

## PROYECTO CAPSTONE

Laboratorio remoto para cursos de física básica  
(PLANO INCLINADO)

## PARTICIPANTES

Eduardo Rodríguez Martínez (Representante)

Fernando Ramírez Rojas

## RESPONSABLE DEL CURSO

Hugo Vargas

Marzo, 2022

# CONTENIDO

1. RESUMEN .....	3
2. OBJETIVO GENERAL .....	3
3. OBJETIVOS ESPECIFICOS .....	3
4. JUSTIFICACION .....	4
5. DESCRIPCION DEL PROYECTO .....	5
5.1 USUARIO [CLIENTE MQTT] .....	7
5.2 BROKER PUBLICO .....	12
5.3 LABORATORIO REMOTO [CLIENTE MQTT] .....	14
5.3.1 ELEMENTOS DEL LABORATORIO REMOTO .....	16
5.4 MICROCONTROLADOR (5) .....	20
5.5 COMPUTADORA (4): DIAGRAMA DE CLASES .....	21
6. LISTA DE MATERIALES .....	22
7. CONCLUSIONES .....	23
8. REFERENCIAS .....	24

## 1. RESUMEN

En este proyecto se propone la realización de un experimento de física básica: el plano inclinado sobre el cual se desliza una esfera de vidrio.

El experimento del plano inclinado podrá ser observado de forma remota mediante un servidor de video. El usuario podrá ajustar el ángulo de inclinación a través de una interfáz gráfica, en donde también se desplegarán los resultados del experimento mediante curvas de desplazamiento, velocidad y aceleración.

La detección del movimiento de la esfera sobre el plano se realizará con técnicas de visión por computadora sobre cada una de las tramas de video capturado con una cámara web.

## 2. OBJETIVO GENERAL

- Desarrollar un prototipo de laboratorio a distancia (Laboratorio Remoto) con experimentos de bajo costo para instituciones educativas nacionales; que permita reanudar las actividades experimentales suspendidas por la pandemia por COVID-19.

## 3. OBJETIVOS ESPECIFICOS

1. Construir un prototipo de plano inclinado sobre el cual se deslice una esfera
2. Detección del movimiento de una esfera mediante técnicas de visión por computadora.
3. Medición de parámetros asociados con el movimiento de la esfera
4. Diseño e implementación de una interfáz de usuario para el control del ángulo de inclinación y visualización de los resultados del experimento.

## 4. JUSTIFICACION

Los laboratorios juegan un papel esencial en la formación de los estudiantes de ciencias básicas e ingenierías, pero debido al confinamiento, producto de la pandemia por COVID-19, estos se han suspendido de forma indeterminada.

Una de las alternativas que permite a los alumnos obtener la formación experimental son los laboratorios a distancia. Estos se presentan generalmente en dos modalidades: i) aquellos que usan la realidad virtual para simular los experimentos (Laboratorio Virtual) [1] y ii) aquellos en los que el estudiante puede interactuar con el experimento físico mediante una plataforma web (Laboratorio Remoto) [2], [3].

La desventaja de los laboratorios que usan realidad virtual es que este tipo de laboratorios no capturan completamente los efectos físicos o químicos de los experimentos, ya que se basan en fórmulas teóricas para su simulación. Por otro lado, la segunda modalidad de los laboratorios a distancia se ha implementado por compañías que ofrecen sus servicios mediante licencias y suscripciones de alto costo, limitando su aplicación a las instituciones que los pueden pagar.

Este proyecto pretende desarrollar un prototipo de laboratorio a distancia (Laboratorio Remoto) con experimentos de bajo costo para instituciones educativas nacionales.

Una innovación que se presenta en este proyecto es el uso de un algoritmo computacional para el “reconocimiento de formas y colores”, para la detección y medición de parámetros de una esfera que se desliza sobre el plano inclinado, a diferencia de proyectos similares que utilizan “sensores de posición” para detectar el desplazamiento, velocidad y aceleración del móvil.

## 5. DESCRIPCION DEL PROYECTO

El diagrama a bloques general del proyecto se muestra en la Figura 1.

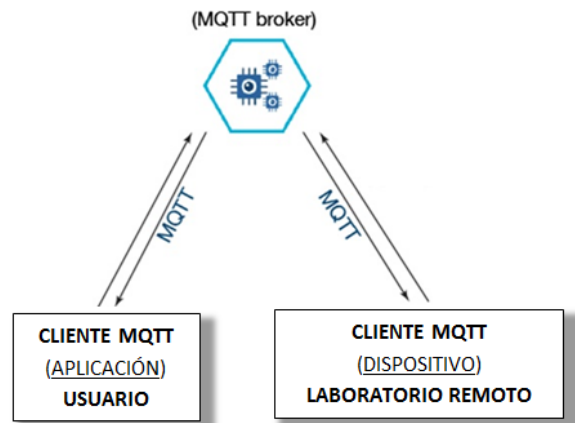


Figura 1. Diagrama a bloques general del proyecto

De acuerdo a este diagrama a bloques, el **USUARIO** puede acceder al **LABORATORIO REMOTO** a través de una estructura de **comunicación por Internet** mediante el concepto de la IoT, utilizando un **bróker** público mediante el protocolo **MQTT**.

La actividad del usuario se reduce a seleccionar un determinado ángulo de inclinación del plano inclinado y, presionando un botón, enviar este parámetro al laboratorio remoto a través del bróker público.

En el laboratorio remoto se encuentra el sistema formado por el plano inclinado, una esfera y una cámara web. El plano inclinado se coloca automáticamente con el ángulo de inclinación solicitado por el usuario, la esfera inicia su desplazamiento y una cámara web captura todo el proceso del experimento para, posteriormente, devolver al usuario los resultados de dicho experimento.

En la figura 2 se muestra el diagrama a bloques detallado del proyecto.

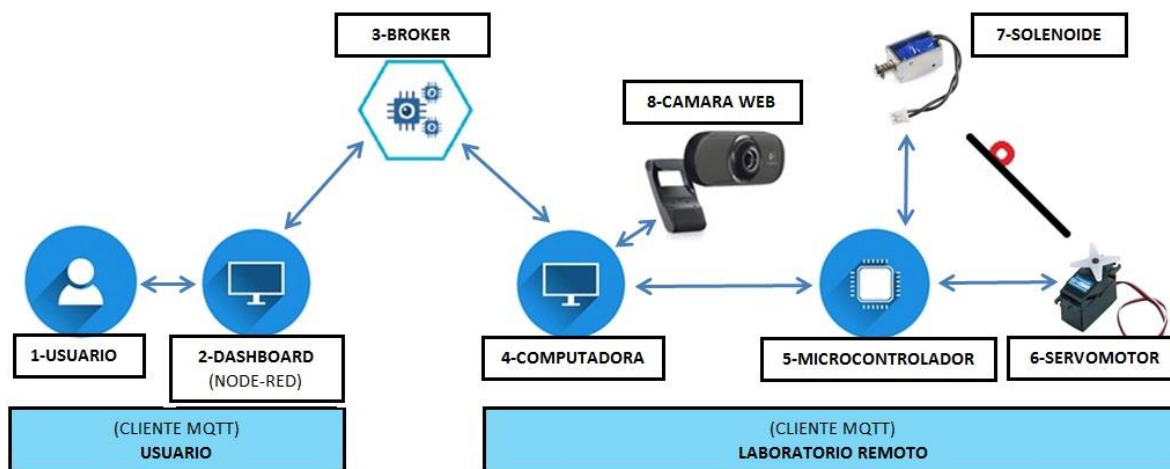


Figura 2. Diagrama a bloques detallado del proyecto

En la figura anterior se pueden observar los siguientes módulos:

**1-USUARIO.-** El usuario participa en las siguientes tres funciones:

- a. Controla de manera remota el ángulo de inclinación del plano inclinado
- b. Recibe visualmente la información sobre el estado del experimento
- c. Recibe visualmente los resultados del experimento.

**2-DASHBOARD.-** En este bloque se encuentra una computadora corriendo NODE-RED y presentando al **USUARIO (1)** un dashboard con la información visual necesaria para realizar las funciones de controlar la inclinación del plano inclinado y recibir los resultados del experimento realizado en el **LABORATORIO REMOTO**.

**3-BROKER.-** Este módulo se encarga de comunicar bidireccionalmente a los módulos del **2-DASHBOARD (2)** y del **LABORATORIO REMOTO**, a través del bloque **COMPUTADORA (4)**, a través de un **BROKER (3)** público mediante el protocolo MQTT, dentro del contexto de la IoT.

**4-COMPUTADORA.-** En la computadora se encuentra activo un programa en python, el cual se encarga de realizar varias funciones como son: establecer la comunicación bidireccional con el **BROKER (3)** público, enviar la información necesaria al **MICROCONTROLADOR (5)** para llevar a cabo el experimento y procesar la información recibida a través de la **CAMARA WEB (8)**.

**5-MICROCONTROLADOR.-** El programa residente en el microcontrolador se encarga de manejar un **SERVOMOTOR (6)** y un **SOLENOIDE (7)**. Ambos dispositivos se encuentran acoplados a los extremos del plano inclinado.

**6-SERVOMOTOR.-** El servomotor se encarga de controlar la inclinación del plano inclinado.

**7-SOLENOIDE.-** El solenoide se encarga de capturar y liberar la esfera que se deslizará sobre el plano inclinado.

**8-CAMARA WEB.-** La cámara WEB se encarga de capturar los tramas de video de todo el experimento con el objetivo de que el programa residente en la computadora del laboratorio remoto, pueda procesar dichas imágenes.

## 5.1 USUARIO [CLIENTE MQTT]

El **USUARIO (1)** como se explicó en la sección anterior, participa en el (a) control remoto del ángulo de inclinación del plano inclinado, y recibe, a su vez, (b) la información sobre el estado del experimento y (c) los resultados del experimento, todo esto de manera gráfica a través de un **DASHBOARD (2)**, como se muestra en la figura 3.

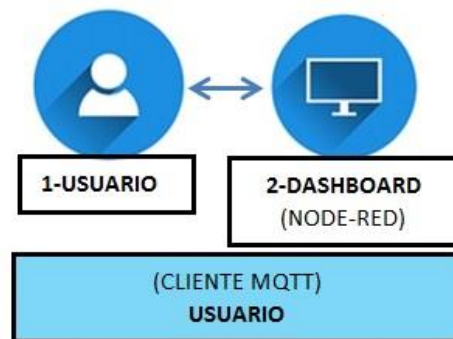


Figura 3 . USUARIO [CLIENTE MQTT]

En la computadora que se muestra en la figura 3, se encuentra corriendo NODE-RED para presentarle al **USUARIO (1)** un **DASHBOARD (2)** con la información visual necesaria para realizar las funciones de controlar la inclinación del plano inclinado (a) y recibir información tanto del estado del experimento (b), como de los resultados del mismo (c), realizados en el **LABORATORIO REMOTO**, como se muestra en la figura 4.

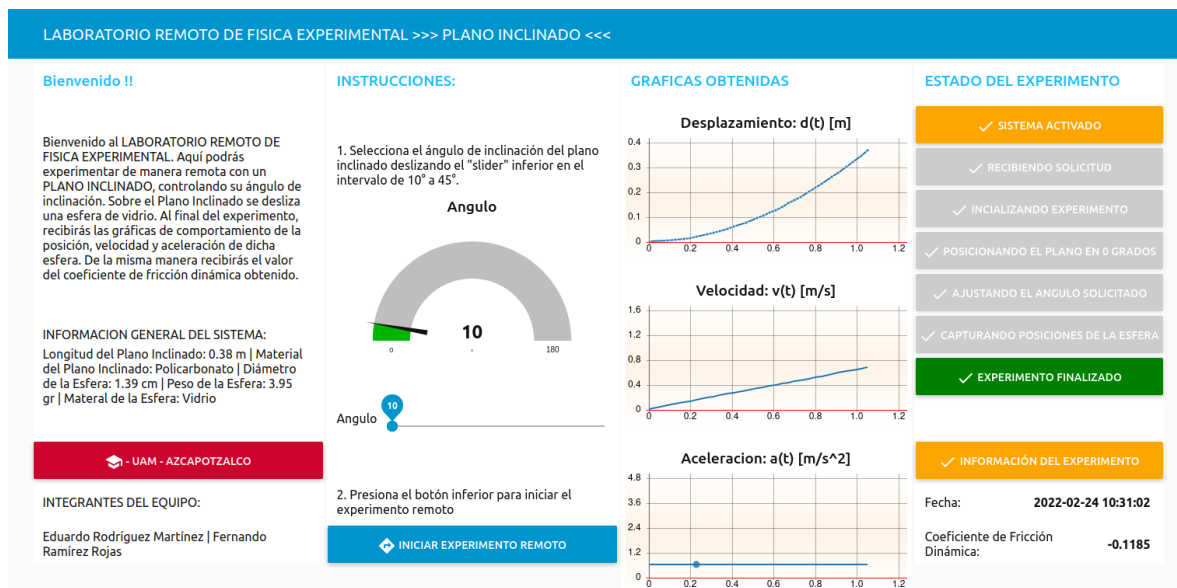


Figura 4 . DASHBOARD generado en Node-Red

El **DASHBOARD (2)** mostrado en la figura 4, está formado por cuatro columnas:

La primera columna presenta un mensaje de bienvenida, indicando además en qué consiste el experimento y el tipo de resultado que se espera obtener. Asimismo se da una breve descripción de los elementos que forman el experimento, así como los nombres de los integrantes del proyecto, como se observa en la figura 5

## Bienvenido !!

Bienvenido al LABORATORIO REMOTO DE FÍSICA EXPERIMENTAL. Aquí podrás experimentar de manera remota con un PLANO INCLINADO, controlando su ángulo de inclinación. Sobre el Plano Inclinado se desliza una esfera de vidrio. Al final del experimento, recibirás las gráficas de comportamiento de la posición, velocidad y aceleración de dicha esfera. De la misma manera recibirás el valor del coeficiente de fricción dinámica obtenido.

### INFORMACION GENERAL DEL SISTEMA:

Longitud del Plano Inclinado: 0.38 m | Material del Plano Inclinado: Policarbonato | Diámetro de la Esfera: 1.39 cm | Peso de la Esfera: 3.95 gr | Material de la Esfera: Vidrio

 - UAM - AZCAPOTZALCO

### INTEGRANTES DEL EQUIPO:

Eduardo Rodríguez Martínez | Fernando Ramírez Rojas

Figura 5 . Primera columna del DASBOARD



Figura 6. Segunda columna del DASHBOARD

En la segunda columna se encuentran las instrucciones para ajustar el control deslizable que permite establecer el ángulo de inclinación del plano inclinado, y el botón para enviar dicha solicitud al LABORATORIO REMOTO (vía MQTT a través del **BROQUER (3)**), como se muestra en la figura 6.



En la tercera columna se presentan los resultados del experimento mediante 3 gráficas:

La gráfica superior corresponde al desplazamiento de la esfera en función del tiempo:  **$d(t)$  vs.  $t$** .

La gráfica de la parte media corresponde a la derivada de la gráfica anterior, es decir, a la velocidad de la esfera en función del tiempo:  **$v(t)$  vs.  $t$** .

Finalmente, la gráfica inferior corresponde a la segunda derivada del desplazamiento, es decir, a la aceleración de la esfera:  **$a(t)$  vs.  $t$** .

#### GRAFICAS OBTENIDAS

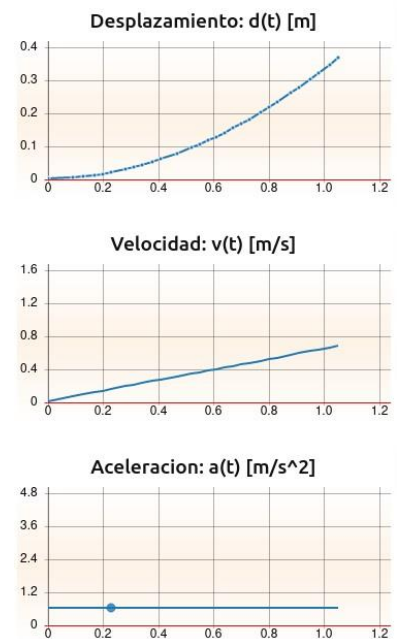


Figura 7. Tercera columna del DASHBOARD



Figura 8 . Cuarta columna del DASHBOARD

En la cuarta columna se presentan dos secciones. En la parte superior se muestra un conjunto de botones informativos sobre el estado por los que va transitando el experimento. El primero botón indica si el sistema está activado, seguido de los botones indicadores de cada uno de los estado del sistema:

- **RECIBIENDO SOLICITUD**
- **INICIALIZANDO EXPERIMENTO**
- **POSICIONANDO EL PLANO EN CERO GRADOS**
- **AJUSTANDO EL ANGULO SOLICITADO**
- **CAPTURANDO POSICIONES DE LA ESFERA**
- **EXPERIMENTO FINALIZADO**

En la parte inferior se muestra la sección correspondiente a la información del experimento, en donde se reporta tanto la fecha de realización del mismo como el coeficiente de fricción dinámica obtenido.

El FLOW correspondiente al **DASHBOARD (2)** descrito anteriormente se muestra en la figura 9

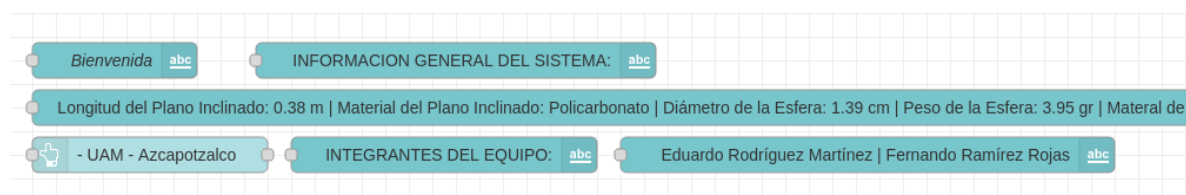


Figura 9a . Flow correspondiente a la bienvenida, descripción e integrantes del proyecto

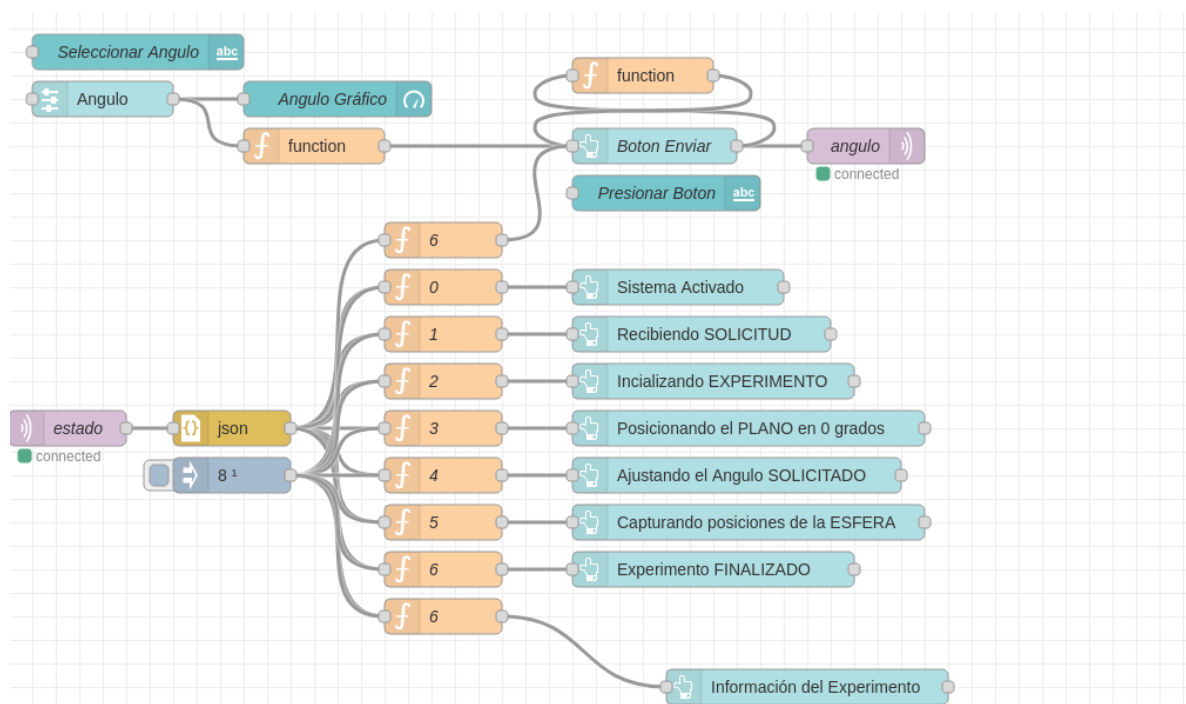


Figura 9b . Flow correspondiente a la selección del ángulo de inclinación y el estado del experimento

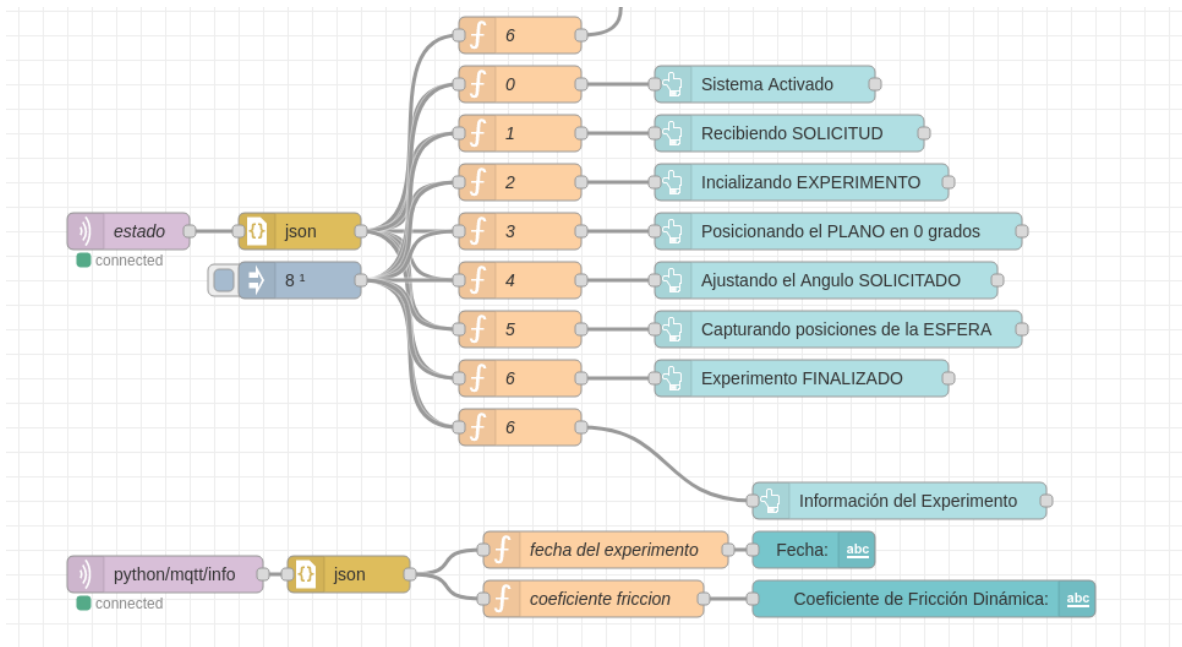


Figura 9c . Flow correspondiente a la información del experimento: la fecha y el valor del coeficiente de fricción dinámica

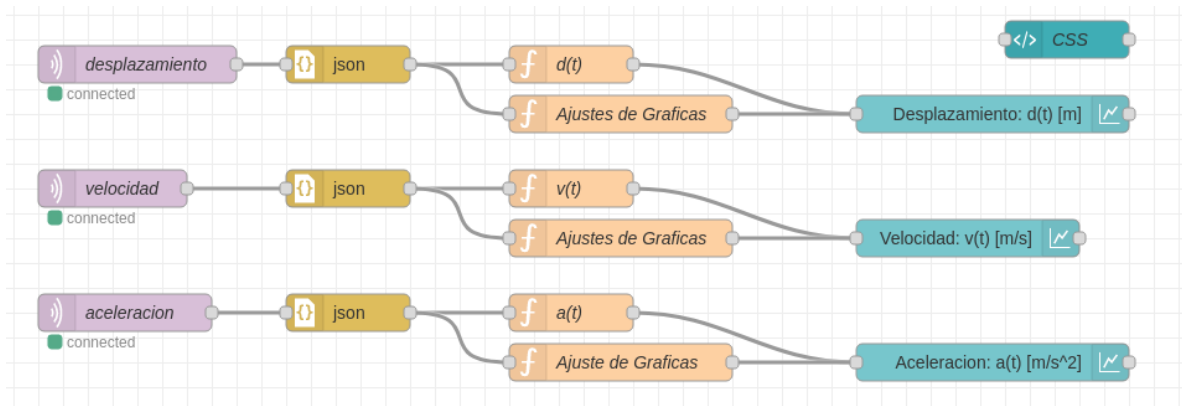


Figura 9d . Flow correspondiente a las gráficas resultantes del experimento

# 5.2 BROKER PUBLICO

El **BROKER (3)** público se encargan de la recepción y distribución de mensajes entre los clientes MQTT que se encuentran enlazados a él, como se muestra en la figura 10.

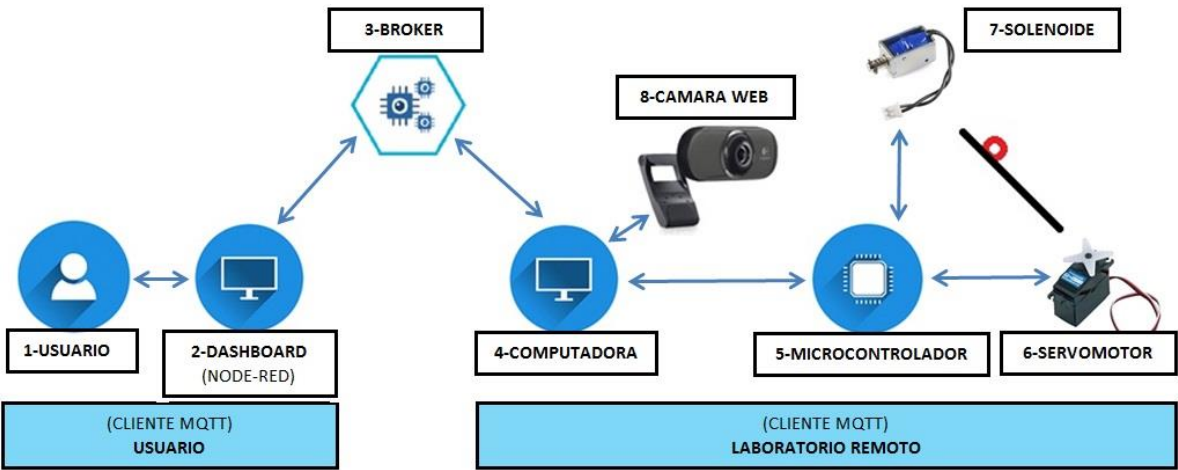


Figura 10. BROKER público

En la tabla 1 se muestra la relación de temas de publicación entre el **DASHBOARD (2)** y la **COMPUTADORA (4)** del **LABORATORIO REMOTO** a través del **BROKER (2)**:

USUARIO (CLIENTE MQTT)		LABORATORIO REMOTO (CLIENTE MQTT)	
TEMAS		TEMAS	
PUBLICACIÓN	SUSCRIPCIÓN	PUBLICACIÓN	SUSCRIPCIÓN
	python/mqtt/estado	python/mqtt/estado	
	python/mqtt/info	python/mqtt/info	
	python/mqtt/desplazamiento	python/mqtt/desplazamiento	
	python/mqtt/velocidad	python/mqtt/velocidad	
	python/mqtt/aceleracion	python/mqtt/aceleracion	
python/mqtt/angulo			python/mqtt/angulo

Tabla 1 . Conjunto de temas PUBLICACION – SUSCRIPCION manejados por el BROKER público

Cada tema definido tanto en el USUARIO como en el LABORATORIO REMOTO, maneja un grupo de datos específicos, como se explica a continuación:

1. El **LABORATORIO REMOTO** envía información al **DASHBOARD (2)** del **USUARIO (1)** sobre:

- **python/mqtt/estado** .- El estado que va teniendo el experimento conforme se desarrolla, mediante los siguientes mensajes de texto:

- **SISTEMA ACTIVADO**
- **RECIBIENDO SOLICITUD**
- **INICIALIZANDO EXPERIMENTO**
- **POSICIONANDO EL PLANO EN CERO GRADOS**
- **AJUSTANDO EL ANGULO SOLICITADO**
- **CAPTURANDO POSICIONES DE LA ESFERA**
- **EXPERIMENTO FINALIZADO**

- **python/mqtt/info** .- La fecha de realización del experimento y el valor resultante del coeficiente de fricción dinámico.

- **python/mqtt/desplazamiento** .- Envía un objeto json tipo “diccionario” de python con las coordenadas de los datos a graficar con el siguiente formato: {“x”: nn, “y”: mm}, donde “nn” y “mm” representan números reales correspondientes a las coordenadas “x”, “y”.

- **python/mqtt/velocidad** y **python/mqtt/aceleración** .- Envía un objeto json tipo “diccionario” de python con las coordenadas de los datos a graficar con el mismo formato utilizado en el tema de desplazamiento.

2. El **DASHBOARD (2)** del **USUARIO (1)** envía información al **LABORATORIO REMOTO** sobre:

- **python/mqtt/angulo** .- El ángulo de inclinación solicitado para el plano inclinado.

## 5.3 LABORATORIO REMOTO [CLIENTE MQTT]

El LABORATORIO REMOTO se encarga de la comunicación con el BROKER (3), la comunicación con el MICROCONTROLADOR (5) y la CAMARA WEB (8), y el procesamiento de las imágenes recibidas a través de esta última, como se muestra en la figura 11.

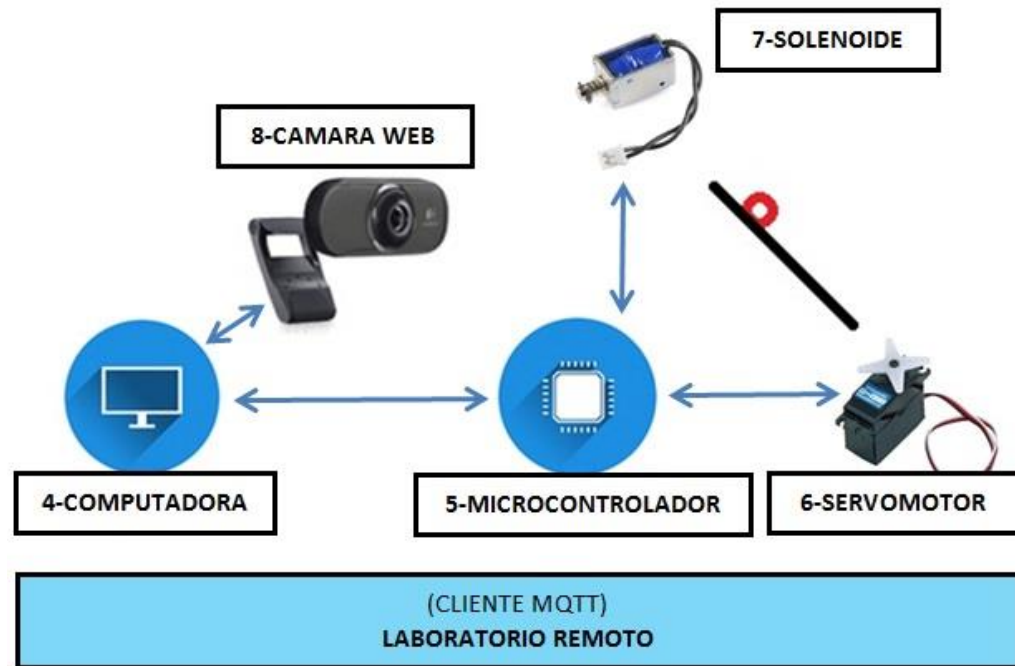


Figura 11 . LABORATORIO REMOTO

La **COMPUTADORA (4)** recibe del **BROKER (3)** público (a través del protocolo MQTT), la solicitud de posicionamiento del ángulo de inclinación del plano inclinado [ **RECIBIENDO SOLICITUD** ]. A su vez, ésta le envía la información de ajuste del ángulo solicitado al **MICROCONTROLADOR (5)**, quien envía las señales necesarias al **SERVOMOTOR (6)** acoplado al plano, para que se incline el ángulo solicitado [ **INICIALIZANDO EXPERIMENTO** ] [ **POSICIONANDO EL PLANO EN CERO GRADOS** ] [ **AJUSTANDO EL ANGULO SOLICITADO** ]. El **MICROCONTROLADOR (5)** también envía las señales correspondientes al **SOLENOIDE (7)** para que capture o libere la esfera que se desplaza por el plano inclinado. En todo este proceso mecánico, la **CAMARA WEB (8)** captura las imágenes del experimento, permitiendo a la **COMPUTADORA (4)** detectar la posición de la esfera [4] y de los marcadores correspondientes [5] del plano inclinado [ **CAPTURANDO POSICIONES DE LA ESFERA** ]. Posteriormente, con las muestras capturadas del movimiento de la esfera, la **COMPUTADORA (4)** genera un modelo matemático (ecuación) del desplazamiento de la esfera en función del tiempo [6],[7], mediante técnicas de regresión lineal ( $d(t)$ ). Una vez obtenido este modelo, se calculan las derivadas primera y segunda de dicho modelo para generar los modelos (ecuaciones) de la velocidad ( $v(t)$ ) y aceleración ( $a(t)$ ) respectivamente [ **EXPERIMENTO FINALIZADO** ].

Cada uno de estos modelos matemáticos se evalúan en un intervalo de tiempo fijo de 1.2 segundos, y las muestras obtenida de cada modelo se envían a través de **BROKER (3)** público de regreso al **USUARIO (1)** a través del **DASHBOARD (2)**, para presentar dichas muestras en forma de tres gráficas correspondientes al

desplazamiento, velocidad y aceleración.

Los modelos utilizados para cada función son:

$$\text{Desplazamiento: } \mathbf{d(t)} = mt^2 + nt + q$$

$$\text{Velocidad: } \mathbf{v(t)} = \frac{d}{dt}d(t) = 2mt + n$$

$$\text{Aceleración: } \mathbf{a(t)} = \frac{d^2}{dt^2}d(t) = 2m$$

Donde:

**m**, **n** y **q** son constantes cuyos valores se obtienen mediante técnicas de regresión, a partir de las muestras obtenidas correspondientes a la posición de la esfera mientras ésta se va desplazando por el plano inclinado. Estas muestras se capturan a través de la **CAMAR WEB (8)** y se procesan en la **COMPUTADORA (4)** mediante técnicas de procesamiento de imágenes.

Con los datos obtenidos del experimento se calcula el coeficiente de fricción dinámica "**u**" correspondiente al experimento realizado, y se envía de igual forma al **USUARIO (1)** a través del **BROKER (3)** público. La ecuación para el cálculo de **u**, es la siguiente:

$$u = \frac{a(t) - g\cos(\emptyset)\tan(\emptyset)}{g\cos(\emptyset)}$$

Donde:

**u** = Coeficiente de fricción dinámica

**a(t)** = Aceleración de la esfera

**g** = Aceleración de la gravedad

**∅** = Angulo del plano inclinado

### 5.3.1 ELEMENTOS DEL LABORATORIO REMOTO

En la figura 12 se muestran los elementos que constituyen el **LABORATORIO REMOTO**.

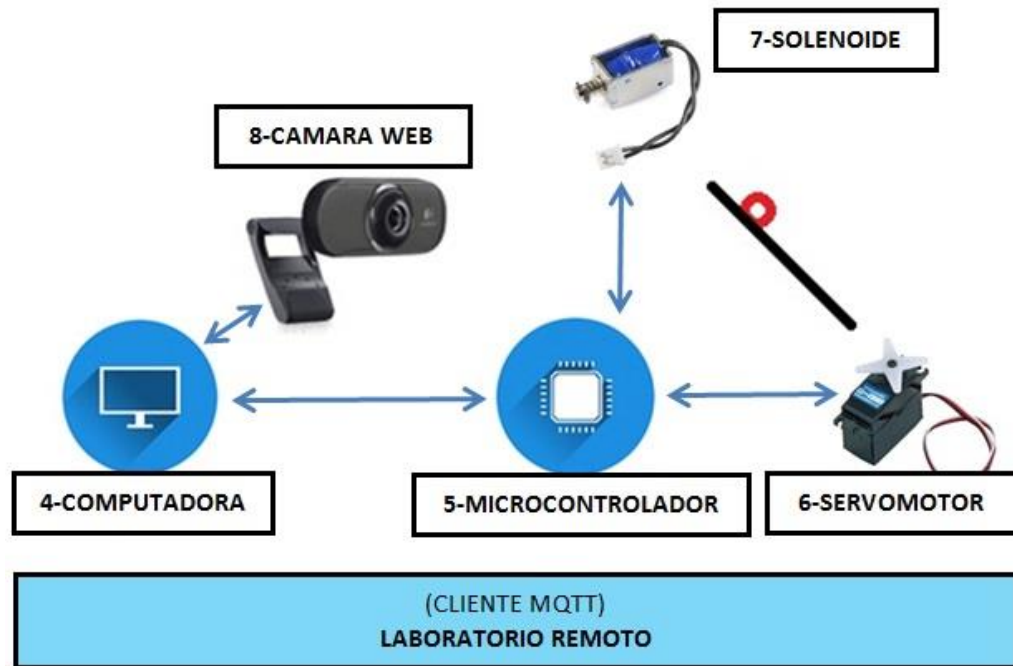


Figura 12 . LABORATORIO REMOTO

**COMPUTADORA (4)** .- En la computadora se tiene corriendo un programa en python que se encarga de:

- i) Establecer la comunicación con el **BROKER (3)** público mediante el protocolo MQTT
- ii) Recibir, por parte del **USUARIO (1)**, las solicitudes del ángulo de inclinación del plano inclinado.
- iii) Enviar, mediante transmisión serial USB al **MICROCONTROLADOR (5)**, dicha solicitud para iniciar el experimento.
- iv) Recibir el estado del experimento indicado por el **MICROCONTROLADOR (5)**, para, a su vez, enviar dicha información por MQTT al **DASHBOARD (2)** del **USUARIO (1)**.
- v) Mediante procesamiento de imágenes (reconocimiento de formas geométricas y colores), toma muestras de la posición de la esfera, así como de ciertos marcadores colocados en el plano inclinado, con el fin



Figura 13a . Computadora



de calcular los modelos matemáticos de las funciones de desplazamiento, velocidad y aceleración, para mostrar las gráficas respectivas, así como calcular el coeficiente de fricción dinámica.

vi) Enviar, mediante el protocolo MQTT, la información resultante del experimento al **DASHBOARD (2)** correspondiente en donde se encuentra el **USUARIO (1)**.

Tanto la **COMPUTADORA (4)** como las gráficas resultantes se muestran en las figuras 13a y 13b.

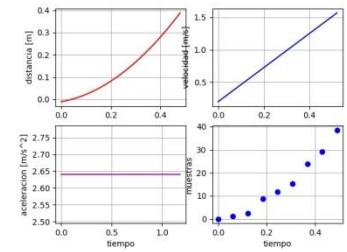


Figura 13b . Gráficas resultantes

**PLANO INCLINADO** .- El plano inclinado se construyó utilizando una base metálica armada mediante un “meccano”, sobre el cual se instaló un servomotor acoplado al plano inclinado (tira de plástico utilizada como empaque de circuitos integrados). En el otro extremo de la tira de plástico se instaló un solenoide que permite “capturar” y/o “liberar” la esfera de vidrio (canica) de color amarillo. Cuando el plano inclinado se encuentra en modo de posicionamiento, el solenoide indicado bloquea el recorrido de la esfera y solamente cuando el plano ha alcanzado su posición indicada, se libera la esfera para que inicie su recorrido, y de esta manera se capturen sus posiciones (muestras) con respecto al tiempo.



Figura 14 . Plano inclinado

**MICROCONTROLADOR (5)** .- El microcontrolador mostrado en la figura 14a, se encuentra conectado a la computadora mediante comunicación serial (USB). El programa residente en el microcontrolador se encarga de manejar un servomotor y un solenoide, ambos acoplados a los extremos del plano inclinado.

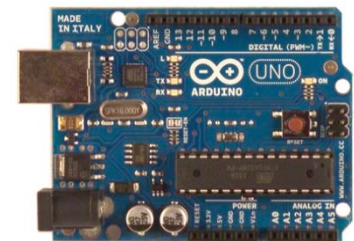


Figura 15a . Microcontrolador

La secuencia de acciones que lleva a cabo el microcontrolador son:

i) **Captura de la esfera.**- Mover el servomotor (plano inclinado) a un ángulo negativo respecto a la horizontal, con el objetivo de capturar la esfera en el extremo del plano inclinado en dónde se encuentra el solenoide activado en modo de captura (figura 15b).



Figura 15b . Captura de la esfera

ii) **Angulo CERO** .- Mover el servomotor (plano inclinado) a un ángulo de CERO grados respecto a la horizontal, con el objetivo de iniciar la colocación del plano inclinado en el ángulo solicitado. La esfera se mantiene en el extremo del plano inclinado en dónde se encuentra el

solenoides activados en modo de captura (figura 15c).

iii) **Angulo solicitado** .- Mover el servomotor (plano inclinado) al ángulo solicitado por el usuario. La esfera se mantiene en el extremo del plano inclinado en donde se encuentra el solenoide activado en modo de captura (figura 15d).

iv) **Libera la esfera** .- Manteniendo el plano inclinado en la posición solicitada por el usuario, se activa el solenoide en el modo de liberación de la esfera. Al liberarse ésta y comenzar a moverse, se toman muestras de su posición (figura 15d).



Figura 15c . Angulo CERO.

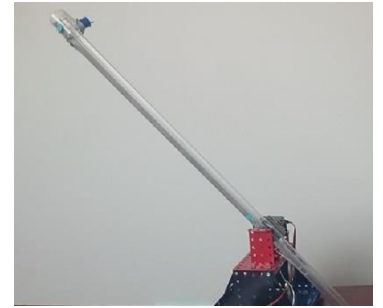


Figura 15d . Angulo solicitado y liberación de la esfera

**SERVOMOTOR (6)** .- La función del servomotor es la mover al plano inclinado tanto en su posición para la captura de la esfera en el extremo del plano inclinado, como en sus posiciones de ángulo CERO (para inicializar el experimento) y ángulo solicitado (para comenzar el experimento). El servomotor utilizado se muestra en la figura 16.



Figura 16 . Servomotor

**SOLENOIDE (7)** .- La función del solenoide es la de capturar, asegurar y liberar la esfera en las diferentes posiciones del plano inclinado. Se muestra en la figura 17.



Figura 17 . Solenoide

**CAMARA WEB (8)** .- La cámara WEB se encarga de capturar los “frames” del video de todo el experimento. Junto con el programa python que reside en la computadores, se procesan las imágenes para generar los modelos del comportamiento del experimento. La cámara se muestra en la figura 18.



Figura 18 . Cámara WEB

## 5.4 MICROCONTROLADOR (5)

El **MICROCONTROLADOR (5)** se encarga de generar las señales necesarias para controlar tanto al **SERVOMOTOR (6)** como al **SOLENOIDE (7)**, como se muestra en la figura 19.

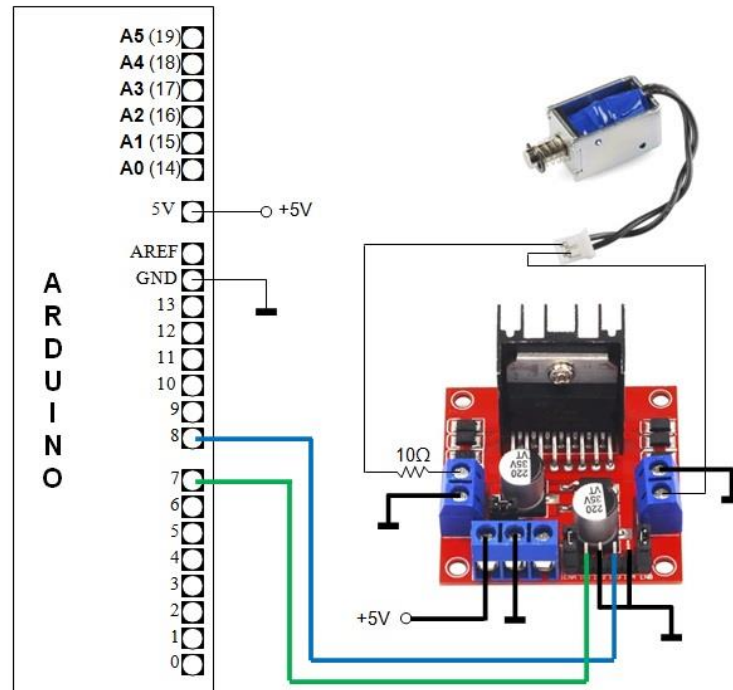


Figura 19 . Conexión del MICROCONTROLADOR al SERVOMOTOR y el SOLENOIDE

De acuerdo a la secuencia explicada en las secciones anteriores, el código completo del **MICROCONTROLADOR (5)** se encuentra en el repositorio correspondiente a este proyecto.

# 5.5 COMPUTADORA (4): DIAGRAMA DE CLASES

El diagrama de clases del programa en python que reside en la **COMPUTADORA (4)** del Laboratorio Remoto, se muestra en la figura 20. El código completo se encuentra en el repositorio correspondiente a este proyecto.

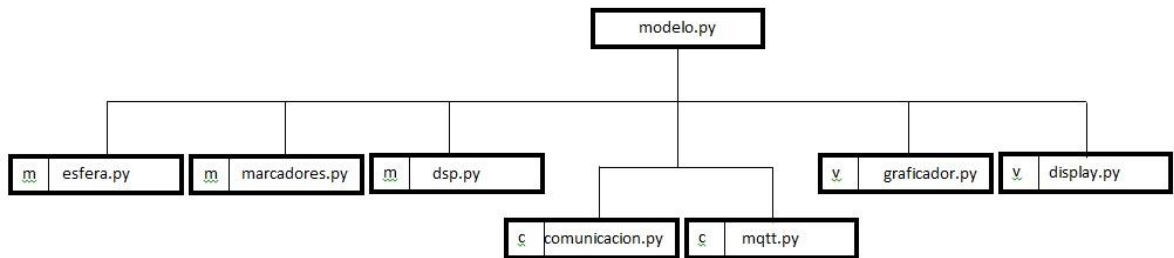


Figura 20 . Diagrama de Clases del programa en python que reside en la COMPUTADORA (4) del Laboratorio Remoto

PROGRAMA PRINCIPAL		DESCRIPCION	
Modelo.py		Genera los estado secuenciales del experimento: <b>SISTEMA ACTIVADO, RECIBIENDO SOLICITUD, INICIALIZANDO EXPERIMENTO, POSICIONANDO EL PLANO EN CERO GRADOS, AJUSTANDO EL ANGULO SOLICITADO, CAPTURANDO POSICIONES DE LA ESFERA, EXPERIMEMNTO FINALIZADO.</b>	
PAQUETE	CLASE	OBJETO	DESCRIPCION
m	Esfera	esfera	Clase reconocedora de la esfera amarilla
	Marcadores	marcador	Clase reconocedora de dos maracdores azules de referencia localizados en los extremos del plano inclinado. Se utilizan como auxiliares para medir la longitud del plano y el ángulo que foma con la horizontal
	DSP	dsp	Clase encargada de realizar el procesamiento digital de los datos obtenidos en el experimento
c	Comunicacion	com	Clase encargada de enviar y recibir información de ARDUINO a través del puerto serial USB
	MQTT	mqtt	Clase encargada de la comunicación con el bróker MQTT
v	Graficacion	graficas	Clase encargada de generar las gráficas de desplazamiento, velocidad y aceleración de la esfera sobre el plano inclinado
	Display	info	Clase encargada de mostrar la información resultante del experimento en el lateral derecho de la imagen

## 6. LISTA DE MATERIALES

La lista de material utilizado para elaborar este proyecto se muestra a continuación:

### PROCESADORES, ACTUADORES Y SENSORES:

- 1 Computadora con Node-Red instalado y capacidad de conectarse a un bróker con el protocolo MQTT
- Bróker publico disponible
- 1 Computadora con la aplicación de Python desarrollada en este proyecto instalada
- 1 Microcontrolador ATMEGA 328P con la aplicación desarrollada en este proyecto instalada
- 1 cable serial USB (computadora <-> microcontrolador)
- 1 Puente H L293
- 1 Solenoide a 5 ó 3.3 volts
- 1 Servomotor JR NES L-401 ó similar
- 1 Fuente de alimentación de 5 V
- 1 Cámara WEB con resolución de 720p/30fps

### ESCENARIO:

- 1 Fomboard blanco como fondo del plano inclinado (1.00 x 1.80 m)
- 1 Tripié para montaje de la cámara WEB (colocado a 1.00 m del fondo y 0.80 m del plano inclinado)
- 1 Nivel de burbuja (para calibración)
- Iluminación natural (10:00 a 17:00 hrs.)

### PLANO INCLINADO:

- 1 Base para el plano inclinado (construido con “meccano”)
- 1 Acoplamiento del servomotor con “meccano” (realizado con una pieza de LEGO)
- 1 Contenedor plástico para circuitos integrados de 60 cm de longitud
- 1 Mica transparente flexible
- 1 Papel azul para crear los marcadores visuales para el plano inclinado (longitud de referencia)
- 1 Canica amarilla de vidrio de 1.4 cm de diámetro
- Piezas varias para adaptaciones

## 7. CONCLUSIONES

En el proyecto presentado cumplió en su mayor parte con los objetivos establecidos correspondientes a utilizar el contexto de la IoT para controlar un sistema remoto, en este caso particular, un plano inclinado.

El contenido de estos objetivos se puede interpretar como el conjunto de procesos que permiten controlar, enviar, generar y recibir información a través de un servidor (bróker) y dos nodos (clientes MQTT) entre los cuales se realiza dicho intercambio de información.

El único punto que no fue cumplido en su totalidad, fue el de transmitir el video del experimento a través de un servidor de video, debido a problemas intrínsecos en el manejo del video, el cual es un tema que podría ser investigado en proyectos futuros

El proyecto propuesto abre las puertas a futuros proyectos relacionados con laboratorios remotos utilizando el mismo esquema de transferencia de información entre dos o más nodos (clientes MQTT).

Consideramos que esta primera experiencia con el manejo de los conceptos y dispositivos asociados al IoT, fue una experiencia muy enriquecedora.

## 8. REFERENCIAS

- [1] <https://www.labster.com/>
- [2] <https://labsland.com/en/labs/inclined-plane>
- [3] <https://www.youtube.com/watch?v=7eHHTk2psLs>
- [4] <https://pyimagesearch.com/2015/09/21/opencv-track-object-movement/>
- [5] <https://pyimagesearch.com/2016/02/01/opencv-center-of-contour/>
- [6] [https://www.w3schools.com/python/python\\_ml\\_multiple\\_regression.asp](https://www.w3schools.com/python/python_ml_multiple_regression.asp)
- [7] <https://www.geeksforgeeks.org/numpy-poly-in-python/>

