

**DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA WEB TIPO FOSS, PARA
REALIZAR PRÁCTICAS DE CONTROL DE POSICIÓN A UN SISTEMA VIGA –
BOLA**



BRANDON ANDRÉS VALENCIA SANTACRUZ

**UNIVERSIDAD DEL CAUCA
FACULTAD DE CIENCIAS NATURALES, EXACTAS Y DE LA EDUCACIÓN
DEPARTAMENTO DE FÍSICA
INGENIERÍA FÍSICA
POPAYÁN
2022**

**DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA WEB TIPO FOSS, PARA
REALIZAR PRÁCTICAS DE CONTROL DE POSICIÓN A UN SISTEMA VIGA –
BOLA**

BRANDON ANDRÉS VALENCIA SANTACRUZ

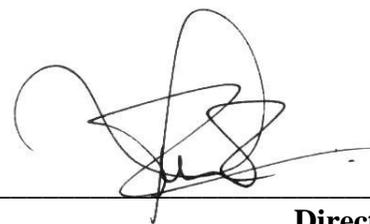
Tesis presentada como requisito para optar al título de Ingeniero Físico.

Director: Ing. Germán Arturo Bacca.

UNIVERSIDAD DEL CAUCA
FACULTAD DE CIENCIAS NATURALES, EXACTAS Y DE LA EDUCACIÓN
DEPARTAMENTO DE FÍSICA.
INGENIERÍA FÍSICA
POPAYÁN
2022

Nota de aceptación

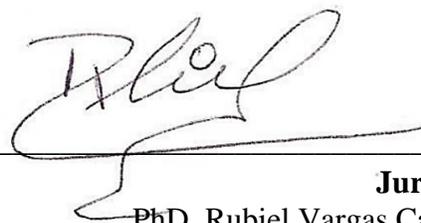
Una vez revisado el documento final del trabajo de grado titulado “Diseño e implementación de una plataforma web tipo FOSS, para realizar prácticas de control de posición a un sistema viga – bola”, realizado por el estudiante Brandon Andrés Valencia Santacruz, se autoriza seguir los trámites correspondientes para optar al título de Ingeniero Físico de la Universidad del Cauca.



Director
PhD. Germán Arturo Bacca Bastidas
Universidad del Cauca



Jurado
Mg. Luis Fernando Echeverry
Facultad de Ciencias Naturales, Exactas y de la
Educación
Universidad del Cauca



Jurado
PhD. Rubiel Vargas Cañas
Facultad de Ciencias Naturales, Exactas y de la
Educación
Universidad del Cauca

Agradecimientos

Quiero dar las gracias a Dios por su guía y bendiciones en este proyecto. Gracias a su providencia, he podido completar esta tesis con éxito. También quiero agradecer a mi madre Sandra Santacruz por su apoyo constante y por ser mi pilar de fuerza en todo momento. A mis tíos Wilson Santacruz y Jonathan Santacruz, por ser una fuente de inspiración y motivación para mí. A mi esposa Luisa Henao, por su comprensión y apoyo incansable durante este proceso. Gracias por ser mi roca y por estar siempre ahí para mí. Quiero dar las gracias a mi director de tesis por su liderazgo y orientación durante este proceso. Gracias por compartir su conocimiento y experiencia conmigo.

Finalmente, quiero agradecer a toda mi familia y amigos por su apoyo y motivación constante durante este proceso. Gracias por ser una fuente de alegría y bendición en mi vida.

Espero que este trabajo sea un reflejo de mi aprendizaje y dedicación. Gracias por su tiempo y consideración.

Resumen

Este proyecto describe el desarrollo de una plataforma web que permite realizar prácticas de control de posición a un sistema viga – bola, bajo el concepto de internet de las cosas –IoT por sus siglas en inglés–, para lo cual se caracteriza un sensor infrarrojo (*GY-530 VL53L0X*), con el fin de obtener el estado del componente eléctrico que se encarga de tomar el valor de la distancia a la que se encuentra la bola del punto de referencia, para utilizarlo como señal de entrada del sistema de control el cual tiene la función de establecer el ángulo de la viga con un servomotor. Adicional a esto, el proyecto cuenta con una interfaz gráfica alojada en la nube que permite al usuario controlar los valores de funcionamiento del sistema desde cualquier dispositivo conectado a internet.

Tabla de contenido

Introducción.....	16
Aspectos generales	16
Capítulo 1	19
1.1 Definición del problema.....	19
1.2. Objetivos.....	20
1.2.1 Objetivo general	20
1.2.2 Objetivos específicos.....	20
1.3. Contribuciones.....	21
1.4. Marco teórico.....	22
1.4.1 Marco conceptual	22
1.4.1.1 Internet de las cosas.....	22
1.4.1.2 Modelo OSI y arquitectura de redes IoT.....	23
1.4.1.3 Protocolos de comunicación IoT.....	24
1.4.1.4 Hypertext Transfer Protocol (HTTP).....	25
1.4.1.5 Protocolos WAN	26
1.4.1.6 Protocolos LPWAN.....	27
1.4.1.7 Tecnologías web.....	27
1.4.1.8 Bases de datos	28
1.4.1.9 Sensor	29

1.4.1.10 Actuador	30
1.4.1.11 Sistemas de supervisión y control	31
Capítulo 2	35
2.1. Estado del arte	35
2.1.1. Referencias internacionales	35
2.1.1.1 Diseño e implementación de laboratorio inteligente basado en IoT (Poongothai, Muthu Subramanian & Rajeswari, 2018)	35
2.1.1.2 Integración de IoT con controlador LQR-PID para vigilancia en línea, control de flujo y presión en un sistema de transporte de fluidos (Priyanka, Maheswari, Thangavel & PonniBala, 2020).....	35
2.1.1.3 Implementación de algoritmos de control estocástico en la nube mediante herramientas de programación IoT para un motor DC (Escobar, 2021).....	36
2.1.1.4 Control y monitoreo IMC-PID en tiempo real del proceso de extracción de aceites esenciales mediante IoT (Siti Hasinah, Mohd Fazalul, Ramli Adnan & Mazidah Tajjudin, 2020).....	36
2.1.1.5 Monitoreo y control PID de un sistema RC usando IoT mediante Raspberry (Velásquez et al, 2020).....	37
2.1.2 Referencias nacionales	37
2.1.2.1 Simulación del sistema de control en la represa de Hidroitungo para controlar el nivel de agua del río Cauca utilizando IoT (Rodríguez & Olarte, 2019).....	37

2.1.2.2	Diseño de un sistema de monitoreo y control agroclimático para lombricultivo en la Universidad Autónoma de Occidente, Valle del Cauca (Becerra & Stenberg, 2020)	38
2.1.2.3	Sistema de riego basado en la Internet de las Cosas (Anaya, A. J., Ríos, J., Carvajal, D. A., Peluffo D. H. & Castro, J. A., 2016).	38
2.1.3	Referencias regionales	39
2.1.3.1	Estrategia metodológica para la transferencia de tecnologías basadas en internet de las cosas con aplicación en la caficultura del Cauca (Quiñones, 2021)	39
2.1.3.2	Desarrollo de un sistema IoT para monitoreo y apoyo en el dimensionamiento de consumo energético para la Industria Licorera del Cauca (Guevara, 2020)	39
2.1.4	Comparación de proyectos	40
Capítulo 3	42
3.1.	Metodología.....	42
3.1.1	Reconocimiento de equipos del sistema viga – bola	43
3.1.1.1	Viga.....	43
3.1.1.2	Bola	43
3.1.1.3	Eje o pivote.....	44
3.1.1.4	Soporte	45
3.1.2	Caracterización eléctrica del equipo y componentes del sistema IoT	45
3.1.2.1	Sensores y actuadores	45
3.1.2.2	Tarjeta de adquisición de datos	47
3.1.2.3	Caracterización de sensor y actuador	56

3.1.3 Diseño e implementación del sistema de control para la posición.....	57
3.1.3.1 Interfaz de usuario.....	59
3.1.3.2 Control interactivo de la planta.....	60
3.1.3.2 Control teórico de la planta.....	60
3.1.3.3 Documentación.....	60
3.1.3.4 Prueba.....	60
Capítulo 4	61
4.1 Análisis y resultados.....	61
4.1.1 Dimensionado del sensor y el actuador	61
4.1.1.1 Caracterización y calibración del sensor.....	61
4.1.2 Implementación del servomotor	65
4.1.2.1 Calibración de señal PWM.....	65
4.1.2.2 Implementación del servomotor.....	67
4.1.3 Análisis del sistema viga – bola	67
4.1.3.1. Función de transferencia de la planta.....	67
4.1.3.2. Simulación del sistema.....	71
4.1.4. Implementación del control PID en la tarjeta ESP32.....	72
4.1.4.1. Instrucciones fundamentales	73
4.1.4.2. Programación de software PID en el ESP32.....	75
4.1.5. Interfaz web.....	76

4.1.5.1. Funcionamiento de la interfaz de usuario	78
4.1.5.2. Modos de trabajo en la interfaz	80
4.1.6. Comunicación con la base de datos	81
4.1.6.1. Comunicación del sistema físico con Firebase	81
4.1.6.2. Comunicación de la interfaz web con Firebase.....	82
4.1.6.3. Comunicación entre la interfaz web y la planta física	83
4.1.7. Evaluación	85
4.1.7.1. Sintonización del controlador PID para la plataforma.	86
4.1.7.2. Despliegue y prueba final de la plataforma.....	89
Conclusiones.....	91

Lista de figuras

Figura 1. Internet de las cosas	22
Figura 2. Capas del modelo OSI.....	23
Figura 3. Comparación de protocolos según ancho de banda de sus datos y alcance	25
Figura 4. Componentes de gestor de almacenamiento y procesador de consultas	29
Figura 5. Ejemplo de sensores	30
Figura 6. Servomotor	30
Figura 7. Esquema de supervisión y control.....	31
Figura 8. Sistema de control de lazo cerrado.....	32
Figura 9. Sistema de control de lazo abierto	32
Figura 10. Función de transferencia en lazo abierto y cerrado.....	33
Figura 11. Control PID	34
Figura 12. Desarrollo metodológico.....	42
Figura 13. Viga de la planta.....	43
Figura 14. Bola	44
Figura 15. Eje de la viga.....	44
Figura 16. Soporte del sistema viga – bola.....	45
Figura 17. Sensor de distancia GY-530 VL53L0X	46
Figura 18. Funcionamiento Sensor de distancia láser	46
Figura 19. Servomotor FUTABA S3003.....	47
Figura 20. Módulo ESP32	48
Figura 21. Montaje del sistema viga – bola de prueba	49
Figura 22. Entrada y salida de la función de transferencia del sistema.....	49

Figura 23. Esquema del sistema para el modelado rueda sin deslizarse	50
Figura 24. Representación de la trayectoria de la bola en la barra.....	52
Figura 25. Registro de los valores de distancia	56
Figura 26. Ángulo del servomotor según PWM.....	57
Figura 27. Esquema del sistema de control	57
Figura 28. Diagrama de flujo de sistema de control.....	58
Figura 29. Ventana de visualización de la interfaz web	59
Figura 30. Implementación sistema a sensar	65
Figura 31. Señal PWM con un ángulo de 1°	66
Figura 32. Señal PWM con un ángulo de 50°	66
Figura 33. Señal PWM con un ángulo de 200°	66
Figura 34. Señal PWM con un ángulo de 255°	66
Figura 35. Sistema de sensado con servomotor.....	67
Figura 36. Código en MatLab para encontrar la función de transferencia	71
Figura 37. Función de transferencia del sistema viga – bola	71
Figura 38. Respuesta de la función de transferencia en lazo abierto ante una entrada escalón	72
Figura 39. Implementación del controlador PID en el ESP32	75
Figura 40. Dependencias del front-End.....	76
Figura 41. Estructura de carpetas para el Front-End	77
Figura 42. Página principal.....	78
Figura 43. Página de inicio de sesión	79
Figura 44. Página de documentación.....	79
Figura 45. Página de teoría.....	80
Figura 46. Panel de control.....	80

Figura 47. Diagrama de la interfaz web	81
Figura 48. Código del ESP32 y Firebase, configuración inicial	82
Figura 49. Código del ESP32 y Firebase, conexión con Realtime Database	82
Figura 50. Código de la plataforma web y Firebase, conexión con Realtime Database	83
Figura 51. Estructura de datos en Realtime Database	83
Figura 52. Datos tipo string a cadena de números con JavaScript	85
Figura 53. Comunicación de la plataforma web y el ESP32 con firebase.....	85
Figura 54. Respuesta del sistema con $K_p=100$, $K_i = K_d = 0$	86
Figura 55. Respuesta del sistema con $K_p=10$, $K_i = K_d = 0$	86
Figura 56. Respuesta del sistema con $K_d=100$, $K_p = K_i = 0$	87
Figura 57. Respuesta del sistema con $K_d=50$, $K_p = K_i = 0$	87
Figura 58. Respuesta del sistema con $K_i=5$, $K_p = K_d = 0$	88
Figura 59. Respuesta del sistema con $K_p=10$, $K_p = 50$, $K_i = 5$	88
Figura 60. Interfaz en producción utilizando Vercel.com	89
Figura 61. Ingreso a la plataforma desde un dispositivo móvil.....	90
Figura 62. Control PID al sistema viga – bola remotamente	90

Lista de tablas

Tabla 1. Descripción de capas modelo OSI.....	24
Tabla 2. Comparación de proyectos en el marco referencial.	40
Tabla 3. Características de la viga del sistema.	43
Tabla 4. Características de la bola.	44
Tabla 5. Características del sensor de distancia GY-530 VL53L0X.....	46
Tabla 6. Características del servomotor FUTABA S3003.	47
Tabla 7. Características Módulo ESP32.	48
Tabla 8. Nombre de las variables.	59
Tabla 9. Datos del sensor con objeto acercándose.	62
Tabla 10. Datos del sensor con objeto alejándose.	63
Tabla 11. Datos promedio del sensor.	64
Tabla 12. Estructura de Carpetas de la interfaz web.	77
Tabla 13. Tipos de datos de Realtime Database.....	84

Lista de gráficos

Gráfico 1. Curva de calibración del sensor acercándose.....	62
Gráfico 2. Curva de calibración del sensor alejándose.....	63
Gráfico 3. Curva de calibración promedio del sensor	64

Introducción

Aspectos generales

El Informe anual de Internet de Cisco (2018-2023) revela que los dispositivos conectados a Internet superarán los 30 000 millones para 2023 (*Cisco Annual Internet Report (2018–2023) White Paper*, 2022). Además, la duración de las baterías es hasta diez veces más extensa en los dispositivos de comunicación masiva y hay una reducción de hasta cinco veces menos de latencia entre la conexión de un extremo a otro (Osseiran *et al.*, 2014); por esta razón, las tendencias muestran a los investigadores un gran interés en la combinación de varias tecnologías, que permiten agilizar y dinamizar el control de diferentes procesos de la vida cotidiana. Desde el control de la energía del hogar remotamente, hasta aplicaciones para conectar la industria con la ciudad en áreas de: transporte, salud, energía, logística, medio ambiente entre otros (Botta *et al.*, 2016).

Actualmente, estamos en lo que se ha llamado ‘la cuarta revolución industrial’ o ‘industry 4.0’, donde se emplea la tecnología para la automatización y digitalización de los procesos y las industrias (Fajardo, Aguilera & Guerrero-Carbacas, 2019), siendo uno de los pilares de esta tecnología el Internet de las Cosas o IoT.

Ahora bien, Colombia es un país que se encuentra atrasado en cuanto IoT se refiere (Porkodi, 2014), por lo que la Ley 1955 de 2019, la cual expide el Plan Nacional de Desarrollo 2018- 2022 “Pacto por Colombia, pacto por la equidad”, las TIC se incluyen dentro del Pacto por la transformación digital de Colombia a empresas y hogares conectados. Según el artículo 310 de la Ley 1955 de 2019, se indica que en el MinTIC tendrá prioridad por iniciativas para acceso a internet público para zonas más pobre y vulnerables, zonas apartadas. El gobierno motiva a las entidades públicas e instituciones educativas a que financien sus necesidades a lo que se refiere conexiones a internet. Tendrá iniciativas para la implementación de oferta y demanda en servicios

de telecomunicaciones principalmente para zonas pobres y vulnerables. Iniciativa que se espera aumente la cobertura en internet. En el Plan Nacional de Desarrollo 2018-2022, la transformación digital publica requiere que las entidades incorporen tecnologías emergentes como DLT – Distributed Ledger Technology–, análisis masivo de datos –Big Data–, Inteligencia Artificial –AI–, Internet de las Cosas –IoT–, Robótica y similares que permitan dar servicios con la cuarta revolución industrial al Estado mediante estas tecnologías emergentes. Por parte del gobierno nacional y el MinTIC flexibiliza la gestión y aprovechamiento de datos públicos, adopción de modelos, uso de software público para tener un mejor aprovechamiento en estas tecnologías emergentes con fin de dar confianza y seguridad digital. Según la Ley 1341 de 2009 del sector TIC en el inciso 2 se indica “Las tecnologías de la información y las comunicaciones deben servir al interés general y es deber del Estado promover su acceso eficiente y en igualdad de oportunidades, a todos los habitantes del territorio nacional”. Por último, la Ley 1540 de 2011 en su artículo 56 adopta el principio de ‘neutralidad en Internet’, en otras palabras, no es posible el bloqueo, interferir, discriminar, ni restringir el derecho a cualquier usuario de internet para utilizar, enviar o recibir contenido, aplicación o servicios lícitos de internet.

Es importante resaltar que el objetivo final de IoT es introducir la tecnología PnP –Plug and Play por sus siglas en inglés– "enchufar, conectar y usar" ya que proporciona al usuario final, facilidad de operación, control de acceso remoto y capacidad de configuración.

Con el reciente aumento del internet de las cosas, se han introducido varias plataformas diferentes de IoT para que los investigadores y desarrolladores faciliten la gestión y el control de varios dispositivos. Por esta razón, el trabajo de grado presentado a continuación trata acerca de la implementación de un sistema de control basado en IoT, que tiene como característica principal la aplicación de un control PID a un sistema físico viga – bola a través de internet, esto tiene dos ventajas: una es la conexión con un sistema físico a través de cualquier dispositivo y, la segunda,

reducir significativamente el proceso de configuración, adaptación e instalación del software encargado de generar la orden de control, ya que al estar en la nube, permite que cualquier persona con conexión a internet pueda ingresar a la interfaz de usuario.

Este trabajo de investigación busca aportar un sistema que, además de aplicarle un control PID, también se pueda utilizar como mecanismo para realizar prácticas de laboratorio, en presencial o remoto.

Por lo anterior, el trabajo de grado está sujeto a cubrir las necesidades de estudiante y el docente con el fin de brindar una mejor experiencia en sistemas de control.

Capítulo 1

1.1 Definición del problema

El componente práctico es realmente importante en el proceso de enseñanza y aprendizaje, debido a que los fundamentos teóricos se afianzan logrando así, la aprehensión de los conocimientos en cualquier área del saber (Álvarez, 2015). Sin embargo, existen algunos inconvenientes con esta situación, uno de ellos es que el programa de Ingeniería Física –PIF– de la Universidad del Cauca no cuenta con una plataforma de software libre y una planta de prueba, que permita a los estudiantes mejorar el proceso de aprendizaje en los tópicos de control abordados, además, también se encuentra que los software más utilizados en desarrollar rutinas para pruebas automatizadas son Matlab y Labview, según la revista IEEE spectrum en su ranking anual (IEEE spectrum, *s.f*) , los cuales tienen un costo pues requieren de licencias (Santos & Farias, 2010) o algunos de ellos están sujetos a un sistema operativo determinado, a esto se le suma el hecho que varios de estos software tienen un proceso de instalación y configuración previa a su uso. Puesto que una gran parte de los desarrolladores de aplicaciones y los usuarios valoran la máxima simplicidad del flujo de trabajo y una interfaz de usuario de apuntar y hacer clic, se tiene un problema de accesibilidad y experiencia de usuario en plataformas de programación y control de procesos.

Por otra parte, se deben sumar las implicaciones del aislamiento. No solo aparece como desafío lograr la reducción de las brechas digitales existentes que dificultan el acceso a las clases virtuales, sino que se deben implementar prácticas para disminuir la deserción (Garcés Fuenmayor & Mora Bolaños, 2020), como lo son las TIC en prácticas de laboratorios virtuales o remotos.

1.2. Objetivos

1.2.1 Objetivo general

- Diseñar e implementar una plataforma web utilizando componentes FOSS, para realizar prácticas de control de posición a un sistema físico viga – bola.

1.2.2 Objetivos específicos

- Diseñar una interfaz gráfica de usuario tipo web, que permita un control sencillo de la planta.
- Construir la planta de prueba viga – bola, con ayuda de un modelo matemático a partir de las leyes físicas.
- Seleccionar un protocolo de comunicación e implementarlo para realizar la conexión entre el microcontrolador y la interfaz web.
- Alojarse la interfaz de usuario en un servicio de Hosting, para ser accedida desde otros dispositivos.

1.3. Contribuciones

El presente proyecto de una plataforma web de bajo costo, basada en componentes FOSS, podría permitir a docentes y estudiantes realizar prácticas de control para un sistema viga – bola, utilizando controladores digitales P, I, D, PD y PID clásico, con estructura en paralelo, permitiendo aplicar los conocimientos teóricos adquiridos en cursos como sistemas de control y potencia, control analógico, identificación de sistemas e instrumentación industrial, además el sistema hardware y software de pruebas será compatible con todos los ordenadores, ya que no dependerá de un sistema operativo, ni de instalaciones de tercero, ampliando el rango de asequibilidad a los usuarios, así, el estudiante dedicará el máximo tiempo posible a los aspectos relacionados con el análisis e interpretación de los resultados alcanzados al realizar las prácticas con los diferentes equipos.

Una de las características principales de este trabajo es que consta de dos partes, una parte teórica y otra parte práctica, lo que daría la posibilidad al usuario de reforzar sus conocimientos teóricos en cualquier momento o, realizar directamente el control al sistema viga – bola, por medio de la interfaz de usuario. El sistema también contará con un mecanismo de seguridad que limitará el acceso a la planta, dando al docente la posibilidad de regular el ingreso a la planta física, evitando conflictos en la orden de control, daños al microcontrolador o al sitio web.

Por último, se permitirá la realización de prácticas de laboratorios remotos, fortaleciendo la educación a distancia, el ahorro en costos de infraestructura física, la reducción en los tiempos de desplazamiento, la accesibilidad y el desarrollo de las competencias transversales como el uso de las TIC.

1.4. Marco teórico

1.4.1 Marco conceptual

1.4.1.1 Internet de las cosas

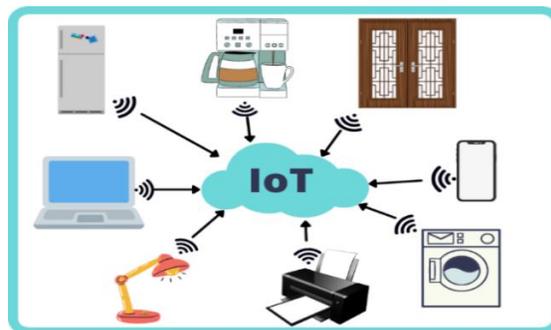
El Internet de las cosas es una red de **interconexión digital entre dispositivos**, personas y la propia Internet que permite el intercambio de datos entre ellos, haciendo que se pueda capturar información clave sobre el uso y el rendimiento de los dispositivos y los objetos para detectar patrones, hacer recomendaciones, mejorar la eficiencia y crear mejores experiencias para los usuarios (Alonso, 2021).

IoT ha evolucionado desde la convergencia de tecnologías inalámbricas, Sistemas Microelectromecánicos –MEMS–, microservicios e internet. Wigmore (2021) habla que “La convergencia ha ayudado a derribar las paredes entre la tecnología operativa (OT) y la tecnología de la información (TI), permitiendo que los datos no estructurados generados por máquinas sean analizados para obtener información que impulse mejoras”.

El IoT (figura 1) se extiende a todo tipo de objetos (físicos y digitales), a los animales, a las personas y a los entornos o ambientes.

Figura 1

Internet de las cosas.

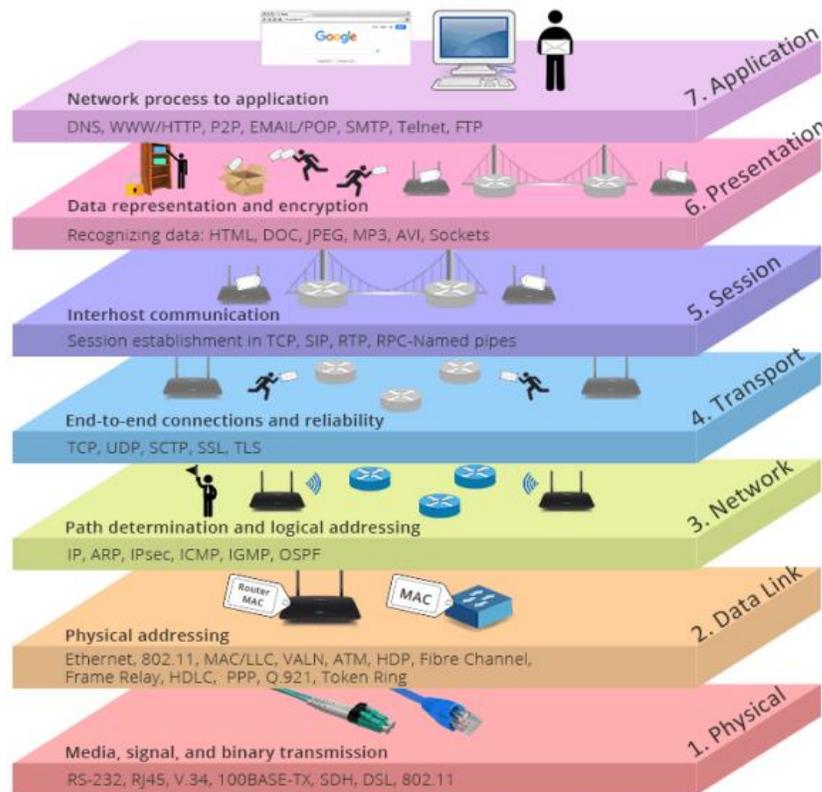


1.4.1.2 Modelo OSI y arquitectura de redes IoT

La mayoría de los conjuntos de protocolos de red se estructuran en capas. El modelo OSI describe una estructura con siete capas para las actividades de red (ver figura 2). Las capas representan las operaciones de transferencia de datos comunes a todos los tipos de transferencias de datos entre las redes de cooperación (*Modelo de referencia OSI (Guía de administración del sistema: servicios IP)*, s. f.) (Tabla 1 Modelo de referencia de Interconexión de Sistemas Abiertos).

Figura 2

Capas del modelo OSI.



Fuente: *Platzi - Cursos online profesionales de tecnología*, s. f.-b

Tabla 1.

Descripción de capas modelo OSI.

No. De carga	Nombre de capa	Descripción
7	Aplicación	Se compone de los servicios y las aplicaciones de comunicación estándar que puede utilizar todo mundo.
6	presentación	Se asegura de que la información se transfiera al sistema receptor de un modo comprensible por el sistema.
5	Sesión	Administra las conexiones y terminaciones entre los sistemas que cooperan.
4	Transporte	Administra la transferencia de datos. Así mismo, garantiza que los datos recibidos sean idénticos a los tramitados.
3	Red	Administra las direcciones de datos y la transferencia entre redes.
2	Vínculo de dato	Administra la transferencia de datos en el medio de red.
1	Física	Define las características de hardware de red.

1.4.1.3 Protocolos de comunicación IoT

Un protocolo de comunicación es una serie de normas que definimos para que dos o más dispositivos puedan comunicarse y entenderse.

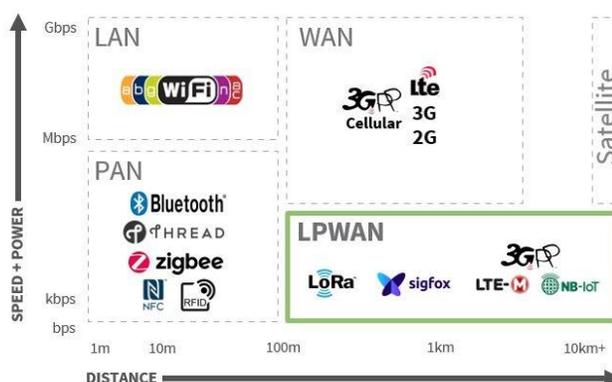
Existe una gran variedad de opciones de conectividad a la hora de hablar de protocolos de IoT. Muchas son tecnologías bien conocidas, como WiFi, Bluetooth, ZigBee y celulares, pero también se encuentran varias opciones de redes alternativas que se están implementando en varios lugares del mundo para el desarrollo de aplicaciones IoT.

Dependiendo de la aplicación es necesario analizar factores como el alcance, la transmisión de datos, la seguridad, la energía y la duración de las baterías. Todo un conjunto de aspectos que dictarán la elección de una tecnología o varias combinadas (Goto IoT | Introducción a protocolos IoT, s. f.).

Existen varios protocolos de la capa de transporte que se suele usar con los programas, como los navegadores web, los programas de transferencia de ficheros en red local e Internet y muchos más, algunos de estos protocolos son HTTP / HTTPS, WAN, LPWAN (ver figura 3).

Figura 3

Comparación de protocolos según ancho de banda de sus datos y alcance.



Fuente: Goto IoT | Introducción a protocolos IoT, s. f..

1.4.1.4 Hypertext Transfer Protocol (HTTP)

Es utilizado por navegadores web para solicitar archivos HTML de parte de los servidores remotos. Así, los usuarios podrán interactuar con dichos archivos mediante la visualización de las páginas web que cuentan con imágenes, música, vídeos, texto, etc.

El protocolo HTTP tiene como base TCP, el cual implementa un modelo de comunicación cliente-servidor. Existen tres tipos de mensajes que HTTP utiliza:

HTTP GET. Se envía un mensaje al servidor que contiene una URL con o sin parámetros.

HTTP POST. Se envía un mensaje al servidor que contiene datos en la sección «body» de la solicitud. Esto es hecho para evitar el envío de datos a través de la propia URL.

HTTP HEAD. Aquí se hace énfasis en la respuesta por parte del servidor. Este mensaje restringe lo que el servidor responde para que solamente responda con la información de la cabecera.

Cabe destacar el protocolo HTTPS, proporciona seguridad punto a punto (entre el cliente y el servidor web). El protocolo HTTPS utiliza el protocolo TLS (Transport Layer Security) que también utiliza TCP por encima. (Fernández, 2022)

1.4.1.5 Protocolos WAN

Los protocolos de capa física WAN (Wide Area Network) describen cómo proporcionar conexiones eléctricas, mecánicas, operacionales y funcionales para los servicios de una red de área amplia. Estos servicios se obtienen en la mayoría de los casos de proveedores de servicio WAN tales como las compañías telefónicas, portadoras alternas, y agencias de Correo, Teléfono, y Telégrafo.

Inicialmente, las WAN se construyeron con redes malladas de líneas privadas compradas a los operadores de telecomunicaciones, pero las arquitecturas WAN han avanzado para incluir servicios de conmutación de paquetes como la retransmisión de tramas y ATM, así como MPLS.

Con estos servicios, una sola conexión a un sitio puede ser conectada a muchas otras a través de la conmutación dentro de las redes de proveedores de servicios. Estos tipos de conexiones proporcionan medios de comunicación directos y en gran parte privados para sus diversas LAN. Eso le da velocidad y seguridad, pero no es barato. Para ciertos tipos de tráfico, Internet también puede entretrejerse para proporcionar conexiones WAN menos costosas (Computerworld, 2020).

1.4.1.6 Protocolos LPWAN

“Es un tipo de red de telecomunicaciones inalámbricas, diseñada para permitir comunicaciones de hasta más de 10 kilómetros de distancia con línea de vista y con una tasa de transferencia de datos baja, entre los 100 bps y los 200 kbps” (Porras & Salah, 2019, p 7)

Son tres las características técnicas de LPWAN que lo hacen coincidir con los requerimientos de IoT. Estas son:

- El alcance geográfico: LPWAN está diseñado para el transporte inalámbrico de datos entre dispositivos separados por distancias en el rango de kilómetros y no de metros.
- La cantidad de data transmitida: La idea de LPWAN es regular el transporte no constante de pequeñas cantidades de datos.
- El bajo consumo eléctrico: El protocolo se fundamenta en el uso de dispositivos cuyas baterías permiten una duración de años en lugar de semanas y meses.

Estas características además diferencian a LPWAN de otras tecnologías inalámbricas como Wi-Fi, Bluetooth, 3GPP y Zigbee. (La rosa 2021).

1.4.1.7 Tecnologías web

“Las tecnologías Web sirven para acceder a los recursos de conocimiento disponibles en Internet o en las intranets utilizando un navegador. Están muy

extendidas por varias razones: facilitan el desarrollo de sistemas de Gestión del Conocimiento (en lo adelante GC), su flexibilidad a la hora de expandir el sistema; su sencillez de uso y que imitan la forma de relacionarse de las personas, al poner a disposición de todos los conocimientos de los demás, por encima de jerarquías, barreras formales u otras cuestiones. Estas tecnologías pueden llegar a proporcionar recursos estratégicos, pero, evidentemente, no por la tecnología en sí misma, que está disponible ampliamente, sino por lo fácil que es personalizarla y construir con ella sistemas de GC propietarios de la empresa”. (Pérez, s.f)

Para el desarrollo de aplicaciones web se han generado múltiples tecnologías entre las que se encuentran: HTML, CSS, JAVASCRIPT.

1.4.1.8 Bases de datos

Una base de datos es una colección de datos almacenados y organizados de forma que un programa del ordenador pueda seleccionarlos rápidamente y capaces de ser: recobrados, actualizados, insertados y borrados. Cada base de datos ha sido diseñada para satisfacer los requisitos de información de una empresa u otro tipo de organización (Marqués, 2011).

Un sistema de bases de datos se divide en módulos que se encargan de cada una de las responsabilidades del sistema completo. Las funciones de un sistema de bases de datos se pueden dividir a grandes rasgos en dos componentes:

1.4.1.8.1 Gestor de almacenamiento

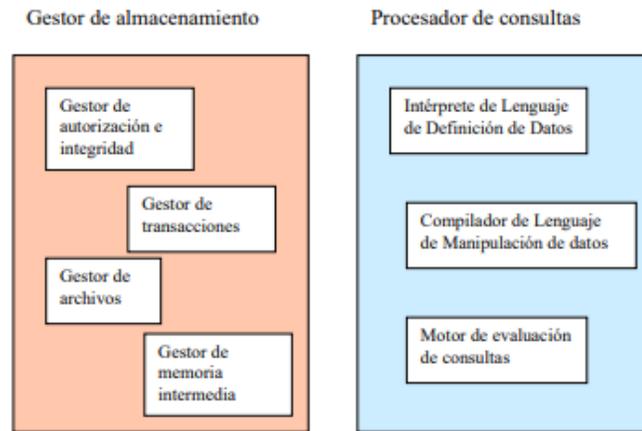
Se divide a su vez en varios módulos, como se indica en la figura 4. La función principal del gestor de almacenamiento es minimizar el movimiento entre el disco y la memoria principal.

1.4.1.8.2 Procesador de consulta

El procesador de consultas es importante porque ayuda al sistema de bases de datos a simplificar y facilitar el acceso a los datos (figura 4).

Figura 4

Componentes de gestor de almacenamiento y procesador de consultas.



Fuente: elaboración propia a partir de Gómez (2013).

1.4.1.9 Sensor

Un sensor es un dispositivo que detecta el cambio en el entorno y responde a alguna salida en el otro sistema. Se convierte un fenómeno físico en un voltaje analógico medible (o, a veces, una señal digital) convertido en una pantalla legible para humanos o transmitida para lectura o procesamiento adicional (Maloy, 2000)

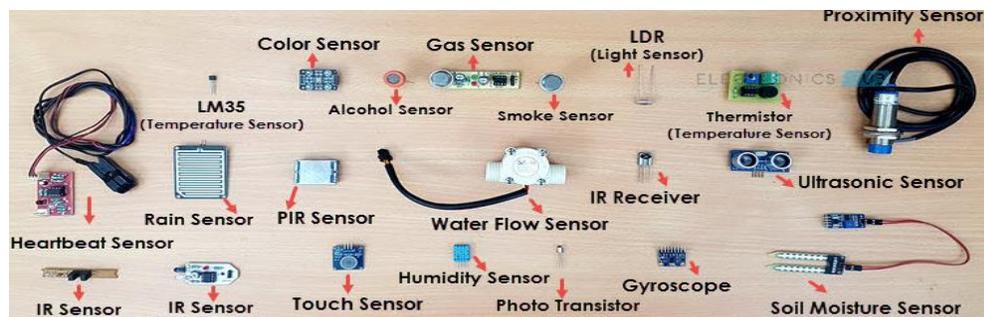
El diseño de sensores y transductores siempre involucra alguna ley o principio físico o químico que relaciona la cantidad de interés con algún evento medible. Los sistemas de monitorización y control requieren sensores para medir cantidades físicas tales como posición lineal, posición angular, desplazamiento entre otras (Colomer Barbera, 2018, p.79).

1.4.1.9.1 Sensores de proximidad

Pueden determinar la presencia o ausencia de un objeto, se produce mediante una salida digital de encendido/apagado. También puede proporcionar la posición a lo largo del tiempo de una manera continua mediante una señal eléctrica proporcional a la posición (ver figura 5) (mecatrónicaLatam, 2021).

Figura 5

Ejemplos de sensores.



Fuente: tomado de Mecatrónica Latam (2021).

1.4.1.10 Actuador

Es un dispositivo con capacidad de generar una fuerza que ejerce un cambio de posición, velocidad o estado de algún tipo sobre un elemento mecánico, a partir de la transformación de energía (Ramírez et al., 2014) (figura 6).

Figura 6

Servomotor.



Fuente: Llamas (2019).

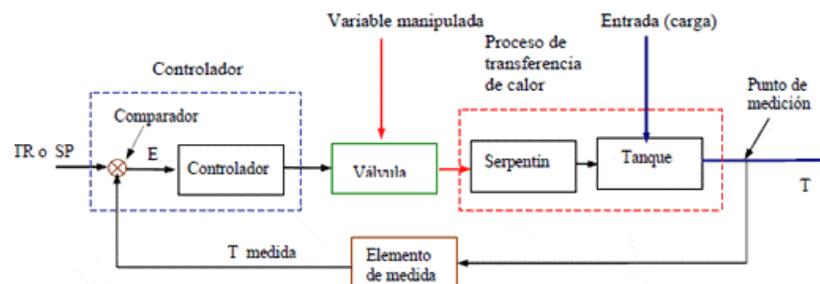
1.4.1.11 Sistemas de supervisión y control

Los sistemas de control y supervisión son independientes e integrados. La parte de control es la responsable de manejar los procesos, mediante una interfaz gráfica que comunique al usuario con los equipos. Mientras que los sistemas de supervisión son los encargados de promover la interfaz entre los sistemas de control, los usuarios y otros sistemas, así como realizar operaciones como registro de datos y exhibición de alarmas (Mikati, Santos & Armenta, 2012, p. 272).

Estos sistemas se encargan de la adquisición de datos, que sirve para obtener las señales que representan los fenómenos físicos a través del tiempo. Para esto se debe tener sensores o transductores actuadores. Las etapas de supervisión y el control se realizan mediante una interfaz gráfica de fácil acceso, la cual permite tomar decisiones y modificar el proceso si es requerido, así se vigila continuamente que el sistema cumpla con los parámetros establecidos sin la necesidad constante de un operador humano (ver figura 7) (Rodríguez, 2013).

Figura 7

Esquema de supervisión y control.



Fuente: (Villajulca, 2020).

1.4.1.11.1 Control en lazo cerrado

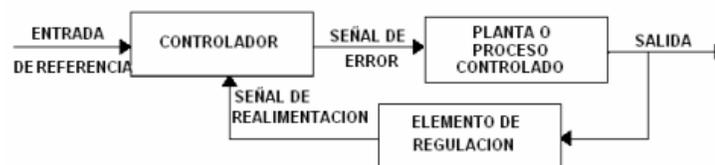
En la práctica, los términos control realimentado y control en lazo cerrado se usan indistintamente. En este tipo de sistemas se alimenta al controlador la señal de error de actuación,

que es la diferencia entre la señal de entrada y la señal de retroalimentación, con el fin de reducir el error y llevar la salida del sistema a un valor deseado (Ogata, 2010).

En otras palabras, el termino lazo cerrado implica el uso de acción de realimentación negativa para reducir el error del sistema (figura 8).

Figura 8

Sistema de control de lazo cerrado.



1.4.1.11.2 Control en lazo abierto

En un sistema de control de lazo abierto la salida no se mide ni se retroalimenta para compararla con la entrada. Los sistemas de control de lazo abierto son sistemas de control en los que la salida no tiene efecto sobre la señal o acción de control (figura 9) (Pérez et al, 2007).

Figura 9

Sistema de control de lazo abierto.



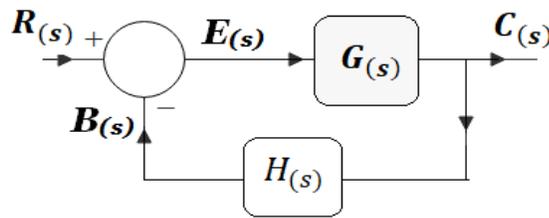
1.4.1.11.3 Función de transferencia

En los sistemas de control, las funciones son utilizadas para caracterizar las relaciones de entrada-salida de los componentes de un sistema; matemáticamente una función de transferencia

es la representación matemática que describe el modelo del sistema y que permite establecer su valor de salida ante una determinada entrada o perturbación al mismo. (Figura 10) (Ogata, 2010).

Figura 10

Función de transferencia en lazo abierto y cerrado.



Fuente: Ogata (2010).

- **Sistemas no lineales:** son sistemas que en su controlador carecen de un comportamiento estable alrededor de un punto de referencia, al saturarse con las señales de entradas grandes (Ogata, 2010).
- **Sistemas lineales:** un sistema se denomina lineal si se le aplica el principio de superposición. El principio establece que la respuesta producida por la aplicación simultánea de dos funciones de entradas diferentes es la suma de las dos respuestas individuales. Por tanto, para el sistema lineal, la respuesta a varias entradas se calcula tratando una entrada cada vez y sumando los resultados (Ogata, 2010).

1.4.1.11.4 Control PID

Es la estrategia de control más usada en las aplicaciones industriales; se estima que más del 90% de los lazos de control utilizan control PID, dado que es una estrategia simple, efectiva y no requiere una gran fundamentación teórica para su utilización. El control PID proporciona un

tratamiento tanto para la respuesta de estado estacionario como la respuesta transitoria, y ofrece una solución genérica y eficiente a los problemas de control del mundo real.

El diseño de controladores PID se puede lograr desde diferentes enfoques, que van desde métodos de ensayo y error, como acercamientos basados en el modelo del sistema (Lozano, Rodríguez y Giraldo, 2012).

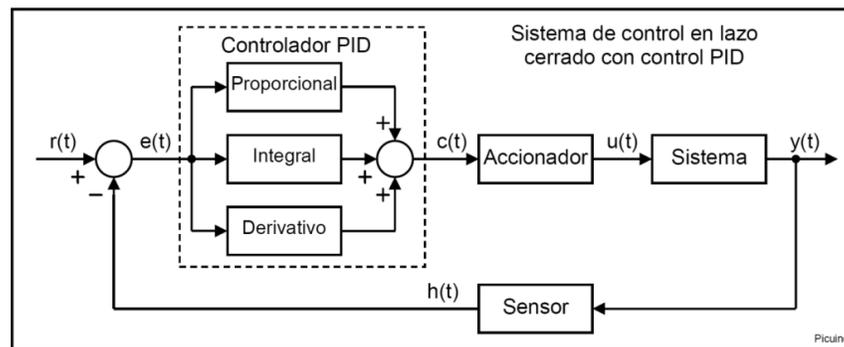
El controlador PID básico es aquel que combina las acciones proporcionales, derivativas e integrales mediante el algoritmo de control (ver ecuación 1).

$$u(t) = K_C \left(e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de(t)}{dt} \right) = P + I + D \quad (1)$$

El término proporcional contribuye a la reducción del error en régimen permanente, la acción integral tiene un efecto cualitativo sobre el error en régimen permanente, ya que aumenta el tipo de sistema y garantiza la anulación de este cuando la referencia es de tipo escalón y el termino derivativo permite cierta predicción del futuro error y por tanto juega un papel anticipativo (figura 11). (Ogata, 2010).

Figura 11

Control PID.



Fuente: (Controlador PID - Control Automático - Picuino, s. f.).

Capítulo 2

2.1. Estado del arte

2.1.1. Referencias internacionales

2.1.1.1 Diseño e implementación de laboratorio inteligente basado en IoT (Poongothai, Muthu Subramanian & Rajeswari, 2018)

Este trabajo tiene como fin desarrollar un sistema de laboratorio inteligente basado en IoT y tecnologías de aplicaciones móviles para monitorear las ocupaciones en general del laboratorio, integrado el consumo de energía, la implementación de dispositivos y fronteras del medio ambiente por medio de sensores, proporcionando de esta forma un ámbito inteligente con eficiencia energética. Este sistema controla y monitorea los dispositivos del laboratorio IoT usando herramientas Node-RED y Aplicación Móvil en ANDROID STUDIO.

Esto es posible mediante el comando que se envía desde el teléfono móvil al servidor MQTT. Luego, el servidor publica el comando en el respectivo módulo ESP8266 presente en un kit IoT al que están conectados todos los dispositivos en el laboratorio.

2.1.1.2 Integración de IoT con controlador LQR-PID para vigilancia en línea, control de flujo y presión en un sistema de transporte de fluidos (Priyanka, Maheswari, Thangavel & PonniBala, 2020)

Este proyecto de investigación se enfoca en desarrollar una arquitectura inteligente basada en IoT integrado para realizar el monitoreo y control en línea de la presión y el caudal en el sistema de transporte de fluidos. Por lo tanto, se propone un sistema SCADA con controlador LQR-PID que monitorea y controla la planta, a través de una interfaz IoT de servidor en línea de alto nivel

para identificar fugas y grietas en la tubería en la etapa inicial y apagar de emergencia el sistema para detener la bomba antes de que conduzca a situaciones catastróficas.

2.1.1.3 Implementación de algoritmos de control estocástico en la nube mediante herramientas de programación IoT para un motor DC (Angos, 2021)

En este trabajo de titulación se presenta el desarrollo de algoritmos de control óptimos estocásticos en la nube para el control de un motor DC, mediante herramientas de programación IoT. Se establecieron los parámetros de funcionamiento de la planta, la ecuación del sistema mediante un muestreo y estimación de datos con la ayuda del software Matlab, luego, se desarrolló los algoritmos de control partiendo de las matrices de estado que se originan de la función de transferencia; con estas características se programó los controles en la herramienta Node-Red con el análisis previo de configuración del protocolo MQTT y la herramienta Remoteit.

Se realizó la comparativa de operación de cada uno de los algoritmos de control aplicados, por medio de pruebas de comunicación y tiempos de respuesta que se puede apreciar en cualquier dispositivo inteligente mediante el uso de la aplicación RemoteRED.

2.1.1.4 Control y monitoreo IMC-PID en tiempo real del proceso de extracción de aceites esenciales mediante IoT (Hasinah Binti Johari et al., 2021).

Este estudio propone un sistema de monitoreo remoto de temperatura para mejorar el control y monitoreo de la temperatura de destilación de vapor utilizando IoT. Además, tener un esquema de control de modelo interno (IMC) en el controlador PID brinda una ventaja adicional en el sistema. En el desarrollo del software se utilizó el servidor Blynk que conecta el ESP8266 NodeMCU con la aplicación Blynk en el teléfono inteligente. La implementación de IoT a través de esta aplicación facilita el monitoreo del proceso de destilación sin el operador en el sitio.

2.1.1.5 Monitoreo y control PID de un sistema RC usando IoT mediante Raspberry (Palomino, 2020b)

Este proyecto presenta la implementación de un sistema de monitoreo y control para un circuito RC de primer orden a través de Internet de las cosas (IoT) usando la plataforma Thingspeak. El control y monitoreo se realiza a través de dos canales de la plataforma, donde una GUI en Raspberry sube los parámetros de un controlador PID a uno de los canales y otra Raspberry remotamente descarga los parámetros de control y los aplica al sistema resistivo mediante Arduino, así mismo sube a la plataforma las mediciones en tiempo real del estado actual de la salida, referencia, señal de control y error del sistema.

2.1.2 Referencias nacionales

2.1.2.1 Simulación del sistema de control en la represa de Hidroituango para controlar el nivel de agua del río Cauca utilizando IoT (Rodríguez & Olarte, 2019)

Este proyecto realiza la simulación de un sistema de control por medio de una red de sensores, los cuales serán encargados de generar las respectivas alertas a los municipios aledaños y así prevenir posibles desastres, los actuadores serán encargados de activar las compuertas cuando lo consideren necesario ya sea para llenar las tinajas o para desviar el caudal del río kilómetros después. La simulación con la ayuda del software “Packet Tracer” dará un esquema general de la red de sensores que permite dar alertas del caudal del río Cauca y su inminente emergencia en cuanto a los poblados más cercanos a la ubicación de la represa de Hidroituango. Estos sensores estarán constantemente monitoreando la presión del agua, la profundidad y el caudal, ya que al alterarse una de estas medidas se puede presentar una emergencia.

2.1.2.2 Diseño de un sistema de monitoreo y control agroclimático para lombricultivo en la Universidad Autónoma de Occidente, Valle del Cauca (Becerra & Stenberg, 2020)

Este proyecto diseña un sistema de control que cumple con el mejoramiento del proceso del lombricompostaje permitiendo sensor, captar y almacenar los datos registrados para su posterior análisis bajo una plataforma IoT permanentemente, brindando al usuario un historial de los valores de las variables críticas sensadas y la visualización del comportamiento de estas durante un periodo de tiempo, para el lombricultivo que se encuentra en el vivero de la universidad autónoma de occidente; a través de dos dispositivos principales que son, el coordinador, que se encarga de recibir y controlar el tráfico de datos y el emisor, que será el dispositivo con que contara cada cama para el monitoreo y control permanente.

2.1.2.3 Sistema de riego basado en la Internet de las Cosas (Anaya, A. J., Ríos, J., Carvajal, D. A., Peluffo D. H. & Castro, J. A., 2016).

En este artículo se describe el diseño, elaboración e implementación de un sistema de riego empleando hardware y software libre, redes de sensores inalámbricas (WSN), actuadores, dispositivos de comunicación inalámbrica y herramientas TIC, con el fin de crear un ambiente donde el Internet de las Cosas (IoT) y la Agricultura de Precisión ofrezcan al usuario un mejor control del riego sobre el cultivo teniendo en cuenta la evapotranspiración.

Para el control de los actuadores la Raspberry Pi se comunica con el Arduino y con el apoyo de los módulos Relés que permiten manejar voltajes de 12V o 110 V. Este módulo contiene una copia local del servidor que está en la nube, permitiendo que el sistema pueda funcionar sin conexión a Internet.

2.1.3 Referencias regionales

2.1.3.1 Estrategia metodológica para la transferencia de tecnologías basadas en internet de las cosas con aplicación en la caficultura del Cauca (Quiñones, 2021)

En este artículo se presenta una estrategia metodológica para la transferencia de tecnologías basadas en IoT con aplicación en la caficultura del Cauca con el fin de promover el uso de herramientas TIC para mejorar la productividad y competitividad del sector cafetero del Cauca. Se identificó las principales experiencias en la implementación de tecnologías basadas en internet de las cosas, en el sector cafetero a nivel nacional e internacional; además, se analizó los requerimientos técnicos y socioculturales para implementar tecnologías basadas en internet de las cosas, aplicables al sector cafetero y se estructuró la propuesta metodológica para la implementación de tecnologías basadas en IoT.

2.1.3.2 Desarrollo de un sistema IoT para monitoreo y apoyo en el dimensionamiento de consumo energético para la Industria Licorera del Cauca (Guevara, 2020)

Este trabajo de grado muestra el desarrollo e implementación de un sistema IoT para el monitoreo de variables eléctricas, que permita dimensionar el consumo energético en circuitos internos de la Industria Licorera del Cauca. El diseño del sistema se basa en dos componentes: La medición y la interfaz de usuario.

Se realizó el diseño e implementación de un medidor polifásico de alto rendimiento y precisión basado en tecnología LoRa, así como también la programación del dispositivo y ajuste de parámetros para medición. En segunda instancia se desarrolló una plataforma web para la visualización de datos eléctricos, para identificar consumos en el tiempo.

2.1.4 Comparación de proyectos

Para condensar la información sobre los referentes anteriormente mencionados, se realiza una comparación de los proyectos teniendo en cuenta aspectos como componentes: tipo FOSS, sistema de control PID basado en IoT, monitoreo de variables en tiempo real, interfaz gráfica de usuario, ajustes por usuario y plataforma educativa (ver tabla 2).

Tabla 2.

Comparación de proyectos en el marco referencial.

Proyectos	Componentes tipo FOSS	Sistema de control basado en IoT	de PID en tiempo real	Monitoreo de variables en tiempo real	Interfaz gráfica de usuario	Ajustes por usuario	Plataforma educativa
Internacional							
1	✓	X		✓	✓	✓	✓
2	✓	✓		✓	✓	X	X
3	✓	✓		✓	✓	X	✓
4	✓	✓		✓	✓	X	X
5	X	✓		✓	✓	✓	✓
Nacional							
6	✓	X		✓	✓	X	X
7	✓	X		✓	✓	X	✓
8	✓	X		X	✓	X	X
Regional							
9	✓	X		X	X	X	X
10	✓	X		✓	✓	X	X

Después de analizar el marco referencial, se refleja la importancia de implementar un proyecto como este en cualquier parte del mundo, pues para muchas personas es importante una plataforma que permita entender el funcionamiento de un controlador hasta el de un sistema basado en IoT. Esto se evidencia en algunas referencias internacionales donde muchos países ya cuentan con sistemas de control basados en IoT, como por ejemplo los proyectos (Priyanka, Maheswari, Thangavel & Ponni, 2020), (Hasinah, Fazalul, Adnan & Tajjudin, 2020) y (Escobar & Sulca, 2021), proponen sistemas de control basados en IoT, los cuales llevan un continuo monitoreo de las variables, pero no cuentan con una interfaz que permita al usuario manipular la acción de control en la planta. Para el proyecto (Poongothai, Muthu & Rajeswari, 2018), tiene un sistema de control que cuentan con una interfaz gráfica de usuario, que permite manipular la acción de control a la planta, pero el control que se realiza es solamente ON-OFF es decir en activar y desactivar compuertas, en (Velázquez, Pineda, Méndez y Martínez, 2020) se tienen varios aspectos positivos, pues a pesar de no contar con una plataforma basada en componentes FOSS, si cuenta con un control PID basado en IoT y posee además una interfaz web donde se puede realizar un continuo monitoreo de las variables.

Para el panorama nacional se presenta algo similar en los trabajos (Becerra & Stengber, 2020), (Rodríguez & Olarte, 2019) y (Anaya et al, 2016), los cuales llevan un monitoreo en tiempo real de las variables a través de una interfaz Web, pero esta no proporciona la posibilidad de influir en la acción de control del sistema.

Por último, en el panorama regional el proyecto de Quiñones (2021) se centra mayormente en ver los requerimientos del lugar donde se implementará el sistema, pero al no monitorear las variables en tiempo real, el usuario no puede modificar la acción de control a la planta, lo que podría generar dudas o confusiones sobre el funcionamiento de un controlador o un sistema de IoT.

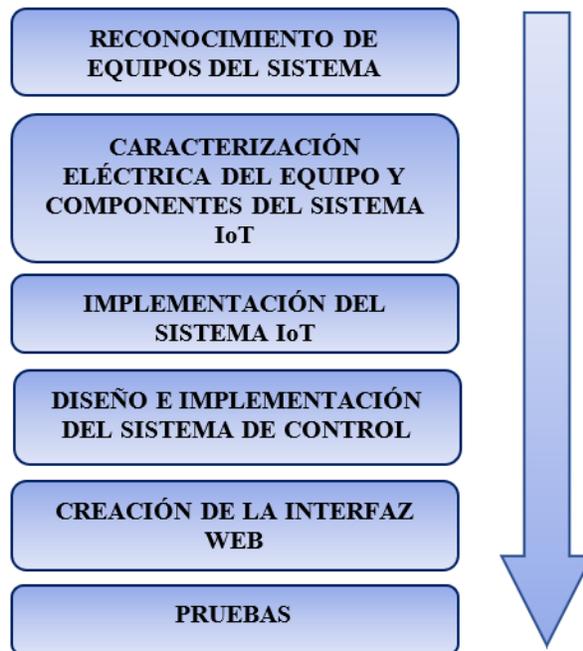
Capítulo 3

3.1. Metodología

El proyecto se desarrolló en seis etapas (figura 12), en las que se describen los procesos mediante los cuales se finalizó el trabajo. En primer lugar se estudia el funcionamiento de los elementos que conforman un sistema viga – bola y su conexión, para establecer el tipo de sensor que permita medir la posición de la bola y poder hacer su respectiva calibración, después se procede a implementar el sistema con todos sus elementos para mirar cómo funcionan los circuitos en conjunto y poder realizar la programación del sistema de control que tiene como objetivo hacer que todo el sistema funcione, finalmente se crea una interfaz web amigable con el usuario con el fin de proporcionarle el control del sistema remotamente.

Figura 12

Desarrollo metodológico.



3.1.1 Reconocimiento de equipos del sistema viga – bola

3.1.1.1 Viga

La viga de madera Triplex (ver figura 13) permite el desplazamiento de la bola que tiene encima, ya que puede ser inclinada con respecto a su eje central, sus características son las siguientes (ver tabla 3).

Figura 13

Viga de la planta.



Tabla 3.

Características de la viga del sistema.

Ficha técnica	
Grosor	4mm
Largo	47,4cm
Ancho	2.5cm
Peso	80g

3.1.1.2 Bola

Se encarga de rodar sobre la viga, afectando la información que capta el sensor (figura 14). Tiene las siguientes características (ver tabla 4).

Figura 14

Bola.

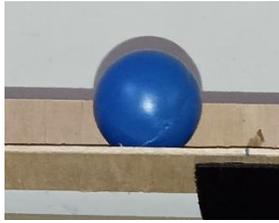


Tabla 4.

Características de la bola.

Ficha técnica	
Radio	1,4cm
Masa	2,75g
Color	Azul

3.1.1.3 Eje o pivote

Es el componente que sirve como centro de giro para la viga, está realizado con PLA a 15% de concentración (figura 15).

Figura 15

Eje de la viga.



3.1.1.4 Soporte

Es el soporte de la bola y viga, además, se encarga de darle altura a la barra para que pueda realizar el movimiento de oscilación, haciendo mover a la bola (figura 16). El soporte está constituido por material PLA a 15% de concentración y un tornillo.

Figura 16

Soporte del sistema viga – bola.



3.1.2 Caracterización eléctrica del equipo y componentes del sistema IoT

3.1.2.1 Sensores y actuadores

Los sensores que se utilizan en el desarrollo del proyecto son los siguientes.

3.2.1.1 Modulo sensor de distancia GY-530 VL53L0X. El sensor GY-530 VL53L0X de distancia laser (figura 17), es utilizado para medir la distancia del centro de la viga a la bola, consiste en el envío de un pulso láser de luz infrarroja, este mide el tiempo que tarda la luz en ir hasta el objetivo y volver de nuevo al sensor. Este sensor incorpora un emisor láser 940nm VCSEL (Vertical Cavity Surface-Emitting Laser), un detector de matriz SPAD (Single Photon Avalanche Diodes) que utiliza su electrónica interna (FlightSense™) para realizar los cálculos de distancia necesarios (figura 18).

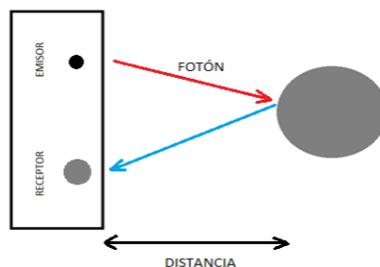
Figura 17

Sensor de distancia GY-530 VL53L0X



Figura 18

Funcionamiento Sensor de distancia láser.



La comunicación entre el sensor y la tarjeta de adquisición de datos se basa en la tecnología I2C. En la tabla 5 se observan las dimensiones de este sensor y los pines de entrada/salida; VCC, GND, SCL, SDA, GPI01 y XSHUT.

Tabla 5.

Características del sensor de distancia GY-530 VL53L0X.

Ficha técnica	
Tamaño	25.0 x 12.20 x 1.00 mm
Rango de medición	50mm – 1200mm
Voltaje de alimentación	2.6V – 5.0V
Temperatura de operación	-20°C – 70°C
Precisión	± 30mm
Interfaz digital	I2C 400kHz
Laser	VCSEL de 940nm

6.2.1.2 Servomotor FUTABA S3003. El actuador utilizado, es un servomotor de RC comercial marca Futaba, referencia S3003 (figura 19), es capaz de ubicarse en cualquier posición dentro de un rango de operaciones (generalmente 180°) y mantenerse en dicha posición, por medio de una señal PWM que se le introduce, este va a modificar su ángulo de giro, modificando así, el ángulo de inclinación de la barra (tabla 6).

Figura 19

Servomotor FUTABA S3003.



Tabla 6.

Características del servomotor FUTABA S3003.

Ficha técnica	
Velocidad	0.23 seg/60 grados
Par de salida	3.2 kg-cm
Dimensiones	40.4 x 19.8 x 36 mm
Peso	37.2 g
Voltaje de operación	4.8 – 6 V

3.1.2.2 Tarjeta de adquisición de datos

3.1.2.2.1 Módulo ESP32. Esta placa electrónica está constituida por varias entradas digitales, analógicas y protocolos de comunicación que permiten realizar sistemas de sensado y

control a un bajo costo (figura 20), cuenta con un módulo WiFi que permite la conexión de la placa a internet. Las especificaciones técnicas se muestran en la tabla 7.

Figura 20

Módulo ESP32.



Tabla 7.

Características Módulo ESP32.

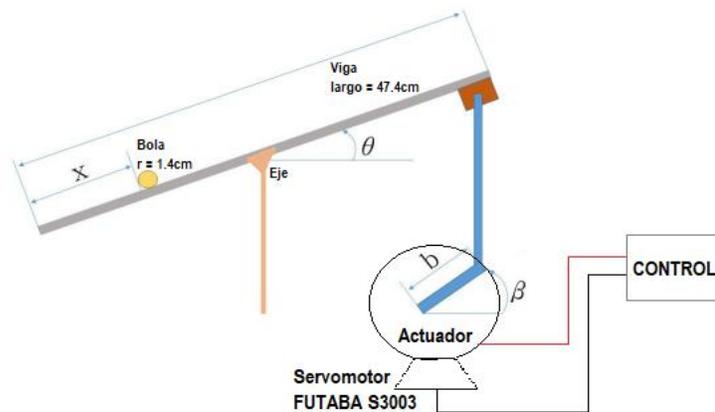
Ficha técnica	
Voltaje de operación	3.3V – 5V
Corriente por cada pin	40Ma
Pines digitales de entrada y salida	2 PWM, 34 digitales y 18 analógicos.
Protocolos de comunicación	I2C, SPI, CAN, PWM
Módulo WiFi.	802.11 b/g/n en la banda de los 2.4GHz
Velocidad de conectividad WiFi	150Mbits/s

3.1.2.2.1 Componentes del sistema viga – bola

En la figura 21, se muestran los componentes que conforman el sistema viga – bola, con las especificaciones técnicas señaladas anteriormente.

Figura 21

Montaje del sistema viga – bola de prueba.



Se utiliza el material triplex como viga, ya que este es liviano y permite el funcionamiento correcto del servomotor, además es un material económico y fácil de conseguir.

3.1.2.3.1 Cálculo del modelo matemático del sistema viga y bola

Se realiza la simulación de este sistema antes de implementarlo en el sistema físico, para obtener las constantes PID que hacen un control óptimo de posición de la bola, o adquirir una primera aproximación de las constantes; para ello, lo primero que se debe hacer es obtener el modelado de la planta, en este caso el modelado de la maqueta barra y bola.

Para obtener el modelado de la planta se utilizan leyes físicas que dan una ecuación diferencial que representa la dinámica del sistema. Teniendo en cuenta el apartado 1.4.1 “Sistemas de supervisión y control”, la entrada y salida de la planta son las descritas por la figura 22.

Figura 22

Entrada y salida de la función de transferencia del sistema.



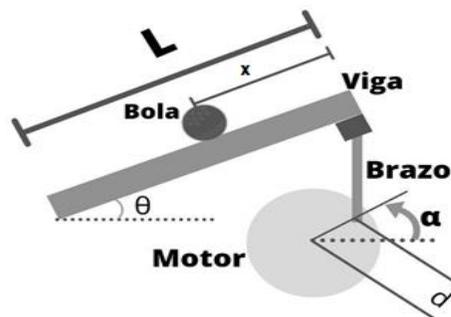
Con lo cual se obtiene la función de transferencia del sistema viga – bola:

$$G(s) = \frac{x(s)}{\theta(s)} \quad (2)$$

Se observa que la entrada es la señal del ángulo de giro del servomotor en radianes, $\theta(s)$ [rad], y la salida de la planta es la señal de posición de la bola en la barra en metros, $X(s)$ [m] (ver figura 23), por tanto, se obtienen los grados de libertad que posee el sistema. En este caso se encuentran dos grados de libertad: la posición de la bola y el ángulo de inclinación de la barra.

Figura 23

Esquema del sistema para el modelado rueda sin deslizarse.



Para obtener los parámetros físicos de la maqueta se han medido las siguientes constantes:

- m : Masa de la bola $\rightarrow m = 27,5\text{g} \rightarrow m = 0,0275\text{kg}$
- R : Radio de la bola $\rightarrow R = 14.0\text{mm}$ ($d = 28.0\text{mm}$) $\rightarrow R = 0.028\text{m}$
- d : Desplazamiento de la biela $\rightarrow d = 3.6\text{cm} \rightarrow d = 0,036\text{m}$
- g : aceleración gravitacional (gravedad) $\rightarrow g = -9,8\text{m/s}^2$
- L : Longitud de la barra $\rightarrow L = 47,4\text{cm} \rightarrow L = 0,474\text{m}$

- J: Momento de inercia de la bola → Sabiendo que para una esfera el momento de inercia con respecto al centro se expresa como: $J = \frac{2}{5} \times m \times R^2 \rightarrow J = 4,4 \times 10^{-6} \text{ kg.m}^2$.
- x: Coordenada de posición de la bola (a la salida de la planta en “m”, a la hora de calcular el error, en “mm”).
- α : Coordenada del ángulo de inclinación de la barra [rad].
- θ : Ángulo de giro del servomotor [rad]

Para el modelado de la planta, se utilizará la ecuación de Lagrange (ver ecuación 3):

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} \right) - \frac{\delta L}{\delta q_i} + \frac{\delta F}{\delta \dot{q}_i} = Q_i \quad (3)$$

Siendo: q_i = Cada una de las coordenadas generalizadas.

$$L = T - V \quad (4)$$

L → La lagrangiana, donde T es la energía cinética y V la potencial.

F = Fuerza de disipación de Rayleigh.

Q = Fuerza generalizada.

Como coordenadas generalizadas se eligen la posición de la bola en la barra “q1” (x) y el ángulo de inclinación de la barra con respecto a la horizontal “q2” (α).

Se calcula la energía cinética total del sistema que será la que aporte la barra más la que aporte la bola, puesto que son los dos elementos en movimiento (ver ecuación 5):

$$T = T_{barra} + T_{bola} \quad (5)$$

Para la barra (ver ecuación 6):

$$T_{barra} = T_{rot} = \frac{1}{2} J_{barra} \omega_{barra}^2 \quad (6)$$

Se reemplaza ω^2 por \dot{q}_2^2 (ver ecuación 7):

$$T_{barra} = \frac{1}{2} J_{barra} \dot{q}_2^2 \quad (7)$$

Para la bola (ver ecuación 8):

$$T_{bola} = T_{rot} + T_{tras} = \frac{1}{2} J_{bola} \omega_{bola}^2 + \frac{1}{2} m v^2 \quad (8)$$

Reemplazando la ecuación 6 y la ecuación 7 en la ecuación 4 se obtiene (ver ecuación 9):

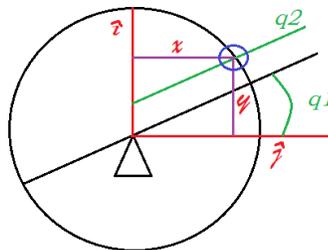
$$T = \frac{1}{2} J_{barra} \dot{q}_2^2 + \frac{1}{2} J_{bola} \omega_{bola}^2 + \frac{1}{2} m v^2 \quad (9)$$

En la figura 24, se puede ver que la bola se desplaza verticalmente gracias a la barra (y), por lo tanto, se obtiene que (ver ecuación 10):

$$v^2 = \dot{x}^2 + \dot{y}^2 \quad (10)$$

Figura 24

Representación de la trayectoria de la bola en la barra.



Donde la variable “x” y la variable “y” se representa como (ver ecuación 11 y 12):

$$x = q_1 \cos(q_2) \quad (11)$$

$$y = q_1 \sin(q_2) \quad (12)$$

Se deriva la ecuación 10 y la ecuación 11 con respecto al tiempo (ver ecuación 13 y 14):

$$\dot{x} = \dot{q}_1 \cos(q_2) - q_1 \dot{q}_2 \sin(q_2) \quad (13)$$

$$\dot{y} = \dot{q}_1 \sin(q_2) + q_1 \dot{q}_2 \cos(q_2) \quad (14)$$

Se reemplaza la ecuación 13 y 14 en la ecuación 10, tenemos (ver ecuación 15):

$$v^2 = (\dot{q}_1 \cos(q_2) - q_1 \dot{q}_2 \sin(q_2))^2 + (\dot{q}_1 \sin(q_2) + q_1 \dot{q}_2 \cos(q_2))^2 \quad (15)$$

Con la siguiente identidad trigonométrica (ver ecuación 16),

$$\sin^2(\theta) + \cos^2(\theta) = 1 \quad (16)$$

Se aplican operaciones algebraicas a la ecuación 15, dando (ver ecuación 17):

$$v^2 = \dot{q}_1^2 + q_1^2 \dot{q}_2^2 \quad (17)$$

La velocidad angular de la bola se calcula con la condición de rodadura sin deslizar, como (ver ecuación 18):

$$v = R\omega \quad (18)$$

Donde se puede obtener la siguiente expresión (ver ecuación 19):

$$\omega = \frac{v}{R} \rightarrow \omega = \frac{\dot{q}_1}{R} \quad (19)$$

Al elevar al cuadrado la ecuación 19 se tiene la siguiente expresión (ver ecuación 20):

$$\omega^2 = \left(\frac{\dot{q}_1}{R}\right)^2 \quad (20)$$

Al sustituir la ecuación 17 y 20 en la ecuación 9, se tiene (ver ecuación 21):

$$T = \frac{1}{2}J_{barrera}\dot{q}_2^2 + \frac{1}{2}J_{bola}\left(\frac{\dot{q}_1}{R}\right)^2 + \frac{1}{2}m(\dot{q}_1^2 + q_1^2\dot{q}_2^2) \quad (21)$$

Aplicando operaciones algebraicas, se reescribe la ecuación 21 por (ver ecuación 22):

$$T = \frac{1}{2}\left(\frac{J_{bola}}{R^2} + m\right)\dot{q}_1^2 + \frac{1}{2}(J_{barrera} + mq_1^2)\dot{q}_2^2 \quad (22)$$

En cuanto a la energía potencial del sistema, siguiendo el mismo procedimiento que para la energía cinética se puede calcular la energía potencial de la barra, tomando como punto de referencia la posición horizontal de la misma (ver ecuación 23).

$$V = V_{barrera} + V_{bola} \quad (23)$$

Así se tiene que (ver ecuación 24),

$$V_{bola} = mgq_1 \sin(q_2) \quad (24)$$

La energía potencial de la barra se aproxima a cero ya que la distancia desde la línea que pasa por el centro de la bola hasta el eje del servomotor se desprecia (ver ecuación 25).

$$V_{barrera} \approx 0 \quad (25)$$

Reemplazando la ecuación 21 y 23 en la ecuación 3 se obtiene la siguiente ecuación general (ver ecuación 26):

$$L = \frac{1}{2} \left(\frac{J_{bola}}{R^2} + m \right) \dot{q}_1^2 + \frac{1}{2} (J_{barras} + m q_1^2) \dot{q}_2^2 - m g q_1 \sin(q_2) \quad (26)$$

La primera ecuación de Lagrange está dada por la ecuación 25, donde se desprecian fuerzas disipativas, por tanto (ver ecuación 27):

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_i} \right) - \frac{\delta L}{\delta q_i} = 0 \quad (27)$$

Resolviendo paso a paso las derivadas, se tiene (ver ecuación 28 y 29):

$$\frac{\delta L}{\delta \dot{q}_1} = \left(\frac{J_{bola}}{R^2} + m \right) \dot{q}_1 \quad (28)$$

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_1} \right) = \left(\frac{J_{bola}}{R^2} + m \right) \ddot{q}_1 \quad (29)$$

Por último, se obtiene que (ver ecuación 30):

$$\frac{\delta L}{\delta q_2} = m q_1 \dot{q}_2 - m g \sin(q_2) \quad (30)$$

Reemplazando la ecuación 28 y la ecuación 29 en la ecuación 26, se obtiene la primera ecuación del movimiento del sistema (ver ecuación 31):

$$\left(\frac{J_{bola}}{R^2} + m \right) \ddot{q}_1 + m g \sin(q_2) - m q_1 \dot{q}_2 = 0 \quad (31)$$

La segunda ecuación de Lagrange está dada por (ver ecuación 32):

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_2} \right) - \frac{\delta L}{\delta q_2} = \tau \quad (32)$$

Donde τ es el torque externo aplicado a la viga. Se resuelve la ecuación 31 similar a la ecuación 27, dando (ver ecuación 33, ecuación 34 y 35):

$$\frac{\delta L}{\delta \dot{q}_2} = (J_{barras} + m q_1^2) \dot{q}_2 \quad (33)$$

$$\frac{d}{dt} \left(\frac{\delta L}{\delta \dot{q}_2} \right) = 2m q_1 \dot{q}_1 \dot{q}_2 + (J_{barras} + m q_1^2) \ddot{q}_2 \quad (34)$$

$$\frac{\delta L}{\delta q_2} = -m g q_1 \cos(q_2) \quad (35)$$

Reemplazando la ecuación 33 y la ecuación 34 en la ecuación 31, se tiene la segunda ecuación de movimiento del sistema bola y barra (ver ecuación 36):

$$2m q_1 \dot{q}_1 \dot{q}_2 + (J_{barras} + m q_1^2) \ddot{q}_2 + m g q_1 \cos(q_2) = \tau \quad (36)$$

Las ecuaciones del movimiento del sistema están representadas por la ecuación 31 y 36.

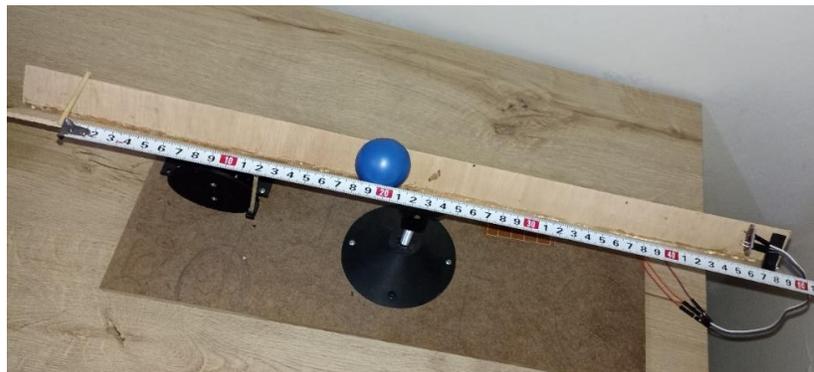
3.1.2.3 Caracterización de sensor y actuador

Una vez implementado el sistema viga – bola, se caracteriza el sensor y actuador usados:

3.1.2.3.1 Sensor de distancia. Para la caracterización del sensor de distancia GY-530 VL53L0X, se instala una librería llamada “VL53L0X.h”, también, se utiliza la bola de radio 1.4cm como objeto de prueba y una cinta métrica. Los valores de distancia se registraron por medio del IDE de Arduino, luego estos valores se comparan con los datos de la cinta métrica (figura 25).

Figura 25

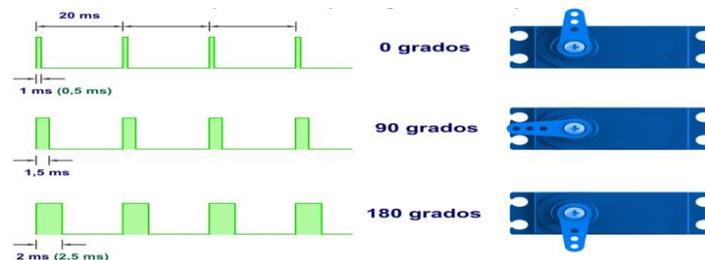
Registro de los valores de distancia.



3.1.2.3.2. Actuador del sistema. En la implementación del servomotor FUTABA S3003, se instaló la librería llamada “ESP32Servo.h”, para tener compatibilidad con la tarjeta de adquisición de datos ESP32, se controla el servomotor mediante señales PWM (ver figura 26). Para ello, se utiliza la función *servo.write (ángulo)* que ajusta la posición del servo con un valor entre 0 y 180 que sería el rango de movilidad del servo de 0° a 180°.

Figura 26

Ángulo del servomotor según PWM

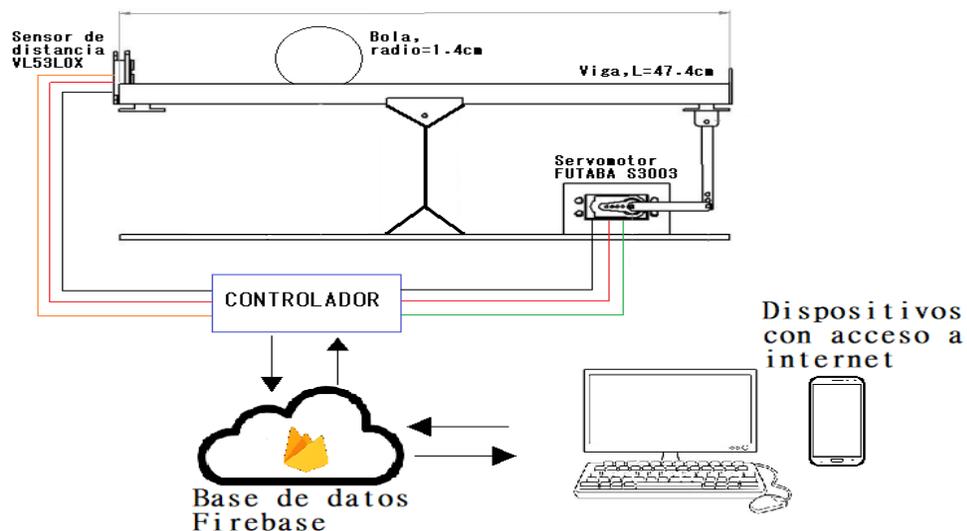


3.1.3 Diseño e implementación del sistema de control para la posición

El sistema de control se basa en el funcionamiento del sistema de sensado de distancia; a continuación, se muestra un esquema general (ver figura 27).

Figura 27

Esquema del sistema de control.



Conociendo la manera como trabajan los sistemas de supervisión y control juntos, fue necesario saber las condiciones que afectan y acortan la vida útil de los componentes del sistema; por ello, se tomaron las acciones de control pertinentes en función de la variable sensada, tanto para proteger el actuador como para garantizar la acción de control (figura 28) (ver tabla 8).

Figura 28

Diagrama de flujo de sistema de control.

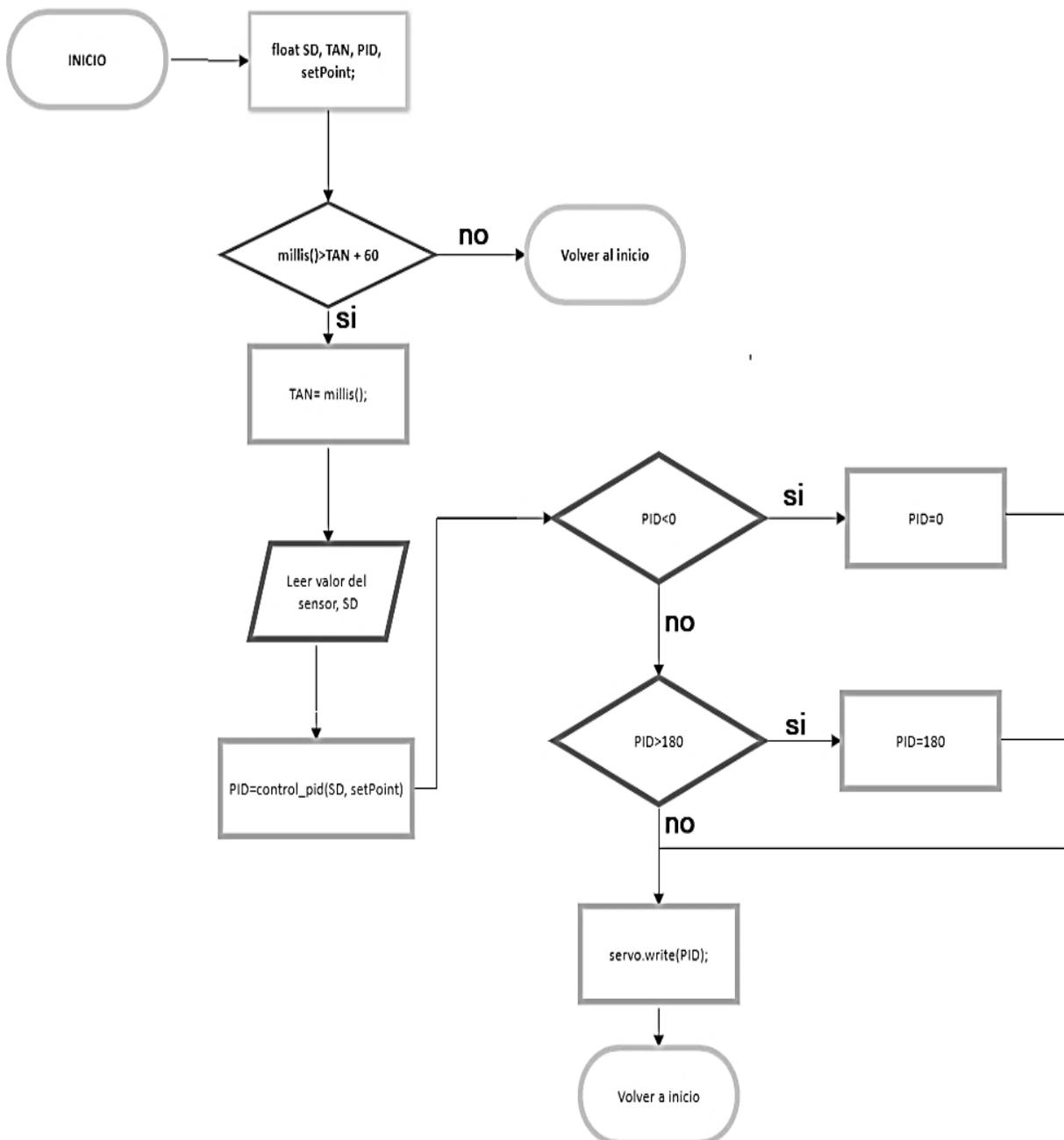


Tabla 8.

Nombre de las variables.

Variable	Significado
SD	Señal de distancia por el sensor
TAN	Tiempo anterior (ms)
PID	Señal del controlador PID
SetPoint	Señal de la distancia deseada

3.1.3.1 Interfaz de usuario

La interfaz se realizó con HTML, CSS y ReactJS como librería de desarrollo Front-end. Se escogió como base de datos Firebase, ya que se ajusta a las necesidades detectadas, pues permite de una manera simple, guardar y visualizar los datos que entrega el sistema de control. ReactJS ofrece un enfoque de programación reactiva que ayuda a visualizar la aplicación, según el estado del usuario, esta librería es de código abierto y es mantenida por la compañía Facebook.inc y la comunidad (Interempresas, 2020). Firebase es un software as a Service (SaaS), permite que sea fácil integrar el hardware, es mantenido por la compañía Google.

Para la interfaz web se crearon tres páginas que permiten al usuario controlar la planta de manera interactiva con credenciales o realizar una simulación de control y documentarse sobre el sistema sin credenciales (ver figura 29).

Figura 29

Ventana de visualización de la interfaz web.



3.1.3.2 Control interactivo de la planta

Permite al usuario manipular los valores K_p , K_i y K_d del controlador PID con fin de seleccionar los óptimos para sean enviados a la base de datos. Genera alertas sobre la planta y además permite visualizar la respuesta del sistema en una gráfica distancia contra tiempo.

3.1.3.2 Control teórico de la planta

Permite al usuario ver el modelado teórico de la planta, aproximar los valores de K_p , K_i y K_d adecuados según el modelado teórico y ver la respuesta teórica del sistema viga – bola, además, da al usuario final una guía sobre el uso de la plataforma, ayudando a evitar malas prácticas cuando se haga una práctica real de laboratorio.

3.1.3.3 Documentación

Este apartado muestra al usuario las tecnologías utilizadas, el código fuente y ofrece una guía básica para realizar una réplica de este proyecto, mostrando el paso a paso de cómo construirlo.

3.1.3.4 Prueba

Para evaluar el sistema de control implementado, primero se compararon los datos obtenidos por el sensor con los instrumentos de referencia, luego se caracterizó el sensor de distancia y después de implementar todo sistema viga – bola, Se realizaron pruebas de control sin utilizar la conexión a internet y así poder tener un estimado del comportamiento del sistema. Luego se conectó el sistema a internet para interactuar con la base de datos y se manipulo el sistema con la interfaz web. Esto se realizó con datos de prueba y posteriormente con los datos reales. Los resultados fueron tratados estadísticamente para obtener las respectivas gráficas y hacer su correspondiente análisis.

Capítulo 4

4.1 Análisis y resultados

4.1.1 Dimensionado del sensor y el actuador

4.1.1.1 Caracterización y calibración del sensor

Sensor de distancia. Para las pruebas realizadas se encuentra que el comportamiento de la salida del sensor está determinado por una función lineal, donde la ecuación característica es programada por Arduino y permite así realizar las mediciones para obtener la calibración.

En este caso se hicieron dos pruebas, una prueba con el objeto alejándose y otra con el objeto acercándose, esto con el fin de obtener un promedio entre las dos curvas y así tener una ecuación característica más precisa, que proporcionará una función de transferencia adecuada para el sistema.

En cada prueba se tuvieron diez resultados, pues las variaciones de distancia del objeto se hicieron de cero a 47.4cm en intervalos de 4.70cm.

A continuación, se muestran las tablas y las gráficas utilizadas en los dos casos para la caracterización (ver tablas 9, 10 y 11) (ver gráficos 1, 2 y 3).

4.1.1.1 Sensor cuando se acerca un objeto

Tabla 9.

Datos del sensor con objeto acercándose.

Sensor con objeto acercándose	
Distancia cinta métrica (cm)	Distancia del sensor (cm)
1.00	1.10
4.70	4.70
9.40	9.90
14.1	14.1
18.8	18.8
23.5	23.9
28.2	28.2
32.9	32.9
37.8	37.1
42.3	42.3
47.0	47.0

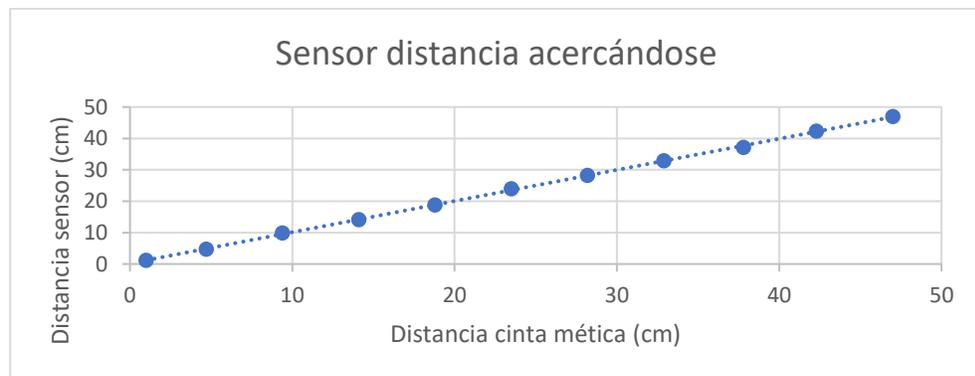


Gráfico 1. Curva de calibración del sensor acercándose.

4.1.1.1.2 Sensor cuando se aleja un objeto

Tabla 10.

Datos del sensor con objeto alejándose.

Sensor con objeto alejándose	
Distancia cinta métrica (cm)	Distancia del sensor (cm)
47.0	46.8
42.3	42.3
37.8	37.8
32.9	33.3
28.2	28.2
23.5	23.5
18.8	18.8
14.1	14.9
9.40	9.40
4.70	4.70
1.00	0.90

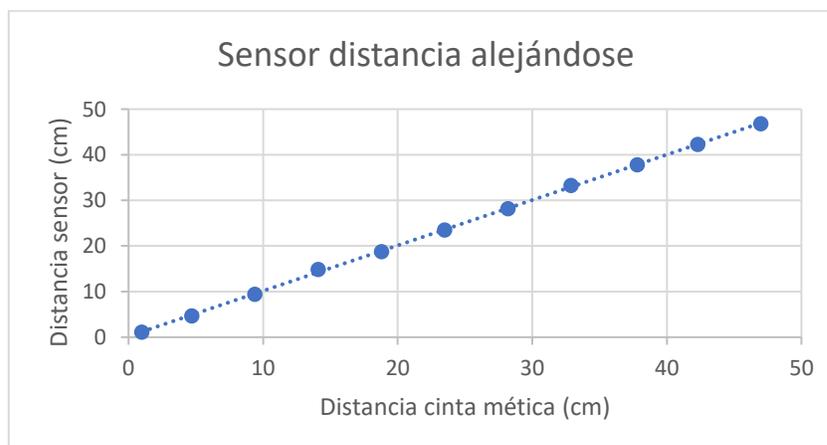


Gráfico 2. Curva de calibración del sensor alejándose.

4.1.1.1.3 Promedio de distancia del sensor cuando el objeto se acerca y se aleja

Tabla 11.

Datos promedio del sensor.

Promedio distancia sensor	
Distancia cinta métrica (cm)	Distancia del sensor (cm)
1.00	1.00
4.70	4.70
9.40	9.40
14.1	14.5
18.8	18.8
23.5	23.5
28.2	28.2
32.9	33.1
37.8	37.8
42.3	42.3
47.0	46.9

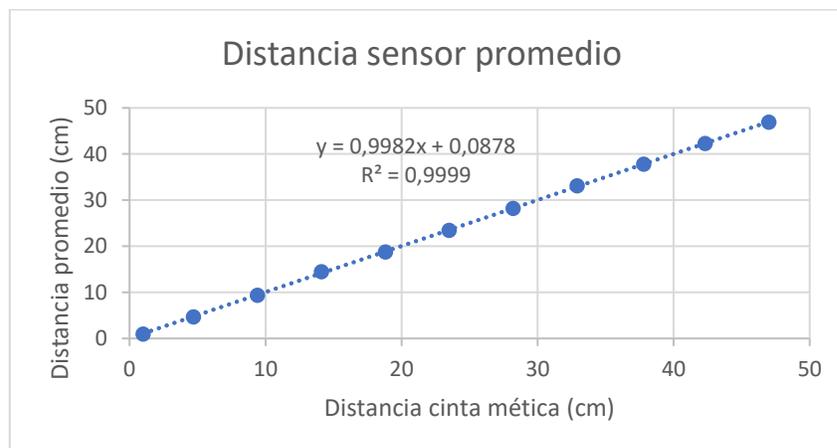


Gráfico 3. Curva de calibración promedio del sensor

Finalmente se obtuvo la ecuación característica de sistema, donde (ver ecuación 37):

$dm = \text{distancia de cinta métrica}$

$ds = \text{distancia del sensor}$

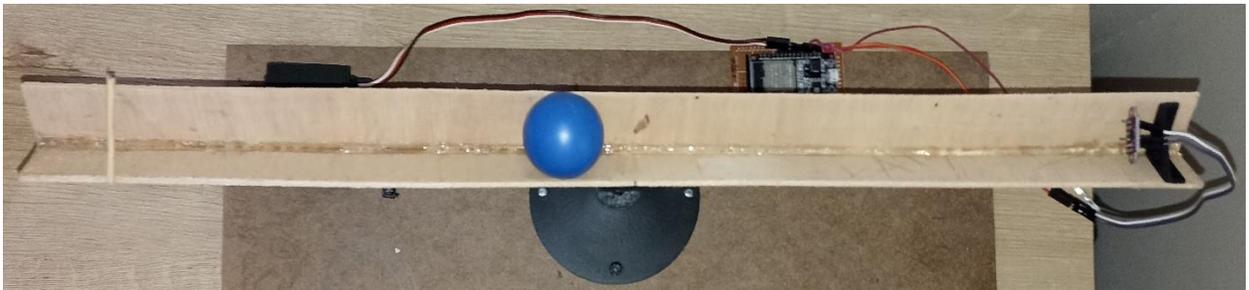
Función de transferencia

$$ds = 0.9982 \cdot dm + 0.0878 \quad (37)$$

Una vez caracterizado y calibrado el sensor de distancia, se procedió a realizar el montaje de la tarjeta de adquisición y se comenzó a tomar datos para comprobar el funcionamiento del sistema (ver figura 30).

Figura 30

Implementación sistema a sensar.



4.1.2 Implementación del servomotor

4.1.2.1 Calibración de señal PWM

Se caracteriza la señal del servomotor para mejorar la precisión del ángulo, para ello se midió la señal PWM según la instrucción del código de Arduino para tener una función que garantice el ángulo de salida respecto a la señal PWM (ver figuras 31, 32, 33 y 34).

Figura 31

Señal PWM con un ángulo de 1°.

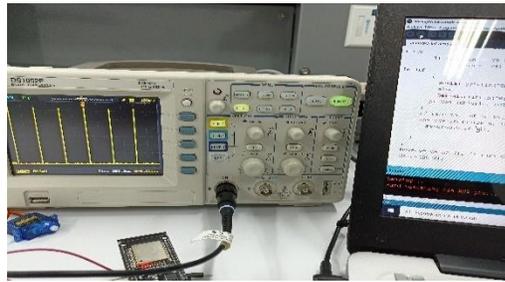


Figura 32

Señal PWM con un ángulo de 50°.



Figura 33

Señal PWM con un ángulo de 200°.



Figura 34

Señal PWM con un ángulo de 255°.

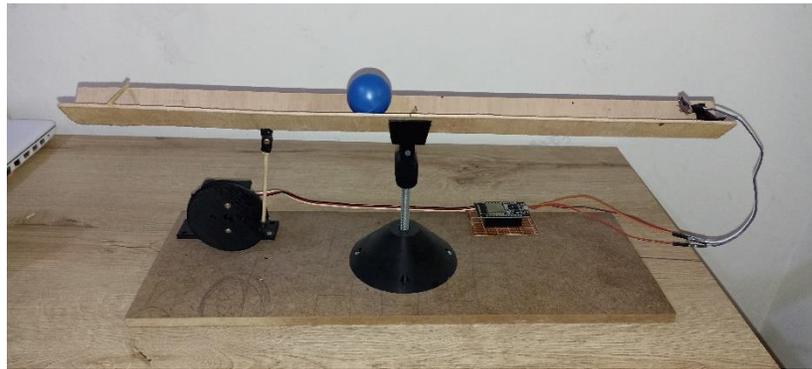


4.1.2.2 Implementación del servomotor.

Una vez caracterizado el servomotor, se procede a realizar su respectiva programación de funcionalidad con ESP32 y, luego, proceder al montaje de los circuitos con la tarjeta de adquisición de datos (ver figura 35).

Figura 35

Sistema de sensado con servomotor



4.1.3 Análisis del sistema viga – bola

4.1.3.1. Función de transferencia de la planta

En el apartado 3.2.2.1 se obtuvieron las ecuaciones que rigen el comportamiento del sistema, la ecuación 31 y la ecuación 36.

Para encontrar la función de transferencia se hará uso de la representación de sistemas en variables de estado y con ayuda del software MatLab se encontrará la respuesta del sistema ante una entrada escalón (ver ecuación 38):

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} = \begin{bmatrix} q_1(t) \\ \dot{q}_1(t) \\ q_2(t) \\ \dot{q}_2(t) \end{bmatrix} \quad (38)$$

Donde q_1 respuesta a “ x ” y q_2 representa a θ . Las ecuaciones de movimiento se pueden reescribir en término de las variables de estado (ver ecuación 39):

$$\dot{x}(t) = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} \quad (39)$$

Despejando $\dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{x}_4$ de la ecuación 30 y 34 se tiene que (ver ecuación 40),

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{mx_1x_4^2 - mg\sin(x_3)}{\frac{J_{bola}}{R^2} + m} \end{aligned} \quad (40)$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{\tau - mgx_1 \cos(x_3) - 2mx_1x_2x_4}{J_{barra} + mx_1^2}$$

Reemplazando la ecuación 35 en la ecuación 36 se obtiene (ver ecuación 41):

$$\dot{x}(t) = \begin{bmatrix} x_2 \\ \frac{mx_1x_4^2 - mg\sin(x_3)}{\frac{J_{bola}}{R^2} + m} \\ x_4 \\ \frac{\tau - mgx_1 \cos(x_3) - 2mx_1x_2x_4}{J_{barra} + mx_1^2} \end{bmatrix} = f(x, \tau) \quad (41)$$

Para analizar el comportamiento ante pequeñas variaciones de la posición de la bola y el ángulo de la barra, se linealiza alrededor del punto de equilibrio (ver ecuación 42):

$$x_0 = \begin{bmatrix} p_0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (42)$$

También se define la entrada de operación requerida para mantener este punto de operación, el par de equilibrio obtenido al igualar $f(x, \tau)$ en la ecuación 36 a cero y evaluar en el punto de operación es (ver ecuación 43):

$$u_{\tau} = m g p_0 \quad (43)$$

En la ecuación 38 se observa el torque requerido para mantener la bola en el punto estacionario P_0 .

Se aplica el Jacobiano a $f(x, \tau)$ para encontrar las matrices A, B y C del espacio de estados, dando como resultado la siguiente matriz (ver ecuación 44).

$$\frac{\delta f(x, \tau)}{\delta x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{m x_4^2}{\frac{J_{bola}}{R^2} + m} & 0 & \frac{-m g \cos(x_3)}{\frac{J_{bola}}{R^2} + m} & \frac{2 m x_1 x_4}{\frac{J_{bola}}{R^2} + m} \\ 0 & 0 & 0 & 1 \\ \frac{\delta f_4}{\delta x_1} & \frac{-2 m x_1 x_4}{m x_1^2 + J_{barra}} & \frac{m g x_1 \sin(x_3)}{m x_1^2 + J_{barra}} & \frac{-2 m x_1 x_2}{m x_1^2 + J_{barra}} \end{bmatrix} \quad (44)$$

Donde la expresión $\frac{\delta f_4}{\delta x_1}$ se representa como (ver ecuación 45),

$$\frac{\delta f_4}{\delta x_1} = \frac{(-2 m x_2 x_4 - m g \cos(x_3))(m x_1^2 + J_{barra}) - (-2 m x_1 x_2 x_4 - m g x_1 \cos(x_3) + \tau)}{(m x_1^2 + J_{barra})^2} \quad (45)$$

La derivada con respecto a τ es (ver ecuación 46):

$$\frac{\delta f(x, \tau)}{\delta \tau} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ \frac{1}{m x_1^2 + J_{barra}} \end{bmatrix} \quad (46)$$

Si se evalúa la matriz Jacobiana en el punto de equilibrio (p_0, u_τ) resulta una matriz de constantes (ver ecuación 47):

$$A = \frac{\delta f}{\delta x}(x_0, \tau_0) = \begin{bmatrix} 0 & 1 & \frac{0}{R^2} & 0 \\ 0 & 0 & \frac{-mg}{R^2 + m} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{-mg}{mp_0^2 + J_{barra}} & 0 & 0 & 0 \end{bmatrix} \quad (47)$$

Para la matriz B (ver ecuación 48):

$$B = \frac{\delta f}{\delta \tau}(x_0, \tau_0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ \frac{0}{mp_0^2 + J_{barra}} \end{bmatrix} \quad (48)$$

La salida del sistema es la posición de la bola por tanto la matriz C es (ver ecuación 49):

$$C = [1 \quad 0 \quad 0 \quad 0] \quad (49)$$

Usando estas matrices, el sistema puede ser escrito en la siguiente representación de espacio de estados (ver ecuación 50):

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad (50)$$

Se reemplaza los parámetros del sistema en la ecuación 39, se tiene (ver ecuación 51):

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -7 & 0 \\ 0 & 0 & 0 & 1 \\ -56.8 & 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 52.6 \end{bmatrix} u(t) \quad (51)$$

Por último, se tiene la salida del sistema en espacio de estados (ver ecuación 52):

$$y(t) = [1 \ 0 \ 0 \ 0]x(t) \quad (52)$$

Se utiliza el software MatLab para encontrar la función de transferencia con los datos de las matrices A, B y C, mediante los siguientes comandos (figura 36):

Figura 36

Código en MatLab para encontrar la función de transferencia.

```

%% definir matrices A, B, C y D:
A=[0 1 0 0; 0 0 -7 0; 0 0 0 1; -56.8 0 0 0];
B=[0;0;0;52.6];
C=[1 0 0 0];
D=0;

%% Encontrar numerados y denominador de la función de transferencia
[num, den] = ss2tf(A,B,C,D);
%% Función de transferencia de la planta
Gs= tf(num, den);

```

El código de la figura 36, da la siguiente función de transferencia (ver figura 37):

Figura 37

Función de transferencia del sistema viga – bola.

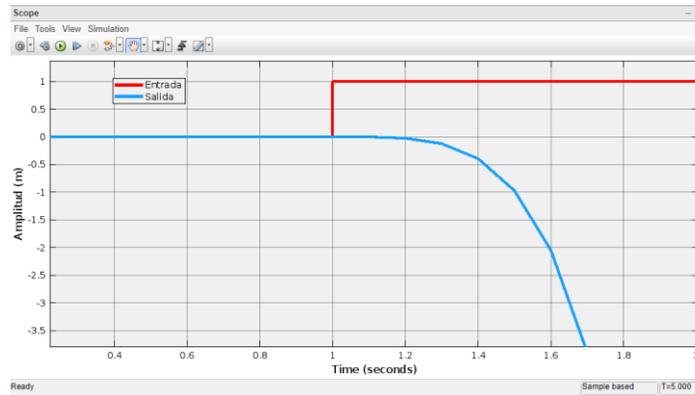
$$G_s = \frac{-368.2}{s^4 + 3.553e-15 s^3 - 7.105e-15 s^2 - 397.6}$$

4.1.3.2. Simulación del sistema

Para analizar la estabilidad de la función de transferencia $G(s)$ de la figura 39, se aplica un método clásico en el cual a partir del modelo se va a aplicar una entrada tipo escalón, y se observará si la salida acaba alcanzando un valor estable en régimen permanente, es decir, para una entrada acotada la salida es acotada. En Matlab se utiliza el comando “step (G(S))”, el cual automáticamente representa la gráfica de la respuesta a una entrada escalón (figura 38).

Figura 38

Respuesta de la función de transferencia en lazo abierto ante una entrada escalón.



Se observa que la planta es inestable en lazo abierto, ya que la salida tiende a infinito, si se hace un análisis sencillo del comportamiento físico, se tiene que el servomotor cambia a inclinación de la barra, por tanto, la bola rodará sin posibilidad de detenerse, por tanto, se observa la necesidad implementar un controlador (en este caso) PID, que permita cerrar el lazo de control y mantener la bola en una posición deseada.

4.1.4. Implementación del control PID en la tarjeta ESP32

Para implementar el control PID al sistema, se debe programar en el IDE de Arduino, es decir, se debe transformar la ecuación del control PID en código que la tarjeta EPS32 pueda entender.

La ecuación del controlador PID que se implementara en la tarjeta ESP32 es:

$$u(t) = K_p \cdot e(t) + K_I \cdot \int_0^t e(t)dt + K_D \cdot \frac{de(t)}{dt} \quad (53)$$

En este caso el control del sistema se debe implementar en tiempo discreto (z), puesto que, se utiliza un microcontrolador ESP32, es decir, un dispositivo digital para controlar el sistema.

En (44), $u(t)$ es la señal del ángulo de giro del servomotor en radianes, lo que se traduce en código como $u(t)$ es la señal PWM que controla el ángulo de giro del servomotor necesario para posicionar la bola en el valor de consigna especificado.

4.1.4.1. Instrucciones fundamentales

Hay varios métodos de aproximación numérica tanto de integración como de diferenciación para discretizar los controladores: método de Euler hacia adelante, método de Euler hacia atrás y método trapezoidal. En este caso, se va a realizar la transformación de ecuaciones diferenciales a ecuaciones en diferencias basándose en el método de Euler hacia atrás.

- **Acción proporcional:** En este caso, la acción proporcional del control PID está dada por (ver ecuación 54):

$$u_p(t) = K_p \cdot e(t) \quad (54)$$

El término proporcional en ecuaciones en diferencias es el siguiente (ver ecuación 55):

$$u_{p_{nT}} = K_p \cdot e_{nT} \quad (55)$$

- **Acción integral:** La acción proporcional está dada por (ver ecuación 56):

$$u_i(t) = K_I \cdot \int_0^t e(t) dt \quad (56)$$

El término integral en ecuaciones en diferencias está dado por (ver ecuación 57):

$$u_{i_{nT}} = K_I \cdot \left(T \cdot \sum_{i=1}^n e_{iT} \right) \quad (57)$$

Reescribiendo la ecuación 46, queda (ver ecuación 58):

$$ui_{nT} = K_I \cdot \left(T \cdot \sum_{i=1}^{n-1} e_{iT} + T \cdot e_{nT} \right) \quad (58)$$

Con en el método de Euler hacia atrás, la ecuación en diferencias es (ver ecuación 59):

$$ui_{nT} = K_I \cdot T \cdot e_{(n-1)T} + K_I \cdot T \cdot e_{nT} \quad (59)$$

En la ecuación 48, se ve como la constante integral afecta a la señal de error acumulada y a la señal de error actual, esto no se quiere ya que, si K_I cambia su valor, esta solo debería actuar sobre señales futuras; por tanto, la acción integral discreta queda (ver ecuación 60):

$$ui_{nT} = T \cdot e_{(n-1)T} + K_I \cdot T \cdot e_{nT} \quad (60)$$

- **Acción derivativa:** Partiendo del término derivativo de la ecuación del controlador PID (ver ecuación 61).

$$ud(t) = K_D \frac{de(t)}{dt} \quad (61)$$

La ecuación en diferencias de la derivada basándose en el método de Euler hacia atrás es la siguiente (ver ecuación 62):

$$ud_{nT} = K_D \cdot \frac{e_{nT} - e_{(n-1)T}}{T} \quad (62)$$

Por tanto, la ecuación del controlador PID queda de la siguiente manera en ecuaciones en diferencias (ver ecuación 63):

$$u_{nT} = K_P \cdot e_{nT} + T \cdot e_{(n-1)T} + K_I \cdot T \cdot e_{nT} + K_D \cdot \frac{e_{nT} - e_{(n-1)T}}{T} \quad (63)$$

Donde T es el periodo de muestreo, para este caso de 60ms, ya que le da tiempo al servomotor de actuar y al sensor de distancia de medir la distancia de la bola.

4.1.4.2. Programación de software PID en el ESP32

La programación del controlador PID que se ha implementado en el ESP32 es la descrita en la figura 39:

Figura 39

Implementación del controlador PID en el ESP32.

```
1 distancia = MedirDist(100);
2 distanciaError = distanciaSetpoint - distancia;
3 PID_p = kp * distanciaError;
4 distDiferencia = distanciaError - distanciaPrevError;
5 PID_d = kd*((distDiferencia)/tiempoMuestreo);
6
7 if(20.7 < distanciaError && distanciaError < 26.7){
8     PID_i = PID_i + (ki * distanciaError);
9 }
10 else{
11     PID_i = 0;
12 }
13
14 PID_total = PID_p + PID_i + PID_d;
15 PID_total = map(PID_total, -150, 150, 0, 150);
16
17 if(PID_total < 20){PID_total = 20;}
18 if(PID_total > 160) {PID_total = 160; }
```

En la primera línea de código mostrado en la figura 40, se realiza un promedio de la distancia para 100 muestras del sensor, con el fin de reducir el ruido, se procede a encontrar el error de la distancia en la línea 2. Con este valor se procede a encontrar las acciones del controlador PID:

- **PID_p:** este caso es el más simple ya que solo se multiplica la constante K_p establecida por el usuario y se multiplica por el error (*distanciaError*) que es la diferencia entre el valor de la distancia actual y el valor de la distancia a la que se quiere llevar la bola.
- **PID_d:** Para la acción derivativa se calcula la diferencia entre el error actual y el error anterior, luego se divide por el tiempo de muestreo que en este caso es 60ms y por último este resultado lo multiplicamos por la constante K_i .

- **PID_i**: En la acción integral, se simplifica T, ya que estamos asegurando que el ESP32 trabajara cada 60ms, además, la acción integral puede generar retrasos, por eso limitamos su acción solo para valores cercanos al “*setPoint*”, como se puede observar en las líneas de la 7 a la 12.

Por último, se suman todas las acciones (*PID_p*, *PID_i* y *PID_d*) para encontrar la acción PID (línea 14). Esta acción de control se ajusta al rango de trabajo del servomotor gracias a las líneas de la 15 a la 18.

4.1.5. Interfaz web

Esta interfaz de usuario fue desarrollada en HTML, CSS y JavaScript, se empleó una librería de código abierto para ello llamada ReactJs, esta librería se basa en JavaScript y se encarga de la parte visual del proyecto. Para iniciar este proyecto con ReactJs, es necesario instalar algunas dependencias (ver figura 40).

Figura 40

Dependencias del Front-End.

```
"dependencies": {  
  "firebase": "^9.6.3",  
  "react": "17.0.2",  
  "react-dom": "17.0.2",  
  "react-scripts": "4.0.3",  
  "recharts": "^2.1.9",  
  "web-vitals": "1.1.2",  
  "wouter": "2.8.0-alpha.1"  
},
```

Se trabaja con la siguiente estructura de carpetas, para tener mejor orden y poder escalar el código fácilmente en un futuro (ver figura 41 y tabla 12).

Figura 41

Estructura de carpetas para el Front-End.

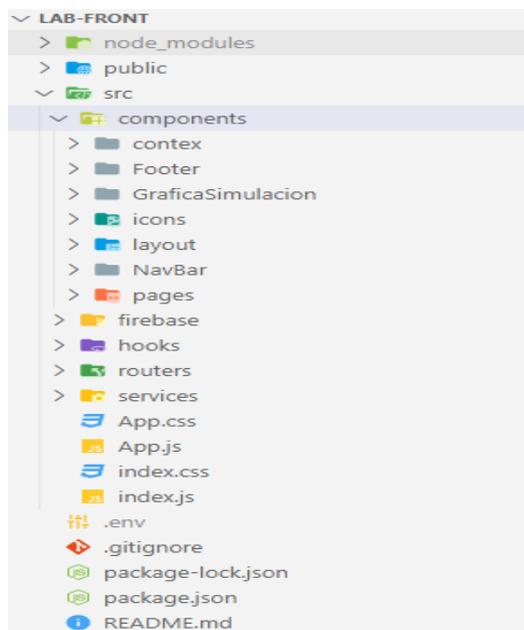


Tabla 12.

Estructura de Carpetas de la interfaz web.

Nombre de la carpeta	Función
node_module	Contiene todos los archivos que son dependencias externas del proyecto.
public	Contiene los archivos estáticos como imágenes, logos, archivos HTML y CSS.
components	Se encuentran los archivos que hacen parte de la interfaz web dividida en pequeñas secciones.
firebase	Tiene los archivos necesarios para comunicarse con la base de datos.
hooks	Son funciones que se pueden ejecutar en cualquier componente de la interfaz.
routers	Contiene las rutas que el usuario puede navegar.
services	Son archivos propios que realizan cierta función específica.

La interfaz web desarrollada posee cuatro pantallas para dar al usuario una vista general o detallada según la necesidad de este.

4.1.5.1. Funcionamiento de la interfaz de usuario

El usuario podrá realizar una navegación sobre la plataforma web, según los objetivos que tenga, tendiendo las siguientes opciones:

4.1.5.1.1. Página principal. En esta el usuario tendrá una vista general de las páginas a las que puede navegar (ver figura 42).

Figura 42

Página principal.

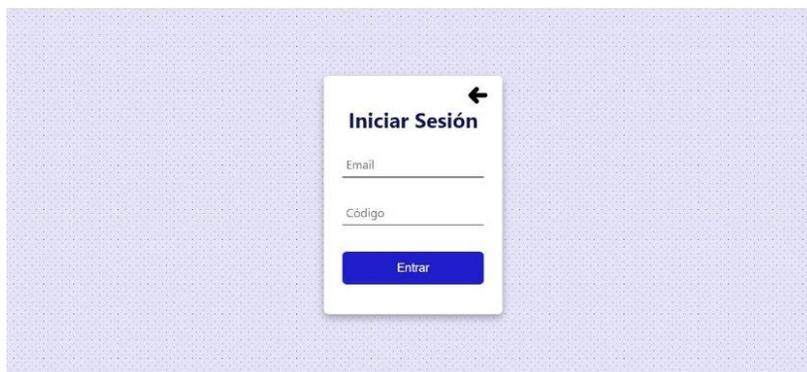


Desde aquí, el usuario podrá navegar a la documentación del proyecto, a los aspectos teóricos del sistema, a la página de inicio de sesión y al panel de control.

4.1.5.1.2. Página de inicio de sesión. El usuario deberá ingresar sus credenciales para tener acceso a la planta física y poder modificar las constantes del controlador PID, estas credenciales podrán ser establecidas por el docente encargado, desde esta página será redirigido el usuario al panel de control (figura 43).

Figura 43

Página de inicio de sesión.



4.1.5.1.3. Página de documentación. Se podrá ver toda la información del código fuente que se implementó para crear este proyecto, también, se podrá ver la configuración de algunas tecnologías implementadas como Firebase y librerías para el ESP32 (ver figura 44).

Figura 44

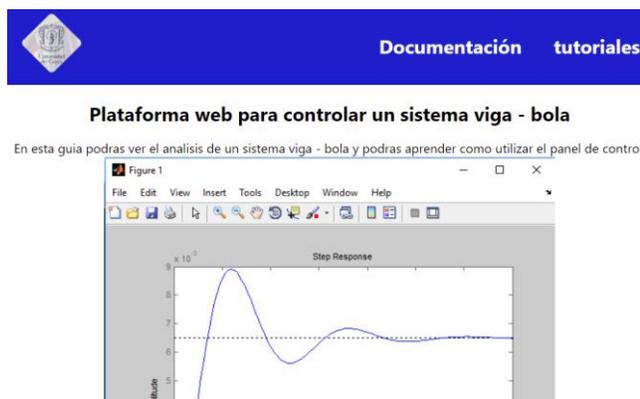
Página de documentación.



4.1.5.1.4. Página sobre la teoría. En esta página el usuario podrá ver todo el tratamiento teórico que se realizó a este sistema viga – bola, también, se podrá ver como es el panel de control, para tener un acercamiento y evitar un mal uso de este (figura 45).

Figura 45

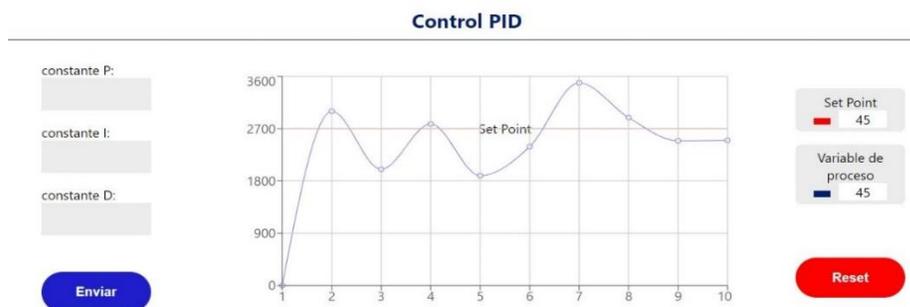
Página de teoría.



4.1.5.1.5. Panel de control. Es la página que controla a la planta física desde cualquier parte del mundo, solo se puede acceder a este panel si se ha iniciado sesión previamente (figura 46).

Figura 46

Panel de control.



En este panel se puede enviar el valor de las constantes K_p , K_i y K_d del controlador PID, estos datos se enviarán a la base de datos y posteriormente, serán leídos por el microcontrolador para ejercer la acción de control pertinente.

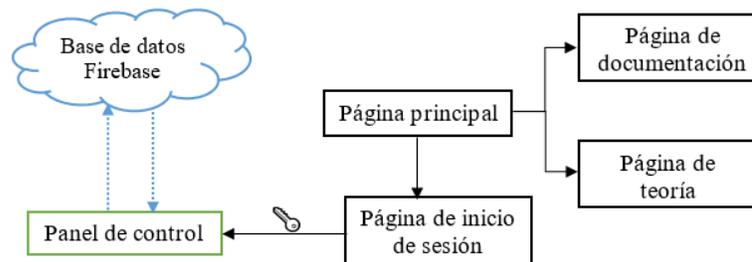
4.1.5.2. Modos de trabajo en la interfaz

La interfaz web tiene dos modos de trabajo (ver figura 47):

- **Modo lectura:** Este modo es solo de lectura, es decir, no necesita de credenciales para acceder a la información de la plataforma y las páginas que trabajan en este modo no se tiene acceso a la base de datos.
- **Modo interactivo:** Para este caso se necesita de credenciales ya que en este modo se hace llamado a la base de datos, tanto para enviar datos como para recibir.

Figura 47

Diagrama de la interfaz web.



4.1.6. Comunicación con la base de datos

Para conectarse con la base de datos es necesario utilizar las llaves secretas de Firebase, con esto el ESP32 y la plataforma web puedan obtener permisos para modificar los datos.

4.1.6.1. Comunicación del sistema físico con Firebase

Para realizar la conexión del ESP32 a Firebase se implementó, primero, una librería llamada “WiFi” que permite al ESP32 conectarse a internet, luego, otra librería llamada “Firebase ESP32 client” que proporciona facilidades para conectarse con Firebase. Para utilizar esta librería es necesario establecer la ruta de la base de datos y las keys de seguridad (figuras 48 y 49).

Figura 48

Código del ESP32 y Firebase, configuración inicial.

```
/* Se llaman las librerías necesarias para que funcione el código */
#include <WiFi.h>
#include <FirebaseESP32.h>
#include <addons/RTDBHelper.h>

/* Aquí se pone el nombre y la contraseña del WIFI*/
#define WIFI_SSID "WIFI_AP"
#define WIFI_PASSWORD "WIFI_PASSWORD"

/* Se definen la llave secreta y la ruta de la base de datos, en este caso
es REALTIME DATABASE
*/
#define DATABASE_URL "URL" //<databaseName>.firebaseio.com or <databaseName>.<region>.firebasedatabase.app
#define DATABASE_SECRET "DATABASE_SECRET"

/* Se crea el objeto para realizar la conexión */
FirebaseData fbdo;
/* Se define un objeto para realizar la autenticación */
FirebaseAuth auth;
/* Se define un objeto para configurar la base de datos */
FirebaseConfig config;
```

Figura 49

Código del ESP32 y Firebase, conexión con Realtime Database.

```
void setup()
{
  Serial.begin(115200);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(".");
    delay(300);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();
  Serial.printf("Firebase Client v%s\n\n", FIREBASE_CLIENT_VERSION);

  /* Se asignan las credenciales al objeto config */
  config.database_url = DATABASE_URL;
  config.signer.tokens.legacy_token = DATABASE_SECRET;

  Firebase.reconnectWiFi(true);

  /* Se inicializa la conexión con Firebase, utilizando
  la autenticación y la configuración
  */
  Firebase.begin(&config, &auth);
}
```

4.1.6.2. Comunicación de la interfaz web con Firebase

Se implementó la librería llamada “Firebase” que proporciona métodos para realizar la conexión y su fácil uso permite integrarla con ReactJs de una manera sencilla (figura 50).

Figura 50

Código de la plataforma web y Firebase, conexión con Realtime Database.

```
// Se llama a la librería Firebase.
import { initializeApp } from "firebase/app";
import { getAuth } from "firebase/auth";
import { getDatabase } from "firebase/database";
// Se establece la configuración de la base de datos
const firebaseConfig = {
  apiKey: process.env.REACT_APP_apiKey,
  authDomain: process.env.REACT_APP_authDomain,
  projectId: process.env.REACT_APP_projectId,
  storageBucket: process.env.REACT_APP_storageBucket,
  messagingSenderId: process.env.REACT_APP_messagingSenderId,
  appId: process.env.REACT_APP_appId,
};
// Se inicializa el objeto Firebase
const app = initializeApp(firebaseConfig);
// Se obtienen los objetos de autenticación y base de datos
export const auth = getAuth(app);
export const db = getDatabase(app);
```

El objeto app, los métodos auth y db, pueden utilizar la base de datos para leer y escribir datos sobre esta; también, proporciona una suscripción que permite actualizar la interfaz web automáticamente si se detecta un cambio en la base de datos.

4.1.6.3. Comunicación entre la interfaz web y la planta física

La planta física y la interfaz web se comunican por medio de la base de datos, por esta razón es necesario establecer las variables que se van a modificar y a leer en la base de datos “Realtime Database” (figura 51 y tabla 13).

Figura 51

Estructura de datos en Realtime Database.

```
https://pid-admin-test-default-rtdb.firebaseio.com/
|
└─ test
   ├── data: ""
   ├── isConnected: false
   ├── kd: 0
   ├── ki: 0
   └── kp: 0
```

Fuente: plataforma Firebase (s. f.)

Tabla 13.*Tipos de datos de Realtime Database.*

Nombre	Tipo de dato	Función
data	Cadena de caracteres (string)	Guardara los datos de la lectura del sensor.
kd	Numérico (number)	Guarda la constante derivativa.
ki	Numérico (number)	Guarda la constante integral.
kp	Numérico (number)	Guarda la constante proporcional.
isConnected	Lógico (boolean)	Guarda el estado de conectado o desconectado de la interfaz web.

La tarjeta ESP32 envía los datos del sensor a la base de datos, mientras tanto, la plataforma web estará pendiente si ocurre algún cambio en la base de datos, en caso de ser así, la interfaz se actualizará automáticamente, mostrando los nuevos valores. Para lograr esto, el ESP32 solo podrá modificar la variable “data” siempre y cuando “isConnected” sea verdadero, es decir, si la plataforma web está conectada a la base de datos, el ESP32 podrá actualizar la variable “data”.

La variable data, a pesar de representar la distancia del sensor, es una cadena de caracteres, esto es por facilidad ya que permitiría agrupar una gran cantidad de valores en una sola variable simple, por ejemplo, si se desea guardar los valores 1, 2, 3, 4 y 5, se podría utilizar un vector (Array) de números, $data[] = \{1, 2, 3, 4, 5\}$, pero la implementación con firebase se vuelve más complicada, utilizando más recursos del ESP32. Por esta razón, se convertirán los datos del sensor en una cadena de texto, separando cada dato con un guion, $data = "1-2-3-4-5"$. Por tanto, el EPS32 enviará la variable data a la base de datos, y la plataforma web por medio de JavaScript será la

encargada de transformar esta variable en una lista de números para ser graficados (ver figuras 52 y 53).

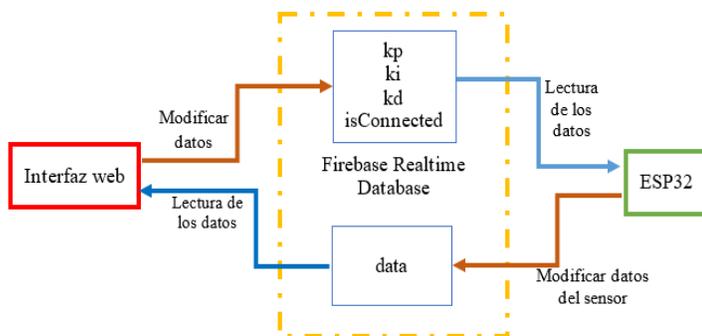
Figura 52

Datos tipo string a cadena de números con JavaScript.

```
const arrayData = dataPid // pregunta si hay datos
? dataPid.data.length > 0
? dataPid.data.split("-") // separar los datos por -
: null // retorna vacío si no hay datos
: null;
const arrayDataPid = arrayData // Se verifica si hay datos
? arrayData.map((item, i) => ({ // crear un array con los datos
  name: i,
  uv: parseFloat(item), // parseFloat para convertir a número
}))
: null; // si arrayData no tiene datos
setDataToGraphics(arrayDataPid); // envía los datos a graficar
```

Figura 53

Comunicación de la plataforma web y el ESP32 con firebase.



4.1.7. Evaluación

Se realizaron diferentes pruebas con la interfaz web en el control del sistema para sintonizar el controlador PID y obtener los valores óptimos de las constantes, mediante el método ensayo y error; luego, se desplegará la plataforma web para ser accedida desde cualquier parte del mundo y se realizó el control PID con las constantes encontradas previamente.

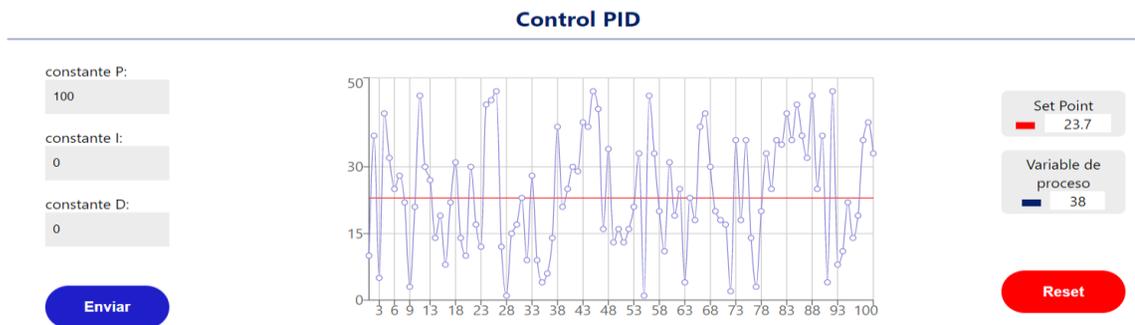
4.1.7.1. Sintonización del controlador PID para la plataforma.

Para sintonizar el controlador se emplearán diferentes pruebas, con el fin de observar el comportamiento del sistema cuando se realice un control P, D, I, PD y PID. Se tomaron 100 muestras, por tanto, el tiempo empleado en la acción de control es de 6 segundos.

Prueba 1: Para este caso se empleará un valor de $K_p=100$, $K_i = 0$, $K_d = 0$ (ver figura 54).

Figura 54

Respuesta del sistema con $K_p=100$, $K_i = K_d = 0$.

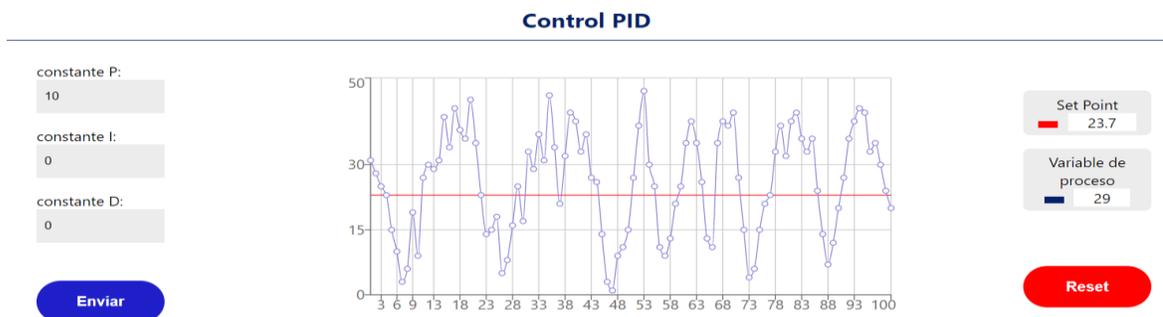


El sistema reacciona de manera violenta, es decir, el servomotor pasa de ángulos pequeños en ángulos grandes rápidamente, esto es porque el error es grande y la acción proporcional también, generando en la salida del servomotor un ángulo grande.

Prueba 2: Para este caso se empleará un valor de $K_p=10$, $K_i = 0$, $K_d = 0$ (ver figura 55).

Figura 55

Respuesta del sistema con $K_p=10$, $K_i = K_d = 0$.

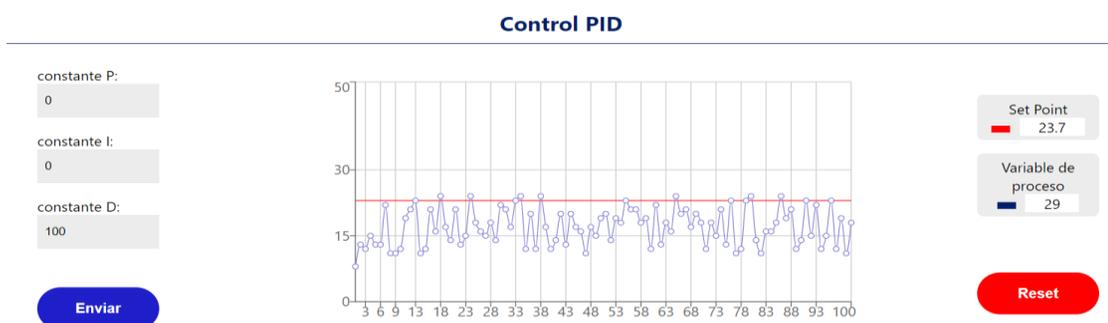


El sistema oscila de una forma menos violenta que en la prueba 1 y de igual manera no se estabiliza, pero este valor de K_p se conservará, ya que provee una oscilación buena porque el servomotor no pasa de ángulos grandes a pequeños tan rápido.

Prueba 3: Para este caso se empleará un valor de $K_p=0$, $K_i = 0$, $K_d = 100$ (ver figura 56).

Figura 56

Respuesta del sistema con $K_d=100$, $K_p = K_i = 0$.

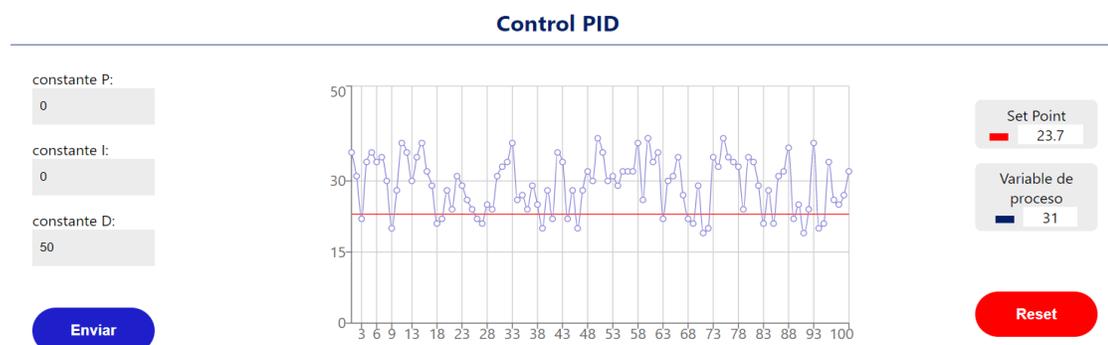


El sistema reacciona de manera muy rápida a la velocidad de la bola, es decir, al mínimo movimiento de la bola, el sistema actúa de forma violenta.

Prueba 4: Para este caso se empleará un valor de $K_p=0$, $K_i = 0$, $K_d = 50$ (ver figura 57).

Figura 57

Respuesta del sistema con $K_d=50$, $K_p = K_i = 0$.

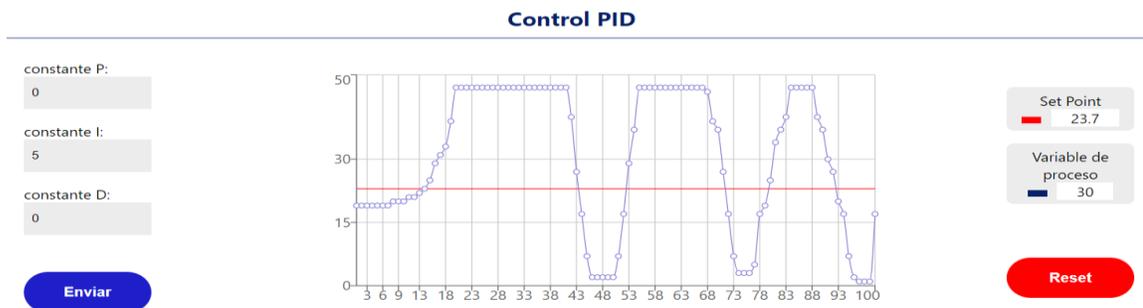


El sistema reacciona de manera rápida a la velocidad de la bola, pero esta oscilación no es tan violenta como en la prueba 3, por tanto, se considera este valor de K_d como candidato.

Prueba 5: Para este caso se empleará un valor de $K_p=0$, $K_i = 5$, $K_d = 0$ (ver figura 58).

Figura 58

Respuesta del sistema con $K_i=5$, $K_p = K_d = 0$.

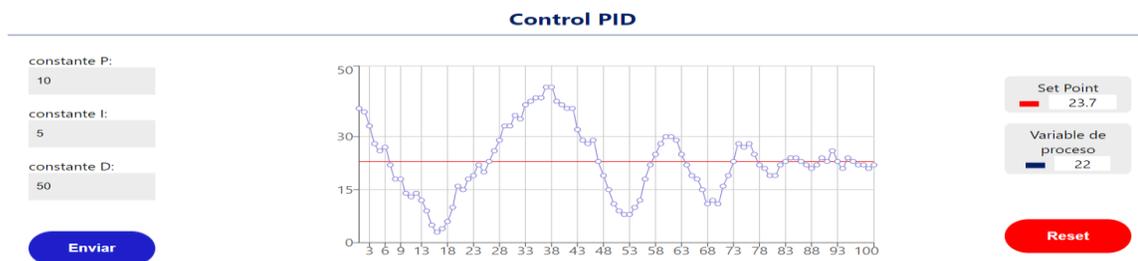


Para este caso, se coloca la bola muy cerca del set Point y se observa como la acción integral hace que el servomotor actúe variando el ángulo cada vez más, según cada iteración ya que el error se sumara cada vez y aumentara la acción integral.

Prueba 6: Para este caso se empleará un valor de $K_p=10$, $K_i = 5$, $K_d = 50$ (ver figura 59).

Figura 59

Respuesta del sistema con $K_p=10$, $K_d = 50$, $K_i = 5$.



El sistema se estabiliza en 4 segundos, lo que nos indica que se pueden mejorar los valores de las constantes, ya que se espera que el sistema se estabilice en un tiempo menor. El sistema de

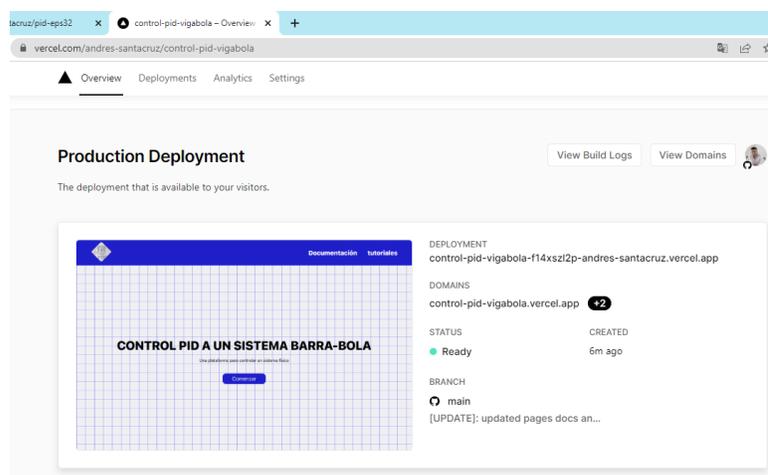
control funciona de manera adecuada, puesto que el error es ínfimo, aunque el control es más lento, es decir, su tiempo de establecimiento es superior a las constantes establecidas, puesto que al aumentar el termino derivativo el amortiguamiento del sistema aumenta, haciéndolo más estable, pero a su vez más lento.

4.1.7.2. Despliegue y prueba final de la plataforma

Una vez realizadas las pruebas, el sistema se despliega en la nube para que pueda ser accedido por cualquier persona en cualquier parte del mundo, para ello se utilizará una plataforma para sitios web llamada Vercel (2022), esta plataforma ya tiene una integración para la biblioteca ReactJs, lo cual facilita el despliegue de la aplicación (ver figura 60).

Figura 60

Interfaz en producción utilizando Vercel.com.



Fuente: plataforma Vercel s. f.

4.7.2.1. Prueba de la plataforma en producción

Se realiza el control PID desde un dispositivo móvil conectado a una red de internet diferente a la red wifi del ESP32, utilizando los valores de K_p , K_i y K_d encontrados en pruebas anteriores, con el fin controlar la planta desde cualquier parte del mundo (ver figuras 61 y 62).

Figura 61

Ingreso a la plataforma desde un dispositivo móvil.



Figura 62

Control PID al sistema viga – bola remotamente.



Conclusiones

- Se diseñó un sistema para sensar la distancia de la bola, con ayuda de una tarjeta ESP32 que realiza el monitoreo de la posición de la bola y envía periódicamente los valores a la base de datos *Realtime Database*.
- Se diseñó un sistema viga – bola el cual es controlado desde una plataforma web remotamente, es decir, desde cualquier parte del mundo con acceso a internet.
- La interfaz web desarrollada en HTML, CSS y JavaScript, permite la configuración de las constantes del control PID que actúa sobre el sistema viga – bola.
- La interfaz web cuenta con un apartado gráfico que muestra la posición de la bola respecto al tiempo del sistema viga – bola, ayudando a observar cómo afecta cada una de las acciones del controlador PID en el comportamiento del sistema.
- La plataforma web implementa reglas de seguridad por medio de autenticación que permiten restringir el control de la planta si no se posee el nombre de usuario y la contraseña.
- Se diseñó un algoritmo que permite transformar diferentes tipos de datos en otros para manipularlos, con el fin de optimizar los recursos computacionales del ESP32 y de la base de datos.
- Se llevó a producción la interfaz web, mediante la plataforma Vercel, permitiendo a cualquier persona en el mundo estudiar el sistema viga – bola y el controlador PID, para ello se debe ingresar al siguiente link: <https://control-pid-vigabola.vercel.app/>.

Bibliografía

- Alonso, R, (2021) ¿Qué es el de las cosas (IoT) y por qué se le llama así?
<https://hardzone.es/reportajes/que-es/internet-cosas-iot/>
- Álvarez, C. (2015) Teoría frente a práctica educativa: algunos problemas y propuestas de solución,
Perfiles Educativos, 37(148).
https://perfileseducativos.unam.mx/iisue_pe/index.php/perfiles/article/view/49320/44373
- Anaya, A, Rios, I, Carvajal, D, Peluffo, D, Castro, J, Espinosa, L. (2016) Sistema de riesgo basado en la internet de las cosas (IoT).
https://www.diegopeluffo.com/publicaciones/2016_JornadasFica_IOT.pdf
- Angos, P. C. G. (2021, 15 septiembre). *Repositorio Institucional de la Universidad Politécnica Salesiana: Implementación de algoritmos de control óptimo estocástico en la nube mediante herramientas de programación IoT para un motor DC*.
<https://dspace.ups.edu.ec/handle/123456789/20873>
- Becerra, A & Senberg, Y (2020) *Diseño de un sistema de monitoreo y control agroclimático para lombricultivo en la Universidad Autónoma de Occidente, Valle del Cauca*. [Pasantía Institucional] Universidad Autónoma de Occidente.
<https://red.uao.edu.co/bitstream/handle/10614/12805/T09573.pdf?sequence=6&isAllowed=y>
- Botta, A. de Donato, W. Persico V. y A. Pescap. (2016). On the Integration of Cloud Computing and Internet of Things. *2014 International Conference on Future Internet of Things and Cloud*, 56, pp. 684-700. 10.1016/j.future.2015.09.021.

Cisco Annual Internet Report (2018–2023) White Paper. (2022, 23 enero). Cisco.
<https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>.

Colomer Barbera, J. C. (2018). Estudio de los sensores para la detección de obstáculos aplicables a robots móviles (tesis de maestría) Universitat Oberta de Catalunya.
<http://openaccess.uoc.edu/webapps/o2/bitstream/10609/80846/6/jacobarTFM0618memoria.pdf>.

Computerworld, (2020) *Definición y ejemplos de WAN para sacarle todo el beneficio.* (2020, 4 diciembre). IDG Communications S.A.U. <https://www.computerworld.es/mundo-profesional/definicion-y-ejemplos-de-wan-para-sacarle-todo-el-beneficio>.

Congreso de Colombia. (Junio 16 de 2010). Ley 1540 de 2011 “Por la cual se expide el Plan Nacional de Desarrollo, 2010-2014”. DO: 48.102.

Congreso de Colombia. (Mayo 25 de 2019). Ley 1955 de 2019 Por el cual se expide el plan nacional de desarrollo 2018-2022 ‘Pacto por Colombia, pacto por la equidad’. DO: 50.964.

Controlador PID - Control Automático - Picuino. (s. f.). Recuperado 24 de marzo de 2022, de <https://www.picuino.com/es/control-pid.html>.

Fajardo, C, Aguilera, A, Guerrero-Carbacas, M. (2019). “Doing More With Less: The Impact of New Technologies on Labor Markets, Economy, and Society” En: *Organizational Transformation and Managing Innovation in the Fourth Industrial Revolution*. 10.4018/978-1-5225-7074-5.ch001

- Fernández, L. (2022) Protocolos de redes: la guía completa con todos los protocolos básicos. Red Zone. <https://www.redeszone.net/tutoriales/internet/protocolos-basicos-redes/>.
- Garcés Fuenmayor, J. & Mora Bolaños, C. (2020). Estrategias de aprendizaje para mitigar la deserción estudiantil en el marco de la COVID-19. *SUMMA. Revista disciplinaria en ciencias económicas y sociales*, 2(Especial), 49-55. <https://doi.org/10.47666/summa.2.esp.06>.
- Gómez, M. (2013) *Material didáctico, Notas del curso bases de datos*. (1ra ed). Casa abierta el campo.
http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas_del_curso_Bases_de_Datos.pdf
- Goto IoT | *Introducción a protocolos IoT*. (s. f.). Recuperado 24 de octubre de 2022, de https://www.gotoiot.com/pages/articles/iot_protocols_intro/index.html.
- Guevara, N. (2020) *Desarrollo de un sistema IoT para monitoreo y apoyo en el dimensionamiento de consumo energético para la Industria Licorera del Cauca* [Trabajo de grado] Corporación Universitaria Autónoma del Cauca.
<https://repositorio.uniautonoma.edu.co/bitstream/handle/123456789/102/T%20E-P%20009%202020.pdf?sequence=1&isAllowed=y>
- Hasinah Johari, S. N., Hezri Fazalul Rahiman, M., Adnan, R., & Tajjudin, M. (2020). *Real-time IMC-PID Control and Monitoring of Essential Oil Extraction Process Using IoT*. 2020 *IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*.
- IEEE Spectrum (s.f) *Los mejores lenguajes de programación 2021*. <https://spectrum.ieee.org/top-programming-languages/>

- La Rosa, A. (2021) LPWAN como base de comunicaciones para IoT. *Pandora fms, Monitoring Blog*. <https://pandorafms.com/blog/es/que-es-lpwan/>
- Llamas, L. (2019). Controlar un servo con Arduino. <https://www.luisllamas.es/controlar-un-servo-con-arduino/>
- Lozano-Valencia, L. F., Rodríguez-García, L. F., & Giraldo-Buitrago, D. (2012). Diseño, implementación y validación de un controlador PID autosintonizado. *TecnoLógicas*, (28), 33-53.
- Maloy, G. (2000) ¿Qué es un sensor y qué hace? Dewesoft. <https://dewesoft.com/es/daq/que-es-un-sensor>
- Marqués, M. (2011) Base de datos. <https://bdigital.uvhm.edu.mx/wp-content/uploads/2020/05/Bases-de-Datos.pdf>
- MecatrónicaLATAM. (2021, 4 mayo). *Sensores*. Mecatrónica LATAM. <https://www.mecatronicalatam.com/es/tutoriales/sensores/>
- Mikati, M., Santos, M. y Armenta, C. (2012). Modelado y simulación de un sistema conjunto de energía solar y eólica para analizar su dependencia a la red eléctrica. *Revista Iberoamericana de Automática e Informática Industrial*. 9(3), 267-281. 10.1016/j.riari.2012.05.010
- Modelo de referencia OSI (Guía de administración del sistema: servicios IP)*. (s. f.). Recuperado 24 de octubre de 2022, de <https://docs.oracle.com/cd/E19957-01/820-2981/ipov-8/index.html>

Ogata, K. (2010) *Ingeniería de control moderna, Vol. 1*. Pearson educación.

Osseiran, A., Boccardi, F, Braun, V, Kusume, K, Marsch, P., Maternia, M., Queseth, O., Schellmann, M. *et al.* (2014). Scenarios for 5G Mobile and Wireless Communications: The Vision of the METIS Project. *Communications Magazine, IEEE*. 52: 26-35. 10.1109/MCOM.2014.6815890.

Palomino, L. Á. V. (2020, noviembre 29). *MONITOREO Y CONTROL PID DE UN SISTEMA RC USANDO IOT MEDIANTE RASPBERRY (MONITORING AND PID CONTROL SYSTEM FOR A RC CIRCUIT THROUGH RASPBERRY)* | Velázquez Palomino | Pistas Educativas. <http://www.itc.mx/ojs/index.php/pistas/article/view/2319>

Pérez, M., Pérez, A. y Pérez, E. (2007) *Introducción a los sistemas de control y modelo matemático para sistemas lineales invariantes en el tiempo*. Universidad Nacional de San Juan. <http://dea.unsj.edu.ar/control1/apuntes/unidad1y2.pdf>

Platzi: *Cursos online profesionales de tecnología*. (s. f.). Recuperado 20 de abril de 2022, de <https://platzi.com/clases/2225-redes/35587-modelo-osi/>

Poongothai, P., Subramanian, M. y Rajeswari, A. (2018). Design and implementation of IoT based smart laboratory. *2018 5th International Conference on Industrial Engineering and Applications (ICIEA)*, pp. 169-173. 10.1109/IEA.2018.8387090.

Porras, W y Salah, O. (2019) Implementación de una red LPWAN de sensores para el monitoreo de condiciones ambientales. https://www.cdtdegas.com/images/Descargas/Nuestra_revista/MetFlu14/9_RedLPWAN.pdf

- Priyanka, C. Maheswari, S. Thangavel, M. y Ponni Bala. (2020). Integrating IoT with LQR-PID controller for online surveillance and control of flow and pressure in fluid transportation system. *Journal of Industrial Information Integration*, 17, ISSN 2452-414X. 10.1016/j.jii.2020.100127.
- Porkodi, R. y Bhuvaneswari, V. (2014). The Internet of Things (IoT) Applications and Communication Enabling Technology Standards: An Overview. *2014 International Conference on Intelligent Computing Applications*, pp. 324-329. 10.1109/ICICA.2014.73.
- Quiñones, R. (2021) *Estrategia metodológica para la transferencia de tecnologías basadas en internet de las cosas con aplicación en la caficultura del Cauca* [Trabajo de grado] Universidad EAN. <https://repository.ean.edu.co/bitstream/handle/10882/11379/QuinonesMauricio2021.pdf?sequence=1&isAllowed=y>.
- Ramírez, L. G. C., Jiménez, G. S. A. & Carreño, J. M. (2014). *Sensores y Actuadores*. Grupo Editorial Patria.
- Rodríguez Penín, A. (2013). *Sistemas SCADA*, tercera edición. Alfaomega Grupo Editor.
- Rodríguez, H y Olarte, C. (2019) *Simulación del sistema de control en la represa de Hidroituango para controlar el nivel de agua del río cauca utilizando IoT* [trabajo de grado] Universidad Católica de Colombia. <https://repository.ucatolica.edu.co/bitstream/10983/24005/1/Simulacion%20del%20sistema%20de%20control%20en%20la%20represa%20de%20Hidroituango%20para%20controlar%20el%20nivel%20de%20agua%20d.pdf>.

Santos Peñas, M. y Farias Castro, G. (2010) Laboratorios virtuales y remotos de procesamiento de señales, *Revista Iberoamericana de Automática e Informática industrial*, 7(1), <https://polipapers.upv.es/index.php/RIAI/article/view/8510/8631>

Vercel.com. (2022). Start with the developer <https://vercel.com>.

Villajulca, J. C. (2020, 14 abril). *Sistema de Control de Lazo Cerrado (“Closed Loop”)*.

Instrumentacion y Automatizacion Industrial. <https://instrumentacionycontrol.net/sistema-de-control-de-lazo-cerrado-closed-loop/>.

Wigmore, I. (2021, 7 abril). *Internet de las cosas (IoT)*. ComputerWeekly.es. <https://www.computerweekly.com/es/definicion/Internet-de-las-cosas-IoT>.