistema final deben ser cómodos, ligeros y estar colocados correctamente en el cuerpo del usuario. Esto permite que los sensores funcionen sin interrupciones, evitando lecturas incorrectas y minimizando alarmas innecesarias. Un posicionamiento adecuado es fundamental para mantener la fiabilidad y consistencia del sistema en su operación diaria.

Previo a su despliegue, es recomendable realizar pruebas y ajustes a todos los componentes del sistema, incluidos sensores y módulos de comunicación, para garantizar un desempeño fiable. Los sensores deben evaluarse en distintas condiciones para comprobar su funcionalidad, mientras que la red de transmisión debe estar configurada para enviar notificaciones sin demoras. Además, la sensibilidad de los algoritmos deben ajustarse según las características específicas de cada usuario.

Para evaluar adecuadamente el impacto del sistema, se sugiere incorporar un mayor número de cámaras para abarcar una mayor área de estudio y pruebas que permitan una apropiada interpretación frente a distintos escenarios, incrementando la cobertura y la cantidad de usuarios para el análisis.

REFERENCIAS

- [1] B. MUNDIAL, *La inclusión de la discapacidad*, Accedido: 15-10-24, 2023. dirección: <https://www.bancomundial.org/es/topic/disability>”.
- [2] O. P. de la salud y O. M. de la salud, *Discapacidad*, Accedido: 14-10-2024, 2024. dirección: <https://www.paho.org/es/temas/discapacidad>.
- [3] M. de salud publica de Ecuador, «3 de diciembre: Día internacional de las personas con discapacidad,» 2023.
- [4] W. E. A. ZAMORA, «Educación Superior en Ecuador con relación a la Inclusión de Personas con Discapacidad,»
- [5] J. Perez y Castro, *Entre barreras y facilitadores las experiencias de lo estudiantes universitarios con discapacidad*. DF Mexico y Mexico: Universidad Nacional Autonomas de Mexico, 2019.
- [6] J. Torbó, «Uso de dispositivos móviles como apoyo Herramienta para involucrar e interesar a los jóvenes Estudiantes en e-Salud,» *Universidad de Agder*, 2019.
- [7] E. A. AQUINO, «SISTEMA PARA ORIENTAR A PERSONAS CON DISCAPACIDAD VISUAL EN ESPACIOS CERRADOS MEDIANTE TECNOLOGÍA BEACON,» 2018.
- [8] J. M. O. A. ROMERO, «DE LOS DISPOSITIVOS, PREDICTORES DEL USO and TELEASISTENCIA,» 2011.
- [9] F. Benedetti, J. Acuña y B. Fabiani, «Teleasistencia: innovaciones tecnológicas para el cuidado de personas a distancia,» 2022.
- [10] J. P. Z. De Paz, C. A. I. Bohorquez, E. K. A. Rivera, F. J. B. Castillo y F. A. C. Velázquez, «SISTEMA DE TELEASISTENCIA MÉDICA TIPO SCADA PARA EL MONITOREO DE PACIENTES UTILIZANDO TECNOLOGÍA GSM,» *Pistas Educativas*, vol. 39, n.º 127, 2018.
- [11] J. Juan, *telemedicina y teleasistencia: ¿Cuál es la diferencia*, Accedido: 24-10-2024, 2023. dirección: <https://invoxmedical.com/blog/telemedicina-teleasistencia/>.
- [12] F. Benedetti, J. Acuña y F. Beatrice, *Teleasistencia: innovaciones tecnologicas para el cuidado de persona a distancia*, Accedido: 24-10-2024, 2022. dirección: <https://publications.iadb.org/es/teleasistencia-innovaciones-tecnologicas-para-el-cuidado-de-personas-distancia>.
- [13] I. Hernandez, *¿Qué es la Teleasistencia para personas mayores? - Cuidado mayor. Cuidado Mayor*. Accedido: 24-10-2024, 2021. dirección: <https://www.cuidadomayor.com/blog/teleasistencia-personas-mayores/>.
- [14] N. Portela, *¿Qué es la teleasistencia? Todo lo que debes saber*, Accedido: 24-10-2024, 2020. dirección: <https://teleasistencia.es/es/blog/asesoramiento-para-la-tercera-edad/que-es-la-teleasistencia-todo-lo-que-debes-saber>.
- [15] N. Kolban, «kolban’s Book on esp32,» 2018, Accedido: 30-11-2024. dirección: https://www.academia.edu/90611466/Kolban_ESP32_sep.
- [16] P. F. S.A., *Manual básico de pruebas del NodeMCU versión ESP32 DevKit V1. Arduino IDE*, Accedido: 24-10-2024, 2022. dirección: <https://www.puntoflotante.net/MANUAL-BASICO-NODEMCU-ESP32-ARDUINO.pdf>.
- [17] P. A. Y. H. Assistant, *ESP32 Wifi + Bluetooth en un solo lugar*, Accedido: 25-10-2024, 2021. dirección: <https://programarfácil.com/esp8266/esp32/#:~:text=ESP32%20es%20una%20familia%20de,y%20otras%20potentes%20caracter%C3%ADsticas%20extra..>
- [18] A. R. B. Saravia, «ESP32 NODE MCU,» 2019. dirección: https://www.microelectronicash.com/downloads/ESP32_MANUAL.pdf.
- [19] Espressif, «ESP32 SERIE Darashett vercio 4.7-2.4 GHz Wi-Fi + Bluetooth® + Bluetooth LE SoC,» *ESP32 SERIE Darashett vercio 4.7*, 2024.
- [20] S. A. O. Cruz, A. S. R. Rodríguez y W. P. Guerra, «Evaluación de cobertura de la comunicación entre microcontroladores ESP32,» *Encuentro Internacional de Educación en Ingeniería*, 2023.
- [21] HeTPro-Tutoriales, *MPU6050 Arduino, Acelerómetro y Giroscopio*, Accedido: 31-10-2024, 2014. dirección: <https://hetpro-store.com/TUTORIALES/modulo-acelerometro-y-giroscopio-mpu6050-i2c-twi/>.
- [22] MakerElectronico, «Acelerometro y Giroscopio MPU6050 — Arduino,» 2015, Accedido: 3-11-2024. dirección: <https://www.makerelectronico.com/mpu6050-acelerometro-y-giroscopio-con-arduino-leonardo/>.

- [23] P. Gokhale, O. Bhat y S. Bhat, «Introduction to IOT,» *International Advanced Research Journal in Science, Engineering and Technology*, vol. 5, n.º 1, págs. 41-44, 2018.
- [24] A. Developers, *Introducción a Android Studio*, Accedido: 3-11-2024, 2024. dirección: <https://developer.android.com/studio/intro?hl=es-419>.
- [25] Arduino.cc, *What is Arduino?* Accedido: 3-11-2024, 2018. dirección: <https://www.arduino.cc/en/Guide/Introduction>.
- [26] Microsiervos, *Wokwi: un simulador para proyectos Arduino, ESP32 y STM32*, Accedido: 3-11-2024, 2023. dirección: <https://www.microsiervos.com/archivo/hackers/wokwi-simulador-proyectos-arduino-esp32-stm32.html#:~:text=Wokwi%20es%20una%20fascinante%20herramienta,qu%C3%A9%20es%20lo%20que%20hace..>
- [27] postman, *What is Postman? Postman API Platform*, Accedido: 26-12-2024, 2022. dirección: <https://www.postman.com/product/what-is-postman/>.
- [28] Y. Muradas, *Qué es Postman y primeros pasos*, Accedido: 12-12-2024, 2022. dirección: "https://openwebinars.net/blog/que-es-postman/#qu%C3%A9-es-postman".
- [29] *ESP32-WROOM-32*. Accedido: 4-2-2025, 2025. dirección: https://octopart.com/es/esp32-wroom-32-espressif+systems-91025511?gad_source=1&gclid=CjwKCAiAh6y9BhBREiwApBLHC1ugyjZaOQLMBYjtybqWbnxNo4twgzRoCsRUQAvD_BwE.
- [30] *arduino uno — Octopart*. Accedido: 4-2-2025, 2025. dirección: <https://octopart.com/es/search?q=arduino+uno¤cy=USD&specs=0>.
- [31] *InvenSense MPU-6050*. Accedido: 4-2-2025, 2025. dirección: <https://octopart.com/es/datasheet/mpu-6050-invensense-19505926>.
- [32] *Pet Camera*. Accedido: 4-2-2025, 2025. dirección: <https://boykeep.com/collections/pet-camera>.
- [33] *18650 Battery Shield V8 Mobile Power Bank 3V/5V For Arduino ESP32 ESP8266 Wifi*. Accedido: 24-1-2025, 2019. dirección: <https://www.diymore.cc/products/18650-16340-lithium-battery-shield-v9-v8-v3-mobile-power-expansion-board-module-5v-3a-3v-micro-usb-type-c-for-arduino-esp8266>.

XII. ANEXOS

Table 1: ESP32-WROOM-32 Specifications

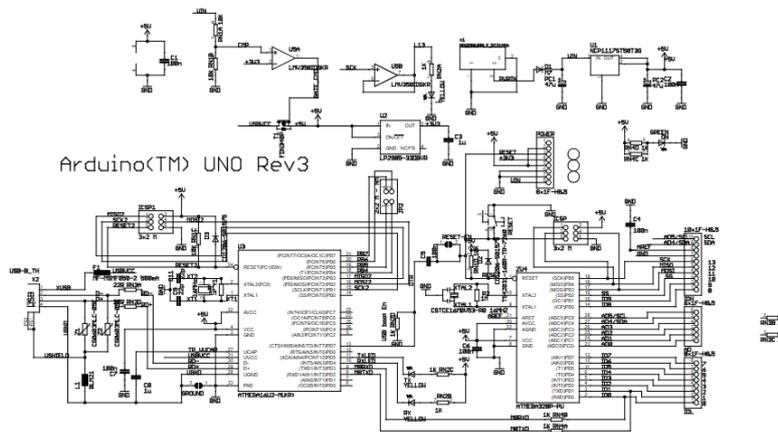
Categories	Items	Specifications
Certification	RF certification	FCC/CE-RED/IC/TELEC/KCC/SRRC/NCC
	Wi-Fi certification	Wi-Fi Alliance
	Bluetooth certification	BQB
	Green certification	RoHS/REACH
Test	Reliability	HTOL/HTSL/uHAST/TCT/ESD
Wi-Fi	Protocols	802.11 b/g/n (802.11n up to 150 Mbps) A-MPDU and A-MSDU aggregation and 0.4 μ s guard interval support
	Frequency range	2.4 GHz ~ 2.5 GHz
Bluetooth	Protocols	Bluetooth v4.2 BR/EDR and BLE specification
	Radio	NZIF receiver with -97 dBm sensitivity
		Class-1, class-2 and class-3 transmitter
		AFH

Espressif Systems

1

ESP32-WROOM-32 Datasheet V2.9

Figura 53. Ficha tecnica del componente ESP32 [29].



Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS. Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information. ARDUINO is a registered trademark.

Figura 54. Ficha tecnica del Arduino UNO [30].

Primary Differences between MPU-6000 and MPU-6050

Part / Item	MPU-6000	MPU-6050
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	n/a	1.71V to VDD
Serial Interfaces Supported	I ² C, SPI	I ² C
Pin 8	/CS	VLOGIC
Pin 9	AD0/SDO	AD0
Pin 23	SCL/SCLK	SCL
Pin 24	SDA/SDI	SDA

Figura 55. Ficha técnica del MPU-6050 [31].



Figura 56. Características de cámara boykeep [32].

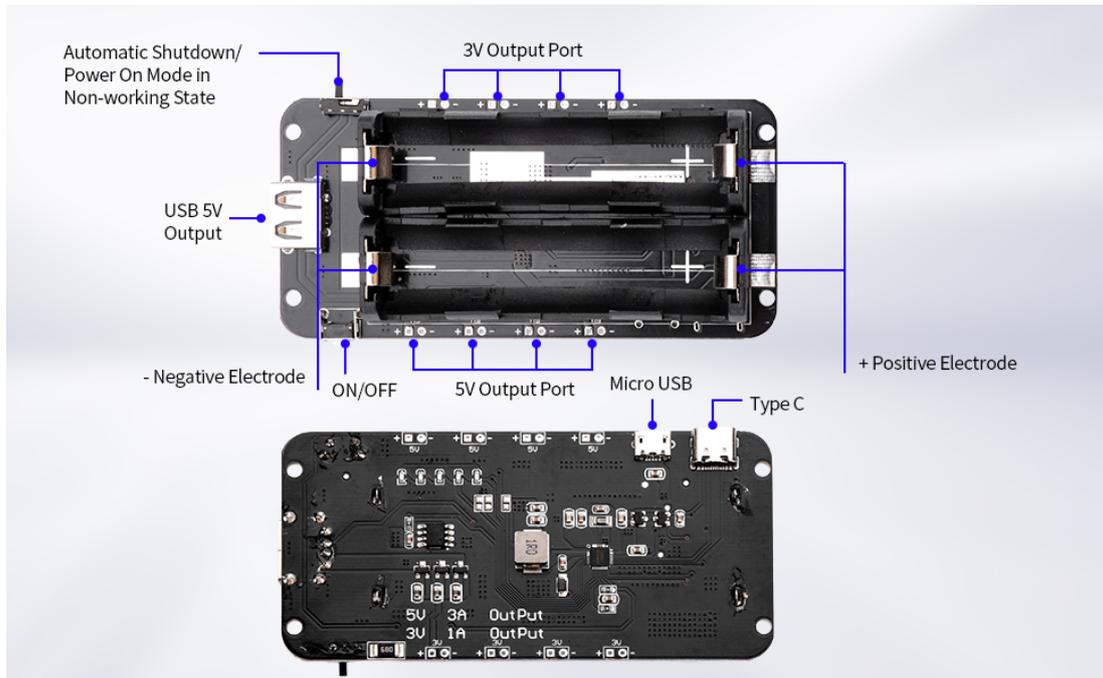


Figura 57. Características de soporte de batería de litio [33].

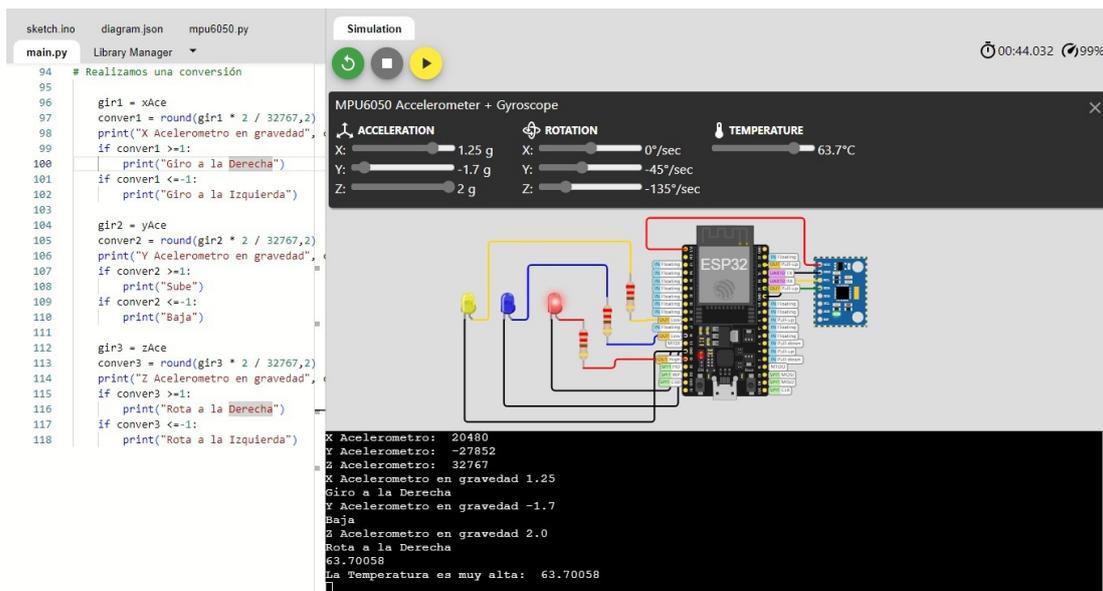


Figura 58. Simulación de las conexiones de prueba, Autor propio.

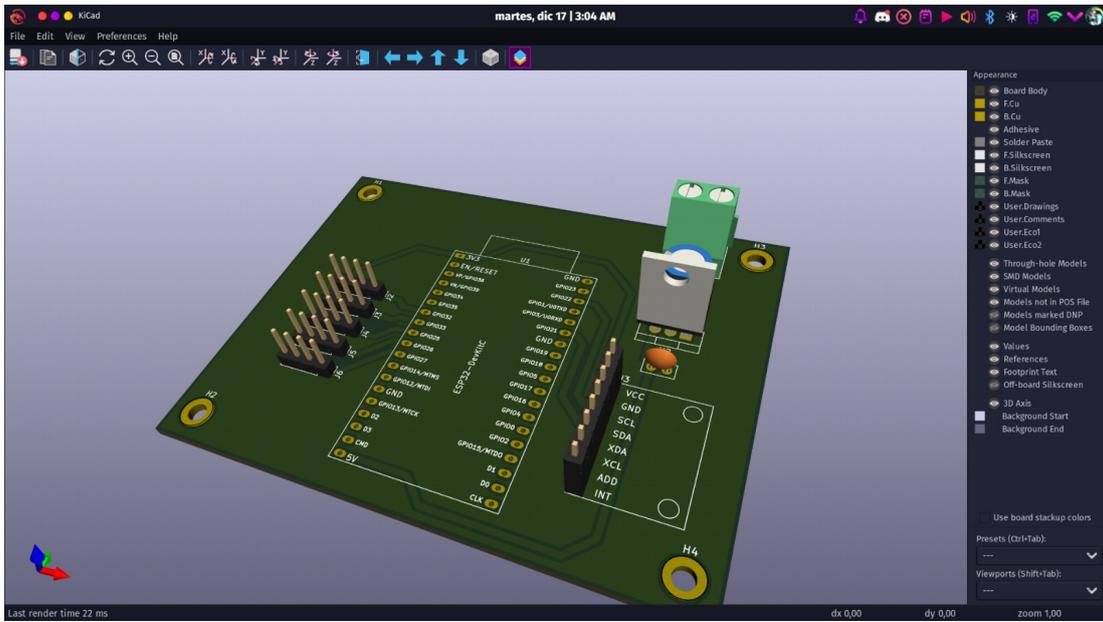


Figura 59. Simulación de la placa fallida, Autor propio.

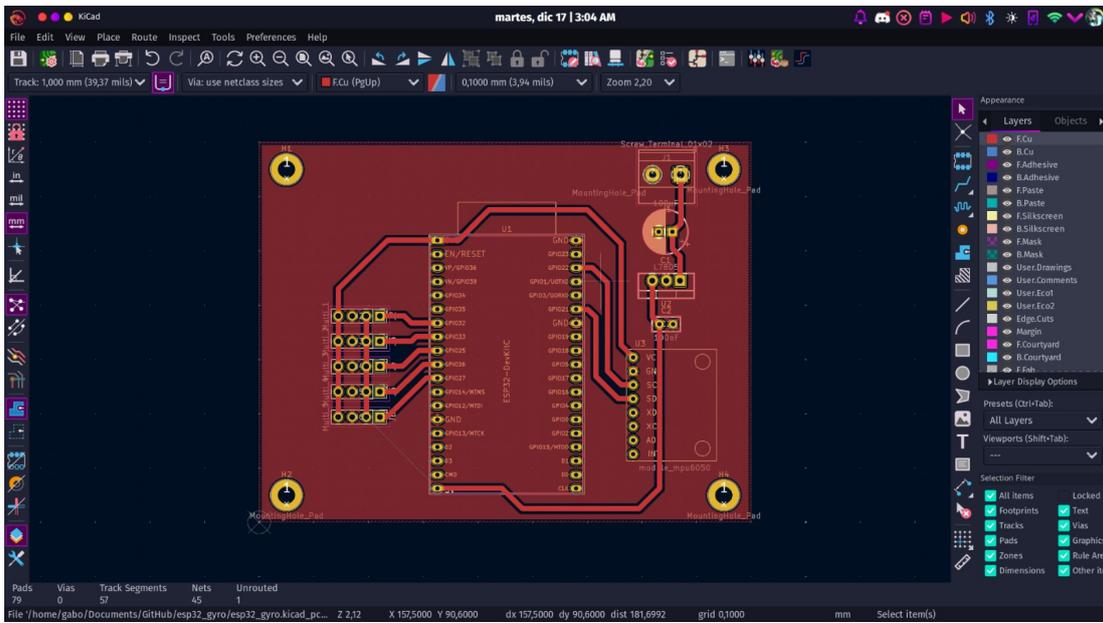


Figura 60. Simulación de la placa PCB fallida, Autor propio