



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE TELECOMUNICACIONES

Tema:

**SISTEMA DE REHABILITACIÓN FÍSICA PARA PACIENTES CON
LESIONES DE TOBILLO EMPLEANDO VISIÓN ARTIFICIAL PARA LA
UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA GABO'S**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la
obtención del título de Ingeniero en Telecomunicaciones

ÁREA: Comunicaciones

LÍNEA DE INVESTIGACIÓN: Tecnologías de la Información y Sistemas de
control

AUTOR: Gilson German Flores Achachi

TUTOR: Ing. Edgar Patricio Córdova Córdova, Mg.

Ambato – Ecuador

Agosto - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: SISTEMA DE REHABILITACIÓN FÍSICA PARA PACIENTES CON LESIONES DE TOBILLO EMPLEANDO VISIÓN ARTIFICIAL PARA LA UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA GABO'S, desarrollado bajo la modalidad Proyecto de Investigación por el señor Gilson German Flores Achachi, estudiante de la Carrera de Telecomunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, agosto 2023.

Ing. Edgar Patricio Córdova Córdova Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación titulado: SISTEMA DE REHABILITACIÓN FÍSICA PARA PACIENTES CON LESIONES DE TOBILLO EMPLEANDO VISIÓN ARTIFICIAL PARA LA UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA GABO'S es absolutamente original, autentico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2023



Gilson German Flores Achachi

C.C. 1804966453

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este Trabajo de Titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, agosto 2023



Gilson German Flores Achachi

C.C. 1804966453

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Gilson German Flores Achachi, estudiante de la Carrera de Telecomunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación titulado: SISTEMA DE REHABILITACIÓN FÍSICA PARA PACIENTES CON LESIONES DE TOBILLO EMPLEANDO VISIÓN ARTIFICIAL PARA LA UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA GABO'S, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2023.

Ing. Elsa Pilar Urrutia Urrutia, Mg
PRESIDENTE DEL TRIBUNAL

Ing. Geovanni Danilo Brito Moncayo, Mg
PROFESOR CALIFICADOR

PhD. Fabián Rodrigo Salazar Escobar
PROFESOR CALIFICADOR

DEDICATORIA

El presente proyecto de titulación va dedicado a mi familia, por todo el apoyo y la confianza que han puesto en el trayecto de mi vida universitaria.

A mis padres Aníbal Flores y Ernestina Achachi quienes me brindaron su ayuda incondicional, por su sacrificio para otorgarme todo lo necesario para concluir mis estudios.

A mis hermanos quienes han estado en los momentos más difíciles de mi vida, alentándome a no rendirme y conseguir siempre mis objetivos.

A mis amigos que me han ayudado en mi etapa universitaria, brindándome consejos para ser una mejor persona.

Gilson Flores

AGRADECIMIENTO

Agradezco a mis padres, hermanos y familiares por estar en todo momento brindándome su apoyo en los momentos difíciles de mi vida, guiándome a cumplir mis metas y ser una persona de bien.

Al Ing. Patricio Córdova por el tiempo y ayuda brindada para el desarrollo del proyecto, por guiarme en el cumplimiento de la investigación.

A la Facultad de Ingeniería en Sistemas, Electrónica e Industrial por brindarme el conocimiento y habilidades para ser un buen profesional.

Mi sincero agradecimiento al Lcdo. Gabriel Moreno y todos quienes conforman la Unidad de Rehabilitación y Fisioterapia Gabo's por la ayuda y la posibilidad de realizar la investigación en sus instalaciones.

Gilson Flores

ÍNDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL	viii
ÍNDICE DE TABLAS	xi
ÍNDICE DE FIGURAS	xii
RESUMEN EJECUTIVO	xv
ABSTRACT.....	xvi
CAPÍTULO I	1
MARCO TEÓRICO	1
1.1 Tema de investigación.....	1
1.1.1 Planteamiento del problema.....	1
1.2 Antecedentes investigativos	3
1.3 Fundamentación teórica	7
1.3.1 Anatomía del tobillo.....	7
1.3.2 Mecanismo de lesión.....	8
1.3.3 Lesión del tobillo.....	9
1.3.4 Medición goniométrica del tobillo	13
1.3.5 Visión artificial.....	14
1.3.6 Sensores.....	15
1.3.7 Captura de movimiento.....	16

1.3.8	Placas SBC (Single Board Computer)	16
1.3.9	Software	17
1.3.10	Protocolo MQTT	18
1.3.11	Plataformas IoT	18
1.3.12	Aplicación web.....	18
1.3.13	Funcionamiento de una aplicación web	19
1.4	Objetivos	20
1.4.1	Objetivo General	20
1.4.2	Objetivos Específicos.....	20
CAPÍTULO II		21
METODOLOGÍA		21
2.1	Materiales	21
2.2	Métodos.....	34
2.2.1	Modalidad de investigación	34
2.2.2	Recolección de información.....	35
2.2.3	Procesamiento y análisis de datos	35
CAPÍTULO III.....		36
RESULTADOS Y DISCUSIÓN		36
3.1	Análisis y discusión de los resultados.....	36
3.2	Desarrollo de la propuesta.....	36
3.2.1	Requerimientos del sistema o especificaciones necesarios para el desarrollo del sistema.....	36
3.2.2	Esquema general del sistema	38
3.2.3	Algoritmo de programación, adquisición y procesamiento	39
3.2.4	Servidor	43
3.2.5	Instalación de Mosquito Broker en Ubuntu	45

3.2.6	Instalación de Node Red	47
3.2.7	Instalación de InfluxDB	48
3.2.8	Instalación de Grafana.....	50
3.2.9	Diseño del entorno virtual.....	52
3.2.10	Diseño de la interfaz de visualización de los ángulos del tobillo	58
3.2.11	Pruebas de funcionamiento	67
3.2.12	Porcentaje de error del sistema de rehabilitación.....	81
3.2.13	Presupuesto	84
CAPÍTULO IV		87
CONCLUSIONES Y RECOMENDACIONES		87
4.1	Conclusiones	87
4.2	Recomendaciones.....	88
BIBLIOGRAFÍA		89
ANEXOS		96
Anexo 1. Validación por el Lcdo. Ramiro Gabriel Moreno Proaño.....		97
Anexo 2. Instalación del Kinect para Windows SDK2.0		98
Anexo 3. Creación del avatar.....		101
Anexo 4. Instalación del Jetpack		103
Anexo 5. Scripts necesarios para el entorno virtual en lenguaje C#.....		106
Anexo 6. Scripts del menú del entorno virtual		109
Anexo 7. Algoritmo para calcular el ángulo de las articulaciones en lenguaje Python.....		112

ÍNDICE DE TABLAS

Tabla 1. Clasificación de los esguinces de tobillo.....	10
Tabla 2. Tratamiento por fases para el esguince de tobillo	13
Tabla 3. Clasificación de la RV.....	15
Tabla 4. Cuadro comparativo de los dispositivos de captación de video	22
Tabla 5. Cuadro comparativo de los motores para entornos virtuales	24
Tabla 6. Comparativa de librerías de Visión artificial	26
Tabla 7. Cuadro comparativo de las microcomputadoras	28
Tabla 8. Protocolos de comunicación.....	30
Tabla 9. Comparativa de las herramientas de recolección	31
Tabla 10. Comparativa de las herramientas de almacenamiento	32
Tabla 11. Comparativa de plataformas graficas.....	33
Tabla 12. Comprobación del entorno virtual.....	68
Tabla 13. Analisis de error del ángulo tobillo flexión plantar.....	82
Tabla 14. Analisis de error del ángulo tobillo flexión dorsal	83
Tabla 15. Error general del sistema.....	84
Tabla 16. Presupuesto de implementación del sistema	86

ÍNDICE DE FIGURAS

Figura 1. Visión lateral de una disección osteoarticular del tobillo:1: ligamento talonavicular dorsal; 2: cápsula anterior del tobillo; 3: ligamento tibiofibular anterior; 4: cápsula posterior del tobillo; 5: tendón calcáneo	7
Figura 2. Esquema de los ligamentos peroneos y tibioperoneos	8
Figura 3. Imagen axial T2 donde se muestran los tendones flexores y extensores.....	8
Figura 4. Movimiento forzado. Inversión y Eversión respectivamente	9
Figura 5. Esguince de tobillo	9
Figura 6. Grados de esguince de tobillo.....	10
Figura 7. Ejercicio propioceptivo.....	12
Figura 8. Movimiento de flexión dorsal y plantar.....	14
Figura 9. Modelado en 3D	16
Figura 10. Funcionamiento del protocolo MQTT.....	18
Figura 11. Esquema de funcionamiento de una aplicación web	19
Figura 12. Detección del esqueleto humano en 2D mediante Kinect Studio.....	23
Figura 13. Detección del esqueleto humano en 3D mediante Kinect Studio.....	23
Figura 14. Creación del avatar	25
Figura 15. Puntos de referencia MediaPipe Pose	27
Figura 16. Recolección y almacenamiento	33
Figura 17. Plataforma para visualización.....	34
Figura 18. Movimiento articular del tobillo	37
Figura 19. Escenario de pruebas	37
Figura 20. Esquema general del sistema	38
Figura 21. Angulo constituido por los 3 puntos de la articulación	41
Figura 22. Conexión SSH mediante PuTTY.....	44
Figura 23. Versión del Sistema Operativo	44
Figura 24. Estado activo de Mosquitto	45
Figura 25. Configuración del archivo mosquitto.conf	46
Figura 26. Publicador.....	47
Figura 27. Suscriptor.....	47
Figura 28. Versión de Node.js.....	47
Figura 29. Servicio activo de Node-Red	48
Figura 30. Versión instalada de influxdb	49

Figura 31. Servicio activo de influxdb	49
Figura 32. Archivo agregado al repositorio	50
Figura 33. Servicio de Grafana activo.....	51
Figura 34. Acceso a Grafana.....	51
Figura 35. Dimensiones del terreno	52
Figura 36. Avatar en la escena de Unity	53
Figura 37. Diseño de la escena.....	53
Figura 38. Escena completa avatar hombre	54
Figura 39. Escena completa avatar mujer	54
Figura 40. Menú principal.....	56
Figura 41. Boton salir en la escena	57
Figura 42. Ejecutable del entorno virtual.....	57
Figura 43. Configuración del nodo influx.....	58
Figura 44. Datos enviados a influxdb	59
Figura 45. Menú de opciones	59
Figura 46. Configuración de influxdb en grafana	60
Figura 47. Dashboard para visualizar resultados	61
Figura 48. Paneles de visualizacion de resultados	61
Figura 49. Interfaz de registro de pacientes	62
Figura 50. Base de datos	63
Figura 51. Modelo entidad-relación.....	63
Figura 52. Tabla de los pacientes registrados en la base de datos	64
Figura 53. Tabla para generar el reporte del paciente.....	64
Figura 54. Flujograma.....	66
Figura 55. Pruebas de funcionamiento en pacientes	67
Figura 56. Sistema implementado.....	70
Figura 57. Distancia del paciente respecto al sistema.....	70
Figura 58. Datos ingresados del paciente sano	71
Figura 59. Ángulo en flexión plantar	72
Figura 60. Ángulo en flexión dorsal	72
Figura 61. Ángulo formado en flexión plantar (paciente sano)	73
Figura 62. Ángulo formado en flexión dorsal (paciente sano).....	73
Figura 63. Reporte del paciente sano	74

Figura 64. Datos ingresados del paciente con lesión	75
Figura 65. Ángulo de los tobillos (paciente lesionado).....	76
Figura 66. Ángulo formado en flexión plantar (paciente lesionado)	76
Figura 67. Ángulo formado en flexión dorsal (paciente lesionado).....	77
Figura 68. Reporte inicial del paciente lesionado	77
Figura 69. Paciente en recuperación	78
Figura 70. Ángulo formado en flexión plantar (paciente lesionado)	79
Figura 71. Ángulo formado en flexión plantar (paciente lesionado)	79
Figura 72. Reporte del paciente.....	80
Figura 73. Paciente iniciando la terapia física.....	80
Figura 74. Paciente realizando ejercicios con el sistema	80
Figura 75. Ángulos calculados por el sistema.....	81
Figura 76. Ángulos medidos con el goniómetro	81

RESUMEN EJECUTIVO

El presente proyecto está orientado a la implementación de un sistema de rehabilitación física para pacientes con lesiones de tobillo empleando visión artificial en la Unidad de Rehabilitación y Fisioterapia Gabo's. Para el entorno virtual se emplea una cámara Kinect V2.0 el mismo que capta los movimientos del paciente y los replica en el avatar que fue desarrollado en Unity, este avatar es presentado en un monitor para que el paciente observe sus movimientos.

El sistema implementado se compone de una minicomputadora Jetson Nano, en el cual se ejecuta el programa de Python para analizar y procesar los datos de las articulaciones del paciente. Se calcula los ángulos formados por los tres puntos de referencia y son enviados empleando el protocolo MQTT a Node-Red que se encarga de recolectar los datos y posteriormente almacenarlos en Influxdb. Se genera la base de datos con la información de los ángulos del tobillo, se realiza la respectiva configuración de influxdb en grafana, para observar la información almacenada en una interfaz gráfica donde se puede apreciar los ángulos generados por los tobillos.

El sistema implementado en la unidad de fisioterapia en comparación con las mediciones obtenidos de forma manual empleando un goniómetro presenta una fiabilidad del 95,28% en promedio total, en la obtención de los ángulos de la articulación.

Palabras clave: Sistema de rehabilitación, entorno virtual, visión artificial, tobillo.

ABSTRACT

This project is oriented to the implementation of a physical rehabilitation system for patients with ankle injuries using artificial vision in the Gabo's Rehabilitation and Physiotherapy Unit. For the virtual environment a Kinect V2.0 camera is used which captures the patient's movements and replicates them in the avatar that was developed in Unity, this avatar is presented on a monitor for the patient to observe their movements.

The implemented system consists of a Jetson Nano minicomputer, which runs the Python program to analyze and process the patient's joint data. The angles formed by the three reference points are calculated and sent using the MQTT protocol to Node-Red, which is responsible for collecting the data and subsequently storing them in Influxdb. The database with the information of the ankle angles is generated, the respective configuration of Influxdb is made in Grafana, to observe the information stored in a graphical interface where the ankle angles generated by the ankles can be appreciated.

The system implemented in the physiotherapy unit in comparison with the measurements obtained manually using a goniometer presents a reliability of 95.28% in total average, in obtaining the angles of the joint.

Keywords: Rehabilitation system, virtual environment, artificial vision, ankle.

CAPÍTULO I

MARCO TEÓRICO

1.1 Tema de investigación

SISTEMA DE REHABILITACIÓN FÍSICA PARA PACIENTES CON LESIONES DE TOBILLO EMPLEANDO VISIÓN ARTIFICIAL PARA LA UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA GABO'S

1.1.1 Planteamiento del problema

A nivel mundial de acuerdo con la Organización mundial de la Salud (OMS) las lesiones articulares están denominadas como afecciones que provocan la discapacidad, siendo la parte inferior con más incidencia en la población activa [1]. El esguince de tobillo es el traumatismo más común del sistema musculo esquelético que afectan tanto a los deportistas como a personas en general, debido al estiramiento o ruptura parcial o total de al menos un ligamento de esta zona anatómica. A pesar de que muchos casos son atendidos en el mismo hogar se calcula una prevalencia de 1 esguince por cada 10.000 habitantes en los servicios de urgencias [2].

En Estados Unidos en un periodo de 5 años surgieron 3 millones de esguinces en los departamentos de emergencia, la mitad de estos sucedieron mientras se realizaban actividades deportivas. Datos de la National Collegiate Athletic Association (NCAA) revelan que los esguinces laterales de tobillo (LAS) son el tipo más usual de esguince de tobillo. En un lapso de estudio de 6 años se documentaron 2.429 LAS, 480 esguinces de tobillo alto y 380 esguinces de tobillo medial. Se obtuvieron las tasas más elevadas de LAS en el baloncesto masculino y femenino con 11,96 y 9,5 respectivamente. Además, en esguinces de tobillo alto se presentaron las tasas más altas en futbol masculino con 2.42, lucha libre 2.11 y hockey 1.77 por 10.000 atletas expuestos [3].

En América Latina las lesiones por esguince de tobillo grado II suceden con más frecuencia en jóvenes adultos, principalmente de 21 a 30 años debido al aumento de

actividades deportivas, estas lesiones corresponden al 12% en ser atendidas en urgencias y también es la lesión peor tratada, cerca del 20% a 30% de todas las lesiones son deportivas. Además, en edades menores o mayores los esguinces suelen evolucionar de la peor forma y la lesión es más grave. Alrededor del 44% presentan secuelas un año después de haber padecido un esguince [4].

En Ecuador de acuerdo con el Instituto Ecuatoriano de Estadísticas y Censos (INEC) mediante el Registro Estadístico de Camas y Egresos Hospitalarios en el año 2020 se presentaron 316 casos de esguinces de tobillo siendo más frecuente en pacientes de 15 a 44 años, El mayor porcentaje se ve reflejado en la población masculina siendo un 57,91%, en comparación con la femenina con un 42,09% [5].

Se calcula que un 55% de las personas que sufren un esguince de tobillo no buscan un diagnóstico o terapia de un profesional de la salud. La desestimación de la seriedad de la lesión juntamente con inapropiados protocolos de curación supone una ineficiente prevención de indicios de prevalencia de dicho traumatismo. Aproximadamente un 60% de pacientes que han tenido un esguince tendrán una inestabilidad funcional del tobillo a causa de varios factores como la falta de propiocepción, incremento de la laxitud del ligamento y debilidad del músculo siendo una afección de gran problemática que perjudica la condición de vida de la población que lo padece [6].

En las sesiones de fisioterapia se realizan ejercicios de estímulo mismas que intentan activar como también reactivar algunos comportamientos, tales como movimientos o acciones, limitaciones y discapacidades. Al ser movimientos repetitivos causan desmotivación en los pacientes conllevando en algunos casos el abandono de la fisioterapia. Con el aumento de la tecnología, la rehabilitación ha aprovechado la realidad virtual para perfeccionar opciones y alternativas de nuevos tratamientos, que mejore el impacto de la fisioterapia. Obteniendo el interés, concentración y entretenimiento del paciente al realizar las actividades físicas dispuestas dentro de su tratamiento [7].

La tele rehabilitación puede ser una opción rentable de servicio a distancia para superar los inconvenientes que pueden dificultar que un paciente reciba atención en una clínica. Con el acceso de esta tecnología se puede evitar viajes estresantes a las unidades de rehabilitación incluso se puede disminuir el costo por las sesiones de terapia física con la seguridad que obtener un adecuado tratamiento de acuerdo con su condición. Además, en el hogar conservando su entorno familiar, el estado anímico ayuda a la recuperación [7].

1.2 Antecedentes investigativos

Después de haber realizado la búsqueda de información en las distintas bases de datos y repositorios de universidades a nivel nacional e internacional, y artículos relacionados con el tema, se analizaron los siguientes casos:

Juan Ayerve y César Rosero, en el año 2018 presenta un trabajo titulado: DESARROLLO DE UN SISTEMA INTERACTIVO PARA ENTRENAMIENTO Y REHABILITACIÓN FÍSICA DE PERSONAS CON DISCAPACIDADES COGNITIVAS DENTRO DE LA FUNDACIÓN FINE, MEDIANTE LA UTILIZACIÓN DEL SENSOR DE MOVIMIENTOS KINECT, en la Universidad Politécnica Salesiana sede Quito. En el proyecto se desarrolla un sistema que ayuda en el proceso de la rehabilitación de motricidad fina y gruesa. Se ha utilizado la rehabilitación motriz juntamente con el desarrollo del Video juego y el dispositivo Kinect 2.0. También se usa un motor de Video Juegos e Interfaces Naturales de Usuarios que permite imitar e interactuar en un entorno semejante al real con los pacientes, de esta manera se consigue motivarlos a cumplir con el proceso de forma amigable y divertida, reduciendo los tiempos en la recuperación. Los tiempos de respuesta son más rápidos debido a que el sensor Kinect 2.0 incorpora mejor resolución de video, mapeo de 6 objetos y mayor rango de distancia de captación permitiendo de esta manera al motor de Videojuegos Unreal Engine renderizar simultáneamente los objetos o assets situados en el entorno, además asigna texturas comparadas con las del mundo real, ocasionando una respuesta efectiva en la experiencia final del usuario [8].

Elodie Medina y Víctor Rofes, en el año 2019 presenta un trabajo titulado: VIRTUAL REALITY REHABILITATION APP (UPPER BODY), en la Universidad Politécnica de Catalunya BarcelonaTech. En la investigación se desarrolla una aplicación de realidad virtual para el tratamiento de las extremidades superiores. Para la realización de la investigación se ha utilizado Unity3D el cual permite la creación de aplicaciones computacionales, sensor Kinect para la captura de movimientos, gafas de realidad virtual (HTC Vive) y phpMyAdmin para crear las bases de datos. La aplicación puede ser utilizada por el paciente y el fisioterapeuta. El paciente accede y selecciona un ejercicio a ejecutar y el fisioterapeuta puede ver el historial de los pacientes y así llevar un seguimiento individualmente, los ejercicios fisioterapéuticos son asignados mediante el movimiento de un avatar el cual el paciente debe imitar. Con esto se logra que los ejercicios sean menos repetitivos y más entretenidos para los pacientes [9].

Laura Pedreros y Martí Fabregat, en el año 2019 presenta un trabajo titulado: APLICACIÓN DE REALIDAD VIRTUAL PARA LA REHABILITACIÓN, en la Universidad Politécnica de Catalunya BarcelonaTech. El proyecto consiste en la rehabilitación motriz mediante un juego, la finalidad es recolectar el mayor número de monedas mientras el paciente se esquivo de varios obstáculos. Por medio de una interfaz inmersiva y entretenida logrando una rehabilitación en un ambiente divertido que motive a realizar la terapia. Se ha empleado la plataforma de creación de aplicaciones computacionales Unity3D donde se desarrolló la programación para el entorno virtual, el sensor Kinect para capturar los movimientos y extraer valores de interés al realizar la terapia, gafas de realidad virtual (HTC Vive) y para la creación de la base de datos el software phpMyAdmin. Consiguiendo una rehabilitación favorable y amenizada en cada sesión, con 3 distintos niveles de dificultad y entornos de realidad virtual [10].

José Jarrín, en el año 2020, presenta un trabajo titulado: SISTEMA DE DETECCIÓN DEL ÁNGULO ARTICULAR EN LOS MOVIMIENTOS DE MIEMBRO SUPERIOR PARA EVALUACIÓN EN FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL, en la Universidad Técnica del Norte. En el proyecto se desarrolla un sistema de medición de ángulos mediante visión artificial con la utilización de un algoritmo de apreciación de pose humana misma que posiciona los puntos articulares

formando extremidades en un esquema de 2D del paciente. Para el tratamiento de imágenes se emplea una CPU, la conexión cableada vía USB entre la cámara web de alta resolución y el servidor, para el almacenamiento de los registros se usa un disco duro; los requerimientos de software se examina la compatibilidad del sistema con las bibliotecas de visión artificial de OpenCV, bibliotecas de aprendizaje automático, software de base de datos y de diseño de interfaces de usuario. Con la utilización de tecnología en ramas de la medicina provee herramientas que facilitan tanto el trabajo de los profesionales, como reducir el tiempo y mejorar los procesos de evaluación y diagnóstico del paciente, el sistema al no ser invasivo para el paciente en la evaluación permite una claridad en el registro de los datos al profesional encargado. También se convierte en un instrumento novedoso debido a que tiene un nivel de confiabilidad del 92,60% [11].

José Cárdenas y Jeyber Gómez, en el año 2021, presenta un trabajo titulado: SISTEMA DE APOYO DE REHABILITACIÓN PARA LA MEDICIÓN DE LOS RANGOS DE MOVILIDAD DE LA MANO USANDO VISIÓN ARTIFICIAL, en la Universidad de Pamplona. El proyecto consiste en una herramienta de soporte para el profesional fisioterapeuta la cual ayuda a tomar las mediciones de los ángulos en tiempo real de forma veloz y eficiente. Para la captura de datos se realiza mediante el software leap motion, usando distintas herramientas facilitadas por el software SDK de leap motion, mismo que permite establecer la posición px, py, pz de las articulaciones y la punta de los dedos en el área de trabajo del sensor. Por medio del software XAMPP se realiza una base de datos para guardar el registro y el historial de cada paciente, todo esto localmente. El uso del sensor Leap motion optimiza la medición de los rangos de movilidad de la mano, facilita la captura y el análisis de los datos del movimiento realizado por el paciente (flexión, extensión, abducción y aducción), dentro de un rango confiable de medida; estos datos son guardados y organizados para el soporte del fisioterapeuta efectuando el respectivo diagnóstico y de esta manera se encargará de valorar la rehabilitación del paciente [12].

David Polit, Carmen Granizo, María Rodríguez y Sandra Cedeño , en el año 2022 presenta un trabajo titulado: TERAPIA DE ARTROSIS CON CELULAS MADRE. NUEVOS AVANCES, en Santo Domingo. En el artículo de investigación se presenta el estudio de células madre. En traumatología las células madre se emplean, principalmente, para tratar la artrosis articular y la pseudoartrosis de huesos. El tratamiento de la artrosis de rodilla con células madre mesenquimales ha obtenido un importante alivio de los síntomas en más del 90% de los pacientes con artrosis avanzada. La terapia lipogems es uno de los últimos y prometedores adelantos en la medicina regenerativa para impedir las enfermedades degenerativas articulares y ayudar en la curación de lesiones en tejidos reparados. Uno de los tratamientos fisioterapéuticos más empleados en el caso de las tendinopatías es el entrenamiento excéntrico de la fuerza, la cual favorece la funcionalidad del paciente, así como la disminución del dolor y demás síntomas como consecuencias de la artrosis. [13]

Bárbara Espuche, en el año 2022 presenta un trabajo titulado: CONSIDERACIONES ÉTICAS Y LEGISLACIÓN SOBRE LA TERAPIA CON CÉLULAS MADRE. En el artículo de investigación se presenta el estudio de células madre. El cuerpo humano cuenta con órganos y organismos que son capaces de regenerarse. La terapia con células madre es una técnica que ha evolucionado en la cual se emplean como medicamentos las propias células, sirviendo ésta como elemento regenerativo ya que estas células sirven de sustitutas a las que están dañadas y son deficientes, regenerándolas. Las terapias celulares se llevan a cabo mediante preparados celulares que incluyen células madre, éstas son sometidas a procedimientos de manipulación para activar su función terapéutica o reparadora de tejidos dañados. Los preparados que se crean contienen sustancias vivas con potencial de diferenciación y proliferativo que siguen con su acción metabólica dentro del organismo que lo recibe, antes de su administración se debe comprobar la seguridad y eficacia mediante la investigación clínica. [14]

1.3 Fundamentación teórica

A continuación, se presenta la fundamentación científica de las variables en estudio de la investigación.

1.3.1 Anatomía del tobillo

El tobillo es una articulación tipo bisagra muy bien adaptado, en la que el astrágalo lo encaja correctamente en las zonas articulares tibial y fibular. Esta conformación permite el movimiento de un solo eje produciendo la flexión plantar y flexión dorsal en un rango de 13-33° para la flexión dorsal y 23-56° para la flexión plantar [15].

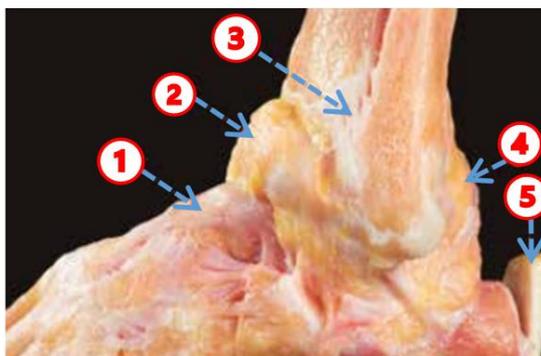


Figura 1. Visión lateral de una disección osteoarticular del tobillo: 1: ligamento talonaviclar dorsal; 2: cápsula anterior del tobillo; 3: ligamento tibiofibular anterior; 4: cápsula posterior del tobillo; 5: tendón calcáneo. [15]

Los huesos

El tobillo está formado por la tibia, peroné, astrágalo y calcáneo, divididos en 2 articulaciones la tibioastragalina compuesto por tibia, peroné y astrágalo y la su astragalina formada por el astrágalo y el calcáneo [16].

Los ligamentos

Fibras densas de tejido conectivo que une 2 huesos entre sí, estas fibras están compuestas de colágeno tipo I en 85% ubicados paralelamente. Los ligamentos del tobillo están divididos en 4 grupos [16]:

- ligamentos colaterales mediales (tibiales),
- laterales (peroneos),
- los del seno del tarso y

- los tibioperoneos.

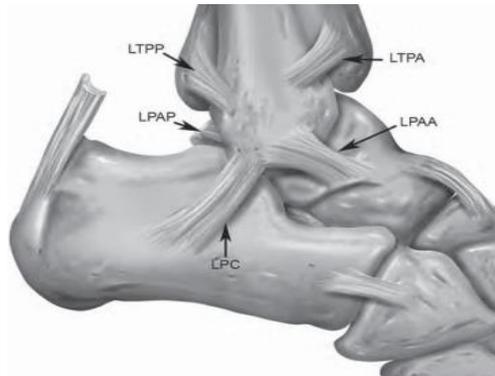


Figura 2. Esquema de los ligamentos peroneos y tibioperoneos. [16]

Los tendones

La finalidad primordial es transmitir el movimiento de forma pasiva de un músculo de contracción a un hueso o fascia. Conformados por fibras de colágeno, elastina y reticulina que les atribuyen resistencia, elasticidad y volumen [16].

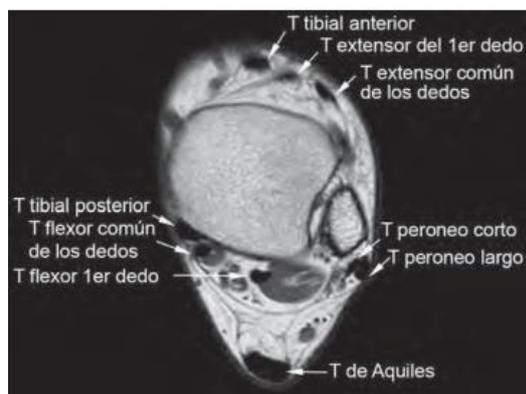


Figura 3. Imagen axial T2 donde se muestran los tendones flexores y extensores [16]

1.3.2 Mecanismo de lesión

El mecanismo lesional es continuamente indirecto, siendo un movimiento forzado torsional el que provoca la distensión de los tejidos capsulo- ligamentosos. El movimiento frecuente es una inversión forzada del pie equino + supinación [17].



Figura 4. Movimiento forzado. Inversión y Eversión respectivamente [17]

El 16-21% de las lesiones traumáticas deportivas es afectado al ligamento lateral externo, concretamente el fascículo peroneoastragalino anterior (PAA) le sigue el ligamento peroneocalcáneo (PC) y si la fuerza no lo detiene se puede lesionar el peroneoastragalino posterior (PAP) [17].

Realizado la investigación acerca de las causas más frecuentes que conllevan a traumatismos se procede a describir todo lo relacionado con las lesiones del tobillo.

1.3.3 Lesión del tobillo

El denominado esguince de tobillo es un estiramiento de ligamentos que sobrepasa el límite anatómico y fisiológico, alargando en exceso al ligamento provocando una rotura parcial o total [4].

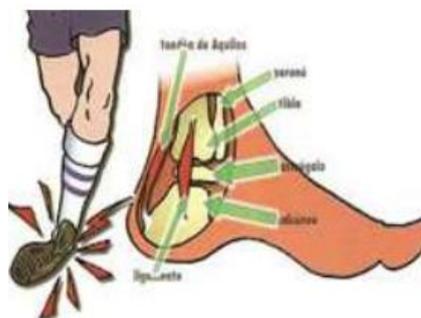


Figura 5. Esguince de tobillo [4]

Clasificación de los esguinces de tobillo

Se han especificado cuantiosos sistemas de clasificación de los grados de un esguince agudo del tobillo anatómicamente en función del número de ligamentos afectados. Convencionalmente se han clasificado basados en signos clínicos y alteración funcional en tres grados [17]:

Tabla 1. Clasificación de los esguinces de tobillo [17]

GRADO	LESIÓN	TUMEFACCIÓN Y EQUIMOSIS	RELACIÓN DOLOR/CARGA
I	Distensión	Mínima	No
II	Rotura Parcial	Moderada	Leve o Moderado
III	Rotura Completa	Severa	Severo o incapacitante

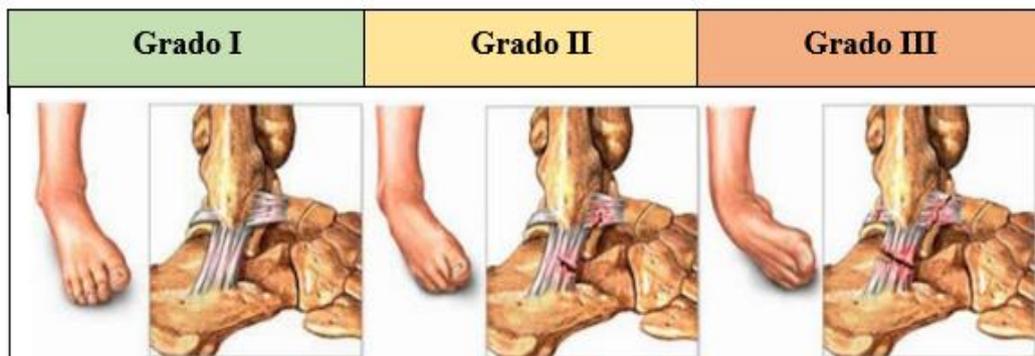


Figura 6. Grados de esguince de tobillo [18]

Síntomas

Los signos y síntomas de un esguince de tobillo varían según la gravedad de la lesión de acuerdo con la Tabla 1 pueden incluir los siguientes aspectos [19]:

- Dolor cuando soporta peso el pie afectado.
- Sensibilidad al tocar el tobillo.
- Hinchazón.
- Aparición de moretones.

- Movimiento limitado.
- Inestabilidad del tobillo.
- Sensación o sonido de chasquido

Causas

Sucede al forzar el tobillo fuera de su posición habitual se sufre un esguince, lo que ocasiona que uno o más ligamentos del tobillo se estiren y se desgarran parcial o totalmente. Para que se ocasione un esguince de tobillo puede comprender las siguientes posibles causas [19]:

- Torcedura del tobillo por una caída.
- Caer mal en un pie después de girar o de saltar
- Realización de actividad cotidiana o deportiva sobre una superficie irregular.
- Durante una actividad deportiva al ser pisado el pie por otra persona.

Complicaciones

El no ser tratado correctamente el esguince de tobillo, hacer actividades muy pronto después de haber sufrido la lesión o sufrir esguinces reiteradamente puede tener los siguientes agravamientos [19]:

- Dolor crónico del tobillo.
- Inestabilidad crónica de la articulación del tobillo.
- Artritis en la articulación del tobillo.

Prevención

Las siguientes sugerencias pueden contribuir a evitar un esguince de tobillo o a prevenir que se repita [19]:

- Antes de realizar ejercicio o practicar deportes hacer un precalentamiento.
- Tener cuidado con las superficies irregulares.
- Utilizar algún tipo de inmovilización o cinta en el tobillo lesionado.
- Usar calzado adecuado.
- Disminuir el uso de zapatos de taco alto.
- No practicar deportes ni participar en actividades para los que no se está preparado.
- Conservar una fuerza y flexibilidad muscular.

- Practicar entrenamiento de equilibrio.

Tratamiento

La finalidad del tratamiento básico es reducir el edema y dolor para conseguir movilización y apoyo tempranos. Se debe efectuar de manera consecutiva, en el transcurso de las fases inflamatoria, reparativa y de remodelación, revalorando la gravedad y estadiaje del esguince [17].

Ejercicios propioceptivos y rehabilitación para esguinces. La propiocepción es el sentido que informa al organismo de la posición de los músculos. Se trata de un aspecto muy importante a desarrollar para evitar lesiones, ya que, cuando mayor sensibilidad propioceptiva se tenga mejor se notará el tobillo [17].



Figura 7. Ejercicio propioceptivo [17]

Ejercicios de rehabilitación para el esguince de tobillo

El tratamiento se lo realiza en 3 fases de acuerdo con el criterio del profesional encargado, estas sugerencias son orientadas para esguinces de grado I y II leves [18].

Primera Fase

En las primeras 72 horas optar por el reposo absoluto manteniendo el pie en alto, aplicación de un vendaje que el medico crea adecuado según el grado de lesión [18].

Segunda Fase

Mantener un reposo relativo a partir del cuarto al séptimo día según el grado de lesión. De ser el caso se puede comenzar con el apoyo gradual con muletas, paso a paso tolerando el dolor. Se puede utilizar tobillera u ortesis semirrígida de tobillo. Otra

opción son los vendajes funcionales o elásticos. En este punto se puede iniciar con ejercicios isométricos [18].

Tercera Fase

Pasado los 15 días utilizar tobillera o media de compresión hasta que los síntomas de dolor desaparezcan. Seguir con ejercicios de fortalecimiento y a partir de la cuarta semana realizar ejercicios propioceptivos. No realizar actividades deportivas hasta pasados 6 semanas [18].

Fisioterapia para el esguince de tobillo

El paciente los puede realizar en casa y es beneficioso para todos los grados de lesión del tobillo. Los ejercicios son los siguientes [18]:

1. Ejercicios isométricos (sin mover la articulación);
2. Ejercicios en descarga de movilización articular.
3. Ejercicios de fortalecimiento con bandas elásticas.
4. Ejercicios propioceptivos.

Tabla 2. Tratamiento por fases para el esguince de tobillo [18].

GRADO ESGUINCE	Ejercicios isométricos	Ejercicios en descarga	Ejercicios en carga	Ejercicios propioceptivos
Grado I	4ª día	7º día	15º día	21º día
Grado II leve	4º día	7º día movilización 15º día elásticos	21º día	4-6 semanas
Grado II moderado	7º día	15º días movilización 21º día elásticos	3-6 semanas	6 semanas
Grado III	Según criterio traumatólogo y rehabilitador y tipo de tratamiento realizado			

1.3.4 Medición goniométrica del tobillo

Se debe partir de la posición neutra para proceder con la medición; la dorsiflexión o flexión dorsal posee un rango de entre 20 y 30 grados y para realizar una marcha

correcta es necesario simplemente 10 grados. En cuanto al rango de movilidad en la plantiflexion o también llamada flexión plantar va de entre 40 y 50 grados con un mínimo de 20 grados para una marcha correcta. En la Figura 8 se puede observar la movilidad del tobillo [20].

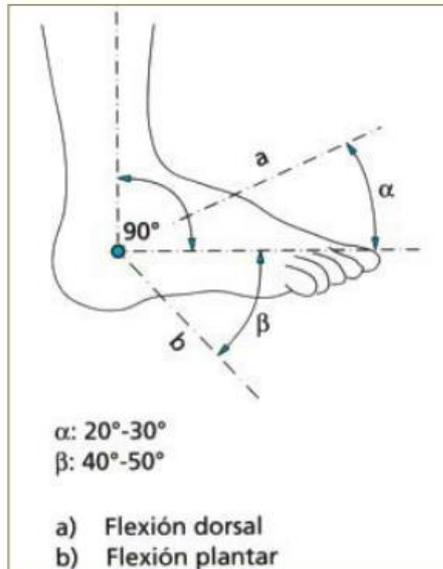


Figura 8. Movimiento de flexión dorsal y plantar [20]

Una vez realizado el estudio de la anatomía del tobillo e identificado las partes susceptibles de lesiones se procederá a revisar el tema de visión artificial.

1.3.5 Visión artificial

La visión artificial es una rama de la inteligencia artificial que se enfoca en la creación de sistemas capaces de interpretar y comprender imágenes y videos. Esta tecnología utiliza algoritmos y modelos de aprendizaje automático para analizar y extraer información útil de imágenes y videos, lo que la convierte en una herramienta valiosa en una variedad de aplicaciones, incluyendo la seguridad, la robótica, la medicina y la industria. La visión artificial también está evolucionando rápidamente gracias a la creciente disponibilidad de datos y la mejora de los procesadores y algoritmos de aprendizaje automático [21].

Realidad Virtual (RV)

Esta tecnología permite crear entornos virtuales mismo que son interactuados por el usuario. Son desarrollados de la manera más realista posible con la finalidad que el usuario no distinga diferencia con la vida real. Utilizada como una herramienta para el aprendizaje y el entrenamiento [22].

Tipos de Realidad Virtual

Se los puede clasificar de acuerdo con el grado de inmersión en que el usuario se encuentre. En la Tabla 3 se observa una comparativa de los tipos de RV [10].

Tabla 3. Clasificación de la RV [10].

Características Principales	RV no inmersiva	RV semi inmersiva	RV inmersiva
Resolución	Alta	Alta	Baja Media
Percepción (Escala)	Baja	Media-Alta	Alta
Sensación de inmersión	Ninguna-Baja	Media-Alta	Alta
Sensación del conocimiento del espacio	Baja	Media	Alta
Retraso	Baja	Baja	Media-Alta
Sensación de observación	Baja	Media	Alta

1.3.6 Sensores

Los sensores son dispositivos que captan una acción externa es decir obtienen y transforman magnitudes físicas las cuales pueden ser luz, magnetismo, presión, temperatura, etc. En variables eléctricas [23].

1.3.7 Captura de movimiento

Es un método empleado para grabar movimiento de personas. La información de los movimientos y acciones es usada para la animación a personajes digitales modelados en 3D [24].

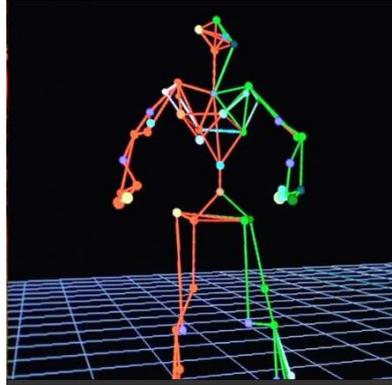


Figura 9. Modelado en 3D [24].

Captura de movimiento sin marcadores

Este método no necesita marcadores de ningún tipo. Es basado en cámaras sensibles a la profundidad y para seguir y grabar a personas y objetos en movimientos se requiere un software especializado [25].

1.3.8 Placas SBC (Single Board Computer)

Jetson Nano

La Jetson Nano es una minicomputadora muy poderosa destinada al desarrollo de Inteligencia Artificial (IA) y a la robótica, siendo capaz de efectuar diferentes redes neuronales en paralelo para realizar la clasificación de imágenes o la detección de objetos. Al contar con una GPU ejecuta aplicaciones con gran velocidad siendo ideal para proyectos de Computer Vision tales como [26]:

- NVIDIA JetBot
- Estimación de pose humana en tiempo real.
- Reconocimiento de matrícula en tiempo real.
- Identificador de recipiente de comida.

- Camara detectora de mascarillas.
- YolactEdge

Raspberry Pi

La Raspberry Pi es una minicomputadora económica con sistema operativo basado en Linux, posee herramientas propias de fácil uso y muy intuitivas, se puede conectar a una Televisión o a un monitor de computadora además que utiliza un teclado y un mouse estándar. Este dispositivo al ser intuitivo permite que las personas de diferentes edades exploren la computación y programen en lenguajes como Python y Scratch. De igual modo puede navegar por internet como una PC de escritorio, reproducir videos HD incluso crear hojas de cálculo. [27]

Beelink SER 5

Beelink SER 5 es una minicomputadora con un procesador móvil similar al de las portátiles, este dispositivo no se calienta y además cuenta con una potente informática interior, el procesador cuenta con excelentes gráficos integrados AMD Vega 6 con 2GB de VRAM, juegos de gama alta no se pueden jugar debido a que no posee una tarjeta gráfica dedicada [28].

1.3.9 Software

Es un conjunto de instrucciones, datos o programas utilizados para operar computadoras y efectuar tareas específicas; también se refiere a aplicaciones, scripts y programas que se ejecutan en un dispositivo. Los tipos más comunes de software son [29]:

- Software de aplicación.
- Software del sistema.
- Software de controlador.
- Middleware.
- Software de programación.

1.3.10 Protocolo MQTT

Este protocolo de transporte de telemetría de cola de mensajes o MQTT tiene el propósito de otorgar una comunicación M2M en la infraestructura IoT puesto que necesita limitado ancho de banda para funcionar en dispositivos en poca RAM y CPU. Es perfecto para sensores y microprocesadores. Para comunicarse los receptores y emisores debe ser necesario que pertenezcan al mismo topic o temas [30].

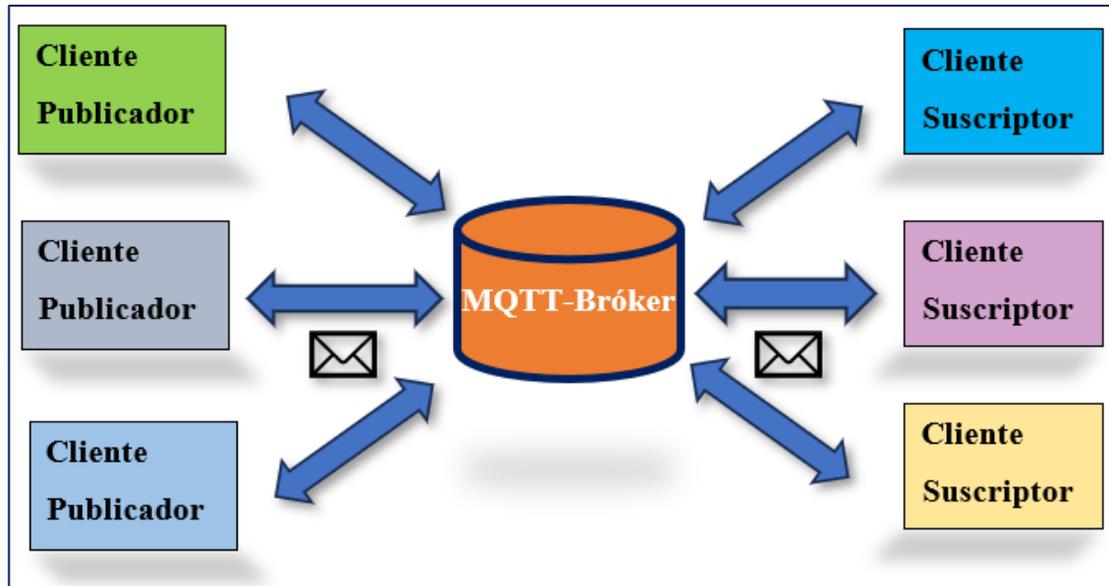


Figura 10. Funcionamiento del protocolo MQTT

Elaborado por: El investigador basado en [31].

1.3.11 Plataformas IoT

La plataforma IoT es un sistema que captura y realiza el tratamiento de datos, mismos que utilizan sistemas embebidos y protocolos de comunicación IoT como lo es MQTT, otorgan realizar sistemas de monitorización en paneles (Dashboard) que realizan funciones de análisis y control de sucesos para gestionar dispositivos inteligentes conectados y para de esta manera supervisar, almacenar y procesar datos desde la nube con la utilización de interfaces [32].

1.3.12 Aplicación web

Una aplicación web es un programa informático que se ejecuta en un servidor web y se accede a través de un navegador web. A diferencia de las aplicaciones de escritorio que se instalan en un ordenador o dispositivo móvil, las aplicaciones web son

accesibles desde cualquier dispositivo que tenga acceso a Internet y un navegador web [33].

Las aplicaciones web pueden ser desarrolladas utilizando diversos lenguajes de programación como: Python, PHP, Java, C#, entre otros, o mediante la selección de un framework tales como Angular, React, Django, Laravel, Spring MVC, etc. HTML se utiliza para estructurar el contenido que se presenta al usuario, CSS para el estilo y JavaScript añade características interactivas. Además, pueden ser alojadas en servidores web privados o en la nube, lo que permite una mayor escalabilidad y flexibilidad [33].

1.3.13 Funcionamiento de una aplicación web

La aplicación web se ejecuta en un servidor web y se accede a través de un navegador web. Cuando un usuario interactúa con la aplicación, establece una conexión TCP al servidor web, envía un comando HTTP (Protocolo de Transferencia de Hipertextos) de petición al servidor, el cual procesa la solicitud y devuelve una respuesta al navegador, que la muestra al usuario. Durante este proceso, la aplicación web puede interactuar con bases de datos, otros servicios en línea o sistemas de autenticación para garantizar la seguridad y la privacidad de los datos del usuario. En general, el funcionamiento de una aplicación web es un proceso continuo de interacción entre el usuario, el navegador web y el servidor web que aloja la aplicación [33].

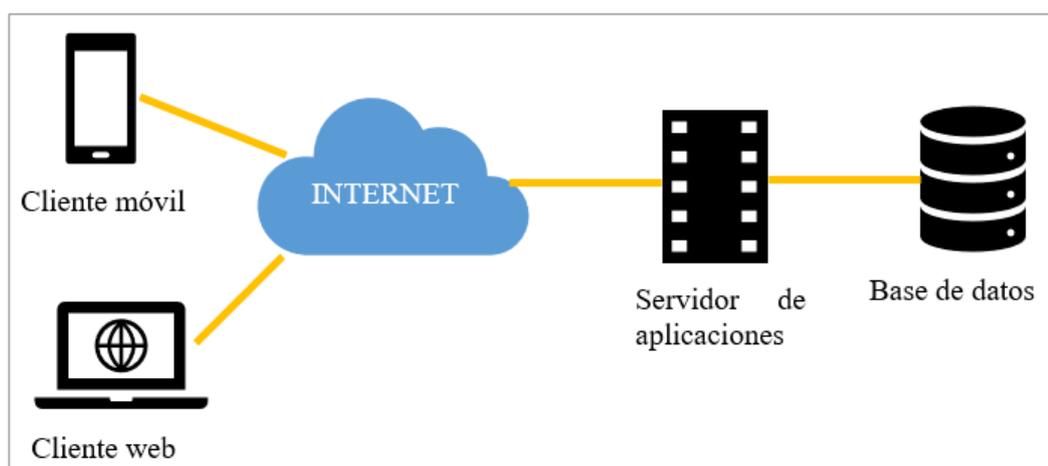


Figura 11. Esquema de funcionamiento de una aplicación web

Elaborado por: El investigador basado en [34].

1.4 Objetivos

1.4.1 Objetivo General

Implementar un sistema de rehabilitación física para pacientes con lesiones de tobillo empleando visión artificial para la Unidad de Rehabilitación y Fisioterapia Gabo's

1.4.2 Objetivos Específicos

- Identificar los métodos empleados para la rehabilitación de lesiones de tobillo.
- Desarrollar un entorno de realidad virtual que contribuya para la rehabilitación de pacientes con lesiones de tobillo.
- Diseñar la interfaz para la visualización y monitoreo del avance en la recuperación del paciente.

CAPÍTULO II

METODOLOGÍA

2.1 Materiales

Para desarrollar el presente proyecto se realiza un análisis de los distintos hardware y software necesarios para desarrollar el entorno virtual y el proceso de visión artificial.

Dispositivo de captación de video

Camara web.- dispositivo que capta la intensidad y color de la luz para transformarlos en datos de pixeles por medio de componentes ópticos. Este elemento siempre es conectado a una computadora para recoger imágenes en tiempo real, mismas que pueden ser procesadas o almacenadas [35].

Kinect. – es un dispositivo conocido como Kinect for Xbox 360 mismo que fue creado por Microsoft y puesto en el mercado en el 2010, este dispositivo permite a los usuarios interactuar con los juegos sin la necesidad de tener un controlador físico. Consigue hasta 30 cuadros por segundo. El campo de visión de este comprende una distancia de 4.5 metros por lo cual es recomendable ubicarse de manera óptima sin interferencias de por medio a una distancia de 2 metros. En cuanto a grados comprende 57 grados en sentido horizontal y 43 en sentido vertical [36].

Kinect v2.- es la revolución de la version 1 mejorada en varios aspectos la cual permite conseguir una mayor precisión para ver la oscuridad similar al infrarrojo, calcula la profundidad dependiendo del tiempo de luz emitida esta tecnología se conoce como LightCoding que realiza una reconstrucción tridimensional de la escena. La distancia del campo de visión es la misma que la versión anterior pero esta versión mejora en cuanto a grados de visión siendo 70 grados en sentido horizontal y 60 grados en sentido vertical [36].

De acuerdo a las descripciones de cada uno de los dispositivos se ha realizado una comparativa. Las características más importantes de cada dispositivo se presentan en la siguiente Tabla 4.

Tabla 4. Cuadro comparativo de los dispositivos de captación de video

Parámetro	Cámara web	Kinect	Kinect v2
Imagen			
Sensores	No	Sensor de profundidad Structured Light	Sensor de profundidad Time of Flight
Resolución	90 fps 4096 px x 2160 px	640×480 @30 fps 1280×960 @12 fps Se denomina ColorStream	1920×1080 @30 fps High Definition Se denomina ColorSource
Campo de visión	90°	57° en horizontal y 43° en vertical	70° en horizontal y 60° en vertical
Sistema de seguimiento	No	Capacidad para detectar 6 personas, pero solo dos se pueden rastrear completamente	Acelerómetro de 3 ejes. Capacidad para detectar 6 cuerpos, todos completamente rastreados.
Sistema de audio	Si	Matriz por fases lineal de 4 micrófonos	Matriz por fases lineal de 4 micrófonos
USB	2.0 y 3.0	2.0	3.0
Sistema operativo	Win 7/8/10, Mac OSX	Win7 o superior	Win 8.1 o superior Sólo 64 bits
Precio	\$30	\$81.97	\$110

Elaborado por: El investigador basado en [35], [36] [37], [38]

De acuerdo con las especificaciones Kinect v2 es seleccionado ya que facilita la integración con el motor de juego, permitió captar mejor el movimiento del paciente puesto que posee un sensor de profundidad y un mayor campo de visión.

Para utilizar el Kinect v2 es necesario instalar el software para Windows SDK 2.0.

Kinect para Windows SDK 2.0

El Kinect para Windows SDK 2.0 es un conjunto de herramientas de software diseñado por Microsoft que posibilita a los desarrolladores la creación de aplicaciones

destinadas a los dispositivos Kinect. Este dispositivo, inicialmente concebido para la consola de videojuegos Xbox 360, consiste en una combinación de sensores de movimiento y fue posteriormente lanzado como un producto independiente para su uso en entornos Windows [9]. Instalación del SDK2.0 revisar el ANEXO 2



Figura 12. Detección del esqueleto humano en 2D mediante Kinect Studio

Elaborado por: El investigador

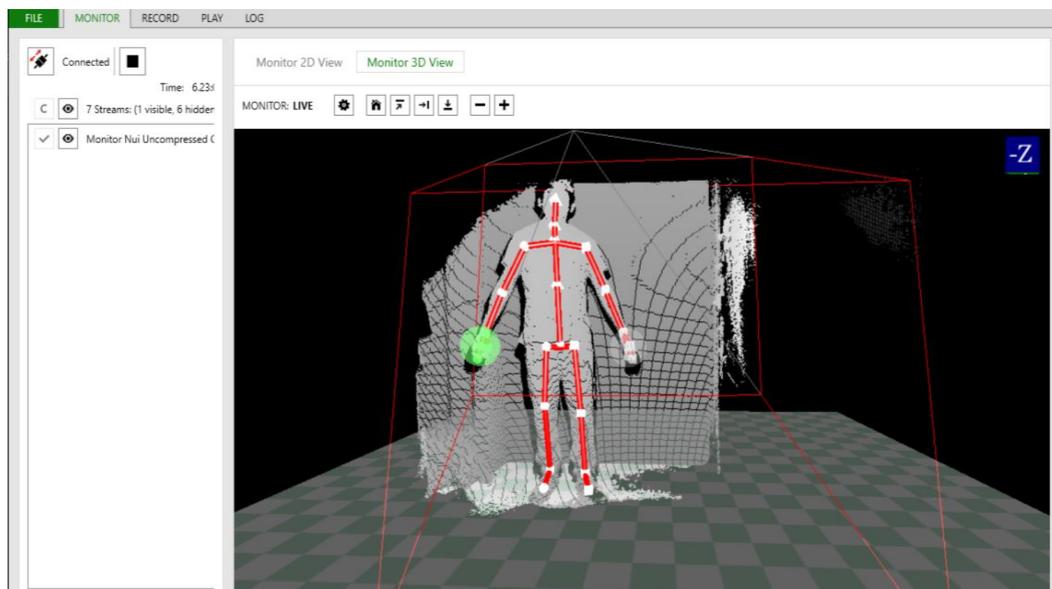


Figura 13. Detección del esqueleto humano en 3D mediante Kinect Studio

Elaborado por: El investigador

El dispositivo Kinect capta “joints” con la cámara RGB y a partir de ellos traza un enlace entre los joints para que simule el cuerpo humano. Las propiedades de cada joint son 11 en las que intervienen color (x, y), profundidad (x, y), cámara (x,y,z) y orientación (x,y,z,w). Lo hace posible debido al código integrado basado en reconocimiento de patrones e imágenes [9].

Selección del motor para entornos virtuales

Tabla 5. Cuadro comparativo de los motores para entornos virtuales

Parámetro	Unity	Unreal Engine	CryEngine
Aprendizaje	Sencilla para principiantes	Complicada	Complicada
Lenguaje de Script	C#, UnityScript	C++, Blueprint	C++, Lua, Schematyc
Comunidad de desarrolladores	Extensa	Inferior a Unity	Limitada
Calidad Grafica	Inferior a Unreal	Buena calidad grafica	Similar a Unreal
Colaboración con otras personas	En el mismo proyecto	En el mismo proyecto	En el mismo proyecto
Licencia	Gratuita y de pago	Gratuita y de pago	Gratuita y de pago
Soporte para RV/RA	Extenso soporte para RV/RA	Extenso soporte para RV/RA	Soporte para RV/RA reducido

Elaborado por: El investigador basado en [39], [40]

Se considera la necesidad de un entorno virtual que sea creado desde 0 con una amplia documentación y que sea intuitiva. Unity ofrece herramientas propicias en escenarios 3D, muy útil en cuanto al aprendizaje;

Por el contrario, Unreal Engine al ser utilizadas por grandes empresas y más profesionales por la calidad de graficas que esta proporciona, tiene una curva de aprendizaje más complicada y no cuenta con mucha documentación. CryEngine es menos accesible para principiantes en comparación con Unity y Unreal Engine pero proporciona soporte y orientación a los desarrolladores de juegos, al poseer una impresionante calidad de grafica utiliza muchos recursos de la PC para poder renderizar el entorno.

Se ha elegido para la creación del entorno virtual al motor Unity debido a que su aprendizaje es amigable y su entorno es intuitivo además de contar con una amplia comunidad de desarrolladores y por consiguiente acceder a documentación importante para llevar a cabo este proyecto.

Creación del avatar

Para crear el avatar se hizo uso de una plataforma web Ready Player Me misma que permite a los usuarios crear y diseñar avatares personalizados, mediante la captura de una foto o realizar los diseños por defecto que esta trae. En la Figura 14 se observa el avatar creado .Se puede evidenciar la creación del avatar paso a paso en el ANEXO 3.

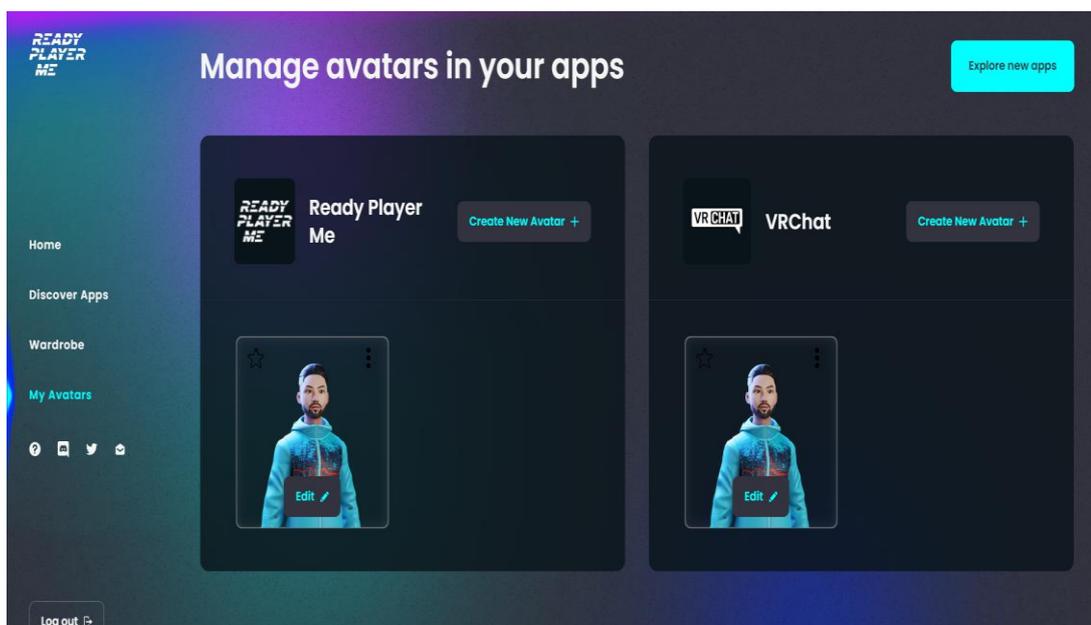


Figura 14. Creación del avatar

Elaborado por: El investigador

Periféricos de entrada y salida

- Mouse
- Teclado
- Monitor

Selección de librería de visión artificial

Las características más importantes de cada librería de visión artificial se presentan en la siguiente Tabla 6.

Tabla 6. Comparativa de librerías de Visión artificial

Parámetro	Tensorflow	PyTorch	OpenCV
Tipo de biblioteca	Redes neuronales, aprendizaje Automático	Aprendizaje Automático	Visión Artificial y Procesamiento de Imágenes
Lenguaje soportado	Python, C++, JavaScript, Swift, Java	Python, C++, JavaScript	Python, C++, C#, Java
Soporte de GPU	Si	Si	Si
Compatibilidad S.O	Windows, Linux, macOS, Android, iOS	Windows, Linux, macOS, Android, iOS	Windows, Linux, macOS, Android, iOS
Documentación	Extensa y detallada	Extensa y detallada	Completa
Rendimiento	Alto rendimiento en tareas de IA	Aumento de la eficiencia computacional	Eficaz y optimo
Aplicaciones	Reconocimientos de imágenes, análisis de sentimiento, diagnóstico medico	Procesamiento y clasificación de imágenes, redes antagónicas generativas	Procesamiento de imágenes, visión por computadora, robótica

Elaborado por: El investigador basado en [41], [42], [43]

La elección de la librería de visión artificial es OpenCV debido a que es adecuada para procesar imágenes y brinda diversos algoritmos y funciones para llevar a cabo el proyecto. Además, de ser eficiente a la hora de captar, procesar, distinguir y estimar las articulaciones del paciente.

MediaPipe: es una solución basada en aprendizaje automático ML (Machine Learning) que permite hacer un seguimiento preciso de la postura corporal, constan de 33 puntos como se muestra en la Figura 15, que se extraen de un cuadro de video. El método implementado logra un rendimiento en tiempo real en una amplia gama de

dispositivos móviles, computacionales y entornos como Python, e incluso se puede utilizar en la web [35].

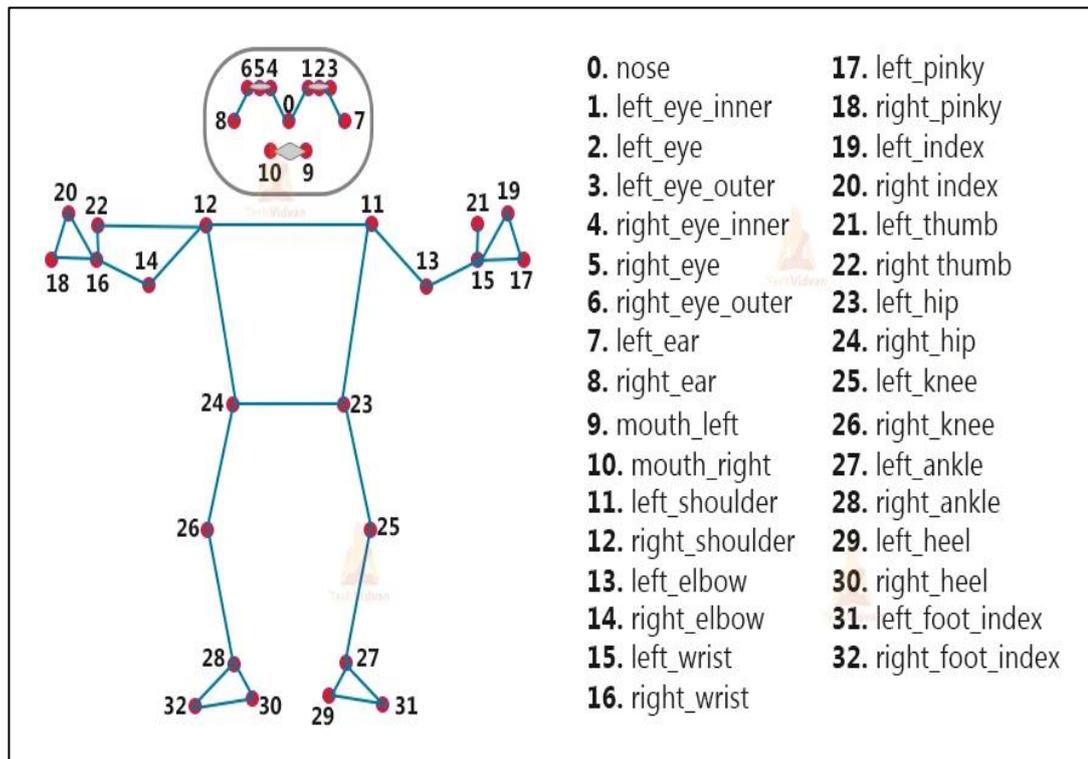


Figura 15. Puntos de referencia MediaPipe Pose [44]

Con la utilización de MediaPipe y OpenCV en el presente proyecto se puede conseguir un mayor rendimiento puesto que MediaPipe se configura para utilizar el modelo BlazePose el cual reconoce y rastrea las poses que realiza el paciente en la terapia física mientras que OpenCV se encarga de procesar las imágenes de video captadas por la cámara.

En consecuencia se pudo captar las articulaciones para ser procesadas y analizadas para su posterior cálculo del ángulo de movilidad del tobillo.

Selección de la minicomputadora

Según lo explicado en la fundamentación teórica acerca de las minicomputadoras de las cuales se detallan 3 SBC. Las características más importantes de cada dispositivo se pueden evidenciar en la siguiente Tabla 7.

Tabla 7. Cuadro comparativo de las microcomputadoras

Parámetros	Jetson Nano	Raspberry Pi 4	Beelink SER5
Imagen			
Fabricante	NVIDIA	Raspberry PI	Beelink
CPU	Quad-Core ARM A57 @ 1.43 GHz	Broadcom BCM2711 Cortex-A72	AMD Ryzen 5 5500U hasta 4.0GHz 6C/12T
GPU	128 Core Maxwell	Broadcom VideoCore VI @ 500MHz	Radeon Graphics 6 1600 MHz
Conectividad	Gigabit Ethernet, Bluetooth	Gigabit Ethernet, Bluetooth 5.0, Wi-Fi 802.11ac	Gigabit Ethernet, WiFi 6 (802.11ax), Bluetooth 5.2
Puertos USB	4 x USB 3.0 1 x USB 2.0	2 x USB 3.0 2 x USB 2.0	2 x USB 3.2 Gen2, 2 x USB 2.0
Otros puertos	Puerto HDMI	Audio Micro HDMI Fuente de alimentación USB-C	Tipo C 2 x HDMI (4K HD), Conector para audio
Dimensiones	2.72 x 1.77 x 1.77 pulgadas	7.87 x 4.88 x 3.31 pulgadas	8,94 x 5,39 x 2,32 pulgadas
Memoria RAM	4GB 64-bit LPDDR4 25.6 GB/s	4GB LPDDR4-3200 SDRAM	16 SO-DIMM DDR4
Sistema operativo	Linux Os	GNU/Linux	W-11 Pro
Precio	\$180	\$236.63	\$388.34

Elaborado por: El investigador basado en [45], [46], [47]

La Tabla 7 presenta las microcomputadoras con sus respectivos parámetros de acuerdo con las necesidades del proyecto. De acuerdo con las características que resaltan se ha

optado por la microcomputadora Jetson Nano debido a que tiene una CPU y GPU óptimas puesto que se emplean para aplicaciones de inteligencia artificial.

La cámara para usar en esta parte del proyecto es la IMX219-77 misma que presenta compatibilidad con la NVIDIA Jetson Nano aprovechando el puerto CSI de la minicomputadora, mide 25 mm x 24 mm x 5.3mm. También dotada de un sensor Sony IMX219 de 8MP para la toma de videos de alta definición.

La empresa NVIDIA proporciona un Jetpack el cual tiene incluido un entorno de desarrollo completo de inteligencia artificial acelerado por hardware. La instalación del Jetpack se puede observar en el ANEXO 4.

Protocolo de comunicación

En la Tabla 8 se observa las características más relevantes de los protocolos de comunicación empleados en el internet de las cosas, son analizados para determinar cuál es el que mejor se acopla a nuestro proyecto.

El protocolo MQTT conocido por su arquitectura cliente//bróker y el modelo de publicar/suscribir permite la comunicación entre múltiples clientes a través de un punto central, convirtiéndole en un protocolo apto para el intercambio de datos en tiempo real, similares a un bus de comunicación. Por otro lado, CoAP se centra en la transferencia de información entre el cliente y el servidor con una comunicación de uno a uno. [48]

HTTP en cambio resalta en el coste de establecer la conexión en comparación con MQTT sin embargo no posee muchas características importantes empleadas en IoT como son las colas, la QoS (Calidad de servicio), mensajes retenidos, entre otros.

En cuestión de mensajes MQTT requiere de todos los clientes conozcan el formato para establecer la comunicación, en cambio CoAP posibilita que los dispositivos capturen una manera de intercambiar datos, este aspecto es conocido como negociación y descubrimiento. [48]

Tabla 8. Protocolos de comunicación

Parámetro	CoAP	HTTP	MQTT
Arquitectura	Cliente/bróker o Cliente/servidor	Cliente/servidor	Cliente/bróker
Modelo	Solicitud/respuesta o Publicar/suscribirse	Solicitud/respuesta	publish/suscribe
QoS	Mensaje confirmable, mensaje no confirmable, asentimiento, reinicio	Limitado (mediante el protocolo de transporte - TCP)	QoS 0: máximo una vez. QoS 1: Por lo menos una vez. QoS 2: Exactamente una vez
Método	GET, PUT, POST, DELETE	Obtener, publicar, encabezar, poner, parche opciones, conectar, eliminar	conectar, desconectar, publicar, suscribir, darse de baja, cerrar.
Normas	Eclipse Foundation, IETF	ETF, W3D	Eclipse Foundation, OASIS
Protocolo de Transporte	UDP, SCTP	TCP	TCP(MQTT-SN puede usar UDP)
Seguridad	NoSec, DTLS	TLS/SSL	TLS/SSL
Puertos predispuestos	5683(coap) 5684(coaps)	80/443(TLS/SSL)	1883/8883(TLS/SSL)
Formato de codificación	Binario	Binario	Binario
Soporte	Cisco, Large Web Community Support	Global Web Protocol Standard	Facebook, Cisco, IBM, Red Hat, Software AG, Tibeo, ITSO, M2Mi, AWS, InduSoft, Fiorano

Elaborado por: El investigador basado en [48], [49]

En consecuencia, MQTT es la elección perfecta para realizar el presente proyecto debido a que se ajusta a las necesidades de transmisión de grandes volúmenes de datos en tiempo real. A través del uso de un bróker, los clientes MQTT pueden establecer conexiones TCP de larga duración, facilitando la comunicación con dispositivos que se encuentran detrás de un NAT.

Recolección y almacenamiento de datos

Tabla 9. Comparativa de las herramientas de recolección

Parámetro	Telegraf	Node-RED	Prometheus
Tipo	Agente recolector de métricas	Capacidad de recolección de métricas	Monitoreo y recolección de métricas
Lenguaje de programación	Go	JavaScript	Go
Modelo de almacenamiento de datos	Remite datos a diversos destinos	Se integra con sistemas externos	Series temporales
Escalabilidad	Si	Si	Si
Arquitectura de recolección	Recopila datos y envía a diferentes destinos	Capacidad de recopilar y enviar métricas	Cliente-servidor
Consultas y lenguaje de consulta	Permite transferir datos a bases de datos que admiten SQL	Manipula y realiza cambios empleando nodos personalizados.	PromQL (Prometheus Query Language)
Compatibilidad con InfluxDB	Si	Si	Si
S.O Compatible	Multiplataforma	Multiplataforma	Multiplataforma
Configuración	Moderada	Fácil	Moderada
Disponibilidad	No	No	Si

Elaborado por: El investigador basado en [50], [51], [52]

Node red es seleccionado debido a que es compatible con influxdb pues tiene la capacidad de recolectar datos del bróker y enviar métricas hacia una base de datos. Además que se puede guardar los datos del paciente realizando una interfaz de registro para posterior generar reportes de avance en la rehabilitación.

Tabla 10. Comparativa de las herramientas de almacenamiento

Parámetro	Influxdb	MySQL	MongoDB
Descripción	DBMS para almacenar series temporales, eventos y métricas	DBMS relacional de código abierto ampliamente utilizado	Guarda los datos en documentos
Lenguaje de programación	Go	C, C++	C++
Modelo de base de datos principal	Serie de tiempo DBMS	DBMS relacional	Almacén de documentos
Capacidades en memoria	Si	Si	Si
Compatibilidad con NodeRed	Si	Si	Si
S.O Compatible	Linux	FreeBSD Linux OS X Solaris Windows	Windows, GNU/Linux, OS X, Solaris
Compatibilidad con Grafana	Si	No	Si
Reporte de métricas	Si	No	No (Existen alternativas)
Integración con IoT	Alta	Baja	Alta

Elaborado por: El investigador basado en [48], [53]

El sistema de rehabilitación necesita ordenar y estructurar la información por tiempo, recopilar datos en función de intervalos continuos registrándose así en orden del tiempo. Por lo cual InfluxDB es ideal a la hora de almacenar amplios volúmenes de datos debido a que es una base de datos de series temporales.

Además, se considera la necesidad de guardar datos del paciente que va a realizar la rehabilitación por tal motivo se estima una base de datos relacional MySQL es la opción óptima para albergar información del paciente y generar reportes, puesto que dedica el rendimiento basado en varias columnas es más complejo y por tal motivo lento. MongoDB almacena documentos llamados sistemas de bases de datos

orientados a documentos, organizan datos sin esquemas; es usado también para reporte de métricas, pero con diferentes alternativas haciéndolo más complicado que InfluxDB.

Se ha seleccionado InfluxDB para implementar el proyecto ya que el sistema proporciona gran cantidad de datos (ángulos de los tobillos) y se requiere tener la información más accesible y analizarlos en tiempo real. Por consiguiente, se usa NodeRed con InfluxDB en la etapa de recolección y almacenamiento de datos respectivamente.

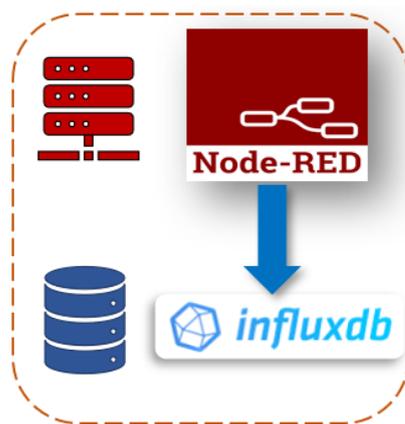


Figura 16. Recolección y almacenamiento

Elaborado por: El investigador

Visualización de datos

Tabla 11. Comparativa de plataformas graficas

Parámetro	Grafana	Kibana	Power BI
Sistema operativo compatible	Linux	Linux	Windows, macOS
Métricas MQTT	Si	Si	No
Costo	Gratis	Gratis	Diferentes planes
Integración con InfluxDB	Si	No	No
Variedad de presentación de datos	Si	Si	Si
Código base	Si	Si	No

Elaborado por: El investigador basado en [48]

En primera instancia se descarta a Power BI ya que no cuenta no métricas MQTT y mucho menos se integran a InfluxDB. Las herramientas de visualización Grafana y Kibana cumplen características semejantes en código base y la variedad de presentar datos, métricas MQTT, los dos son compatibles con Linux además de ser gratuitas. A pesar de ello, Grafana presenta la oportunidad de integrarse con InfluxDB con una elevada compatibilidad con aplicaciones con el mismo código base puesto que son programadas en lenguaje Go.

Por tal motivo se ha elegido a Grafana al ser una plataforma grafica que posibilita el monitoreo de una manera sencilla e intuitiva para el administrador y los usuarios.



Figura 17. Plataforma para visualización

Elaborado por: El investigador basado en [48], [54]

2.2 Métodos

2.2.1 Modalidad de investigación

El presente proyecto se sustentó en una Investigación Aplicada, porque se emplearon los conocimientos adquiridos durante la carrera y se realizaron investigaciones que sustentaron la realización del proyecto; estas investigaciones se lo obtuvieron tanto en las bases de datos como en las bibliotecas virtuales, los mismos que fueron proporcionados por la Universidad.

Investigación fue Bibliográfica, porque el proyecto de titulación se fundamentó con consultas en libros, revistas, artículos y bases de datos científicas de varios repositorios de universidades. Puesto que sirvió de apoyo para la ejecución del trabajo como antecedentes de investigación.

Además, se realizó una Investigación de Campo, porque el desarrollo del proyecto implicó la participación del paciente y, mediante el sistema de rehabilitación se logró monitorear las terapias físicas del paciente.

2.2.2 Recolección de información

Para lograr la recolección de información se emplearon libros, revistas, fuentes online y proyectos desarrollados, asimismo datos de incidencia y prevalencia de las lesiones, por lo que se tomó en cuenta fuentes de información confiables. Así también se obtiene información de especialistas que permitieron el desarrollo del proyecto.

2.2.3 Procesamiento y análisis de datos

Realizada la recolección de información necesaria para el proyecto de investigación, se procedió a realizar los siguientes pasos:

- Revisión de toda la información recolectada
- Estudio de las propuestas de solución planteadas para la rehabilitación de pacientes.
- Selección relevante de información que contribuya con ejecución de la propuesta de solución.
- Planteamiento de la propuesta de solución.
- Control y verificación de los datos obtenidos mediante el testeado del sistema y el dispositivo.

CAPÍTULO III

RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de los resultados

El sistema de rehabilitación física para pacientes con lesiones de tobillo empleando visión artificial, contribuye a la recuperación del paciente que presenta traumatismos en la articulación, con la utilización del entorno virtual el usuario tiene una terapia agradable e interactiva debido a que la cámara capta los movimientos ejercidos por el paciente y los replica en un avatar que es visualizado en una pantalla. Además, el sistema obtiene los ángulos del tobillo por medio de una cámara de alta definición conectada a la minicomputadora estos datos son enviados a una plataforma de visualización para ser analizados y por ende tomar una decisión en que ejercicios se debe aplicar para mejorar la movilidad. El sistema presenta un error absoluto promedio de 4,55 con un nivel de confiabilidad de 95,28% en el cálculo de los ángulos.

3.2 Desarrollo de la propuesta

Para cumplir los objetivos planteados del sistema de rehabilitación física mediante visión artificial para pacientes con lesiones de tobillo se creó un entorno virtual mismo que contribuye en la rehabilitación, posteriormente se implementó los algoritmos de visión artificial en una minicomputadora diseñada precisamente para realizar estos trabajos. Los datos extraídos son ángulos de movilidad del tobillo como son flexión plantar y flexión dorsal, para obtener información del avance en la recuperación del paciente que padece un traumatismo en el tobillo.

3.2.1 Requerimientos del sistema o especificaciones necesarios para el desarrollo del sistema

Rango de movimiento articular del tobillo

La movilidad del tobillo se efectúa principalmente en el plano sagital; el rango de flexoextensión va desde los 43 a 63° sin embargo se necesita solo 30° de este rango

(20° flexión plantar y 10° flexión dorsal) para tener una marcha estable. Este rango de movilidad ocasiona alteraciones de la marcha cuando se padece patologías que limitan la movilidad [55].

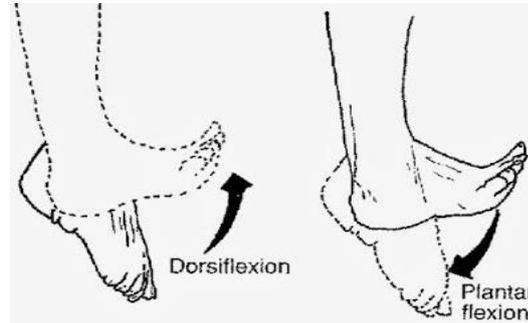


Figura 18. Movimiento articular del tobillo [56]

Escenario para las pruebas de funcionamiento

El escenario para la instalación del sistema de rehabilitación física para pacientes con lesiones de tobillo en la Unidad de Rehabilitación y Fisioterapia Gabo's se presenta en la Figura 19. La habitación debe ser amplia y con una buena iluminación para tener una mejor precisión al momento de realizar los análisis y que el sensor tome la captura del cuerpo. La arquitectura consta de un sensor Kinect que se coloca para tener un enfoque total del paciente, se tiene un monitor para observar los ejercicios que un avatar los realiza en función de lo que haga el paciente, también consta de una cámara adicional misma que captura los ángulos de movilidad para monitorear el avance de la rehabilitación del tobillo; en una mesa pequeña esta la Jetson nano con los periféricos necesarios.



Figura 19. Escenario de pruebas

Elaborado por: El investigador

3.2.2 Esquema general del sistema

El paciente realiza sus terapias frente al Kinect el cual capta los movimientos y los replica en un avatar, el sensor necesita de un adaptador para poder conectarlo a la PC al mismo que son conectados otros periféricos. En la Figura 20 se observa la conexión de los elementos, para el sensor Kinect se tiene un adaptador mismo que está conformado por una fuente de alimentación y un convertidor para conexión USB 3.0, también a la PC es conectado un periférico de salida monitor mismo que se establece la conexión por medio de un cable HDMI para interactuar con el entorno virtual.

El sistema también realiza la medición y procesamiento de los ángulos en las articulaciones del tobillo de los pacientes en la minicomputadora Jetson nano. La Jetson nano tiene una alimentación de 12V a 4A en corriente continua.

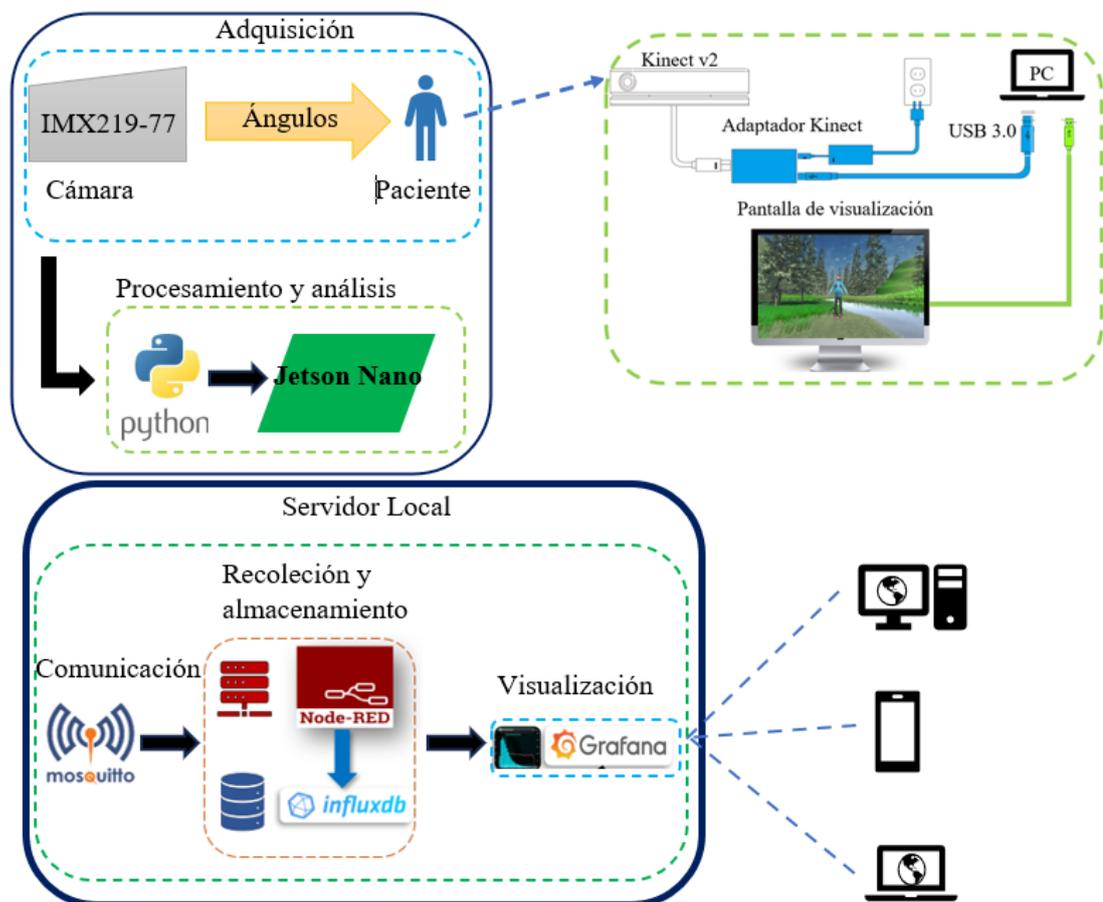


Figura 20. Esquema general del sistema

Elaborado por: El investigador

En la Figura 20 se observa un esquema general del sistema, mismo que está conformado por varios apartados a continuación se explica de forma breve.

Adquisición.- la cámara de alta definición se encarga de adquirir en video al paciente principalmente los tobillos.

Procesamiento y análisis.- empleando el lenguaje de programación Python se realiza el procesamiento y análisis de las articulaciones detectadas de esta manera se calcula lo ángulos de la articulación en estudio.

Comunicación.- todas las instancias están instaladas en la Jetson Nano; los datos obtenidos en la etapa anterior son enviados a través del protocolo MQTT.

Recolección y almacenamiento.- Node Red se encarga de recoger y transmitir las métricas desde el bróker MQTT a la base de datos influxdb.

Visualización.- Finalmente se visualizan los datos en Grafana que se configura para leer los datos InfluxDB, la información es presentada en una interfaz que contiene varios gráficos, se pueden realizar consultas para entender los datos almacenados.

3.2.3 Algoritmo de programación, adquisición y procesamiento

Obtención de los fotogramas

En el siguiente código se observa la variable captura que almacena los frames que son capturados con la cámara, se observa el valor 1 correspondiente a la cámara que se va a utilizar. También se observa el valor mínimo de confianza de detección y el valor mínimo de confianza de movimiento, estos valores son para una detección correcta y para ser rastreados con éxito los puntos de referencia respectivamente.

```
# Ubicamos la cámara conectada a la Jetson Nano

captura = cv2.VideoCapture(1)

# DETECTION = Valor mínimo de confianza de detección
# TRACKING = Valor mínimo de confianza de movimiento

with mp_pose.Pose(
    min_detection_confidence=0.5,min_tracking_confidence=0.5) as pose:

    while captura.isOpened():

        ret, frame = captura.read()
```

Puntos de referencia

Con el siguiente código desarrollado se obtiene los puntos de referencia de los dos tobillos mismos que son necesarios para el cálculo de los ángulos que conforman la extremidad inferior, en este caso los puntos de referencia que son importantes para dicho calculo son los de la rodilla, tobillo y la punta del pie. El ángulo por obtener es en relación a los otros dos puntos.

```
# Obtener las coordenadas
# Puntos para obtener el ángulo del tobillo

knee_left = [landmarks [mp_pose.PoseLandmark.LEFT_KNEE.value].x,
             landmarks [mp_pose.PoseLandmark.LEFT_KNEE.value].y]

ankle_left = [landmarks [mp_pose.PoseLandmark.LEFT_ANKLE.value].x,
              landmarks [mp_pose.PoseLandmark.LEFT_ANKLE.value].y]

foot_left= [landmarks [mp_pose.PoseLandmark.LEFT_FOOT_INDEX.value].x,
            landmarks [mp_pose.PoseLandmark.LEFT_FOOT_INDEX.value].y]
```

Cálculo del ángulo que forma el tobillo

Para calcular el ángulo del tobillo es necesario conocer los otros dos puntos de referencia, empleando cálculos matemáticos y trigonométricos se puede determinar el ángulo que forma el tobillo entre la rodilla y la punta del pie, es necesario establecer las coordenadas (X, Y) [57].

Mediante el producto escalar se logra determinar el ángulo formado por dos vectores, en la Figura 21 se observa dicho ángulo construido por los 3 puntos de referencia antes mencionados, el vector V_1 se forma por la unión de los puntos J1 y J0 mientras que la unión de J2 con J0 forma el vector V_2 [57].

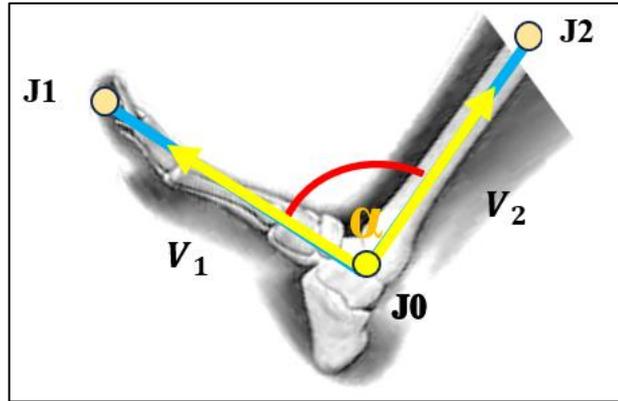


Figura 21. Angulo constituido por los 3 puntos de la articulación

Elaborado por: El investigador basado en [57]

Se obtienen los vectores (1) y (2) con las coordenadas rectangulares en la articulación central:

$$V_1 = (X_1 - X_0, Y_1 - Y_0) \quad (1)$$

$$V_2 = (X_2 - X_0, Y_2 - Y_0) \quad (2)$$

Se determina el producto escalar entre (1) y (2):

$$V_1 V_2 = (V_{1X} V_{2X} + V_{1Y} V_{2Y}) \quad (3)$$

Mediante la definición de producto escalar entre dos vectores $a \cdot b = |a||b|\cos \alpha$. Se iguala (3) aplicando el producto escalar entre (1) y (2) determinando así (4):

$$|V_1||V_2|\cos \alpha = V_{1X} V_{2X} + V_{1Y} V_{2Y} \quad (4)$$

Despejando $\cos \alpha$ en (4):

$$\cos \alpha = \frac{V_{1X}V_{2X} + V_{1Y}V_{2Y}}{|V_1||V_2|} \quad (5)$$

Para obtener la ecuación (6) despejo el ángulo α de (5):

$$\alpha = \cos^{-1} \left[\frac{V_{1X}V_{2X} + V_{1Y}V_{2Y}}{|V_1||V_2|} \right] \quad (6)$$

Sustituyendo las componentes (1) y (2) en (6):

$$\alpha = \cos^{-1} \left[\frac{(X_1 - X_0)(X_2 - X_0) + (Y_1 - Y_0)(Y_2 - Y_0)}{|(X_1 - X_0, Y_1 - Y_0)|| (X_2 - X_0, Y_2 - Y_0) |} \right] \quad (7)$$

Con la ecuación (7) se procede a crear el algoritmo que calcula el ángulo entre los 3 puntos de las articulaciones para ser graficadas en tiempo real. En el siguiente código se muestra la función necesaria para el cálculo, se fija una restricción para que el ángulo no sobrepase los 180° dado que el tobillo tiene un limitado rango de movimiento de aproximadamente 135 a 140° para proporcionar buena estabilidad y soporte del pie.

Con el ángulo obtenido se puede verificar si está en el rango de movilidad o si necesita realizar ejercicios destinados para aumentar dicho rango.

```

# Cálculos de los ángulos del tobillo

def calculate_angle(p, m, f):

    # primero (Rodilla)
    p = np.array(p)

    # medio (Tobillo)
    m = np.array(m)

    # final (Punta del pie)
    f = np.array(f)

    radians = np.arccos2(f[1] - m[1], f[0] - m[0]) - np.arccos2(p[1] -
                                                                m[1], p[0] - m[0])

    angulo = np.abs(radians * 180.0 / np.pi)

    #Restriccion

    if angulo > 180.0:
        angulo = 360 - angulo
    return angulo

```

3.2.4 Servidor

Para el sistema de rehabilitación se emplea un servidor local en la Jetson Nano, donde se instala NodeRed el cual se encarga de recibir datos e interpretarlos estos datos son recibidos del Broker Mosquitto mismo que se encuentra instalado en el mismo sistema el cual recibe los datos enviados desde Python. En la base de datos Influxdb son almacenados la información recibida desde Python, en cuanto a la interfaz gráfica Grafana va a recibir los datos para ser visualizados en diferentes esquemas para tener una buena apreciación y monitoreo.

Nos conectamos al servidor mediante el software PuTTY con la ip del servidor y el puerto 22, como se muestra en la Figura 22.

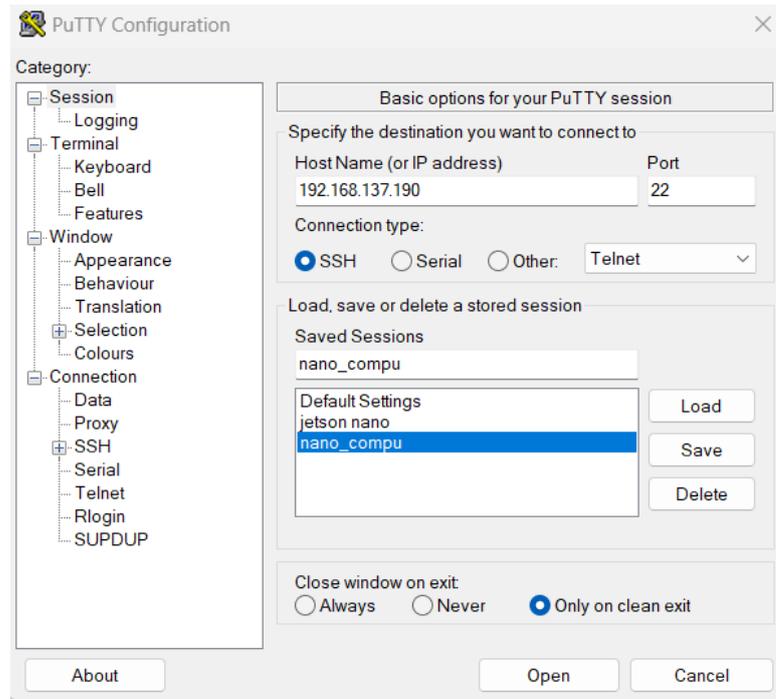


Figura 22. Conexión SSH mediante PuTTY

Elaborado por: El investigador

En la Figura 23 se observa los datos del sistema operativo como es versión de Ubuntu la cual es 20.04.6 LTS con una arquitectura arm64, esta información se consigue con el siguiente comando:

```
jetson@nano:~$ hostnamectl
```

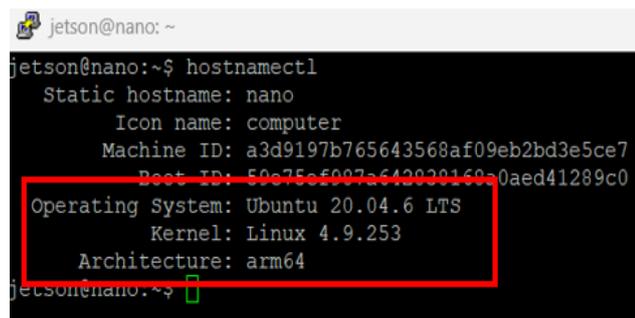


Figura 23. Versión del Sistema Operativo

Elaborado por: El investigador

3.2.5 Instalación de Mosquito Broker en Ubuntu

Antes de la respectiva instalación de Mosquitto primero se debe actualizar el software. Con los comandos que se muestran a continuación:

```
jetson@nano:~$ sudo apt update
```

```
jetson@nano:~$ sudo apt upgrade
```

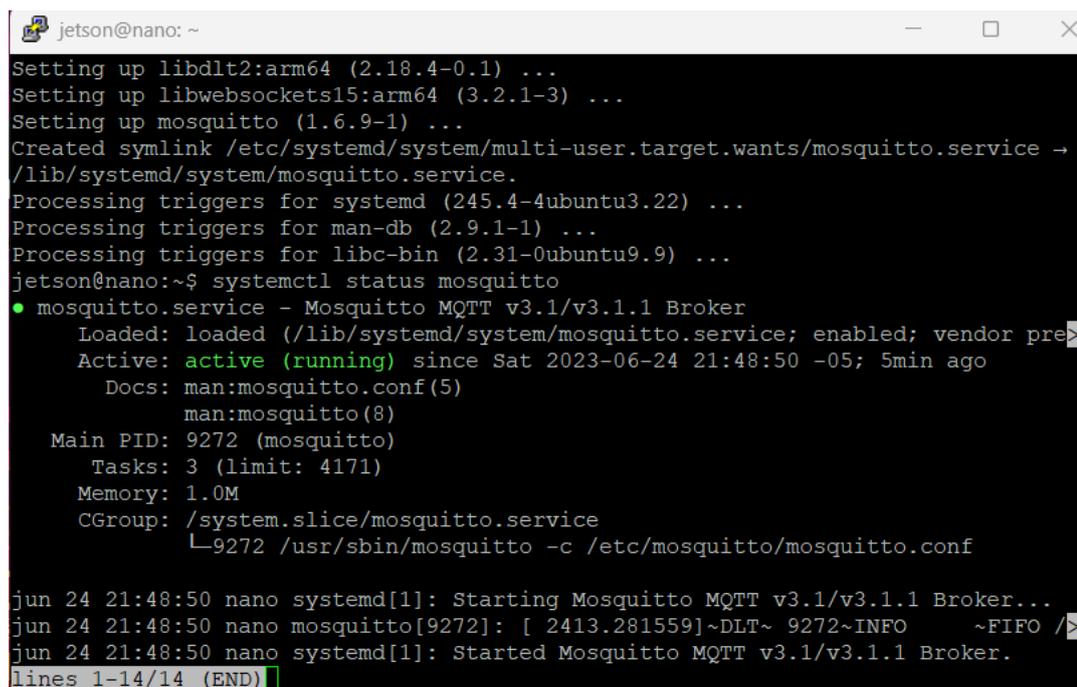
Para instalar el bróker se utiliza el comando siguiente:

```
jetson@nano:~$ sudo apt-get install mosquitto mosquitto-clients
```

Una vez instalado el bróker Mosquitto se le otorga el servicio que arranque con Linux cada vez que se enciende la Jetson Nano, posterior a eso se ubica el comando para ver el estado de Mosquito:

```
jetson@nano:~$ sudo systemctl enable mosquitto.service
```

```
jetson@nano:~$ systemctl status mosquitto
```



```
jetson@nano: ~
Setting up libdlt2:arm64 (2.18.4-0.1) ...
Setting up libwebsockets15:arm64 (3.2.1-3) ...
Setting up mosquitto (1.6.9-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/mosquitto.service →
/lib/systemd/system/mosquitto.service.
Processing triggers for systemd (245.4-4ubuntu3.22) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
jetson@nano:~$ systemctl status mosquitto
● mosquitto.service - Mosquitto MQTT v3.1/v3.1.1 Broker
   Loaded: loaded (/lib/systemd/system/mosquitto.service; enabled; vendor pre
   Active: active (running) since Sat 2023-06-24 21:48:50 -05; 5min ago
     Docs: man:mosquitto.conf(5)
           man:mosquitto(8)
   Main PID: 9272 (mosquitto)
     Tasks: 3 (limit: 4171)
    Memory: 1.0M
    CGroup: /system.slice/mosquitto.service
            └─9272 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

jun 24 21:48:50 nano systemd[1]: Starting Mosquitto MQTT v3.1/v3.1.1 Broker...
jun 24 21:48:50 nano mosquitto[9272]: [ 2413.281559]~DLT~ 9272~INFO ~FIFO /
jun 24 21:48:50 nano systemd[1]: Started Mosquitto MQTT v3.1/v3.1.1 Broker.
lines 1-14/14 (END)
```

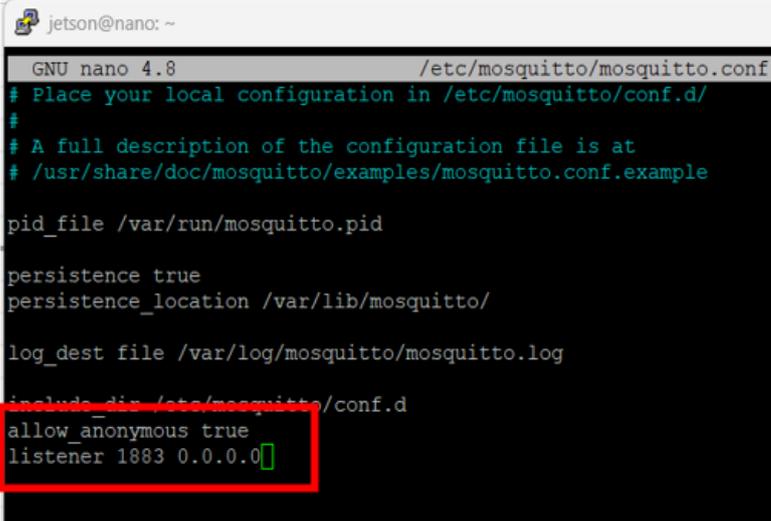
Figura 24. Estado activo de Mosquitto

Elaborado por: El investigador

Comprobado que se haya instalado correctamente el software Mosquitto se debe efectuar unas configuraciones en el archivo mosquitto.conf, para acceder a este archivo se logra con este comando:

```
jetson@nano:~$ sudo nano /etc/mosquitto/mosquitto.conf
```

Al ingresar al archivo se configura el puerto que tiene por defecto MQTT el puerto por defecto es el 1883, por donde el bróker va a recibir toda la información que está admitiendo el publicador desde el suscriptor en este caso Python. Adicional se realiza la configuración para otorgar que los usuarios anónimos se puedan conectar al bróker, estos cambios se pueden evidenciar en la Figura 25 marcada con un recuadro rojo.



```
jetson@nano: ~
GNU nano 4.8 /etc/mosquitto/mosquitto.conf
# Place your local configuration in /etc/mosquitto/conf.d/
#
# A full description of the configuration file is at
# /usr/share/doc/mosquitto/examples/mosquitto.conf.example

pid_file /var/run/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d
allow_anonymous true
listener 1883 0.0.0.0
```

Figura 25. Configuración del archivo mosquitto.conf

Elaborado por: el investigador

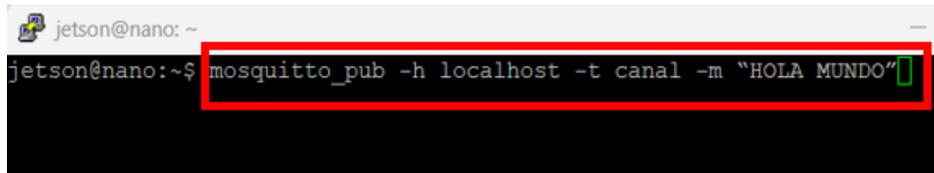
Si queremos comprobar que el protocolo este activo y funcionando se deben abrir dos ventanas en las cuales se van a colocar los comandos siguientes:

Para suscribirse a un tópico se utiliza el comando:

```
jetson@nano:~$ mosquitto_sub -h localhost -t canal
```

Para publicar los datos en el bróker se utiliza el comando siguiente:

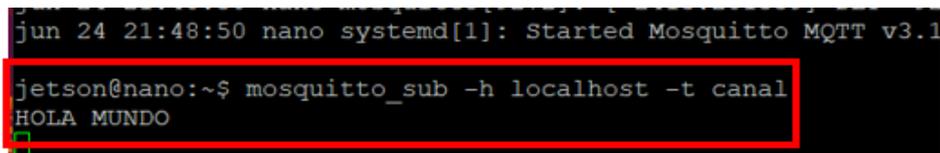
```
jetson@nano:~$ mosquitto_pub -h localhost -t canal -m "HOLA MUNDO"
```



```
jetson@nano: ~
jetson@nano:~$ mosquitto_pub -h localhost -t canal -m "HOLA MUNDO"
```

Figura 26. Publicador

Elaborado por: el investigador



```
jun 24 21:48:50 nano systemd[1]: Started Mosquitto MQTT v3.1
jetson@nano:~$ mosquitto_sub -h localhost -t canal
HOLA MUNDO
```

Figura 27. Suscriptor

Elaborado por: El investigador

3.2.6 Instalación de Node Red

En primera instancia se debe actualizar el software. Con los comandos que se muestran a continuación empleando la consola PuTTY:

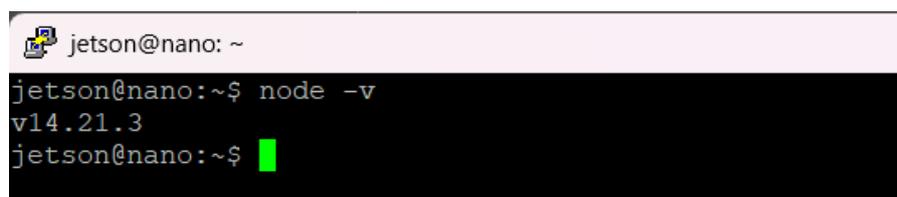
```
jetson@nano:~$ sudo apt update
```

```
jetson@nano:~$ sudo apt upgrade
```

Para que Node-Red se ejecute es necesario instalar Node.js

```
jetson@nano:~$ sudo apt install nodejs
```

```
jetson@nano:~$ node -v
```



```
jetson@nano: ~
jetson@nano:~$ node -v
v14.21.3
jetson@nano:~$
```

Figura 28. Versión de Node.js

Elaborado por: El investigador

Se verifica que este instalado Node.js y se procede con la instalación de Node-Red empleando el comando proporcionado a continuación:

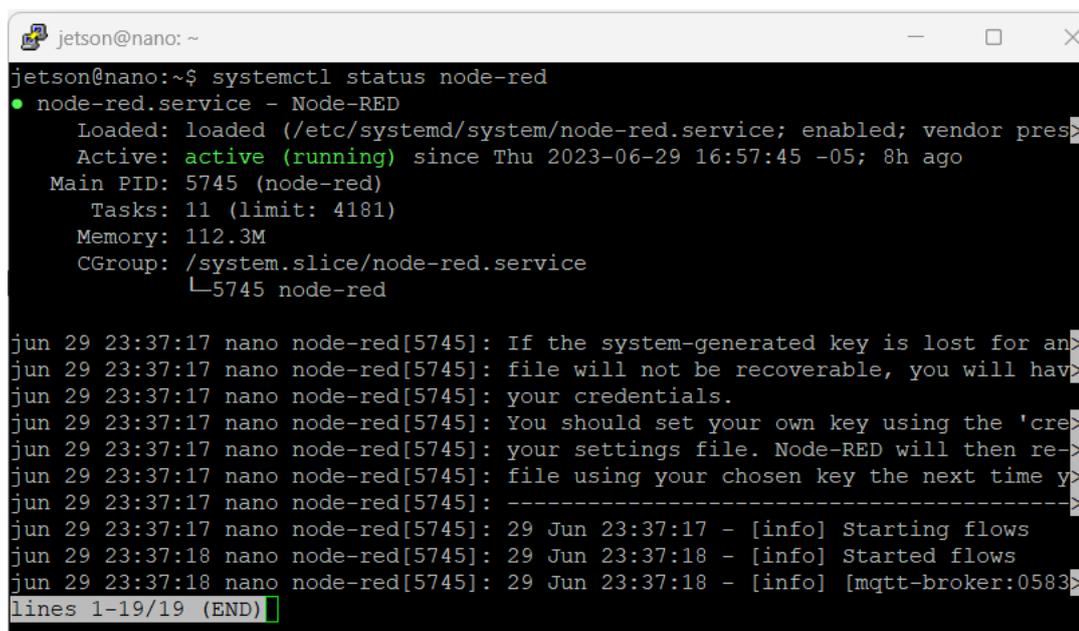
```
jetson@nano:~$ sudo npm install -g --unsafe-perm node-red
```

Realizada la instalación debemos iniciar el servicio:

```
jetson@nano:~$ sudo systemctl start node-red
```

Para verificar que el servicio de Node-Red este activo ingresamos el siguiente comando:

```
jetson@nano:~$ sudo systemctl status node-red
```



```
jetson@nano:~$ sudo systemctl status node-red
● node-red.service - Node-RED
   Loaded: loaded (/etc/systemd/system/node-red.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-06-29 16:57:45 -05; 8h ago
     Main PID: 5745 (node-red)
        Tasks: 11 (limit: 4181)
       Memory: 112.3M
      CGroup: /system.slice/node-red.service
             └─5745 node-red

jun 29 23:37:17 nano node-red[5745]: If the system-generated key is lost for an
jun 29 23:37:17 nano node-red[5745]: file will not be recoverable, you will hav
jun 29 23:37:17 nano node-red[5745]: your credentials.
jun 29 23:37:17 nano node-red[5745]: You should set your own key using the 'cre
jun 29 23:37:17 nano node-red[5745]: your settings file. Node-RED will then re-
jun 29 23:37:17 nano node-red[5745]: file using your chosen key the next time y
jun 29 23:37:17 nano node-red[5745]: -----
jun 29 23:37:17 nano node-red[5745]: 29 Jun 23:37:17 - [info] Starting flows
jun 29 23:37:18 nano node-red[5745]: 29 Jun 23:37:18 - [info] Started flows
jun 29 23:37:18 nano node-red[5745]: 29 Jun 23:37:18 - [info] [mqtt-broker:0583
lines 1-19/19 (END)
```

Figura 29. Servicio activo de Node-Red

Elaborado por: El investigador

3.2.7 Instalación de InfluxDB

Empleando la consola PuTTY se procede a la instalación de InfluxDB para el almacenamiento de los datos (ángulos) se utilizan los siguientes comandos:

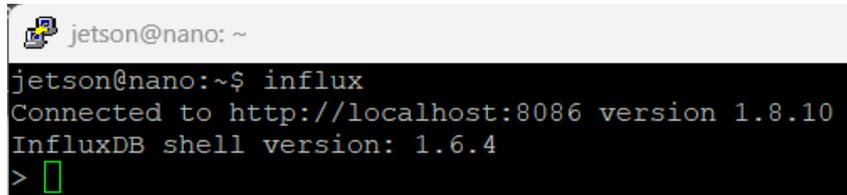
```
jetson@nano:~$ sudo apt update
```

```
jetson@nano:~$ sudo apt upgrade
```

```
jetson@nano:~$ sudo apt install influxdb
```

```
jetson@nano:~$ sudo apt install influxdb-client
```

En la Figura 33 se puede observar la versión instalada en nuestro servidor y el puerto predeterminado para influxdb. Con esto se comprueba que haya tenido una instalación correcta.



```
jetson@nano: ~
jetson@nano:~$ influx
Connected to http://localhost:8086 version 1.8.10
InfluxDB shell version: 1.6.4
> █
```

Figura 30. Versión instalada de influxdb

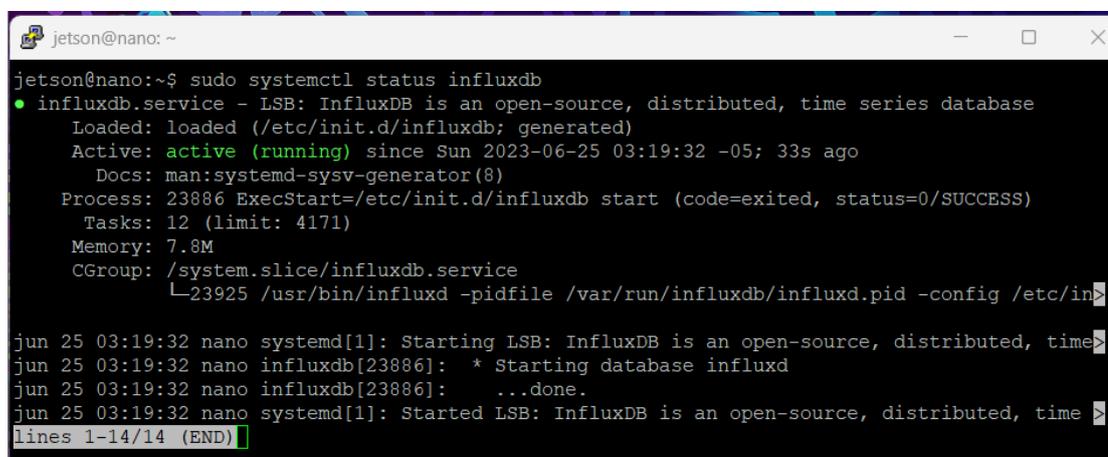
Elaborado por: El investigador

Realizada la instalación debemos iniciar el servicio:

```
jetson@nano:~$ sudo systemctl start influxdb
```

Para verificar que el servicio de influxdb este activo ingresamos el siguiente comando:

```
jetson@nano:~$ sudo systemctl status influxdb
```



```
jetson@nano: ~
jetson@nano:~$ sudo systemctl status influxdb
● influxdb.service - LSB: InfluxDB is an open-source, distributed, time series database
   Loaded: loaded (/etc/init.d/influxdb; generated)
   Active: active (running) since Sun 2023-06-25 03:19:32 -05; 33s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 23886 ExecStart=/etc/init.d/influxdb start (code=exited, status=0/SUCCESS)
    Tasks: 12 (limit: 4171)
   Memory: 7.8M
    CGroup: /system.slice/influxdb.service
            └─23925 /usr/bin/influxd -pidfile /var/run/influxdb/influxd.pid -config /etc/in
jun 25 03:19:32 nano systemd[1]: Starting LSB: InfluxDB is an open-source, distributed, time
jun 25 03:19:32 nano influxdb[23886]: * Starting database influxd
jun 25 03:19:32 nano influxdb[23886]:   ...done.
jun 25 03:19:32 nano systemd[1]: Started LSB: InfluxDB is an open-source, distributed, time
lines 1-14/14 (END) █
```

Figura 31. Servicio activo de influxdb

Elaborado por: El investigador

3.2.8 Instalación de Grafana

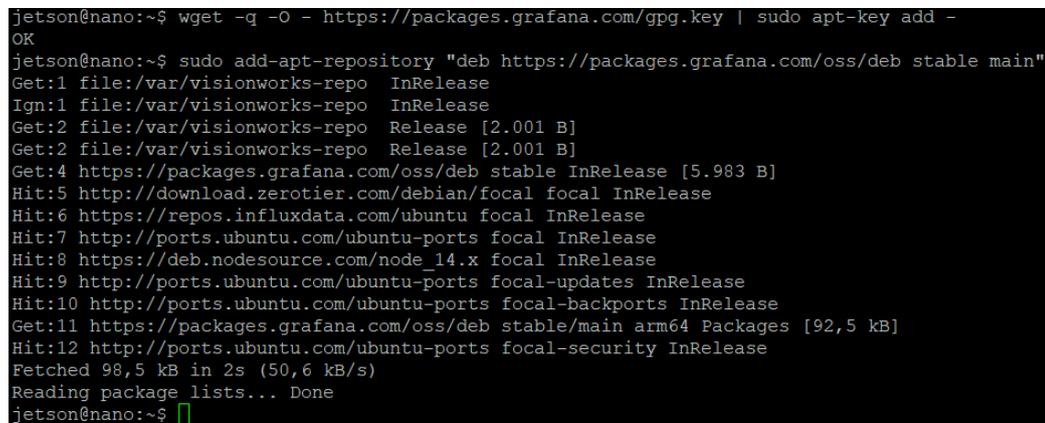
Para realizar la respectiva instalación de Grafana se utiliza la consola de PuTTY empleando los siguientes comandos, la instalación se realiza en base a [58]:

Se descarga la clave gpg y se agrega a la lista donde se encuentran las claves confiables:

```
jetson@nano:~$ wget -q -O - https://packages.grafana.com/gpg.key |  
sudo apt-key add -
```

Agregamos al repositorio de Grafana.

```
jetson@nano:~$ sudo add-apt-repository "deb  
https://packages.grafana.com/oss/deb stable main"
```



```
jetson@nano:~$ wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -  
OK  
jetson@nano:~$ sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"  
Get:1 file:/var/visionworks-repo InRelease  
Ign:1 file:/var/visionworks-repo InRelease  
Get:2 file:/var/visionworks-repo Release [2.001 B]  
Get:2 file:/var/visionworks-repo Release [2.001 B]  
Get:4 https://packages.grafana.com/oss/deb stable InRelease [5.983 B]  
Hit:5 http://download.zerotier.com/debian/focal focal InRelease  
Hit:6 https://repos.influxdata.com/ubuntu focal InRelease  
Hit:7 http://ports.ubuntu.com/ubuntu-ports focal InRelease  
Hit:8 https://deb.nodesource.com/node_14.x focal InRelease  
Hit:9 http://ports.ubuntu.com/ubuntu-ports focal-updates InRelease  
Hit:10 http://ports.ubuntu.com/ubuntu-ports focal-backports InRelease  
Get:11 https://packages.grafana.com/oss/deb stable/main arm64 Packages [92,5 kB]  
Hit:12 http://ports.ubuntu.com/ubuntu-ports focal-security InRelease  
Fetched 98,5 kB in 2s (50,6 kB/s)  
Reading package lists... Done  
jetson@nano:~$
```

Figura 32. Archivo agregado al repositorio

Elaborado por: El investigador

Se actualiza y se continua con la instalación

```
jetson@nano:~$ sudo apt install grafana
```

Una vez realizada la instalación se procede a agregar al servicio de Grafana al inicio automático cuando inicie el sistema.

```
jetson@nano:~$ sudo systemctl enable grafana
```

```
jetson@nano:~$ sudo systemctl start grafana
```

Para ver el estado de grafana se utiliza el siguiente comando:

```
jetson@nano:~$ sudo systemctl status grafana
```

```
jetson@nano:~$ systemctl status grafana-server.service
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; disabled; >
   Active: active (running) since Fri 2023-06-30 01:46:24 -05; 898ms ago
     Docs: http://docs.grafana.org
   Main PID: 98260 (grafana-server)
    Tasks: 9 (limit: 4181)
   Memory: 35.0M
   CGroup: /system.slice/grafana-server.service
           └─98260 /usr/sbin/grafana-server --config=/etc/grafana/grafana.ini>

jun 30 01:46:24 nano grafana-server[98260]: logger=licensing t=2023-06-30T01:46>
jun 30 01:46:24 nano grafana-server[98260]: logger=plugin.manager t=2023-06-30T>
jun 30 01:46:24 nano grafana-server[98260]: logger=plugin.finder t=2023-06-30T0>
jun 30 01:46:24 nano grafana-server[98260]: logger=query_data t=2023-06-30T01:4>
jun 30 01:46:24 nano grafana-server[98260]: logger=live.push_http t=2023-06-30T>
jun 30 01:46:24 nano grafana-server[98260]: logger=server t=2023-06-30T01:46:24>
jun 30 01:46:24 nano grafana-server[98260]: logger=grafanaStorageLogger t=2023->
jun 30 01:46:24 nano grafana-server[98260]: logger=report t=2023-06-30T01:46:24>
jun 30 01:46:24 nano grafana-server[98260]: logger=server t=2023-06-30T01:46:24>
jun 30 01:46:24 nano grafana-server[98260]: logger=tracing t=2023-06-30T01:46:2>
lines 1-20/20 (END)
```

Figura 33. Servicio de Grafana activo

Elaborado por: El investigador

En la Figura 34 se muestra el acceso a la conexión por el puerto 3000.

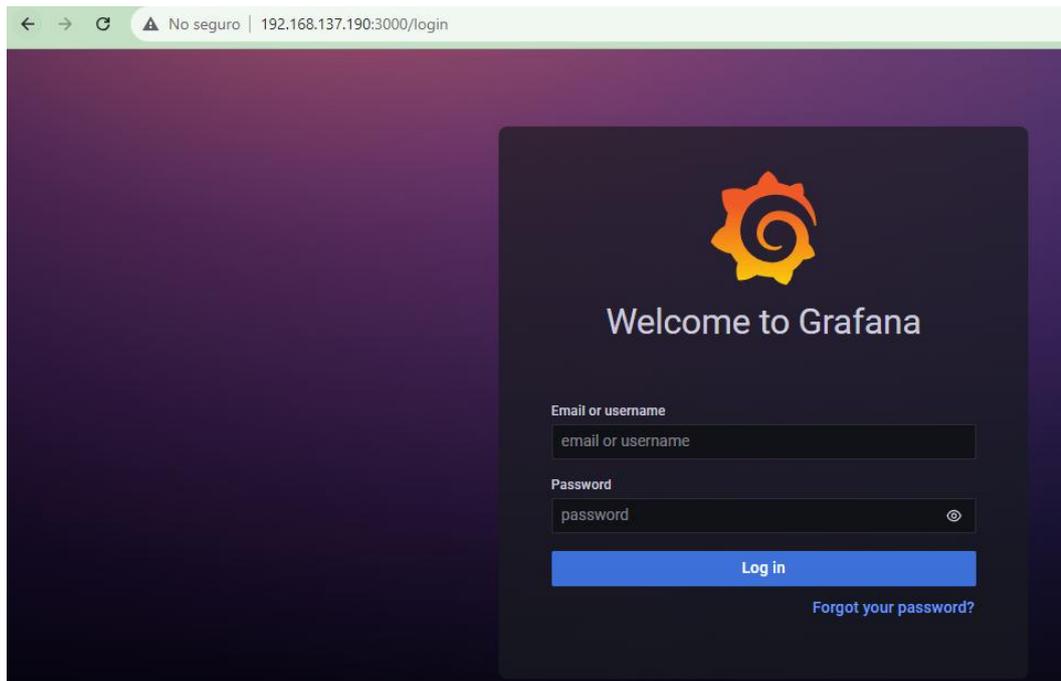


Figura 34. Acceso a Grafana

Elaborado por: El investigador

3.2.9 Diseño del entorno virtual

Desarrollo del entorno virtual

Se procede con la creación de la escena en Unity. Como se observa en la Figura 35 se le asigna una escala de 1 en las tres dimensiones. Al terreno se le asigna las medidas de ancho=200, altura=100 y profundidad =200 debido a que no se requiere de mucha escena sino más bien de crear la apariencia de estar fuera del sitio de rehabilitación.

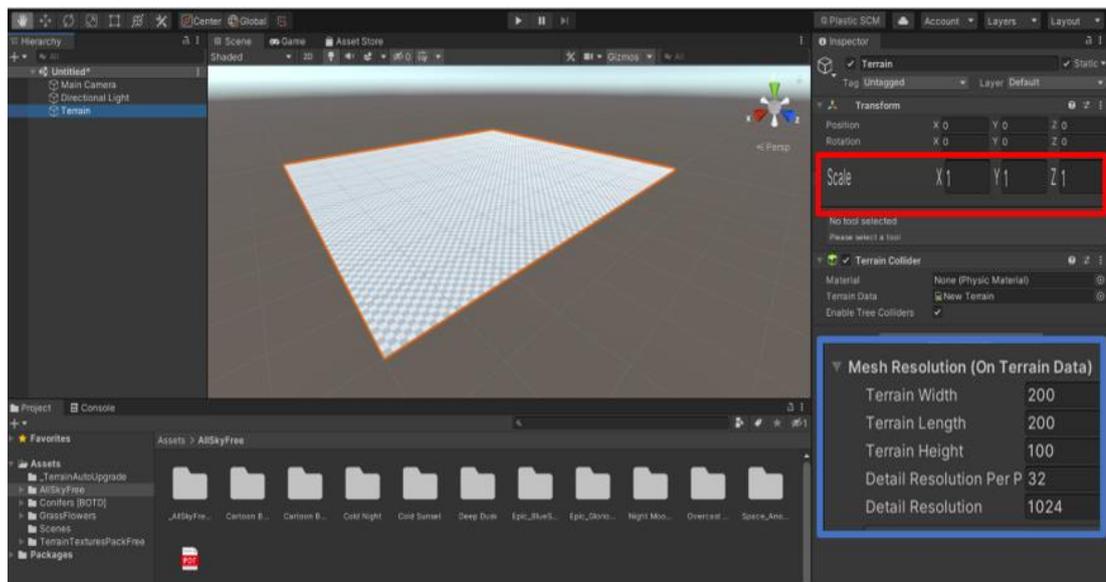


Figura 35. Dimensiones del terreno

Elaborado por: El investigador

En el terreno creado se procede a realizar las elevaciones para asimilar a las montañas e importar el personaje creado con anterioridad, para ubicarlo en la escena de Unity (Figura 36) y así proceder a darle estilos como color de las montañas, incluir árboles entre otras cosas, esto se presenta en la Figura 37.

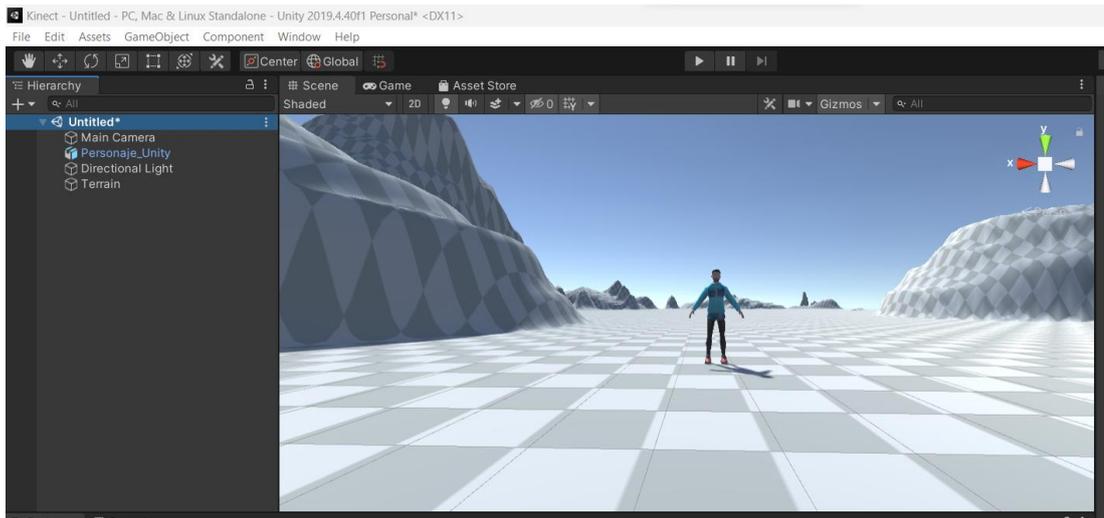


Figura 36. Avatar en la escena de Unity

Elaborado por: El investigador

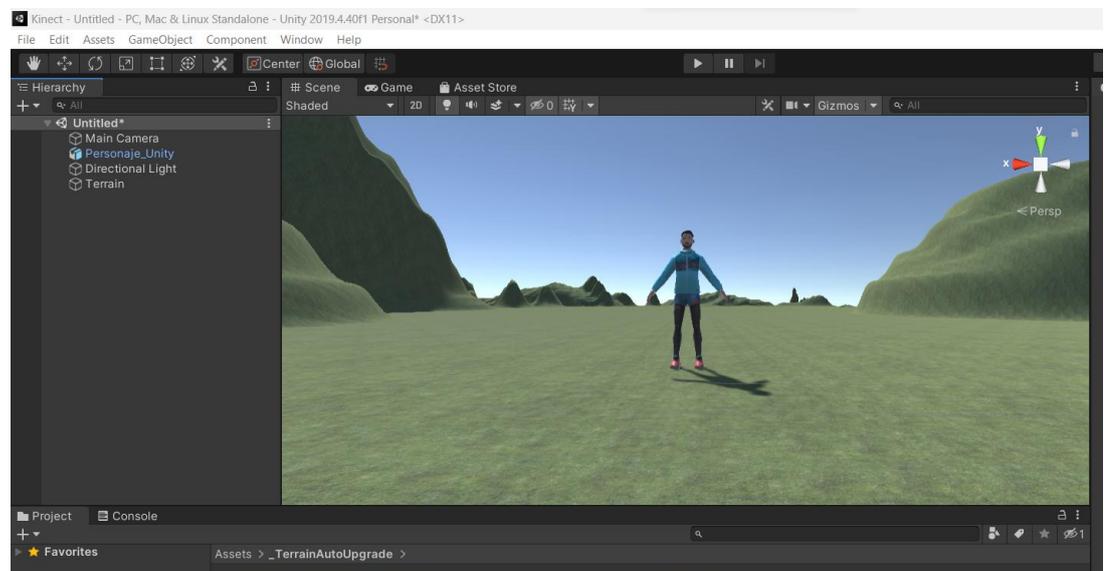


Figura 37. Diseño de la escena

Elaborado por: El investigador

La escena completa se presenta en la Figura 38 donde se puede identificar el terreno con un diseño enfocado a la naturaleza juntamente con el personaje mismo que generara el movimiento captado por el sensor Kinect v2.



Figura 38. Escena completa avatar hombre

Elaborado por: El investigador

Concluida la escena se puede importar más avatares, como se puede evidenciar en la Figura 39 se ubicó un avatar mujer en el mismo espacio que el avatar hombre, esto nos ayuda para empezar con la creación del menú de selección de personaje.



Figura 39. Escena completa avatar mujer

Elaborado por: El investigador

Estructura general del script C# para dar movimiento al avatar

Para dar movimiento a los diferentes avatares se requieren scripts escritos en C#. Estos scripts se encuentran en el ANEXO 5 , mismos que fueron desarrollados en el entorno de Visual Studio Code.

La estructura de los scripts utilizados en C# se divide en las siguientes partes:

- Biblioteca dinámica: permiten la reutilización de código
- Declaración de la clase MonoBehaviour: es una clase base de Unity permite interactuar a un objeto con el entorno de Unity y sus eventos
- void Start(): se ejecuta una vez al comienzo de la ejecución del juego
- void Update(): se ejecuta en cada Frame del juego

En el siguiente código se observa un ejemplo de la estructura en C# abierto en Visual Studio Code.

```
public class NewBehaviourScript : MonoBehaviour
{
    // se ejecuta una vez
    void Start()
    {
    }
    // se ejecuta en cada Frame
    void Update()
    {
    }
}
```

Diseño del menú inicial (Selección del personaje)

Para crear el menú principal se debe crear otra escena, pero en el mismo proyecto de Unity, luego se elige canvas en 2D para poder diseñar tanto el botón de iniciar como los dos recuadros que sirven para colocar a los personajes que se quiera seleccionar (Figura 40), para realizar este menú son necesario 4 scripts. Revisar ANEXO 6.

Con el script GuardarPersonaje se almacena al personaje seleccionado una vez dado clic sobre este, dependiendo de cuál sea la elección se va a guardar el avatar. El script Boton permite que la elección anterior sea guardada y el dato enviado al script

CargarPersonaje mismo que se encarga de ubicar en la escena al personaje seleccionado y de eliminar al que no se ha elegido.

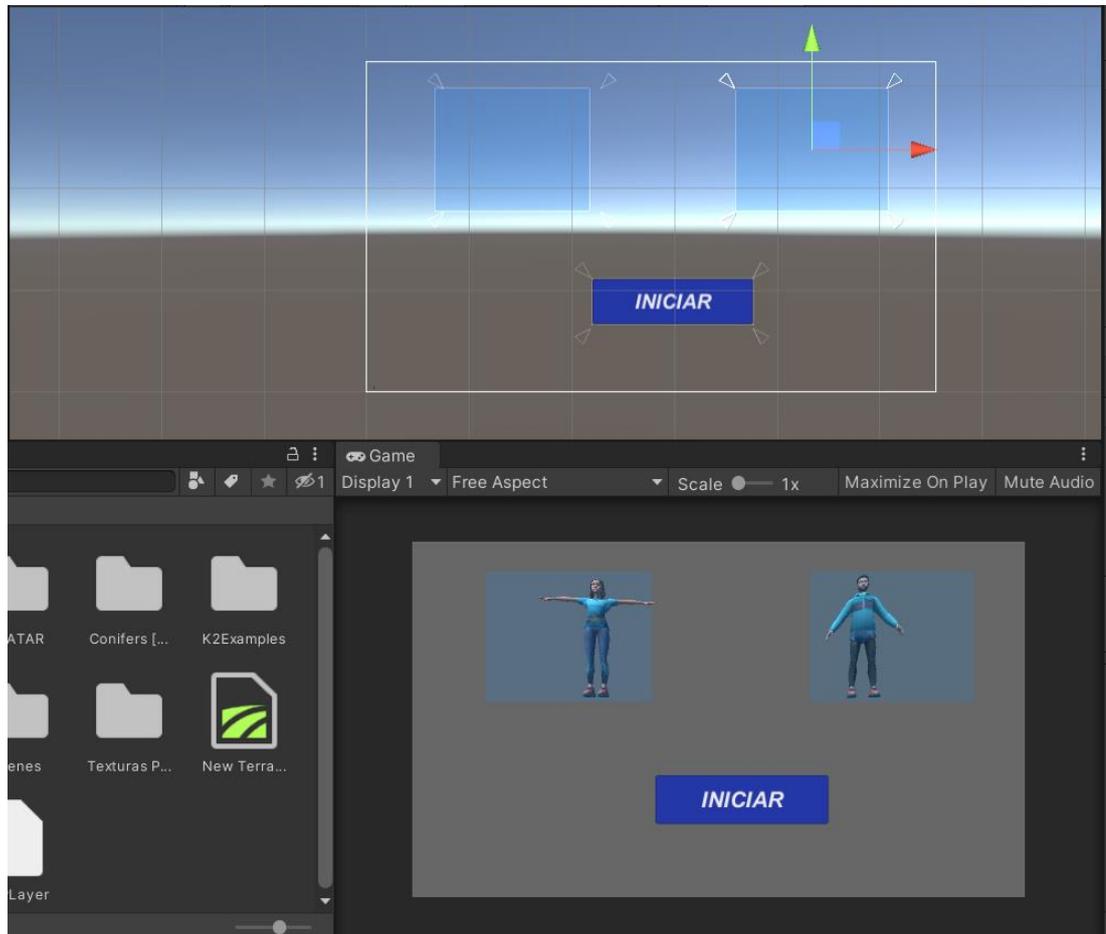


Figura 40. Menú principal

Elaborado por: El investigador

Finalmente, con el script Salir proporcionamos a nuestro entorno virtual una manera de quitar por completo de las escenas. En la Figura 41 se observa el botón salir en la escena completa del entorno virtual.



Figura 41. Boton salir en la escena

Elaborado por: El investigador

Unity permite exportar el entorno virtual y descargarlo en un ejecutable. En la Figura 42 se muestra el archivo ejecutable con el icono de la Unidad de Rehabilitación y Fisioterapias Gabo's, de esta manera se puede instalar en otros equipos y no es necesario que tenga instalado Unity, pero si debe tener el SDK2.0. El entorno virtual se instaló en la PC de la unidad de Rehabilitación.

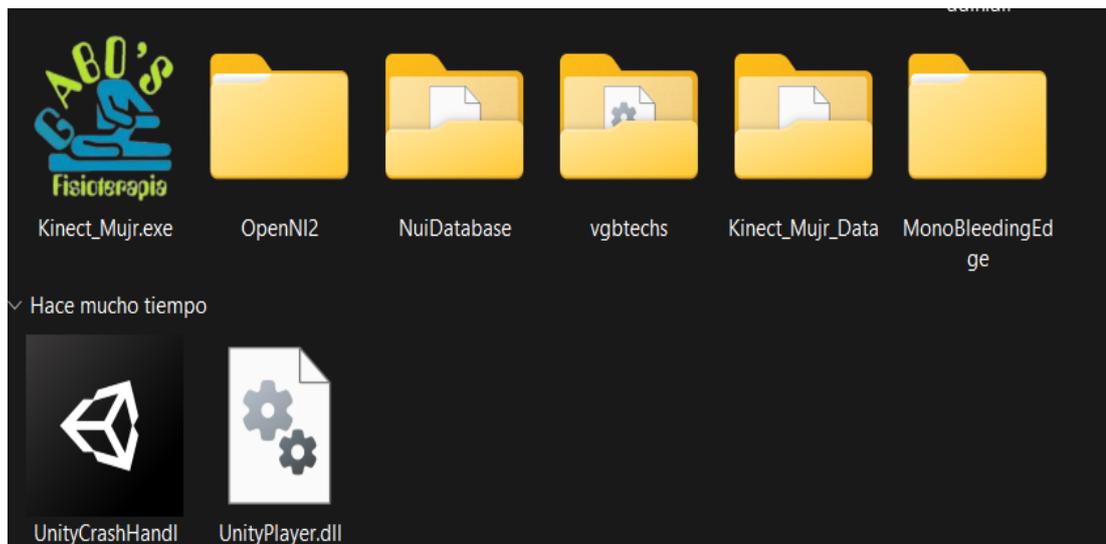


Figura 42. Ejecutable del entorno virtual

Elaborado por: El investigador

3.2.10 Diseño de la interfaz de visualización de los ángulos del tobillo

Para realizar la interfaz se hace uso de las herramientas previamente instaladas como es Node Red para recolectar los datos enviados desde Python y para desarrollar un apartado de registro de los pacientes para posteriormente generar los reportes.

Configuración de influxdb en Node-Red

En influxdb se guardan todos los ángulos obtenidos de los dos tobillos en una base de datos llamada gaboterapia que son enviados desde Python por medio del protocolo MQTT con el topic angle/ankle/left y angle/ankle/right, se debe configurar el nodo de influx designándole un nombre de medida para que sea guardado los datos como se puede observar en la Figura 43 se ha designado gradosD este paso se debe repetir para designar a gradosI.

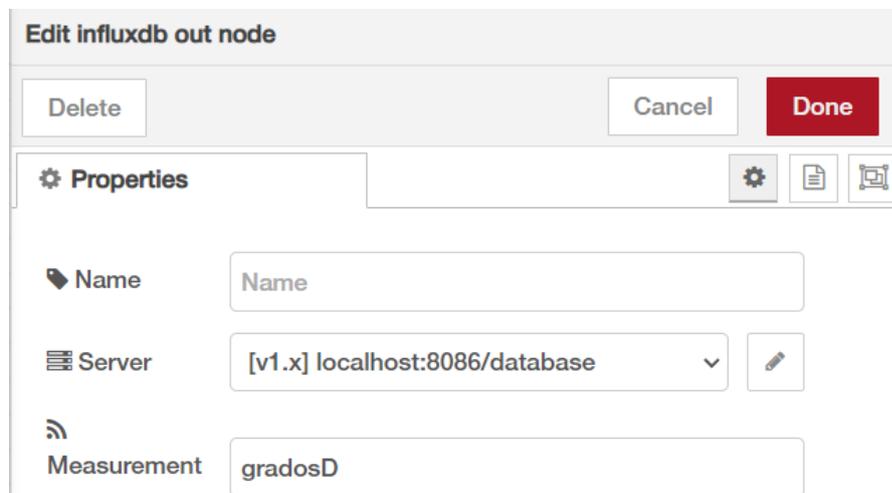


Figura 43. Configuración del nodo influx

Elaborado por: El investigador

En la Figura 44 se muestra cómo queda las conexiones para que los datos de los ángulos tanto del tobillo izquierdo como del derecho se guarden en la base de datos influxdb. Sin olvidarse que después de realizar cualquier cambio se debe presionar el botón de Deploy.

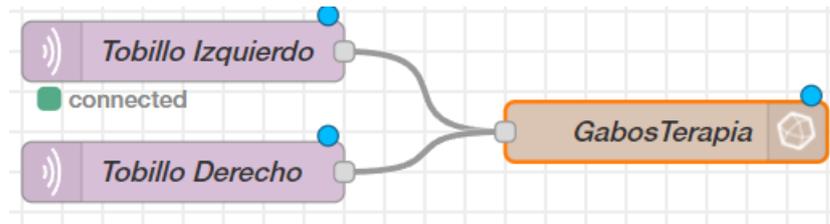


Figura 44.Datos enviados a influxdb

Elaborado por: El investigador

Configuración de grafana para visualizar los datos almacenados por influxdb

Una vez ingresado a la página principal de grafana se debe agregar la base de datos para lo cual hay que ir al menú principal que se encuentra en el panel lateral izquierdo. En la Figura 45 se observa las opciones que hay en el menú, se debe escoger Data sources y posterior InfluxDB.

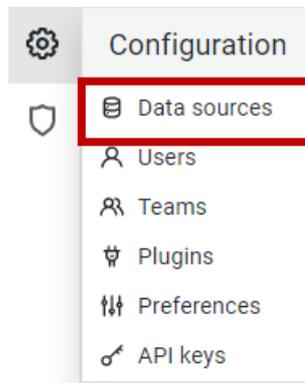


Figura 45. Menú de opciones

Elaborado por: El investigador

En la Figura 46 se observa los datos que se deben configurar para ser utilizado influxdb en grafana:

1. Colocar un nombre a la base de datos.
2. Se debe escribir la dirección IP del servidor
3. Ubicar el puerto de InfluxDB que por defecto es el 8086.
4. Añadir el nombre de la base de datos en InfluxDB
5. Agregar las credenciales de usuario y contraseña

Realizado todos los puntos anteriores se da clic en el botón Save & test si los datos están correctos la base de datos procede a guardarse, es importante escribir

correctamente los datos caso contrario aparece un mensaje de error y por ende la base de datos no se guarda.

Data Sources / InfluxDB
Type: InfluxDB

Settings

Name Default

Query Language

HTTP

URL

Access Help >

Allowed cookies

Timeout

Database

User

Password

HTTP Method

Min time interval

Max series

Figura 46. Configuración de influxdb en grafana

Elaborado por: El investigador

Finalmente se puede observar en la Figura 47 la interfaz en grafana donde se ha colocado por medio de una entrada de texto en HTML se agregó el nombre y una imagen que es el logo de la Unidad de Fisioterapia.



Figura 47. Dashboard para visualizar resultados

Elaborado por: El investigador

A continuación se muestra la Figura 48 en donde se encuentran diferentes paneles de visualización, con el color amarillo los resultados del tobillo izquierdo y con azul los resultados del tobillo derecho. Además, en la parte inferior se observa los dos datos en un solo panel donde se puede realizar una comparativa de los ángulos.



Figura 48. Paneles de visualización de resultados

Elaborado por: El investigador

Registro de pacientes

En parte superior izquierda de la ventana de node red esta un menú de opciones para realizar el respectivo registro de los pacientes como ver el reporte de estos, registran datos como: nombre, apellido, cedula, edad, peso, estatura, lesión. Al ser ingresado los datos completos se guardan y así el paciente se registra en la base de datos. En la Figura 49 se observa la ventana de registro de los pacientes.



Figura 49. Interfaz de registro de pacientes

Elaborado por: El investigador

Bases de datos

En la minicomputadora Jetson nano que funge como servidor se levanta el servicio de apache2, para que almacene los datos de los pacientes y las recomendaciones según el ángulo formado por la articulación toda esta información se guarda en MySQL.

Se crea la base de datos “monitoreo”, luego se crean las tablas requeridas para almacenar información. La primera tabla almacena los datos que los pacientes ingresan a esta se le da el nombre de “pacientes”. La segunda tabla almacena los datos que provienen de Python a esta se le designa “tratamiento_pacientes”.

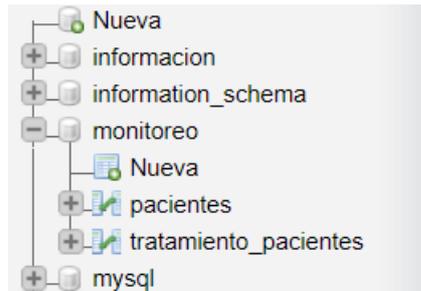


Figura 50. Base de datos

Elaborado por: El investigador

Se utiliza la función de claves foráneas en la base de datos para relacionar las respectivas tablas, misma que emplea una clave primaria generalmente llamada id y una clave foránea misma que es empleada para relacionar los registros únicos en otra tabla. El identificador de la tabla “pacientes” con el nombre de id_pacientes, la cual está relacionada con la tabla “tratamiento_pacientes” con el campo id_pacientes. En la Figura 51 se muestra como están relacionadas las tablas.

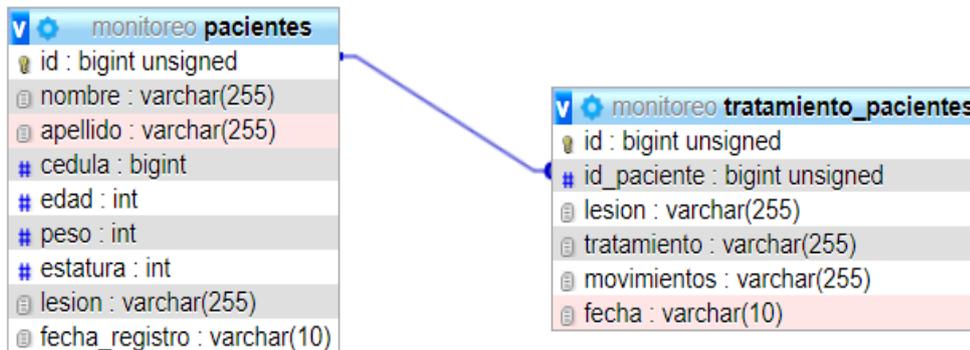


Figura 51. Modelo entidad-relación

Elaborado por: El investigador

Posteriormente, se indican las tablas que se crearon en la base de datos de MySQL, esta información se puede observar al ingresar en el navegador con la dirección de 192.168.137.190 seguidamente de phpmyadmin con la cuenta que se haya registrado en la instalación. En la Figura 52 se muestra la tabla de los datos de los pacientes que se han registrado.

Examinar Estructura SQL Buscar Insertar Exportar Importar Privilegios Operaciones

Mostrando filas 0 - 7 (total de 8, La consulta tardó 0.0011 segundos.)

SELECT * FROM `clientes`

Perfilando [Editar en línea] [Editar] [Explicar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

id	nombre	apellido	cedula	edad	peso	estatura	lesion	fecha_registro
8	Santiago	Bustos	1823456780	45	70	159	Ninguna	06/07/2023
9	Juan	Haro	1823456781	23	65	168	Rotura de pierna D	07/07/2023
10	Diego	Mayorga	1823456782	16	55	170	Esguince de tobillo I	07/07/2023
11	Fabian	Aldaz	1823456783	50	68	168	Esguince de tobillo I	07/07/2023
12	Gino	Torres	1823456784	35	70	173	Esguince de tobillo D	07/07/2023
13	Elena	Alban	1823456785	43	75	150	Esguince de tobillo D	07/07/2023
14	Gilson	Flores	1804966453	27	65	157	Rotura del tendón de aquiles	07/07/2023
15	Andrea	Nuñez	1823456786	27	50	153	Esguince	07/07/2023

Seleccionar todo | Para los elementos que están marcados: Editar Copiar Borrar Exportar

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

Figura 52. Tabla de los pacientes registrados en la base de datos

Elaborado por: El investigador

En la Figura 53 se indica la tabla que guarda los datos del paciente para poder generar el reporte de acuerdo a las recomendaciones para realizar un ejercicio en específico todo esto depende del ángulo generado por el tobillo.

id	id_cliente	lesion	tratamiento	movimientos	fecha
1	8	Ninguna	Correcto	135	06/07/2023
10	8	Ninguno	Correcto	68.7	06/07/2023
11	8	Ninguna	Correcto	138.77	06/07/2023
12	8	Ninguna	Correcto	67.19	07/07/2023
13	8	Ninguna	Correcto	135.97	07/07/2023
14	8	Ninguna	Correcto	69.94	07/07/2023
15	9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	97.54	06/07/2023
16	9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	89.16	06/07/2023
17	9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	95.23	06/07/2023
18	9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	90	06/07/2023
19	9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	95.75	06/07/2023
20	9	Rotura de pierna Derecha	Correcto	72.23	07/07/2023
21	9	Rotura de pierna Derecha	Correcto	131.18	07/07/2023
22	9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	88.67	07/07/2023
23	9	Rotura de pierna Derecha	Correcto	133.42	07/07/2023
24	9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	97.37	07/07/2023
25	9	Rotura de pierna Derecha	Correcto	135.67	10/07/2023
26	9	Rotura de pierna Derecha	Correcto	143.19	10/07/2023
27	9	Rotura de pierna Derecha	Correcto	136.47	10/07/2023

Figura 53. Tabla para generar el reporte del paciente

Elaborado por: El investigador

Flujograma de procesos para la obtención y grafica de los ángulos del tobillo

En este punto se describe los procesos primordiales que constituyen el sistema de rehabilitación.

El proceso comienza al encender la Jetson Nano, el cual ejecuta el programa de Python e inicializa las librerías, a su vez a la microcomputadora está conectada una cámara. En la siguiente etapa se realiza la conexión por MQTT misma que debe contener la ip del bróker y el puerto designado. Al ser la conexión correcta el programa establece la comunicación con el bróker MQTT instalado en el servidor.

Luego que se haya establecido la conexión el programa identifica la cámara designada para detectar las articulaciones, extremidades y ángulos, se calculan los ángulos formados por tres puntos de referencia y son enviados empleando el protocolo MQTT a Node-Red que se encarga de recolectar los datos y posteriormente almacenarlos en Influxdb.

Una vez generada la base de datos con la información de los ángulos del tobillo se realiza la respectiva configuración de influxdb en grafana, realizado este paso se puede observar la información almacenada en una interfaz gráfica donde se puede apreciar los ángulos generados por los tobillos. Todo este procedimiento es efectuado cuando el dispositivo este encendido y por ende el programa ejecutándose.

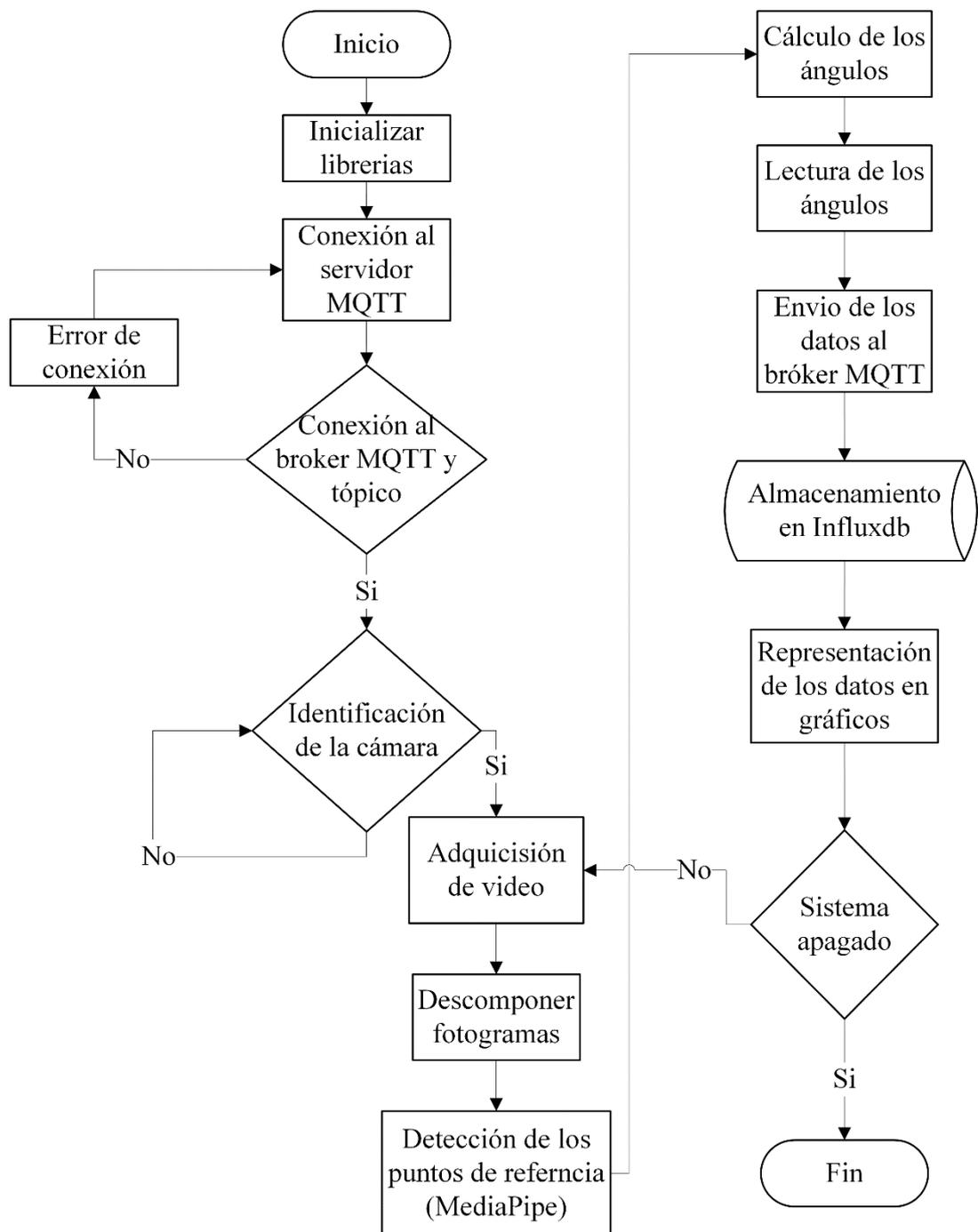


Figura 54. Flujograma

Elaborado por: El investigador

3.2.11 Pruebas de funcionamiento

Para verificar la funcionalidad del sistema se efectúan diversas pruebas con los pacientes que tienen algún traumatismo en el tobillo, el paciente se posiciona frente al sistema de rehabilitación en donde la cámara del Kinect toma los movimientos y la réplica al avatar que se proyecta en la pantalla, del mismo modo la cámara IMX219-77 toma los ángulos del tobillo para realizar el análisis respectivo, para posterior ser observados en un dashboard de Grafana.

En la Figura 55 se observa a los pacientes quienes han sido expuestos a las respectivas pruebas de funcionamiento del sistema de rehabilitación física.



Figura 55. Pruebas de funcionamiento en pacientes

Elaborado por: El investigador

Para continuar con desarrollo de las pruebas se debe verificar que el sistema este correctamente implementado, comprobar si el entorno virtual se ejecuta con normalidad como se muestra en la Tabla 12.

Tabla 12. Comprobación del entorno virtual

Movimientos	
Paciente	Avatar
	
	
	



Elaborado por: El investigador

Además, se debe considerar una distancia óptima para que el sistema capte por completo al paciente como se observa en la Figura 57.



Figura 56. Sistema implementado

Elaborado por: El investigador



Figura 57. Distancia del paciente respecto al sistema

Elaborado por: El investigador

Prueba 1

Registro del paciente sano

Al iniciar con la rehabilitación física principalmente se debe realizar el registro de los datos de los pacientes con la correcta información y sin olvidarse de la lesión que presenta. El paciente registrado se le otorga una id específica para poder realizar las consultas para la generación del reporte. En la Figura 58 se puede observar los datos ingresados del primer paciente.

Registro de datos

Registro

Nombre:
Santiago

Apellido:
[redacted]

Cedula:
18

Edad:
45

Peso(kg):
70

Estatura(cm):
159

Lesión:
Ninguna

GUARDAR LIMPIAR

Figura 58. Datos ingresados del paciente sano

Elaborado por: El investigador

Una vez realizado el registro de los datos el paciente procede a realizar actividad frente al sistema mismo que está captando los movimientos ejercidos y replicándolos en un avatar que se visualiza en una pantalla. De igual manera la otra cámara de alta definición toma los puntos de referencia y los calcula proporcionándonos los ángulos de la articulación y en base a los mismos estimar una recomendación para mejorar la movilidad.

Resultados obtenidos del paciente sano

En la Figura 59 y 60 se ven los ángulos que se obtienen en las articulaciones, se grafica las líneas de las articulaciones con sus respectivos puntos de referencia.



Figura 59. Ángulo en flexión plantar

Elaborado por: El investigador



Figura 60. Ángulo en flexión dorsal

Elaborado por: El investigador

En la Figura 61 se ven los resultados en flexión plantar obtenidos del paciente sano.



Figura 61. Ángulo formado en flexión plantar (paciente sano)

Elaborado por: El investigador

En la Figura 62 se muestra los resultados en flexión dorsal obtenidos del paciente sano.



Figura 62. Ángulo formado en flexión dorsal (paciente sano)

Elaborado por: El investigador

Reporte del paciente sano



Información General

Usuario	:	Santiago [redacted]
Cedula	:	18 [redacted]
Edad	:	45
Peso (Kg)	:	70
Estatura (Cm)	:	159

Información de la Lesión

Ninguna

Historial del Paciente: Santiago [redacted]

ID Paciente	Lesión	Tratamiento	Movimientos	Fecha
8	Ninguna	Correcto	135	06/07/2023
8	Ninguno	Correcto	68.7	06/07/2023
8	Ninguna	Correcto	138.77	06/07/2023
8	Ninguna	Correcto	67.19	07/07/2023
8	Ninguna	Correcto	135.97	07/07/2023
8	Ninguna	Correcto	69.94	07/07/2023

Figura 63. Reporte del paciente sano

Elaborado por: El investigador

Análisis de los resultados obtenidos

Se puede evidenciar en los resultados obtenidos que los ángulos de movilidad del tobillo corresponden a un rango normal tomando en cuenta en ángulo inicial que es de 90 grados, a este se le aumenta de 40 a 50 grados en la flexión plantar. En la Figura 61 se observa los ángulos 143 y 139 grados para el tobillo derecho como para el izquierdo respectivamente.

Para determinar una correcta flexión dorsal se parte del ángulo inicial que es de 90 grados a este se le disminuye de 20 a 30 grados. En la Figura 62 se observa los ángulos 72,5 y 66 grados para el tobillo izquierdo como para el derecho respectivamente.

En la Figura 63 se observa las recomendaciones del tratamiento. Al tratarse de un paciente sano se puede verificar que los ángulos de movilidad están correctos y no necesita realizar un trabajo adicional para aumentar dicha movilidad de los tobillos.

Prueba 2

Registro del paciente con rotura de pierna derecha

Al iniciar con la rehabilitación física principalmente se debe realizar el registro de los datos de los pacientes con la correcta información y sin olvidarse de la lesión que presenta. El paciente registrado se le otorga una id específica para poder realizar las consultas para la generación del reporte. En la Figura 64 se puede observar los datos ingresados del segundo paciente.

Registro de datos

Registro

Nombre:
Juan

Apellido:
[redacted]

Cedula:
18 [redacted]

Edad:
23

Peso(kg):
65

Estatura(cm):
168

Lesión:
Rotura de pierna Derecha

GUARDAR LIMPIAR

Figura 64. Datos ingresados del paciente con lesión

Elaborado por: El investigador

Una vez realizado el registro de los datos el paciente procede a realizar actividad frente al sistema mismo que está captando los movimientos ejercidos y replicándolos en un avatar que se visualiza en una pantalla. De igual manera la otra cámara de alta definición toma los puntos de referencia y los calcula proporcionándonos los ángulos de la articulación y en base a los mismos estimar una recomendación para mejorar la movilidad.

Resultados obtenidos del paciente con rotura de pierna derecha

En la Figura 65 se ven los ángulos que se obtienen en las articulaciones, se grafica las líneas de las articulaciones con sus respectivos puntos de referencia.



Figura 65. Ángulo de los tobillos (paciente lesionado)

Elaborado por: El investigador

En la Figura 66 se observan los resultados en flexión plantar obtenidos del paciente con rotura de la pierna derecha.



Figura 66. Ángulo formado en flexión plantar (paciente lesionado)

Elaborado por: El investigador

En la Figura 67 se observan los resultados en flexión dorsal obtenidos del paciente con rotura de la pierna derecha.



Figura 67. Ángulo formado en flexión dorsal (paciente lesionado)

Elaborado por: El investigador

Reporte del paciente

GABO'S
Fisioterapia
Juan Haro

Información General

Usuario	:	Juan
Cedula	:	18
Edad	:	23
Peso (Km)	:	65
Estatura (Cm)	:	168

Información de la Lesión

Rotura de pierna D

Historial del Paciente: Juan

ID Paciente	Lesión	Tratamiento	Movimientos	Fecha
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	97.54	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	89.16	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	95.23	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	90	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	95.75	06/07/2023

Figura 68. Reporte inicial del paciente lesionado

Elaborado por: El investigador

Análisis de los resultados obtenidos

Se puede apreciar en los resultados obtenidos que los ángulos de movilidad del tobillo corresponden a un paciente que ha sufrido una lesión de consideración, por tal motivo teniendo en cuenta el ángulo inicial que es de 90 grados, a este se le aumenta de 40 a 50 grados en la flexión plantar. En la Figura 66 se observa los ángulos 97.5 y 132 grados para el tobillo derecho como para el izquierdo respectivamente.

Para establecer una correcta flexión dorsal se parte del ángulo inicial que es de 90 grados a este se le disminuye de 20 a 30 grados. En la Figura 67 se observa los ángulos 66.6 y 97.2 grados para el tobillo izquierdo como para el derecho respectivamente.

El paciente al encontrarse en reposo por la lesión ha perdido movilidad del tobillo derecho casi en su totalidad de acuerdo a los datos obtenidos el rango de movimiento es de 2 grados por lo cual se debe realizar ejercicios para aumentar la movilidad tanto en flexión plantar como en flexión dorsal, todo esto de acuerdo al reporte generado.

Resultados obtenidos del paciente después de las recomendaciones

Los ejercicios son aplicados en las siguientes sesiones de fisioterapia, se ha dado seguimiento al paciente mismo que después de varias sesiones de fisioterapia ha conseguido mejorar el rango de movilidad los resultados se presentan en la Figura 69.



Figura 69. Paciente en recuperación

Elaborado por: El investigador

En la Figura 70 se observan los resultados en flexión plantar obtenidos del paciente con rotura de la pierna derecha en recuperación .



Figura 70. Ángulo formado en flexión plantar (paciente lesionado)

Elaborado por: El investigador

En la Figura 70 se observan los resultados en flexión dorsal obtenidos del paciente con rotura de la pierna derecha en recuperación .



Figura 71. Ángulo formado en flexión plantar (paciente lesionado)

Elaborado por: El investigador

Reporte del paciente después de varias sesiones de fisioterapia

Historial del Paciente: Juan				
ID Paciente	Lesión	Tratamiento	Movimientos	Fecha
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	97.54	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	89.16	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	95.23	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	90	06/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	95.75	06/07/2023
9	Rotura de pierna Derecha	Correcto	72.23	07/07/2023
9	Rotura de pierna Derecha	Correcto	131.18	07/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de dorsiflexión	88.67	07/07/2023
9	Rotura de pierna Derecha	Correcto	133.42	07/07/2023
9	Rotura de pierna Derecha	Aumentar ejercicios de plantiflexión	97.37	07/07/2023
9	Rotura de pierna Derecha	Correcto	135.67	10/07/2023
9	Rotura de pierna Derecha	Correcto	143.19	10/07/2023
9	Rotura de pierna Derecha	Correcto	136.47	10/07/2023
9	Rotura de pierna Derecha	Correcto	132.97	10/07/2023
9	Rotura de pierna Derecha	Correcto	73.23	10/07/2023
9	Rotura de pierna Derecha	Correcto	74.67	10/07/2023
9	Rotura de pierna Derecha	Correcto	66.13	10/07/2023

Figura 72. Reporte del paciente

Elaborado por: El investigador

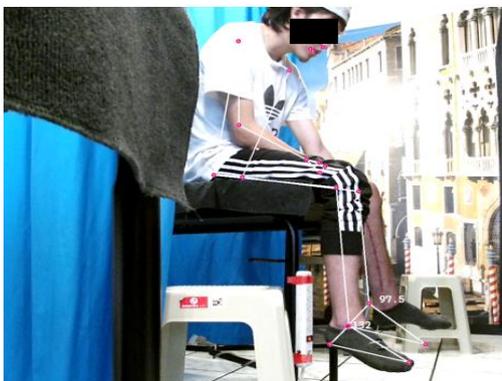


Figura 73. Paciente iniciando la terapia física

Elaborado por: El investigador



Figura 74. Paciente realizando ejercicios con el sistema

Elaborado por: El investigador

En la Figura 72 se puede observar los datos que se generan en el reporte, los ángulos que se obtienen al inicio de la rehabilitación corresponden a un limitado rango de movilidad por cuanto se recomienda realizar ejercicios de flexión plantar y flexión dorsal. Tomando en cuenta las recomendaciones realizadas se puede observar que conforme pasan los días de terapias se va recuperando el rango de movimiento.

Los primeros datos son tomadas del paciente sentado en la camilla como se muestra en la Figura 73, al transcurso de los días al paciente se le observan mejorías, en la Figura 74 se evidencia al paciente realizando los respectivos ejercicios al frente del sistema implementado.

3.2.12 Porcentaje de error del sistema de rehabilitación

Finalizado las pruebas de funcionamiento con pacientes sanos y lesionados se determina la exactitud y el error del sistema en la detección de los ángulos del tobillo, para realizar este procedimiento el profesional efectúa las mediciones respectivas haciendo uso del goniómetro.



Figura 75. Ángulos calculados por el sistema

Elaborado por: El investigador



Figura 76. Ángulos medidos con el goniómetro

Elaborado por: El investigador

Empleando los dos métodos se realizan 20 muestras de la articulación para realizar los respectivos cálculos de error, se ocupan los ángulos determinados por el sistema y los medidos de forma manual tanto en flexión plantar como flexión dorsal.

Tabla 13. Analisis de error del ángulo tobillo flexión plantar

N°	Sistema (grados)	Goniómetro (grados)	Error absoluto	Error relativo	Error porcentual
1	143,12	139	4,12	0,030	2,96%
2	139,63	135,5	4,13	0,030	3,05%
3	138,96	132	6,96	0,053	5,27%
4	132,19	139	6,81	0,049	4,90%
5	124,12	130,5	6,38	0,049	4,89%
6	123,17	125,5	2,33	0,019	1,86%
7	143,59	135	8,59	0,064	6,36%
8	137,56	139,55	1,99	0,014	1,43%
9	120,45	129	8,55	0,066	6,63%
10	92,27	90	2,27	0,025	2,52%
11	110,59	115	4,41	0,038	3,83%
12	115,34	110,5	4,84	0,044	4,38%
13	123,81	120	3,81	0,032	3,18%
14	127,96	120,5	7,46	0,062	6,19%
15	125,17	119	6,17	0,052	5,18%
16	143,46	135	8,46	0,063	6,27%
17	136,47	139,5	3,03	0,022	2,17%
18	129,26	135,5	6,24	0,046	4,61%
19	135,37	140	4,63	0,033	3,31%
20	142,37	135,5	6,87	0,051	5,07%
		Promedio	5,4025	Promedio	4,20%

Elaborado por: El investigador

En la Tabla 13 que pertenece a los ángulos tomados en flexión plantar se evidencia un error del 4,20% del sistema en diferencia con las mediciones realizadas por el fisioterapeuta utilizando el goniómetro. No obstante el error absoluto promedio que se obtiene en el sistema es de 5,4025 en consecuencia se tiene un nivel elevado de confiabilidad del 95,80%.

Tabla 14. Analisis de error del ángulo tobillo flexión dorsal

N°	Sistema (grados)	Goniómetro (grados)	Error absoluto	Error relativo	Error porcentual
1	72,53	70	2,53	0,036	3,61%
2	66,13	63	3,13	0,050	4,97%
3	66,67	70,5	3,83	0,054	5,43%
4	71,37	68,5	2,87	0,042	4,19%
5	69,94	65,5	4,44	0,068	6,78%
6	65,46	70	4,54	0,065	6,49%
7	67,17	62,5	4,67	0,075	7,47%
8	69,22	65,5	3,72	0,057	5,68%
9	74,67	70	4,67	0,067	6,67%
10	76,88	73,57	3,31	0,045	4,50%
11	79,52	85	5,48	0,064	6,45%
12	62,56	65,5	2,94	0,045	4,49%
13	68,43	65	3,43	0,053	5,28%
14	86,64	90	3,36	0,037	3,73%
15	83,56	80,5	3,06	0,038	3,80%
16	79,23	75	4,23	0,056	5,64%
17	72,36	67,5	4,86	0,072	7,20%
18	73,15	70	3,15	0,045	4,50%
19	76,47	72,5	3,97	0,055	5,48%
20	68,28	70	1,72	0,025	2,46%
		Promedio	3,6955	Promedio	5,24%

Elaborado por: El investigador

En la Tabla 14 que pertenece a los ángulos tomados en flexión dorsal se evidencia un error del 5.24% del sistema en diferencia con las mediciones realizadas por el fisioterapeuta utilizando el goniómetro. Sin embargo, el error absoluto promedio que se obtiene en el sistema es de 3,6955 en consecuencia se tiene un nivel elevado de confiabilidad del 94,76%.

Obtenidos los valores tanto para flexión plantar como para flexión dorsal se procede a calcular el error total del sistema. En la Tabla 14 se muestra los errores totales en cada una de las mediciones.

Tabla 15. Error general del sistema

Movimientos	Error absoluto	Error porcentual
Flexión plantar	5,4025	4,203
Flexión dorsal	3,6955	5,625
Total	4,55	4,72%

Elaborado por: El investigador

Como se observa en la Tabla 15 se obtiene un error general promedio del 4,72% en los dos movimientos teniendo en cuenta que hay más error en el movimiento de flexión dorsal. No obstante, se tiene un error absoluto promedio de 4,55 en consecuencia el nivel de confiabilidad general del sistema es de 95,28% en el cálculo de los ángulos.

Validación del sistema

Al realizar las pruebas de funcionamiento conjuntamente con el fisioterapeuta de la Unidad de Rehabilitación y Fisioterapia Gabo's se determina el error del sistema permitiendo de esta manera conocer al profesional la eficiencia del sistema implementado.

En el ANEXO 1 se muestra el certificado que valida el sistema realizado, mismo que fue otorgado por el Licenciado en Terapia Física Gabriel Moreno.

A continuación se procede a realizar el cálculo para determinar el presupuesto del sistema implementado.

3.2.13 Presupuesto

En la determinación del presupuesto del sistema de rehabilitación física para pacientes con lesiones de tobillo se consideró el presupuesto tanto de diseño como de construcción. El diseño de construcción se obtiene con los materiales esenciales para implementar el sistema mientras que para el presupuesto de diseño se establece por el

número de horas destinadas en la elaboración. Se debe tomar en cuenta el salario de un ingeniero en Telecomunicaciones según el Ministerio del Trabajo determina en \$858 en la tabla Sectorial de salarios 2023 [59]. Se estipulan 21 días laborables; a continuación se determina el salario por día:

$$\text{Salario}_{\text{diario}} = \frac{\text{Salario}_{\text{mensual}}}{\text{Días}_{\text{laborables}}}$$

$$\text{Salario}_{\text{diario}} = \frac{858}{21}$$

$$\text{Salario}_{\text{diario}} = 40.85[\text{dólares}]$$

Según el Ministerio de Trabajo las horas laborables son 8, a continuación se obtiene el valor por hora de trabajo.

$$\text{Salario}_{\text{hora}} = \frac{\text{Salario}_{\text{diario}}}{\text{Horas}_{\text{laborables}}}$$

$$\text{Salario}_{\text{hora}} = \frac{40,85}{8}$$

$$\text{Salario}_{\text{hora}} = 5,10[\text{dólares}]$$

Se estima 120 horas de investigación, diseño, implementación y pruebas de funcionamiento, con lo cual se obtiene el presupuesto de diseño del sistema de rehabilitación.

$$\text{Presupuesto}_{\text{total}} = \text{Presupuesto}_{\text{diseño}} * \text{Salario}_{\text{hora}}$$

$$\text{Presupuesto}_{\text{diseño}} = 120[\text{horas}] * 5.10[\text{dólares}]$$

$$\text{Presupuesto}_{\text{diseño}} = 612[\text{dólares}]$$

Los materiales necesarios para implementar el sistema de rehabilitación se detallan en la Tabla 16.

Tabla 16. Presupuesto de implementación del sistema

N°	Detalle	Cantidad	Valor Unitario	Valor total
1	Kit de desarrollo Jetson Nano	1	\$180	\$180
2	Cámara IMX219-77	1	\$25	\$25
3	Kinect v2.0	1	\$110	\$110
4	Monitor	1	\$60	\$60
5	Mouse	1	\$6	\$6
6	Teclado	1	\$14	\$14
7	Adaptador HDMI a VGA	1	\$11	\$11
			Subtotal	\$406
			IVA(12%)	\$48,72
			TOTAL	\$455,72

Elaborado por: El investigador

Empleando la siguiente ecuación se determina el presupuesto total del proyecto de investigación.

$$Presupuesto_{total} = Presupuesto_{diseño} * Salario_{construcción}$$

$$Presupuesto_{total} = 612[dólares] * 455,72[dólares]$$

$$Presupuesto_{total} = 1067,72[dólares]$$

El entorno virtual fue instalado en la computadora de la unidad de fisioterapia por consiguiente no se tomó en cuenta ese rubro para el presupuesto.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Para tratar las lesiones de tobillo en la unidad de rehabilitación el método común es en base al grado de lesión del tobillo, dependiendo de la gravedad se procura tomar reposos y cuidados para impedir movilidad brusca; otra técnica importante es la terapia física realizando ejercicios de estiramiento, fortalecimiento y movimiento tanto en flexión plantar como flexión dorsal, ejercicios que son guiados por el fisioterapeuta, de existir molestias se emplean métodos de electroterapia y crioterapia, sin embargo, conforme avanza la terapia no se tiene un dato cierto en la recuperación del mismo puesto que se toma como referencia el rango de movilidad del tobillo sano. Conllevando en un futuro a presentar nuevamente la lesión, en consecuencia teniendo una condición crónica como lo es la inestabilidad del tobillo.
- El entorno virtual contribuye en la rehabilitación del paciente pues captura los ejercicios realizados y los replica en un avatar que es visualizado en tiempo real en un monitor, se evidencia que la interacción entre el especialista, el usuario y la aplicación virtual desarrollada es entretenida y al no ser invasiva mejora el estado anímico del paciente, disfrutando así de una sesión de rehabilitación funcional divertida puesto que la persona trata de hacer bien los ejercicios mirando al personaje en la pantalla.
- En el presente proyecto se diseñó una interfaz de monitoreo en la cual se van recolectando los datos de los ángulos de movilidad del tobillo en tiempo real según el tiempo designado en la terapia física del paciente, estos datos son obtenidos mediante la cámara y procesados en la minicomputadora Jetson Nano para ser enviados empleando el protocolo MQTT, estos datos son procesados en NodeRed y guardados en InfluxDB para ser visualizados en Grafana, con la utilización de estas herramientas la información presentada es elegante e interactiva.

4.2 Recomendaciones

- Para investigaciones futuras se recomienda realizar el análisis de la articulación de la rodilla debido a que la Unidad de Rehabilitación y Fisioterapia Gabo's se especializa en las articulaciones de tobillo y rodilla, de esta manera se tendrá un sistema completo para la detección de los ángulos de movilidad para las dos articulaciones.
- Para que el avatar realice los movimientos adecuados es recomendable mantener una distancia mínima de 2 metros respecto al sistema, de tal forma que la cámara capte en su totalidad al cuerpo humano sin olvidar que la habitación debe ser amplia y contar con una buena iluminación para tener mayor precisión al realizar el análisis.
- Para mejorar el rango de movilidad del tobillo tanto en flexión plantar como flexión dorsal considerar las observaciones que se presentan en el reporte del paciente, se recomienda analizar el artículo de los profesores en Ortopodología Roberto Jiménez y Agustín Iglesias para poder establecer el ángulo óptimo que se debe obtener en la flexoextensión del tobillo.
- Para observar los datos que se obtienen en las sesiones de rehabilitación desde otro punto que no sea la red de la unidad de fisioterapia se recomienda implementar un servidor en la nube, de esta forma tanto el paciente como el fisioterapeuta pueda acceder a esta información desde la comodidad de sus hogares o desde otro sitio que cuenten con acceso a internet.

BIBLIOGRAFÍA

- [1] L. Culqui, «Inestabilidad articular y esguince de tobillo en pacientes del servicio de medicina física del Hospital Regional Virgen de Fátima, Chachapoyas,» Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas, Chachapoyas, 2022.
- [2] D. Catalán, «Tratamiento de esguince de tobillo grado II en adultos laboralmente activos: Inmovilización contra vendaje funcional,» *Sanidad Militar*, vol. 72, n° 3-4, pp. 240-245, 2018.
- [3] E. Chen, «Ankle Sprains: Evaluation, Rehabilitation, and Prevention,» *Current Sports Medicine Reports*, vol. 18, n° 6, pp. 217-223, 2019.
- [4] G. Changoluisa, «Análisis del tratamiento de esguince de tobillo grado II en jóvenes adultos con inmovilización versus vendaje funcional,» Universidad Central del Ecuador, Quito, 2021.
- [5] INEC, «Archivo Nacional de Datos y Metadatos Estadísticos (ANDA),» 2020. [En línea]. Available: <https://www.ecuadorencifras.gob.ec/camas-y-egresos-hospitalarios/#>. [Último acceso: 24 Enero 2023].
- [6] J. Grajales y N. López, «Identificación y validación de parámetros cinemáticos y factores predictores de esguince de tobillo, mediante un sistema óptico de captura de movimiento durante la marcha,» Universida ECCI, Bogotá, 2019.
- [7] M. Barrios, L. Rodríguez y C. Pachón, «Telerehabilitación funcional en entornos virtuales interactivos como propuesta de rehabilitación en pacientes con discapacidad,» *ESPACIOS*, vol. 40, n° 25, 2019.
- [8] J. Ayerve y C. Rosero, «Desarrollo de un sistema interactivo para entrenamiento y rehabilitación física de personas con discapacidades cognitivas dentro de la Fundación Fine, mediante la utilización del sensor de movimientos Kinect,» Universidad Politécnica Salesiana , Quito, 2018.

- [9] E. Medina y V. Rofes, «Virtual Reality Rehabilitation App (Upper Body),» Universidad Politécnica de Catalunya BarcelonaTech, Barcelona, 2019.
- [10] L. Pedreros y M. Fabregat, «Aplicación de realidad virtual para la rehabilitación,» Universidad Politecnica de Catalunya, Barcelona, 2019.
- [11] J. Jarrín, «Sistema de detección del ángulo articular en los movimientos de miembro superior para evaluación en fisioterapia mediante visión artificial,» Universidad Técnica del Norte, Ibarra, 2020.
- [12] J. Cárdenas y J. Gómez, «Sistema de apoyo de rehabilitación para la medición de los rangos de movilidad de la mano usando visión artificial,» EasyChair Preprint, Pamplona, 2021.
- [13] D. Polit, C. Granizo, M. Rodríguez y C. Sandra, «Terapia de artrosis con células madre. Nuevos avances,» *Recimundo*, vol. 6, nº 2, pp. 630-637, 2022.
- [14] B. Espuche, «Consideraciones éticas y legislación sobre la terapia con células madres,» *Ocronos*, vol. V, nº 10, p. 265, 2022.
- [15] M. Dalmau, F. Malagelada, M. Guelfi y J. Vega, «Anatomía del tobillo,» *ARTROSCOPIA Y CIRUGÍA ARTICULAR*, vol. 27, nº 67, pp. 5-11, 2020.
- [16] K. Zaragoza y S. Fernández, «Ligamentos y tendones del tobillo: anatomía y afecciones más frecuentes analizadas mediante resonancia magnética,» *Anales de Radiología México*, vol. 12, nº 2, pp. 81-94, 2013.
- [17] M. Argueso, M. Campos y E. Iglesias, «ASEPEYO,» 2017. [En línea]. Available: https://www.serme.es/wp-content/uploads/2017/07/guia_clinica_y_imagen.pdf. [Último acceso: 30 Enero 2023].
- [18] H. López, «Medicina de Familia en la Red,» Esguince de tobillo, tratamiento y ejercicios de recuperación, 28 Octubre 2013. [En línea]. Available: <https://www.drlopezheras.com/2013/10/esguince-de-tobillo-tratamiento-y.html>. [Último acceso: 29 Abril 2023].

- [19] Mayo Clinic , «Esguince de tobillo,» 11 Agosto 2022. [En línea]. Available: <https://www.mayoclinic.org/es-es/diseases-conditions/sprained-ankle/symptoms-causes/syc-20353225>. [Último acceso: 30 Enero 2023].
- [20] R. Jiménez y A. Iglesias, «Revisión de la Medición Goniométrica del Tobillo,» *Revista Española de Patología*, vol. I, n° 23, pp. 30-32, 2012.
- [21] J. Valencia, T. Ramirez, L. Castañeda y M. Toro, «Detection of infractions and license plates in motorcycles by means of artificial vision, applied to Intelligent Transport Systems,» *Revista Iberica de Sistemas e Tecnologias de Informacao*, n° 37, pp. 1-15, 30 Abril 2020.
- [22] E. d. I. Fuente y R. Mazaeda, «Visión Artificial, Realidad Virtual y Realidad Aumentada,» Universidad de Valladolid, Valladolid, 2017.
- [23] J. Vivanco, «Sistema de control electrónico para acuarios utilizando tecnologías GSM y VOIP,» Universidad Técnica de Ambato, Ambato, 2017.
- [24] Dario, «Motion Capture,» 16 Marzo 1963. [En línea]. Available: <https://proyectoidis.org/motion-capture/>. [Último acceso: 10 Abril 2023].
- [25] Trigital, «MOCAP, COMO FUNCIONA LA CAPTURA DE MOVIMIENTO Y QUE APLICACIONES TIENE PARA LA INDUSTRIA.,» [En línea]. Available: <https://trigital.es/mocap-como-funciona-la-captura-de-movimiento-y-que-aplicaciones-tiene-para-la-industria/#>. [Último acceso: 10 Abril 2023].
- [26] N. Jiménez, «Evaluación de la plataforma Nvidia Jetson Nano para aplicaciones de visión artificial,» Universidad Autónoma de Madrid, Madrid, 2021.
- [27] Raspberry Pi Foundation, «Raspberry Pi Foundation,» [En línea]. Available: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>. [Último acceso: 21 Julio 2023].
- [28] TECH4ALL, «Revisión de Beelink SER5,» [En línea]. Available: <https://n9.cl/6hrzs>. [Último acceso: 21 Julio 2023].
- [29] L. Rosencrance, «TechTarget,» TechTarget Aplicación Arquitectura, Marzo 2021. [En línea]. Available:

<https://www.techtarget.com/searcharchitecture/definition/software>. [Último acceso: 10 Abril 2023].

- [30] R. Amaguaya, «Análisis comparativo a nivel transaccional de Brokers MQTT (Mosquitto, Mosca, EMQ) con respecto a la disponibilidad en infraestructuras IoT ante ataques DDoS,» Escuela Politécnica Nacional, Quito, 2020.
- [31] E. Borja, «Diseño de una arquitectura usando el protocolo Message Queue Telemetry Transport MQTT) sobre plataformas de bajo coste, para monitorización de procesos Industriales,» Escuela Superior Politécnica de Chimborazo, Riobamba, 2020.
- [32] F. Ferrandez, M. Maciá, M. Platero y D. Silveira, «Plataformas IoT basada en paneles de monitorización (Dashboard,» Universidad de Alicante, Alicante, 2020.
- [33] E. Abril, «Módulo inteligente para la estimulación multisensorial en niños con síndrome de Down del Centro de Rehabilitación Física y Neurológica "Bendiciones" de la ciudad de Ambato,» Universidad Técnica de Ambato, Ambato, 2022.
- [34] Strapp Inc, «Strapp,» ¿Que es una aplicación web?, 30 julio 2019. [En línea]. Available: <https://www.strappinc.com/blog/strapp-datos/que-es-una-aplicacion-web>. [Último acceso: 19 Abril 2023].
- [35] D. Briseño, «SISTEMA ELECTRÓNICO PARA LA DETECCIÓN DE POSICIÓN ANGULAR Y ERGONOMÍA DE CICLISTAS EMPLEANDO VISIÓN ARTIFICIAL,» UNIVERSIDAD TÉCNICA DE AMBATO, Ambato, 2023.
- [36] C. Núñez, D. Chicaiza, A. Larrea y S. Acosta, «Comparación de tecnología Kinect v1, v2 y Azure enfocada en la motricidad infantil,» *Revista Ibérica de Sistemas y Tecnologías de la Información*, nº 43, pp. 338-350, 2021.
- [37] Microsoft, «Microsoft,» Comparación entre Azure Kinect y Kinect para Windows v2, 05 Diciembre 2022. [En línea]. Available:

<https://learn.microsoft.com/es-es/azure/kinect-dk/windows-comparison>.
[Último acceso: 5 Mayo 2023].

- [38] E. Duque, Diferencias entre Kinect V1 y Kinect V2, 5 Febrero 2015. [En línea]. Available: <https://edwinnui.wordpress.com/2015/02/05/diferencias-entre-kinect-v1-y-kinect-v2-2/>. [Último acceso: 5 Mayo 2023].
- [39] E. Pardos, «BaboonLab,» Unreal Engine 4, el motor gráfico que ofrece realismo al máximo, 21 Septiembre 2017. [En línea]. Available: <https://n9.cl/6s9025>. [Último acceso: 28 Junio 2023].
- [40] A. J. «Noobs,» Unity, CryEngine O Unreal, 30 Mayo 2020. [En línea]. Available: <https://noobs.es/unity-cryengine-o-unreal/>. [Último acceso: 28 Junio 2023].
- [41] J. L. Alonso, «Incentro,» ¿Qué es TensorFlow y para qué sirve?, 15 Junio 2022. [En línea]. Available: <https://www.incentro.com/es-ES/blog/que-es-tensorflow>. [Último acceso: 10 Julio 2023].
- [42] J. L. Blasco, «OpenWebinars,» PyTorch 2.0: Innovaciones en el marco de trabajo de Machine Learning, 13 Junio 2023. [En línea]. Available: <https://n9.cl/ye48b>. [Último acceso: 13 Julio 2023].
- [43] H. Rodríguez, «Crehana,» ¿Qué es OpenCV?: ¡Descubre todo acerca de la visión artificial!, 28 Abril 2021. [En línea]. Available: <https://www.crehana.com/blog/transformacion-digital/que-es-opencv/>. [Último acceso: 10 Julio 2023].
- [44] TechVidvan, «TechVidvan,» Human Pose Estimation using OpenCV & Python, [En línea]. Available: <https://techvidvan.com/tutorials/human-pose-estimation-opencv/>. [Último acceso: 29 Junio 2023].
- [45] E. Rodriguez, «Xataka,» Raspberry Pi, 18 Septiembre 2018. [En línea]. Available: <https://www.xataka.com/makers/cero-maker-todo-necesario-para-empezar-raspberry-pi>. [Último acceso: 5 Mayo 2023].

- [46] O. Gonzales, «BricoGeek,» Jetson Nano de NVIDIA, 21 Junio 2019. [En línea]. Available: <https://blog.bricogeek.com/noticias/robotica/todo-lo-que-necesitas-saber-sobre-jetson-nano-de-nvidia/>. [Último acceso: 5 Mayo 2023].
- [47] F. Castañeda, «Topes de Gama,» Mini PC Beelink SER5, 4 Abril 2023. [En línea]. Available: <https://topesdegama.com/ofertas/amazon/ordenador-mini-beelink-ser5-abril-2023>. [Último acceso: 5 Mayo 2023].
- [48] C. Bautista y S. Teneda, «SISTEMA ELECTRÓNICO DE MONITOREO Y CONTROL DE CAPTACIÓN DE ENERGÍA SOLAR Y ELECTROMAGNÉTICA PARA APLICACIONES IoT,» Universidad Técnica de Ambato, Ambato, 2022.
- [49] N. Naik, «Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP,» *IEEE International Systems Engineering Symposium (ISSE)*, pp. 1-7, 2017.
- [50] Influxdata, «Características Telegraf,» Agente de servidor de código abierto , [En línea]. Available: <https://www.influxdata.com/time-series-platform/telegraf/>. [Último acceso: 10 Julio 2023].
- [51] Nodered, «Documentación oficial de Node-RED,» [En línea]. Available: <https://nodered.org/docs/>. [Último acceso: 10 Julio 2023].
- [52] Prometheus, «Prometheus Documentation,» [En línea]. Available: <https://prometheus.io/docs/introduction/overview/>. [Último acceso: 10 Julio 2023].
- [53] A. Robledano, «OpenWebinars,» Qué es mongoDB, 28 Octubre 2019. [En línea]. Available: <https://openwebinars.net/blog/que-es-mongodb/>. [Último acceso: 29 Junio 2023].
- [54] B. Lankheet, «JLCP,» Diagramas de procesos, fluxos e mapas no Grafana com Flowcharting, 25 Junio 2022. [En línea]. Available: <https://www.jlcp.com.br/diagramas-de-processos-fluxos-e-mapas-no-grafana-com-flowcharting/>. [Último acceso: 30 Junio 2023].

- [55] M. Monteagudo, P. Martínez y E. Maceira, «Anatomía funcional, biomecánica y patomecánica de la estabilidad del tobillo,» *Revista del Pie y Tobillo*, vol. 2016, n° 8, pp. 7-16, 2016.
- [56] Fisioterapia, «La importancia de la Dorsiflexión de tobillo para la carrera y el Squat,» Noviembre noviembre 2013. [En línea]. Available: <http://fisioterapia.blogspot.com/2013/11/la-importancia-de-la-dorsiflexion-de.html>. [Último acceso: 28 Junio 2023].
- [57] J. Carranco, F. Salgado, C. Zeas y O. Alvarado, «SISTEMA DE DETECCIÓN Y CLASIFICACIÓN DE POSTURA DE CICLISTAS, BIKE FIT, MEDIANTE KINECT V.2 Y REDES NEURONALES.,» Universidad del Azuay, Cuenca, 2016.
- [58] B. Moore, «Server Space,» Instalación de Grafana en Ubuntu 20.04, 12 Septiembre 2022. [En línea]. Available: <https://serverspace.io/es/support/help/grafana-ubuntu-20-04/>. [Último acceso: 1 Julio 2023].
- [59] M. Delgado, «SMS Ecuador Auditores y Asesores Gerenciales,» Tabla Sectorial de Salarios Mínimos Ecuador 2023, [En línea]. Available: <https://n9.cl/d6nm2>. [Último acceso: 13 Julio 2023].

ANEXOS

Anexo 1. Validación por el Lcdo. Ramiro Gabriel Moreno Proaño

**UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA
GABO'S**



CERTIFICADO

Ambato, 24/07/2023

Ingeniera, Mg
Pilar Urrutia
DECANA
Facultad de Ingeniería en Sistemas, Electrónica e Industrial
Presente

Yo, **Ramiro Gabriel Moreno Proaño**, con cedula de ciudadanía **1804302444** con registro del Senecyt N° 1010-2017-1826645 Licenciado en Terapia Física, certifico que he revisado el proyecto de investigación con el tema: "SISTEMA DE REHABILITACIÓN FÍSICA PARA PACIENTES CON LESIONES DE TOBILLO EMPLEANDO VISIÓN ARTIFICIAL PARA LA UNIDAD DE REHABILITACIÓN Y FISIOTERAPIA GABO'S", el mismo que fue desarrollado en su totalidad por el señor **Gilson German Flores Achachi**, portador de la cedula de ciudadanía **1804966453**, estudiante de la Carrera de **Telecomunicaciones** de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

Es así como después de haber realizado las pruebas necesarias me permito verificar la funcionalidad del sistema y puedo validar que los resultados obtenidos hasta la fecha son confiables y de gran ayuda en la rehabilitación física y monitoreo de pacientes con lesiones de tobillo.

Saludos cordiales.

Atentamente,


.....
Lic. Ramiro Gabriel Moreno Proaño
Fisioterapeuta
1804302444
(03) 2521493
0992970670
gabriel91moreno@hotmail.com
REG: 1010-2017-1826645

Lic. Gabriel Moreno
Fisioterapeuta

Anexo 2. Instalación del Kinect para Windows SDK2.0

Descarga el SDK2.0 y lo abrimos

Kinect for Windows SDK 2.0

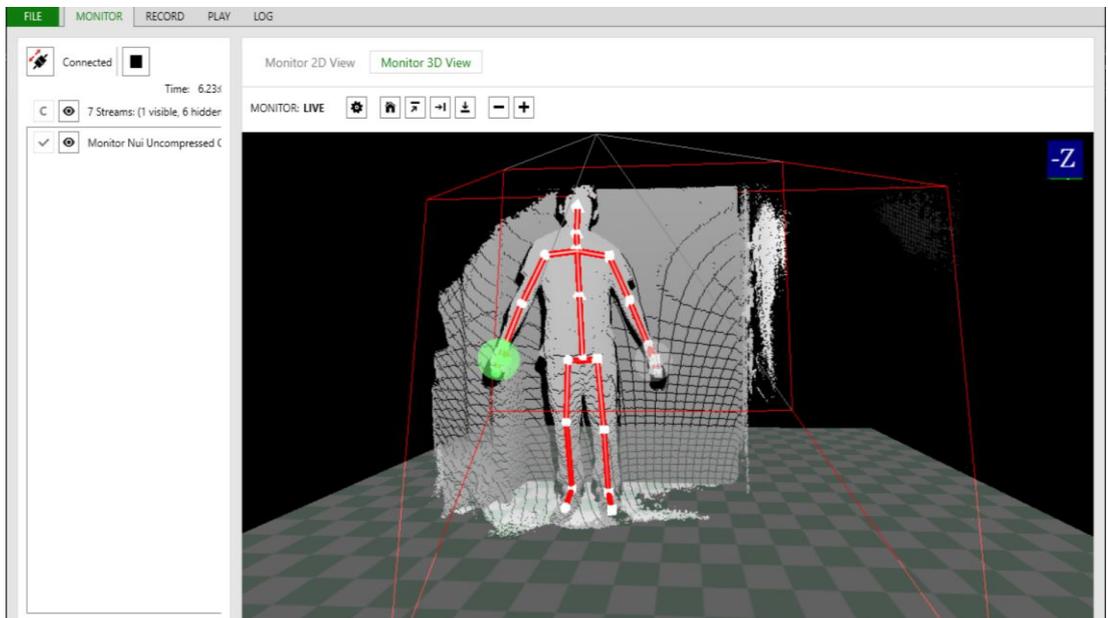
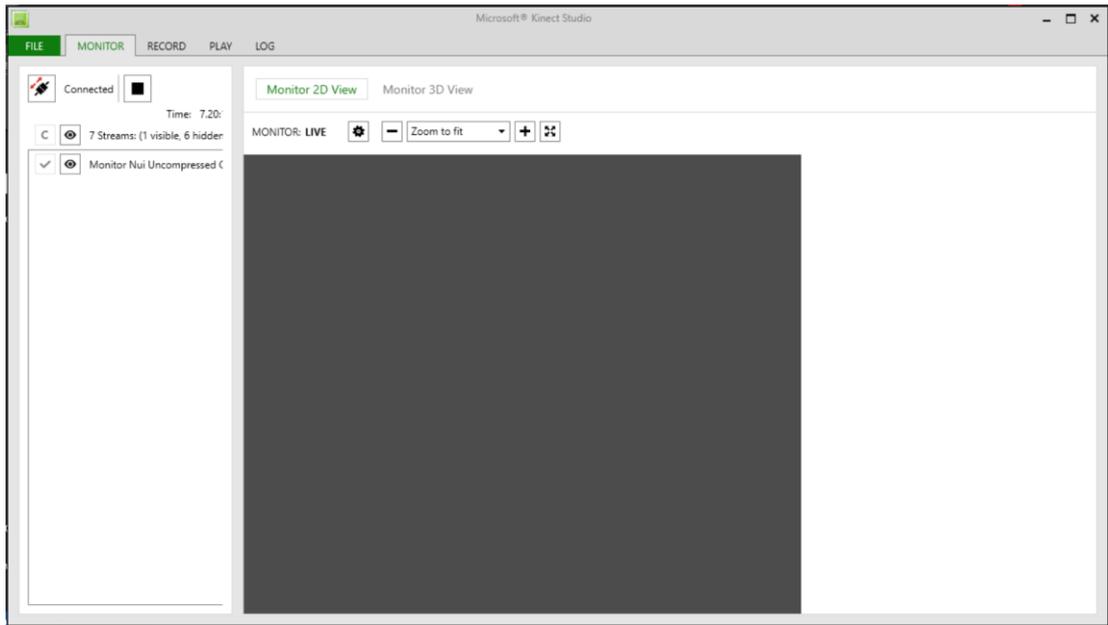
Important! Selecting a language below will dynamically change the complete page content to that language.

Language: **English** [Download](#)

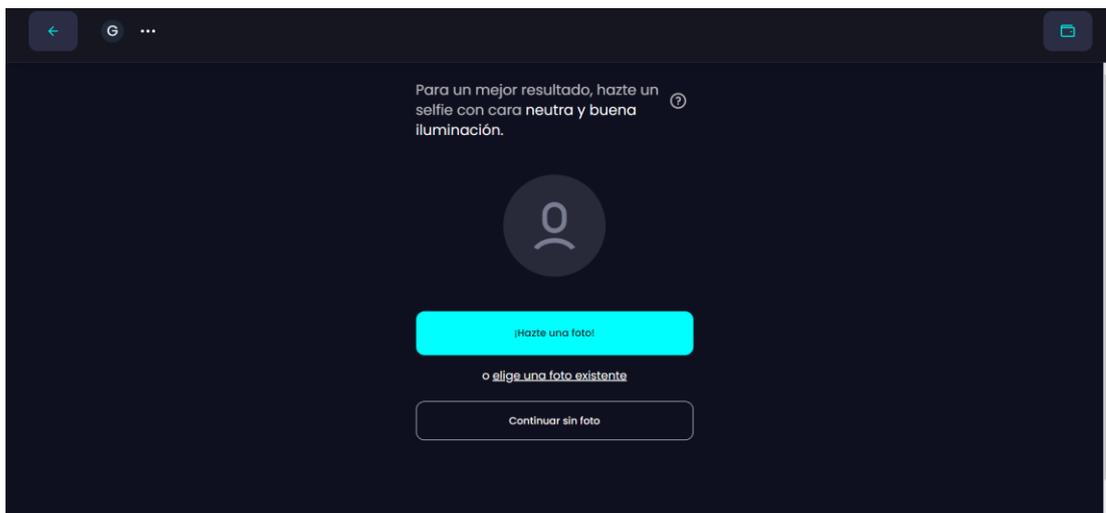
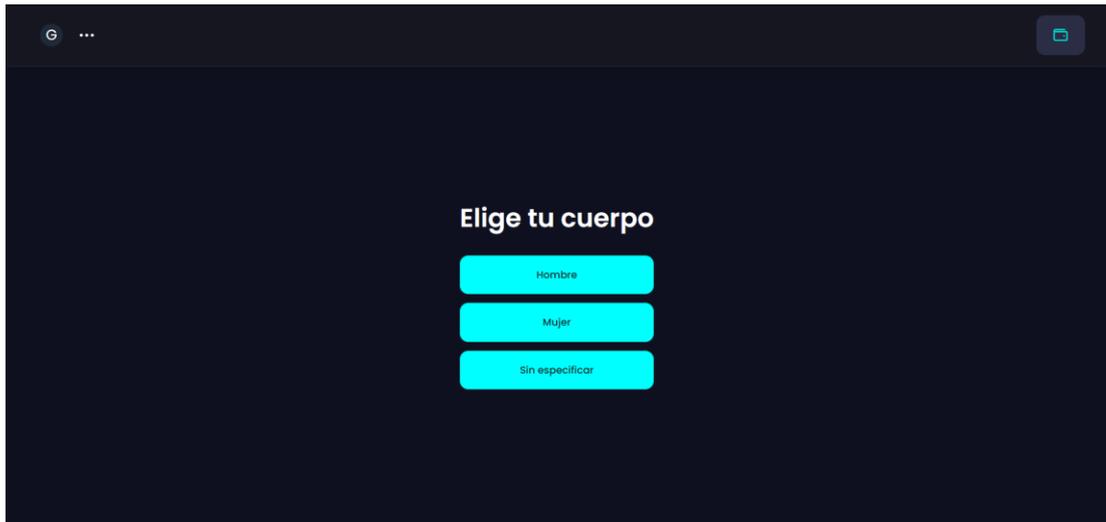
The screenshot shows the 'SDK Browser (Kinect for Windows) v2.0' window. The main header features the 'KINECT for Windows' logo and the text 'Resources & Samples'. Below the header is a navigation menu with options: 'All', 'Components', 'Samples: C#', 'Samples: C++', 'Samples: Windows Store', 'SDKs', and 'Tools'. The main content area displays a list of four items, each with a 'KINECT for Windows' logo, a title, a description, a difficulty level, a language selection dropdown, and a 'Run' button:

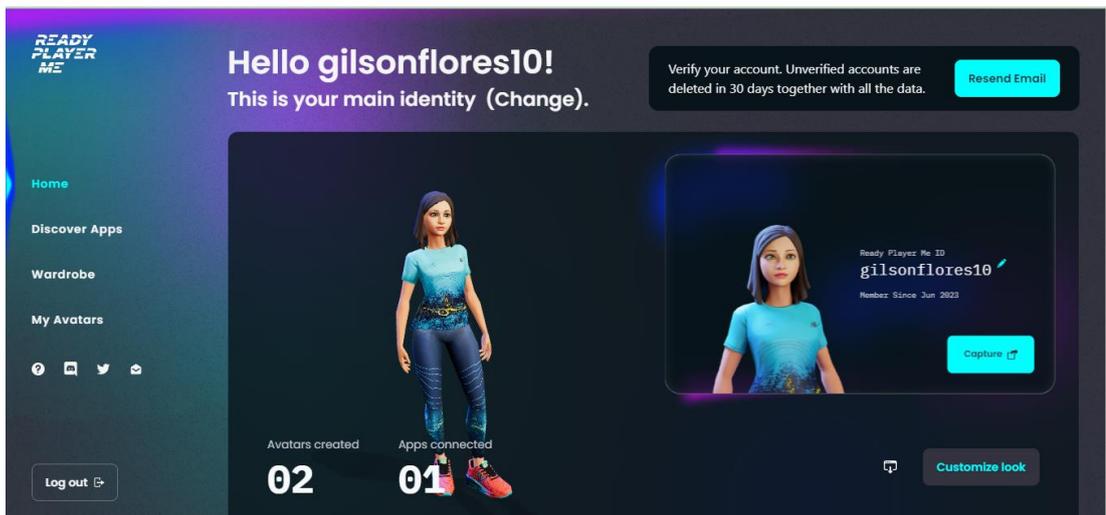
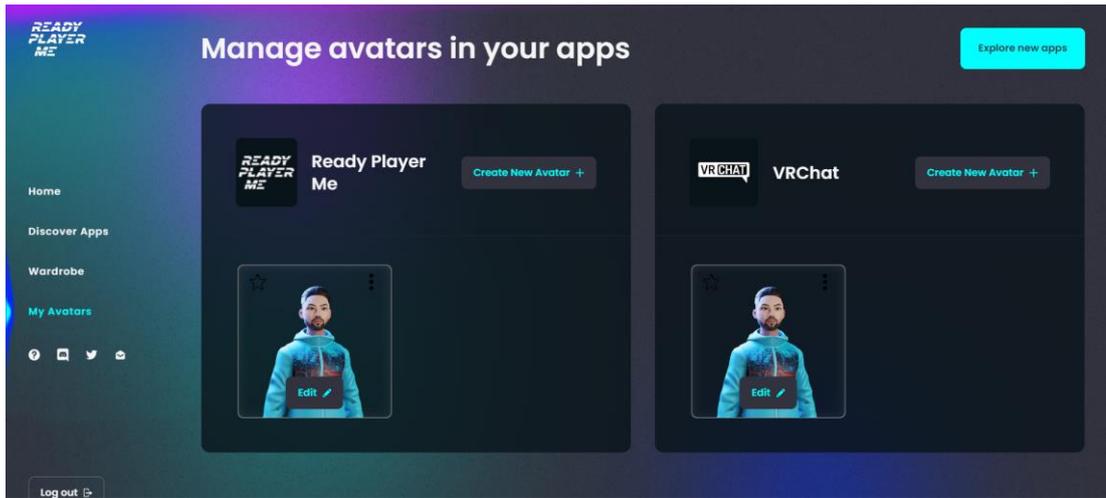
- Kinect Configuration Verifier**: The Kinect Configuration Verifier analyzes your computer for known hardware incompatibilities and verifies communication with the Kinect sensor. Difficulty: Beginner. Language: [dropdown]. Run button.
- Kinect for Windows Documentation**: Documentation for Kinect for Windows. Difficulty: Beginner. Language: [dropdown]. Go To Web Page button.
- Kinect Studio**: Kinect Studio can be used to monitor, record and playback data from the Kinect sensor to enable better testing and debugging of your application. Difficulty: Intermediate. Language: [dropdown]. Run button.
- Visual Gesture Builder - PREVIEW**: Visual Gesture Builder can be used to tag, train and analyze body-based gesture detectors using machine-learning technologies. Difficulty: Advanced. Language: [dropdown]. Run button.

Conectamos el Kinect y presionando el botón de Play se nos descargara el software Kinect Studio para comprobar el funcionamiento del sensor.

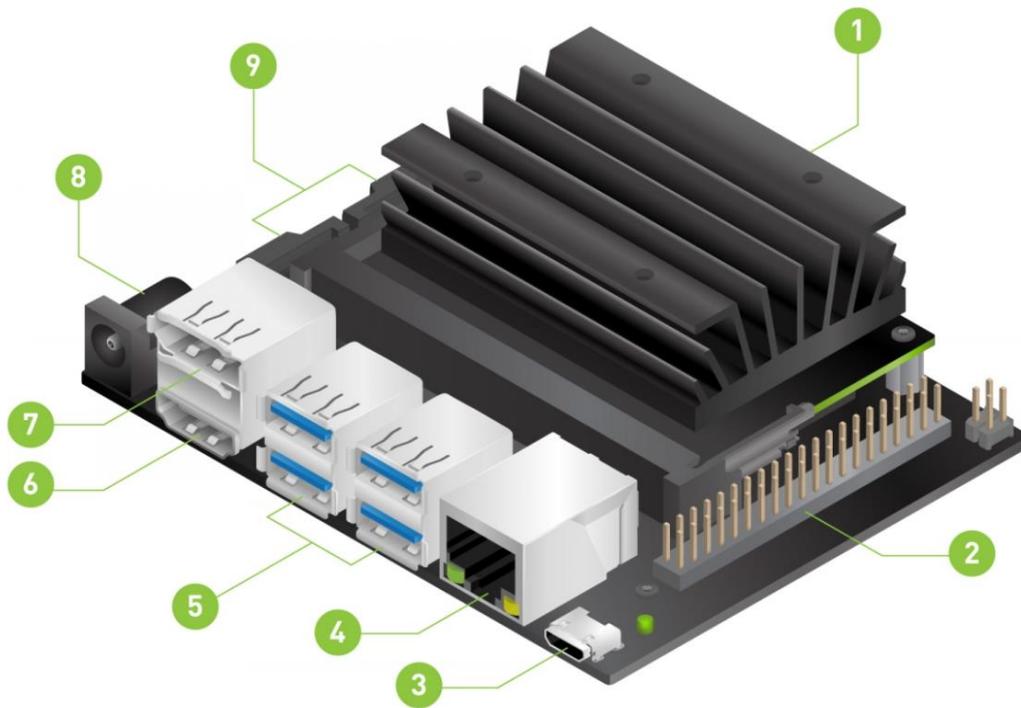


Anexo 3. Creación del avatar



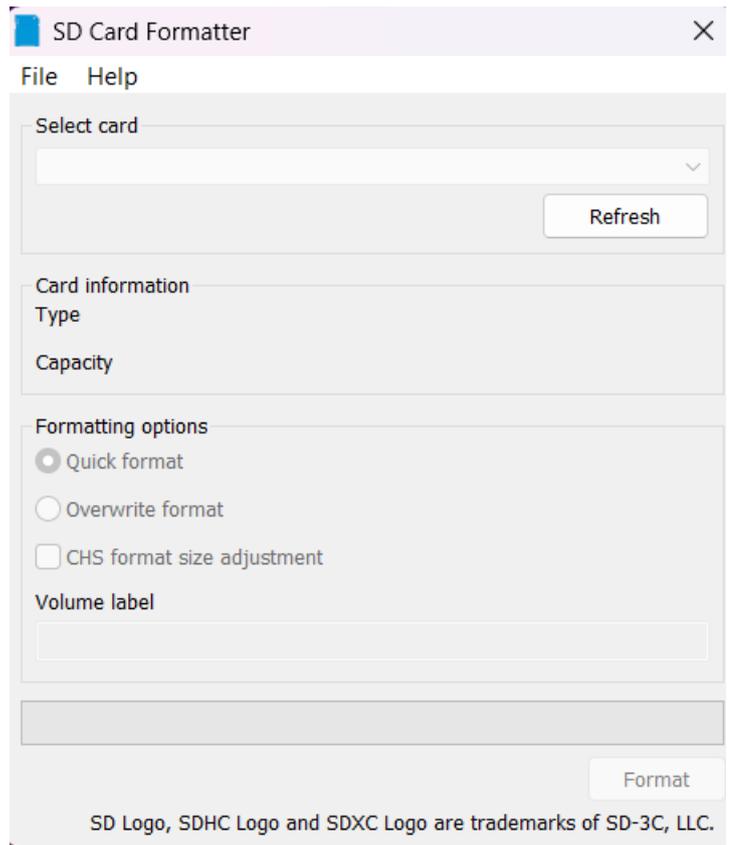


Anexo 4. Instalación del Jetpack

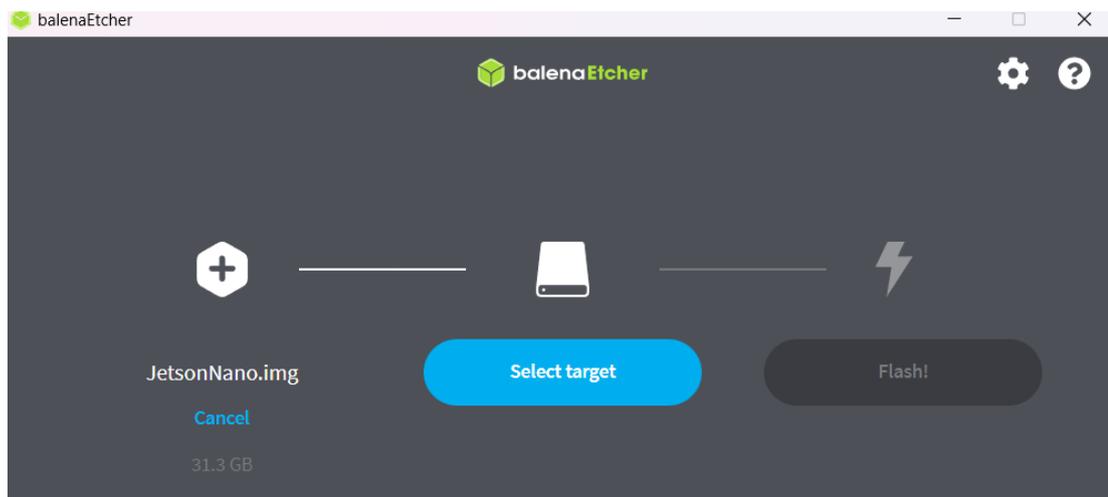


- | | | | |
|---|---|---|---|
| 1 | Ranura para tarjeta microSD para almacenamiento principal | 5 | Puertos USB 3.0 (x4) |
| 2 | encabezado de expansión de 40 pines | 6 | Puerto de salida HDMI |
| 3 | Puerto micro-USB para entrada de alimentación de 5 V o para modo de dispositivo | 7 | Conector DisplayPort |
| 4 | Puerto Ethernet Gigabit | 8 | Conector DC Barrel para entrada de alimentación de 5V |
| | | 9 | Conectores de cámara MIPI CSI-2 |

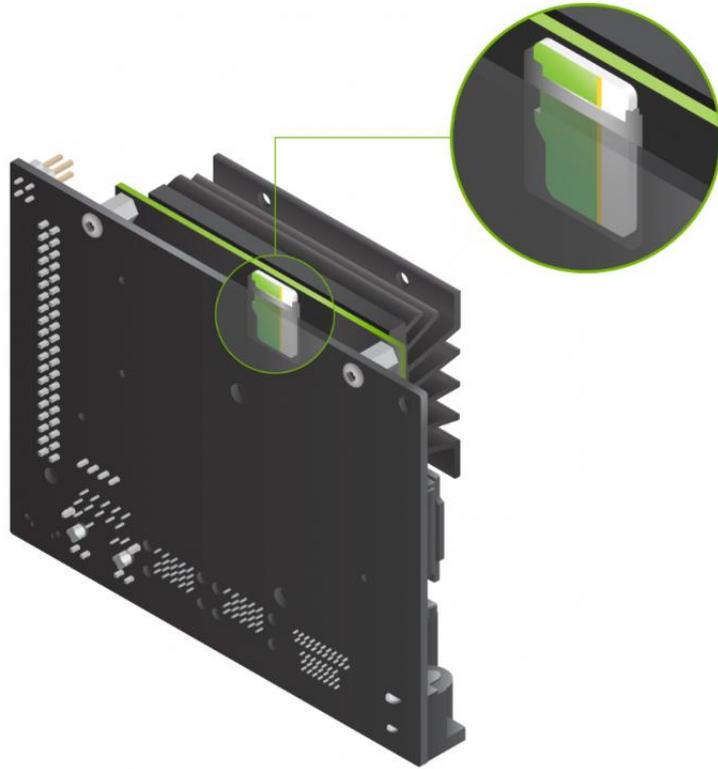
Se recomienda una tarjeta microSD de al menos 32GB y se procede a formatearla para lo cual se utiliza SD MemoryCard Formatter para Windows



Luego se procede a flashear la imagen utilizando Etcher



Se inserta la tarjeta microSD en la abertura destinada como se observa en la siguiente figura



Realizado el anterior paso se procede con la instalación y configuración inicial

1. Revise y acepte el EULA del software NVIDIA Jetson
2. Seleccione el idioma del sistema, la distribución del teclado y la zona horaria
3. Crear nombre de usuario, contraseña y nombre de la computadora
4. Seleccione el tamaño de partición de la aplicación: se recomienda utilizar el tamaño máximo sugerido

Anexo 5. Scripts necesarios para el entorno virtual en lenguaje C#

KinectManager: inicializa, controla el sensor Kinect y captura los datos del cuerpo

```
//inicio de directivas para utilizar clases y funciones
using UnityEngine;
//permite acceder a las clases y funciones proporcionadas por el SDK
de Kinect para Windws.
using Windows.Kinect;
public class BodySourceManager : MonoBehaviour
{
//almacena una referencia al sensor Kinect
    private KinectSensor _Sensor;
//lee los fotogramas de datos de los cuerpos detectados por el
sensor kinect
    private BodyFrameReader _Reader;
//arreglo de objetos, almacena la información de los cuerpos
detectados
    private Body[ ] _Data = null;
//contiene la información de los cuerpos detectados por el sensor
kinect
    public Body[ ] GetData()
    {
        return _Data;
    }
//guarda información sobre el plano del suelo detectado por el
sensor kinect
    public Vector4 FloorPlanes { get; private set; }
//inicializa el sensor Kinect y abre un lector de fotogramas
    void Start()
    {
        _Sensor = KinectSensor.Default();
//si el sensor está disponible se abre el lector de fotogramas y el
sensor Kinect
        if (_Sensor != null)
        {
            _Reader = _Sensor.BodyFrameSource.OpenReader();
            if (!_Sensor.IsOpen)
            {
                _Sensor.Open();
            }
        }
    }
//se obtienen los datos actualizados de los cuerpos detectados por
el sensor kinect
    void Update()
    {
        if (_Reader != null)
        {
            var frame = _Reader.AcquireLatestFrame();
            if (frame != null)
            {
                if (_Data == null)
                {
                    _Data = new Body[_Sensor.BodyFrameSource.BodyCount];
                }
                frame.GetAndRefreshBodyData(_Data);
                FloorPlanes = frame.FloorClipPlane;
            }
        }
    }
}
```

```

//libera el fotograma para evitar fugas de memoria
        frame.Dispose();
        frame = null;
    }
}
}
//se ejecuta cuando la aplicación se cierra para evitar fugas de
memoria
void OnApplicationQuit()
{
    if (_Reader != null)
    {
        _Reader.Dispose();
        _Reader = null;
    }
    if (_Sensor != null)
    {
//verifica si el sensor kinect esta abierto
        if (_Sensor.IsOpen)
        {
//cierra el sensor kinect para liberar recursos
            _Sensor.Close();
        }
        _Sensor = null;
    }
}
}
}
}
}

```

AvatarController: Controla al avatar para que se mueva según los datos capturados por el sensor

```

//inicio de directivas para utilizar clases y funciones
using UnityEngine;
using System;
using System.Collections;
using System.Collections.Generic;
using System.Runtime.InteropServices;
using System.IO;
using System.Text;
public class AvatarController : AvatarController
{
//variables publicas que almacenan las referencias de los objetos
del juego
//Representan los huesos en 3D del humanoide
    public Transform HipCenter;
    public Transform Spine;
    public Transform ShoulderCenter;
    public Transform Neck;

    public Transform ClavicleLeft;
    public Transform ShoulderLeft;
    public Transform ElbowLeft;
    public Transform HandLeft;
    public Transform FingersLeft;
    public Transform ThumbLeft;
}

```

```

    public Transform ClavicleRight;
    public Transform ShoulderRight;
    public Transform ElbowRight;
    public Transform HandRight;
    public Transform FingersRight;
    public Transform ThumbRight;

    public Transform HipLeft;
    public Transform KneeLeft;
    public Transform FootLeft;

    public Transform HipRight;
    public Transform KneeRight;
    public Transform FootRight;
//variable opcional para almacenar una referencia
//se utiliza para controlar el movimiento y rotación del modelo
completo

    [Tooltip("The body root node (optional).")]
    public Transform BodyRoot;

//función protegida para mapear los huesos del humanoide 3D
    protected override void MapBones()
    {
//A cada hueso se le asigna a la posición correspondiente del
arreglo
        bones[0] = HipCenter;
        bones[1] = Spine;
        bones[2] = ShoulderCenter;
        bones[3] = Neck;
        bones[5] = ShoulderLeft;
        bones[6] = ElbowLeft;
        bones[7] = HandLeft;
        bones[11] = ShoulderRight;
        bones[12] = ElbowRight;
        bones[13] = HandRight;
        bones[17] = HipLeft;
        bones[18] = KneeLeft;
        bones[19] = FootLeft;
        bones[21] = HipRight;
        bones[22] = KneeRight;
        bones[23] = FootRight;
        bones[25] = ClavicleLeft;
        bones[26] = ClavicleRight;
        bones[27] = FingersLeft;
        bones[28] = FingersRight;
        bones[29] = ThumbLeft;
        bones[30] = ThumbRight;
        bodyRoot = BodyRoot;
    }
}

```

Anexo 6. Scripts del menú del entorno virtual

Guardar el personaje que se elige

```
//inicio de directivas para utilizar clases y funciones
using UnityEngine;
public class GuardarPersonaje : MonoBehaviour
{
//variables publicas/ visibles en el inspector de Unity
    public bool hombre;
    public bool mujer;
//se ejecuta en cada frame del juego
    private void Update() {
//verifica la selección del personaje
        if(hombre ==false && mujer == false)
        {
//por defecto se selecciona el personaje hombre
            hombre = true;
        }
//permite cargar el personaje que se ha elegido previamente
        hombre = PlayerPrefs.GetInt("hombreselect") == 1;
        mujer = PlayerPrefs.GetInt("mujerselect") == 1;
    }
//se selecciona el personaje mujer
    public void personajeMujer(){

        hombre = false;
        mujer = true;
        guardar(); //almacena en las PlayerPrefs
    }
//se selecciona el personaje hombre
    public void personajeHombre(){

        hombre = true;
        mujer = false;
        guardar();//almacena en las PlayerPrefs
    }

//función que guarda la selección del personaje hombre/mujer ->
1/0
    public void guardar(){

        PlayerPrefs.SetInt("hombreselect", hombre ? 1 :0);
        PlayerPrefs.SetInt("mujerselect", mujer ? 1 :0);

    }
}
```

Carga el personaje en la escena después de presionar le botón de inicio

```
//inicio de directivas para utilizar clases y funciones
using UnityEngine;
public class CargarPersonaje : MonoBehaviour
{
//se almacenan los objetos del juego en las variables publicas
    public GameObject Man;
    public GameObject Girl;
//variables para determinar si el personaje elegido es hombre o
mujer
    public bool hombre;
    public bool mujer;
    private void Update() {
//obtienen los valores almacenados en las PlayerPrefs
        hombre = PlayerPrefs.GetInt("hombreselect") == 1;
        mujer = PlayerPrefs.GetInt("mujerselect") == 1;
//selección del personaje hombre
        if(hombre == true)
        {
            Man.SetActive(true);
            Destroy(Girl);
        }
//selección del personaje mujer
        if(mujer == true)
        {
            Girl.SetActive(true);
            Destroy(Man);
        }
    }
}
```

Boton que inicia el entorno cargando al personaje elegido en el paso anterior

```
//inicio de directivas para utilizar clases y funciones
using UnityEngine; //namespace

using UnityEngine.SceneManagement; //acceso al namespace

public class Boton : MonoBehaviour
{
    public int numeroEscena; //será visible en el inspector de
unity
    public void iniciar()//llama a la función
    {
        SceneManager.LoadScene(numeroEscena);
    }
}
```

Boton para salir del entorno, se presiona y cierra la ventana

```
//inicio de directivas para utilizar clases y funciones
using UnityEngine;
public class Salir : MonoBehaviour
{
    //Se encarga de cerrar la aplicación del juego

    public void ExitGame()
    {
        //Comprueba si el juego se está ejecutando

        #if UNITY_EDITOR
            UnityEditor.EditorApplication.isPlaying = false;

        //Si el juego no está ejecutándose se aplica la condición

        #else
            Application.Quit();

        //Permite ejecutar diferentes bloques

        #endif
    } //Fin de la función cerrar el juego en Unity
}
```

Anexo 7. Algoritmo para calcular el ángulo de las articulaciones en lenguaje Python

```
# Bibliotecas necesarias para el proyecto
import cv2
import mediapipe as mp
import numpy as np
import paho.mqtt.client as mqtt

# Simplificar el código con un alias
mp_drawing = mp.solutions.drawing_utils
mp_pose = mp.solutions.pose

# Configuración de MQTT
mqtt_broker = "192.168.137.190"
mqtt_port = 1883
mqtt_topic_left = "angle/ankle/left"
mqtt_topic_right = "angle/ankle/right"
mqtt_topic_estado = "estado/reporte"

# Instancia para conectarse y comunicarse con el bróker MQTT
mqtt_client = mqtt.Client()

# Variable contador
contador = 0

# Cálculos de los ángulos del tobillo
def calculate_angle(p, m, f):

    # primero (Rodilla)
    p = np.array(p)

    # medio (Tobillo)
    m = np.array(m)

    # final (Punta del pie)
    f = np.array(f)

    radians = np.arctan2(f[1] - m[1], f[0] - m[0]) -
               np.arctan2(p[1] - m[1], p[0] - m[0])
    angulo = np.abs(radians * 180.0 / np.pi)

    #restriccion

    if angulo > 180.0:
        angulo = 360 - angulo

    return angulo

# Ubicamos la cámara conectada a la Jetson Nano
captura = cv2.VideoCapture(1)

# DETECTION = Valor mínimo de confianza de detección
# TRACKING = Valor mínimo de confianza de movimiento

# Para liberar correctamente los recursos al finalizar la detección
de la postura
with mp_pose.Pose(
    min_detection_confidence=0.5, min_tracking_confidence=0.5) as pose:
```

```

# Evento de conexión al bróker MQTT

def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Conectado al broker MQTT")
    else:
        print("Error de conexión al broker MQTT")

# Conexión al broker MQTT utilizando la dirección IP y puerto
mqtt_client.on_connect = on_connect
mqtt_client.connect(mqtt_broker, mqtt_port)
mqtt_client.loop_start()

# Bucle principal, se ejecuta cuando la camara este abierta
# Captura los fotogramas de la camara y se procesa utilizando
MediaPipe para detectar las posturas

while captura.isOpened():
    ret, frame = captura.read()

    # Cambia el color de la imagen a RGB
    image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    image.flags.writeable = False

    # Realiza la detección
    results = pose.process(image)

    # Volver al color BGR
    image.flags.writeable = True
    image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)

    # Se extraen los puntos de referencia
    try:
        landmarks = results.pose_landmarks.landmark

    # Obtener las coordenadas

    # Puntos para obtener el ángulo del tobillo izquierdo
    knee_left= [landmarks[mp_pose.PoseLandmark.LEFT_KNEE.value].x,
                landmarks[mp_pose.PoseLandmark.LEFT_KNEE.value].y]
    ankle_left = [landmarks[mp_pose.PoseLandmark.LEFT_ANKLE.value].x,
                 landmarks[mp_pose.PoseLandmark.LEFT_ANKLE.value].y]
    foot_left= [landmarks[mp_pose.PoseLandmark.LEFT_FOOT_INDEX.value].x,
               landmarks[mp_pose.PoseLandmark.LEFT_FOOT_INDEX.value].y]

    # Puntos para obtener el ángulo del tobillo derecho
    knee_right = [landmarks[mp_pose.PoseLandmark.RIGHT_KNEE.value].x,
                 landmarks[mp_pose.PoseLandmark.RIGHT_KNEE.value].y]
    ankle_right = [landmarks[mp_pose.PoseLandmark.RIGHT_ANKLE.value].x,
                  landmarks[mp_pose.PoseLandmark.RIGHT_ANKLE.value].y]
    foot_right=[landmarks[mp_pose.PoseLandmark.RIGHT_FOOT_INDEX.value].x,
               landmarks[mp_pose.PoseLandmark.RIGHT_FOOT_INDEX.value].y]

    # Calcular el ángulo del tobillo izquierdo
    ankle_lefti = round(calculate_angle(knee_left, ankle_left,
                                       foot_left), 2)
    angle_ankle_left = 180 - ankle_lefti
    # Visualizar el ángulo del tobillo izquierdo
    cv2.putText(image, str(angle_ankle_left),
                tuple(np.multiply(ankle_left, [640, 480]).astype(int)),

```

```

        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2,
cv2.LINE_AA)

        # Calcular el ángulo del tobillo derecho
ankle_rightd = round(calculate_angle(knee_right, ankle_right,
        foot_right), 2)
angle_ankle_right = 180 - ankle_rightd
        # Visualizar el ángulo del tobillo derecho
cv2.putText(image, str(angle_ankle_right),
        tuple(np.multiply(ankle_right, [640, 480]).astype(int)),
        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255), 2,
cv2.LINE_AA)

        # Publicar los ángulos del tobillo por MQTT
mqtt_client.publish(mqtt_topic_left, str(angle_ankle_left))
mqtt_client.publish(mqtt_topic_right, str(angle_ankle_right))

        # Imprimir los datos en la consola
print("Ángulo del tobillo izquierdo:", angle_ankle_left)
print("Ángulo del tobillo derecho:", angle_ankle_right)
#flexion dorsal 20-30
#flexion plantar 40-50
##### TOBILLO IZQUIERDO #####

# Condicionales para generar el reporte del paciente tanto en
flexión plantar como flexión dorsal

    if contador < 20:
        if 60 <= angle_ankle_left <= 70:
            mqtt_client.publish(mqtt_topic_estado, 'flexión
dorsal I: correcta')

            if angle_ankle_left < 60:
                mqtt_client.publish(mqtt_topic_estado, 'flexión
dorsal I: Aumentar ejercicio de dorsiflexión')
            if 120 <= angle_ankle_left <= 140:
                mqtt_client.publish(mqtt_topic_estado, 'flexión
plantar I: correcta')
            if (90 < angle_ankle_left < 120):
                mqtt_client.publish(mqtt_topic_estado, 'flexión
plantar I: Aumentar ejercicio de plantiflexion')
            ##### TOBILLO DERECHO #####
            if 60 <= angle_ankle_right <= 70:
                mqtt_client.publish(mqtt_topic_estado, 'flexión
dorsal D: correcta')
            if angle_ankle_right < 60:
                mqtt_client.publish(mqtt_topic_estado, 'flexión
dorsal D: Aumentar ejercicio de dorsiflexión')
            if 120 <= angle_ankle_right <= 140:
                mqtt_client.publish(mqtt_topic_estado, 'flexión
plantar D: correcta')
            if (90 < angle_ankle_right < 120):
                mqtt_client.publish(mqtt_topic_estado, 'flexión
plantar D: Aumentar ejercicio de plantiflexion')
            contador +=1

    except:
        pass

```

```

        # Procesar las detecciones
        mp_drawing.draw_landmarks(image, results.pose_landmarks,
        mp_pose.POSE_CONNECTIONS, mp_drawing.DrawingSpec(color=(128, 0,
        250), thickness=2, circle_radius=3), mp_drawing.DrawingSpec(color=(255
        , 255, 255), thickness=2))
        # Se muestra la imagen procesada con los ángulos superpuestos en
        una ventana llamada "Ángulos del tobillo"

        cv2.imshow('Ángulos del tobillo', image)

# Al presionar la tecla s el bucle se detiene y la cámara se cierra

        if cv2.waitKey(10) & 0xFF == ord('s'):
            break

# Se liberan los recursos asociados con la cámara
# Se cierran todas las ventanas
# Se desconecta del bróker MQTT

cap.release()
cv2.destroyAllWindows()
mqtt_client.loop_stop()
mqtt_client.disconnect()

```