

UNIVERSIDAD NACIONAL DE MOQUEGUA
INGENIERÍA DE SISTEMAS E INFORMÁTICA



**ROBOT MÓVIL AUTÓNOMO SEGUIDOR DE
LÍNEA CON SELECCIÓN DE DESTINO POR
BLUETOOTH PARA ENTORNOS CONTROLADOS**

Presentado por:

Moises Ronaldo Atencio Ccallalla

Sayli Denilayda Ayala Mamani

Docente:

ROSAS CUEVAS, YESSICA

Ilo, 30 de julio de 2025

Índice

1. Resumen	5
2. Objetivos	6
2.1. Objetivo General	6
2.2. Objetivos Específicos	6
3. Planteamiento del Problema y Solucion	7
3.1. Problema	7
3.2. Justificación	8
3.3. Solución Propuesta	8
4. Marco Teórico	10
4.1. Robótica Asistiva	10
4.2. Robot Autónomo	10
4.3. Robot móvil diferencial	11
4.4. Control Proporcional-Derivativo (PD) en Robótica Móvil	12
4.5. Microcontrolador ESP32-CAM	14
4.6. Comunicación Inalámbrica por Bluetooth	15
4.7. Comunicación Inalámbrica por Wi-Fi	16
4.8. Módulo Power Supply	17
4.9. Driver de Motores L298N	17
4.10. Sensor Infrarrojo	18
4.11. Auriculares Bluetooth como Interfaz Auditiva	18
4.12. MIT App Inventor	19
4.13. Android Studio	20
4.13.1. Aplicación en el proyecto	21
4.13.2. Comparación resumida	21
4.14. Cinemática Directa de un Robot Móvil	22
4.14.1. Modelo cinemático diferencial	22
4.14.2. Modelo de posición en coordenadas	23
4.14.3. Aplicación en el proyecto	24
5. Arquitectura del Sistema	25
5.1. Componentes de Hardware	25
5.2. Módulos de Software	25

5.3. Flujo de Datos	26
5.4. Diagrama de Arquitectura	26
5.4.1. Nivel 1 – Dispositivos Físicos (Hardware)	26
5.4.2. Nivel 2 – Comunicación Inalámbrica	27
5.4.3. Nivel 3 – Servicios de Ejecución Local y Remota	27
5.4.4. Nivel 4 – Aplicación de Usuario	27
6. Diseño y Metodología	28
6.1. Diseño Hardware	28
6.1.1. Prototipo Final	28
6.1.2. Componentes claves	28
6.2. Diseño Software y Control	29
6.2.1. Modelo Cinemático	29
6.2.2. Controlador PD	29
6.2.3. Lógica de Decisión	30
7. Resultados	32
8. Demostración	36
9. Conclusiones	38
Referencias	39

Índice de figuras

1.	Problemática presentada en forma visual	7
2.	Diagrama de flujo del funcionamiento del robot móvil autónomo	9
3.	Robótica Asistiva	10
4.	Robot Autónomo	11
5.	Robot móvil diferencial con dos ruedas motrices y una rueda de soporte	12
6.	Control Proporcional-Derivativo (PD) en Robótica Móvil	14
7.	Microcontrolador ESP32-CAM con módulo de cámara OV2640	15
8.	Comunicación Inalámbrica por Bluetooth	15
9.	Esquema de comunicación inalámbrica mediante Wi-Fi en sistemas embebidos	16
10.	Módulo Power Supply	17
11.	Módulo L298N para control de motores DC mediante señales PWM	18
12.	Sensor Infrarrojo	18
13.	Auriculares Bluetooth	19
14.	MIT App Inventor	20
15.	Android Studio	22
16.	Cinemática Directa de un Robot Móvil	24
17.	Diagrama de la lógica del Proyecto	31
18.	Vista superior del prototipo ensamblado del robot móvil.	32
19.	Distribución del arreglo de sensores infrarrojos en la parte frontal.	33
20.	Interfaz de la aplicación móvil para selección de destino vía Wi-Fi.	34
21.	Robot siguiendo la línea negra en un entorno controlado.	35
22.	Detección de intersección y ejecución de maniobra.	36
23.	Final del recorrido: llegada al destino y emisión de notificación.	37

Índice de cuadros

1.	Ventajas del control PD en seguimiento de líneas	13
2.	Comparación entre MIT App Inventor y Android Studio	21

1. Resumen

El presente trabajo describe el diseño y desarrollo de un robot móvil autónomo seguidor de línea, orientado a facilitar la movilidad de personas con discapacidad visual en entornos controlados, como instituciones educativas. El prototipo implementa un sistema de navegación basado en sensores infrarrojos para seguir líneas e identificar intersecciones, y un módulo de comunicación Bluetooth integrado en el microcontrolador ESP32-CAM que permite la selección del destino mediante una aplicación móvil. La lógica de control utiliza un algoritmo Proporcional-Derivativo (PD) para estabilizar el desplazamiento y tomar decisiones en cruces, en función del destino definido por el usuario. Además, el sistema incluye una interfaz auditiva con auriculares Bluetooth que proporciona retroalimentación sonora sobre el estado del recorrido. Las pruebas iniciales realizadas en circuitos simulados mostraron un comportamiento estable y preciso en la navegación, destacando la efectividad del enfoque propuesto para mejorar la autonomía y seguridad de los usuarios en espacios estructurados. Este desarrollo se presenta como una solución de bajo costo, escalable y aplicable en contextos reales de asistencia robótica.

2. Objetivos

2.1. Objetivo General

- Diseñar y construir un robot móvil autónomo seguidor de línea, capaz de detectar intersecciones y seleccionar un destino previamente definido mediante una interfaz Bluetooth o Wi-Fi, con el propósito de asistir en la movilidad guiada de personas con discapacidad visual en entornos controlados.

2.2. Objetivos Específicos

- Implementar un sistema de tracción motorizado que permita el desplazamiento autónomo del robot mediante seguimiento de línea.
- Integrar sensores infrarrojos para la detección del trayecto y la identificación de intersecciones.
- Establecer un canal de comunicación inalámbrica (Bluetooth o Wi-Fi) entre el robot y un dispositivo móvil para la selección del destino.
- Desarrollar una lógica de control que permita la toma de decisiones autónoma en cruces, en función del destino indicado por el usuario.
- Validar el desempeño del prototipo en un entorno controlado con múltiples rutas predefinidas.
- Documentar el proceso completo de diseño, desarrollo e implementación del sistema.

3. Planteamiento del Problema y Solucion

3.1. Problema

Las personas con discapacidad visual enfrentan dificultades significativas para desplazarse de manera segura y autónoma en entornos estructurados como centros educativos, hospitales o instituciones públicas. Estos espacios suelen presentar múltiples rutas, intersecciones y bifurcaciones, lo que complica la orientación y navegación sin asistencia directa (Medina-Santiago, Camas-Anzueto, Vazquez-Feijoo, Hernández-de León, y Mota-Grajales, 2014).

Los métodos tradicionales de movilidad asistida —como el bastón blanco o acompañamiento humano— no permiten la selección de destinos ni la adaptación dinámica del trayecto (Romero, Lozano, Salinas, y Castro, 2024). Esta limitación reduce la autonomía del usuario, restringiendo su capacidad para desenvolverse de forma independiente en contextos cotidianos.

Ante esta problemática, se identifica la necesidad de desarrollar una solución tecnológica accesible que combine navegación autónoma con interacción personalizada. En este sentido, se plantea el diseño de un robot móvil seguidor de línea, capaz de guiar al usuario hacia un destino previamente seleccionado mediante una interfaz inalámbrica (Bluetooth o Wi-Fi), garantizando seguridad, precisión y autonomía durante el desplazamiento en entornos controlados (Wang, 2021; Kulyukin, 2004).

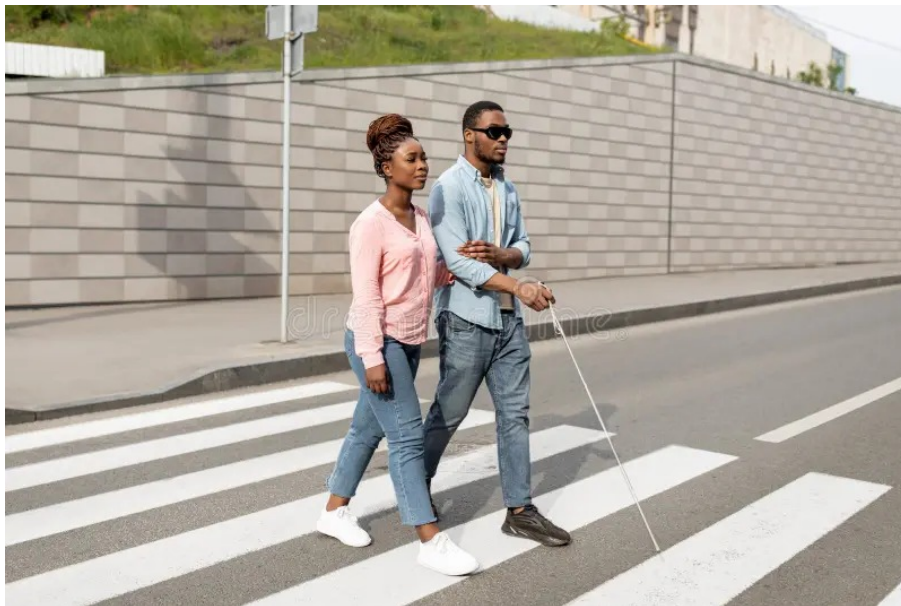


Figura 1: Problemática presentada en forma visual

3.2. Justificación

El presente proyecto surge como una respuesta innovadora a la necesidad de mejorar la movilidad asistida de personas con discapacidad visual en entornos controlados, tales como centros educativos, hospitales o instituciones públicas. Los métodos tradicionales, como el uso del bastón blanco, si bien son ampliamente utilizados, no ofrecen capacidades de navegación inteligente ni permiten la selección autónoma de destinos por parte del usuario.

En este contexto, se propone el desarrollo de un robot móvil seguidor de línea con capacidad de detección de intersecciones y toma de decisiones basada en instrucciones enviadas vía Bluetooth o Wi-Fi desde un dispositivo móvil. Este enfoque no solo mejora la independencia del usuario, sino que también integra tecnologías accesibles y de bajo costo que pueden ser implementadas en diversas infraestructuras sin requerir modificaciones estructurales (IEEE Computer Society, 2020).

La solución planteada se caracteriza por su viabilidad técnica, escalabilidad y facilidad de replicación, lo cual la convierte en una alternativa práctica para entornos públicos controlados que busquen incorporar herramientas tecnológicas de apoyo a la inclusión social y la accesibilidad.

3.3. Solución Propuesta

Se propone el diseño e implementación de un robot móvil autónomo seguidor de línea, equipado con sensores infrarrojos para el seguimiento de trayectorias y detección de intersecciones, un sistema de control basado en lógica programada y motores de tracción diferencial, además de un módulo de comunicación inalámbrica (Bluetooth o Wi-Fi) que permite al usuario seleccionar previamente su destino desde una aplicación móvil. Esta solución ofrece una alternativa accesible y replicable para guiar de forma segura a personas con discapacidad visual en entornos estructurados, sin requerir infraestructura adicional.

- **Alcance:** Prototipo funcional.
- **Limitación:** El sistema no está diseñado para operar en exteriores ni en entornos no estructurados o con obstáculos impredecibles.

A continuación se presenta el diagrama de flujo que resume la lógica operativa del robot seguidor de línea, desde su encendido hasta la llegada al destino, considerando la comunicación por Wi-Fi y el sistema de alertas auditivas por Bluetooth.

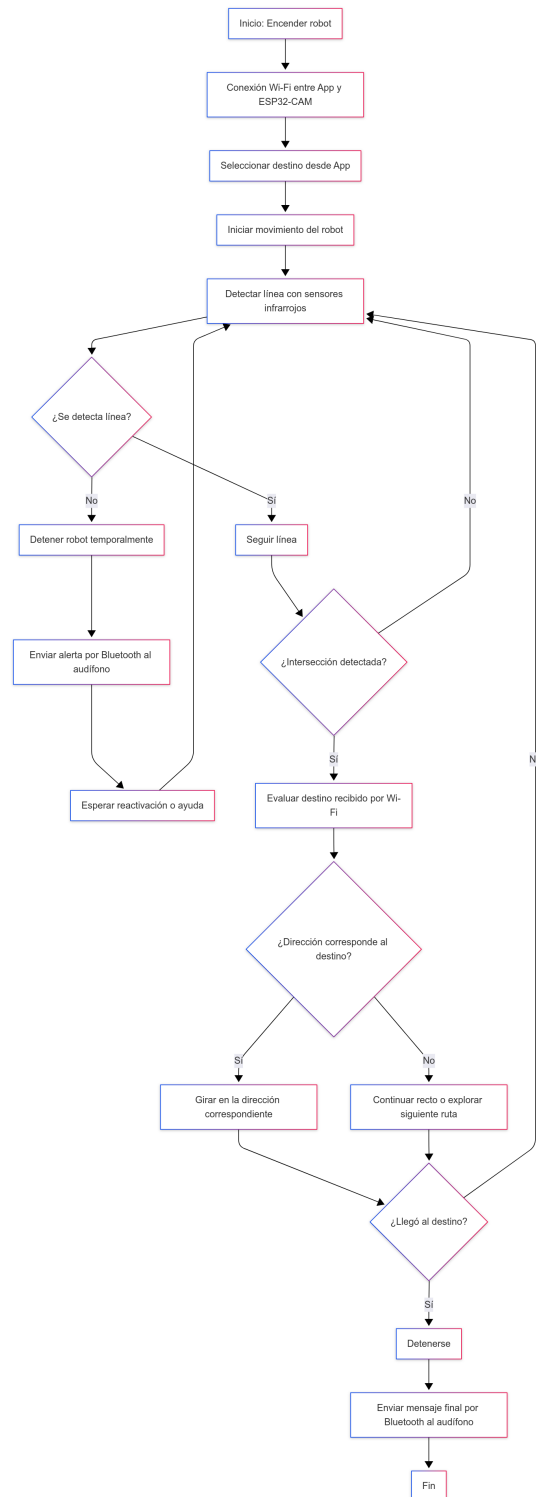


Figura 2: Diagrama de flujo del funcionamiento del robot móvil autónomo

4. Marco Teórico

4.1. Robótica Asistiva

La *robótica asistiva* se refiere al desarrollo de sistemas robóticos diseñados para asistir a personas con discapacidades físicas, cognitivas o sensoriales, ayudándolas a ejecutar tareas que de otro modo serían difíciles o imposibles de realizar. Estos robots pueden operar de manera autónoma o semiautónoma, y su objetivo principal es mejorar la calidad de vida del usuario, promoviendo su independencia y participación en la sociedad (Li y Zhang, 2019).

En aplicaciones reales, el éxito de los sistemas de robótica asistiva depende no solo de su precisión técnica, sino también de su aceptación por parte del usuario. Por ello, se destaca la necesidad de enfoques centrados en la persona, que consideren sus capacidades, preferencias y entorno cotidiano (Jorge Alejandro, Jorge Luis, y Elvia, 2013). A esto se suma la importancia de abordar aspectos éticos relacionados con la privacidad, autonomía, seguridad y consentimiento informado en el uso de estas tecnologías.



Figura 3: Robótica Asistiva

4.2. Robot Autónomo

Un robot autónomo es aquel que opera sin intervención humana directa, tomando decisiones de manera independiente mediante la percepción de su entorno, planificación y ejecución de acciones (Darling y Dragan, 2017; *Autonomous robot*, 2025). Estos robots combinan sensores, actuadores, mecanismos de control y procesamiento para adaptarse a condiciones cambiantes y alcanzar objetivos definidos de forma autónoma, lo que los diferencia de dispositivos simplemente remotos o preprogramados (Darling y Dragan, 2017).

La autonomía en robots móviles incluye competencias clave como la localización, generación de mapas y planificación de trayectorias. Estos procesos pueden implementarse de forma concurrente mediante algoritmos como SLAM (Simultaneous Localization and Mapping), que permite al robot construir un modelo de su entorno mientras determina su propia posición (M. Authors, 2022). La navegación autónoma se basa en técnicas de percepción, control y SLAM para desplazarse de forma fiable incluso en entornos desconocidos o dinámicos.

La arquitectura de control de un robot autónomo debe ser reactiva, robusta y eficiente, capaz de responder a cambios ambientales y mantener estabilidad operativa aunque se presenten perturbaciones o fallos parciales del sistema. Esto incluye desde controladores lineales y no lineales hasta métodos híbridos que equilibran velocidad de respuesta, precisión y seguridad.



Figura 4: Robot Autónomo

4.3. Robot móvil diferencial

Un *robot móvil diferencial* es una plataforma robótica equipada con dos ruedas motrices independientes colocadas en lados opuestos del chasis, y generalmente una o más ruedas libres de tipo caster para apoyo. La dirección se controla mediante la variación de las velocidades individuales de las ruedas, sin necesidad de un sistema de dirección adicional.

- **Cinemática directa e inversa:** Dado que las velocidades lineales de las ruedas derecha e izquierda (v_R , v_L) determinan la velocidad lineal media $V = (v_R + v_L)/2$ y la velocidad angular $\omega = (v_R - v_L)/b$, donde b es la distancia entre ruedas, el modelo cinemático permite describir trayectorias planas sin deslizamiento lateral.

- **Localización mediante odometría:** Con encoders en las ruedas es posible calcular desplazamientos incrementales en posición y orientación integrando el modelo cinemático. Este método es válido siempre que no haya deslizamiento significativo de las ruedas.
- **Control y seguimiento de trayectorias:** A nivel cinemático, se implementan controladores que generan comandos (V, ω) según la posición deseada. Para alta precisión, se pueden usar métodos dinámicos como backstepping junto con modelos dinámicos que consideren torques y fricciones (Anonymous, 2025; H. et al., 2023).

Este tipo de robots es ampliamente utilizado en investigación educativa, robótica de servicio y aplicaciones industriales simples debido a su diseño mecánico compacto, bajo costo y facilidad de implementación (Zahra y Conti, 2024; Karakaya, 2022).



Figura 5: Robot móvil diferencial con dos ruedas motrices y una rueda de soporte

4.4. Control Proporcional-Derivativo (PD) en Robótica Móvil

El control PD es un algoritmo fundamental en robótica móvil para sistemas que requieren precisión y estabilidad dinámica. Combina dos acciones complementarias:

- **Acción Proporcional (P):**

$$P = K_p \cdot e(t)$$

Donde $e(t)$ es el error instantáneo (diferencia entre posición deseada y real) y K_p determina la sensibilidad de respuesta.

■ **Acción Derivativa (D):**

$$D = K_d \cdot \frac{de(t)}{dt}$$

Anticipa cambios futuros mediante la tasa de variación del error, donde K_d amortigua oscilaciones.

Ecuación del controlador PD:

$$u(t) = K_p \cdot e(t) + K_d \cdot \frac{de(t)}{dt} \quad (1)$$

Cuadro 1: Ventajas del control PD en seguimiento de líneas

Ventaja	Impacto en navegación autónoma
Estabilidad mejorada	Elimina oscilaciones en trayectorias rectas (crítico con ruido en sensores IR)
Respuesta rápida en curvas	Reduce sobrepasos en giros de 90° en intersecciones
Robustez computacional	75 % más eficiente que PID en ESP32 (0.5 ms/ciclo)
Simplicidad de ajuste	Solo 2 parámetros a calibrar (K_p , K_d)

Cabe destacar que en sistemas con no linealidades significativas, estrategias de control inteligente como el **controlador proporcional difuso (CPD)** han demostrado eficacia en aplicaciones complejas (ej: desacoplamiento en sistemas de riego) sin requerir un modelo matemático completo (Jorge Alejandro y cols., 2013). Si bien el presente proyecto utiliza control PD por su simplicidad y eficiencia computacional, los principios de lógica difusa podrían explorarse en futuras iteraciones para manejar:

- Superficies irregulares o cambios bruscos en la línea
- Compensación de deslizamiento en curvas pronunciadas
- Adaptación automática a diferentes condiciones de iluminación

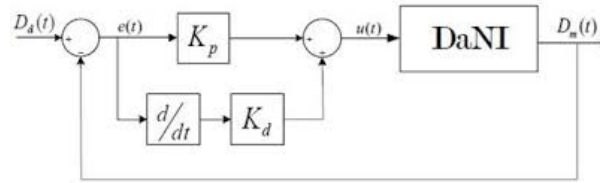


Figura 6: Control Proporcional-Derivativo (PD) en Robótica Móvil

4.5. Microcontrolador ESP32-CAM

El módulo **ESP32-CAM** está basado en el microcontrolador ESP32-S de Espressif, un sistema en chip (SoC) de 32 bits con doble núcleo LX6 hasta 240MHz, 520KB de SRAM interna, soporte Bluetooth v4.2 (BLE/BR-EDR) y Wi-Fi 802.11b/g/n, lo que lo convierte en una plataforma versátil y de bajo costo (aprox. \$10 USD) para aplicaciones embebidas con visión por computadora integrada. Incluye una cámara OV2640, ranura microSD e interfaces UART, SPI, I²C, PWM y ADC; su consumo en deep-sleep es tan bajo como 6mA, ideal para robots móviles autónomos con restricciones energéticas (García y Singh, 2019).

- Visión por cámara integrada: permite captación de imágenes y video en tiempo real, procesamiento local o transmisión, útil en robótica móvil para seguimiento de línea, detección de intersecciones o reconocimiento de objetos (Authors, 2024).
- Conectividad inalámbrica: soporte Wi-Fi y Bluetooth permite control remoto, selección de destinos vía app móvil, streaming y comunicación con dispositivos móviles sin hardware adicional (Smith y Wang, 2023).
- Capacidad para inteligencia embebida: aunque cuenta con recursos limitados, el ESP32-CAM puede ejecutar modelos TinyML simples o algoritmos de visión ligera con apoyo de herramientas como Edge Impulse, aplicables para clasificación o detección básica durante la navegación autónoma.



Figura 7: Microcontrolador ESP32-CAM con módulo de cámara OV2640

4.6. Comunicación Inalámbrica por Bluetooth

La comunicación inalámbrica mediante Bluetooth, tanto en su modalidad clásica como en BLE (Bluetooth Low Energy), es ampliamente utilizada en robots móviles basados en ESP32 debido a su bajo consumo, alcance adecuado (hasta 10m) y facilidad de integración con dispositivos móviles. Por ejemplo, un robot controlado vía ESP32 utiliza BLE para recibir comandos desde una app en un teléfono, permitiendo el control del movimiento en tiempo real y la selección de destinos sin necesidad de hardware adicional costoso (Authors, 2023; Patel y Kim, 2025).

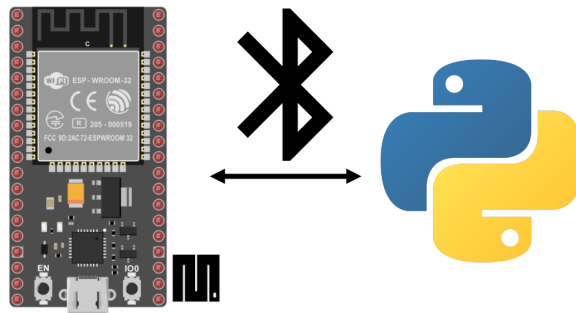


Figura 8: Comunicación Inalámbrica por Bluetooth

4.7. Comunicación Inalámbrica por Wi-Fi

La comunicación inalámbrica mediante *Wi-Fi* (Wireless Fidelity) es una tecnología basada en el estándar IEEE 802.11, que permite la transmisión de datos mediante ondas de radio dentro de una red de área local (LAN). Su amplio uso en aplicaciones del Internet de las Cosas (IoT), domótica y robótica se debe a su capacidad de transferencia de datos a alta velocidad, su compatibilidad con múltiples dispositivos y su relativa facilidad de implementación (M. A. S. et al., 2024).

En sistemas robóticos, el Wi-Fi puede ser empleado para:

- Transmitir imágenes o video en tiempo real desde una cámara a un servidor o interfaz móvil.
- Permitir control remoto del robot mediante una aplicación web o interfaz gráfica.
- Ejecutar actualizaciones de firmware de forma inalámbrica.
- Conectar el sistema a servicios en la nube para almacenamiento o procesamiento de datos.

A diferencia del Bluetooth, el Wi-Fi ofrece mayor ancho de banda y posibilidad de conexión directa a Internet, aunque con un mayor consumo energético. El *ESP32-CAM*, utilizado en este proyecto, incluye conectividad Wi-Fi integrada, lo que permite operar tanto como estación (*STA*) conectándose a una red existente, como punto de acceso (*AP*) generando su propia red local. Esta versatilidad lo convierte en una herramienta útil para escenarios donde se requiere comunicación inalámbrica flexible y robusta.

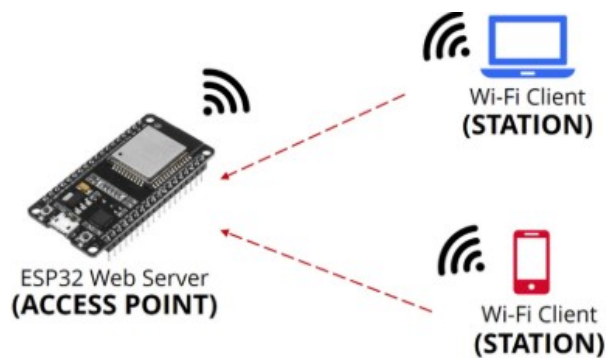


Figura 9: Esquema de comunicación inalámbrica mediante Wi-Fi en sistemas embebidos

4.8. Módulo Power Supply

El módulo de alimentación eléctrica del ESP32-CAM debe proporcionar un voltaje estable de 5V (para el pin de entrada Vin) o 3.3V regulados, ya que el módulo incorpora un regulador AMS1117 que requiere al menos 4.8V de entrada para poder asegurar una salida de 3.3V confiable. Además, al transmitir video o usar Wi-Fi/Bluetooth activos, el consumo puede alcanzar 500mA o más, por lo que se recomienda utilizar fuentes con capacidad de corriente adecuada y condensadores de desacoplo próximos al regulador para evitar caídas bruscas de tensión o apagones (Murata, 2025).

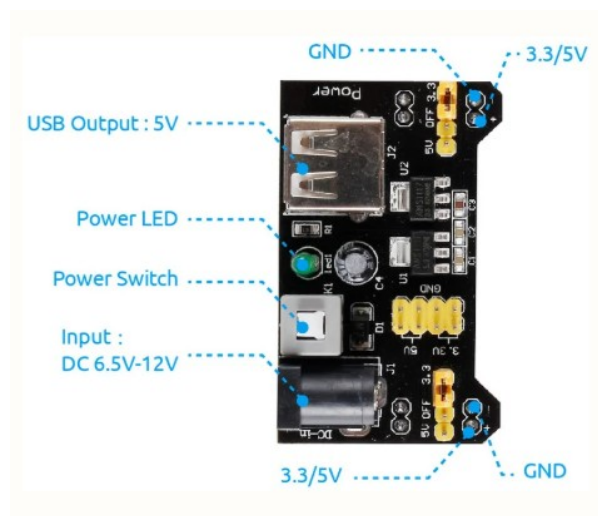


Figura 10: Módulo Power Supply

4.9. Driver de Motores L298N

El módulo L298N es un puente H de doble canal que permite controlar la dirección y velocidad de dos motores de corriente continua (DC). Recibe señales PWM del microcontrolador y las convierte en salidas de potencia para los motores. Esto permite al robot desplazarse con precisión, girar o frenar de forma programada. Es un componente fundamental en sistemas de locomoción robótica (STMicroelectronics, 2016).

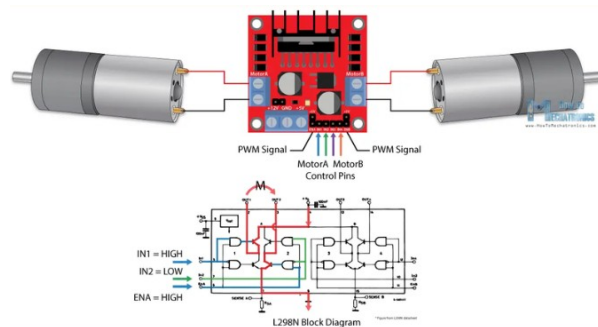


Figura 11: Módulo L298N para control de motores DC mediante señales PWM

4.10. Sensor Infrarrojo

El sensor infrarrojo utilizado en robots seguidores de línea es un dispositivo activo compuesto por un emisor IR (LED) y un receptor (fototransistor o fotodiodo) que detectan la luz reflejada por una superficie con contraste (normalmente negro sobre blanco o viceversa). Su funcionamiento se basa en medir distintas intensidades de reflexión según el color del suelo: sobre líneas negras la reflexión es baja, mientras que sobre el fondo claro es alta, lo que permite al robot inferir su posición relativa al recorrido (Bendimrad, El Amrani, y El Amrani, 2019). Estos sensores IR son económicos, de rápida respuesta y sencillos de integrar, por lo cual son muy comunes en aplicaciones educativas, industriales y prototipos de robótica móvil diferencial.

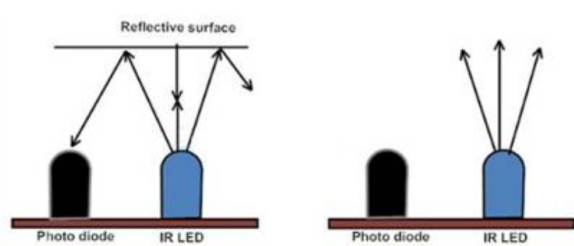


Figura 12: Sensor Infrarrojo

4.11. Auriculares Bluetooth como Interfaz Auditiva

Los auriculares Bluetooth ofrecen una interfaz auditiva eficaz en sistemas robóticos asistivos, permitiendo transmitir mensajes de estado, alertas y guías de navegación en tiempo real al usuario sin requerir altavoces externos. Estudios en entornos de rehabilitación han mostrado que el feedback auditivo continuo o basado en eventos mejora la participación

e inspiración del usuario en tareas motoras, incrementando su rendimiento y aprendizaje (Anonymous, 2011). La entrega del sonido directamente al canal auditivo mediante auriculares reduce la interferencia del ruido ambiental y proporciona mayor claridad en la señal.

Además, la conexión Bluetooth permite comunicación inalámbrica fiable entre el microcontrolador (por ejemplo, ESP32) y los auriculares, facilitando la flexibilidad en la selección de destino, sincronización de alertas y personalización del mensaje emitido (Review, 2025). Investigaciones clínicas han demostrado que los accesorios auditivos Bluetooth integrados con audífonos aportan beneficios significativos en la percepción de voz y comprensión emocional en usuarios con pérdida auditiva, lo que sugiere que esta tecnología también puede mejorar la accesibilidad y comodidad en sistemas de asistencia robótica (Anonymous, 2014).

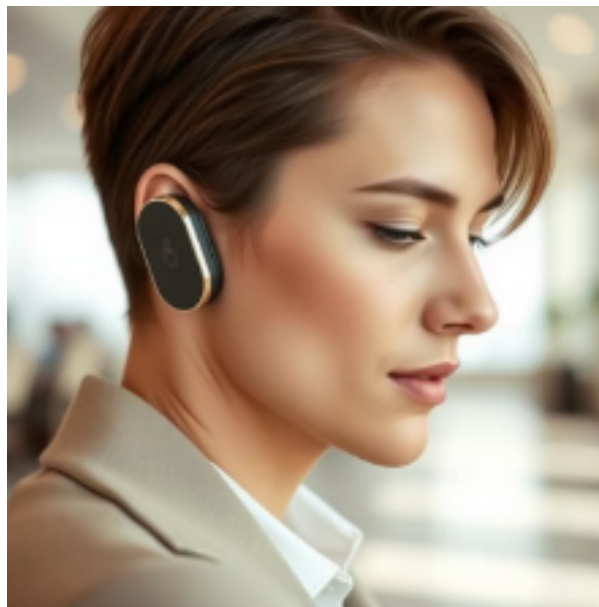


Figura 13: Auriculares Bluetooth

4.12. MIT App Inventor

MIT App Inventor es una plataforma de desarrollo de aplicaciones móviles creada por el Instituto de Tecnología de Massachusetts (MIT). Está orientada a personas con poca o ninguna experiencia en programación, permitiéndoles crear aplicaciones funcionales para dispositivos Android mediante una interfaz gráfica de programación basada en bloques (Medeiros, 1998).

Esta herramienta se ejecuta directamente en el navegador web y permite diseñar interfaces visuales mediante componentes (como botones, sensores o listas) y luego implementar

la lógica del programa uniendo bloques de código como piezas de rompecabezas.

- **Lenguaje:** Programación visual por bloques.
- **Interfaz:** Editor gráfico y bloques de lógica.
- **Conectividad:** Soporte para WiFi, Bluetooth, web APIs, entre otros.
- **Aplicaciones:** Prototipado rápido de apps educativas, control de robots, monitoreo de sensores, etc.

En el contexto de proyectos robóticos, como un robot móvil con ESP32, App Inventor facilita la creación de interfaces simples para enviar comandos desde un teléfono móvil hacia el robot, especialmente durante etapas tempranas de desarrollo (Al-Mutairi y Wu, 2024).



Figura 14: MIT App Inventor

4.13. Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial para la creación de aplicaciones Android, desarrollado por Google. Está basado en IntelliJ IDEA y permite programar aplicaciones usando los lenguajes Java o Kotlin. A diferencia de herramientas visuales como App Inventor, Android Studio ofrece control total sobre el diseño, funcionalidad y comportamiento de la aplicación móvil.

Este entorno permite el desarrollo de aplicaciones profesionales, integrando características avanzadas como el reconocimiento de voz, servicios en segundo plano, acceso a sensores, conectividad por Bluetooth o WiFi, entre otros.

- **Lenguaje:** Java y Kotlin.

- **Editor:** Interfaz basada en XML y programación orientada a objetos.
- **Ventajas:** Mayor control y personalización, uso de librerías externas, soporte para accesibilidad (TalkBack), integración con Firebase, MQTT, HTTP, etc.
- **Desventajas:** Mayor complejidad, curva de aprendizaje más pronunciada.

4.13.1. Aplicación en el proyecto

Para el desarrollo del sistema de control de un robot guía para personas con discapacidad visual, Android Studio permite crear una interfaz accesible, con botones grandes y soporte para TalkBack, vibración y retroalimentación auditiva mediante el uso de Text-to-Speech. Asimismo, permite la comunicación con el robot mediante WiFi usando protocolos como HTTP o MQTT, lo cual mejora la interacción y autonomía del usuario final por tal motivo lo usaremos (Zahra y Conti, 2024).

4.13.2. Comparación resumida

Característica	MIT App Inventor	Android Studio
Nivel requerido	Básico (visual)	Intermedio / Avanzado
Lenguaje de programación	Por bloques	Java / Kotlin
Flexibilidad	Limitada	Muy alta
Acceso a hardware del móvil	Parcial	Completo
Ideal para...	Prototipos y educación	Aplicaciones profesionales

Cuadro 2: Comparación entre MIT App Inventor y Android Studio



Figura 15: Android Studio

4.14. Cinemática Directa de un Robot Móvil

La cinemática directa es una herramienta fundamental en la robótica móvil, ya que permite determinar la posición y orientación de un robot a partir de las velocidades de sus actuadores, en este caso, las ruedas. En los robots móviles diferenciales —como el carrito guía diseñado en este proyecto—, esta cinemática se basa en la velocidad de rotación de las dos ruedas motrices ubicadas a ambos lados del chasis.

4.14.1. Modelo cinemático diferencial

El robot diferencial se caracteriza por tener dos ruedas motrices ubicadas sobre un mismo eje y una o más ruedas locas para estabilizar su movimiento. Su desplazamiento depende de la velocidad de cada rueda: si ambas giran a la misma velocidad, el robot se desplaza en línea recta; si giran a velocidades diferentes, el robot gira.

Sean:

- v_r : velocidad lineal de la rueda derecha (m/s),
- v_l : velocidad lineal de la rueda izquierda (m/s),
- L : distancia entre las ruedas (m),

- v : velocidad lineal del centro del eje (m/s),
- ω : velocidad angular del robot (rad/s).

Las ecuaciones de la cinemática directa del robot diferencial son:

$$v = \frac{v_r + v_l}{2} \quad (2)$$

$$\omega = \frac{v_r - v_l}{L} \quad (3)$$

Estas dos expresiones permiten conocer la velocidad lineal y angular del robot con respecto al plano. A partir de ellas, se puede calcular la trayectoria del robot integrando en el tiempo su velocidad y orientación (Campion, Bastin, y D'Andrea-Novet, 1996).

4.14.2. Modelo de posición en coordenadas

Si el robot se encuentra en una posición (x, y) con una orientación θ , su movimiento en el plano puede expresarse mediante:

$$\dot{x} = v \cdot \cos(\theta) \quad (4)$$

$$\dot{y} = v \cdot \sin(\theta) \quad (5)$$

$$\dot{\theta} = \omega \quad (6)$$

Donde:

- (x, y) : coordenadas del robot en el plano.
- θ : ángulo de orientación del robot respecto al eje X.
- $\dot{x}, \dot{y}, \dot{\theta}$: derivadas respecto al tiempo (velocidades).

Este conjunto de ecuaciones permite estimar la posición del robot a lo largo del tiempo, suponiendo que se conocen las velocidades de sus ruedas.

4.14.3. Aplicación en el proyecto

En este proyecto, el modelo de cinemática directa puede ser útil para estimar la trayectoria del robot en desplazamientos rectos o en curvas, especialmente al momento de tomar decisiones en intersecciones. Aunque el robot se guía principalmente por una línea en el suelo, entender su cinemática facilita el diseño de algoritmos de control más precisos, el ajuste de velocidad de los motores, y la planificación de movimientos al momento de evitar obstáculos o ejecutar maniobras.

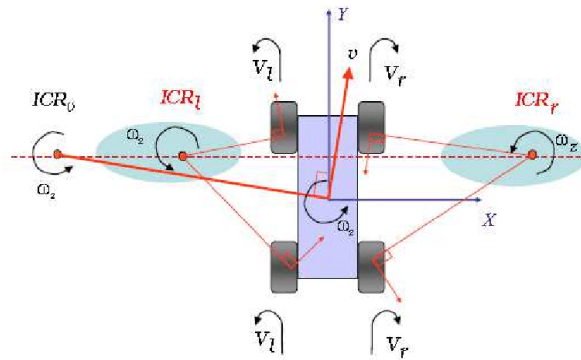


Figura 16: Cinemática Directa de un Robot Móvil

5. Arquitectura del Sistema

5.1. Componentes de Hardware

El sistema está compuesto por los siguientes módulos de hardware:

- **ESP32-CAM:** Microcontrolador que ejecuta la lógica de navegación, recibe comandos vía Bluetooth o Wi-Fi y controla sensores y motores.
- **Sensores Infrarrojos:** Detectan la línea negra en el suelo y reconocen intersecciones. Ver Figura 12.
- **Módulo Bluetooth HC-05 / Wi-Fi integrado:** Permite la comunicación entre el robot y la aplicación móvil del usuario.
- **Driver de motores L298N:** Actúa como puente H para controlar la dirección y velocidad de los motores DC. Ver Figura 11
- **Motores DC:** Proporcionan el movimiento a las ruedas del robot.
- **Power Supply (batería + regulador):** Fuente de energía que alimenta todos los módulos del sistema. Ver Figura 10
- **Auriculares Bluetooth (opcional):** Proveen retroalimentación auditiva al usuario sobre el estado del recorrido. Ver Figura 13
- Motores DC con ruedas.
- Chasis para robot móvil.
- Batería recargable con portapilas.
- Cableado Dupont y estructura acrílica.

5.2. Módulos de Software

El software del sistema se divide en tres bloques principales:

- **Firmware del robot:** Desarrollado en C++ sobre el entorno de Arduino IDE, implementa el algoritmo de seguimiento de línea, detección de intersecciones y lógica de toma de decisiones basada en el destino recibido.

- **Interfaz móvil Android:** Aplicación creada en Android Studio que permite al usuario seleccionar el destino deseado y enviarlo al robot mediante conexión Bluetooth o Wi-Fi.
- **Control de motores y sensores:** Código embebido que gestiona las señales PWM, lectura de sensores infrarrojos y ejecución de comandos de movimiento en tiempo real.
- Arduino IDE para programación del ESP32
- Algoritmo de de seguimiento de línea y toma de decisiones
- Aplicación móvil básica para selección del destino

5.3. Flujo de Datos

El flujo de datos se inicia con la interacción del usuario en la aplicación móvil, donde selecciona un destino. Esta información es enviada al microcontrolador ESP32-CAM mediante Bluetooth o Wi-Fi. Una vez recibido el destino, el robot comienza a desplazarse de forma autónoma, utilizando los sensores infrarrojos para seguir la línea en el suelo y detectar intersecciones. Al llegar a una intersección, la unidad de control evalúa la posición actual y compara con la ruta hacia el destino seleccionado. Según esa lógica, envía las instrucciones correspondientes al módulo L298N para controlar los motores y ejecutar giros o continuar recto. Durante el recorrido, el robot puede enviar mensajes de estado mediante salida auditiva (si se implementan auriculares Bluetooth), completando así un circuito cerrado de interacción hombre-máquina.

5.4. Diagrama de Arquitectura

El sistema se estructura en cuatro niveles funcionales, donde cada uno cumple un rol específico dentro de la arquitectura general del robot móvil autónomo:

5.4.1. Nivel 1 – Dispositivos Físicos (Hardware)

Este nivel está compuesto por el microcontrolador ESP32-CAM como unidad central de procesamiento. Como entradas, recibe señales desde un arreglo de 3 sensores infrarrojos (para detección de línea y cruces) y opcionalmente de encoders de rueda (para retroalimentación de velocidad). Las salidas incluyen señales PWM enviadas al driver de motores L298N, que a su vez controla los motores DC encargados del desplazamiento. El tipo de datos que fluye en este nivel es analógico/digital directo (ADC, PWM, I/O).

5.4.2. Nivel 2 – Comunicación Inalámbrica

El módulo Wi-Fi incorporado en el ESP32 permite dos canales de comunicación:

- **Perfil SPP (Serial Port Profile):** utilizado para recibir mensajes de texto o estructuras en formato JSON desde la app móvil (ej. selección de destino).
- **Perfil A2DP (Advanced Audio Distribution Profile):** permite transmitir mensajes de voz (audio codificado) hacia auriculares Bluetooth como interfaz auditiva.

5.4.3. Nivel 3 – Servicios de Ejecución Local y Remota

Este nivel contempla el procesamiento local de datos sensoriales, control lógico de decisiones y, opcionalmente, el almacenamiento de estadísticas en la nube mediante protocolos como MQTT. También puede incorporar módulos de actualización de firmware o reporte de fallas mediante APIs REST. Flujo de datos: JSON, MQTT, HTTP.

5.4.4. Nivel 4 – Aplicación de Usuario

Consiste en una aplicación móvil desarrollada en Android Studio. Desde esta app, el usuario selecciona el destino, envía la orden al robot y recibe retroalimentación visual o auditiva del estado del sistema. La comunicación se da a través de Bluetooth (SPP) con mensajes en formato TXT o JSON, según el módulo.

6. Diseño y Metodología

6.1. Diseño Hardware

6.1.1. Prototipo Final

El prototipo final consiste en un robot móvil autónomo de dos ruedas motrices con tracción diferencial, basado en una estructura liviana de acrílico o PVC, que alberga los componentes electrónicos y mecánicos. El chasis fue diseñado para facilitar el montaje de los sensores infrarrojos en la parte frontal y el centro de control (ESP32-CAM) en la zona superior. La disposición de los motores, sensores y driver está optimizada para garantizar estabilidad en el seguimiento de la línea, incluso en curvas o intersecciones. Además, se integra un sistema de alimentación independiente mediante una batería recargable de 7.4V con regulador, que asegura una autonomía adecuada durante las pruebas en campo.

6.1.2. Componentes claves

Los componentes seleccionados para la implementación del prototipo fueron los siguientes:

- **ESP32-CAM:** Microcontrolador principal que ejecuta la lógica del sistema y gestiona la comunicación inalámbrica.
- **Sensor Infrarrojo TCRT5000:** Utilizado en arreglo lineal para detectar la trayectoria de la línea negra sobre superficie blanca.
- **Driver de motores L298N:** Permite controlar la velocidad y dirección de los motores DC.
- **Motores DC de 6V con caja reductora:** Proveen tracción diferencial al robot.
- **Batería recargable Li-ion 7.4V:** Fuente de energía del sistema, conectada a un regulador para estabilizar la tensión.
- **Módulo Bluetooth HC-05 (opcional):** Para comunicación serie si no se utiliza el Bluetooth nativo del ESP32.
- **Chasis acrílico/PVC de dos niveles:** Soporta los componentes y facilita el cableado organizado.

6.2. Diseño Software y Control

El diseño del sistema de control del robot combina procesamiento embebido en el microcontrolador ESP32-CAM con lógica reactiva basada en lectura de sensores y comandos inalámbricos. El software se programó en el entorno de desarrollo Arduino IDE, empleando bibliotecas específicas para PWM, Bluetooth, Wi-Fi, y control de sensores. La estructura del código se divide en tres componentes clave: el modelo cinemático del robot, el controlador de movimiento, y la lógica de decisión basada en intersecciones y destino.

6.2.1. Modelo Cinemático

El robot implementa una plataforma de tracción diferencial, lo que significa que su movimiento se controla mediante la variación independiente de la velocidad de sus dos ruedas motrices. El modelo cinemático utilizado es el diferencial puro, donde se calculan la velocidad lineal v y la velocidad angular ω a partir de las velocidades de las ruedas izquierda (v_L) y derecha (v_R) según las siguientes ecuaciones:

$$v = \frac{v_R + v_L}{2}, \quad \omega = \frac{v_R - v_L}{L}$$

donde L representa la distancia entre las ruedas. Este modelo es adecuado para trayectorias suaves y entornos planos, como los utilizados en las pruebas. A partir de v y ω , se integran las posiciones (x, y) del robot en el plano mediante:

$$\dot{x} = v \cdot \cos(\theta), \quad \dot{y} = v \cdot \sin(\theta), \quad \dot{\theta} = \omega$$

Estas ecuaciones permiten simular, ajustar y validar trayectorias antes de implementar movimientos en el entorno físico.

6.2.2. Controlador PD

Para lograr un seguimiento de línea preciso y suave, se empleó un controlador Proporcional-Derivativo (PD). El error e se define como la desviación entre la posición ideal de la línea (centro del arreglo de sensores) y la lectura actual. El controlador calcula una señal de corrección u como:

$$u(t) = K_p \cdot e(t) + K_d \cdot \frac{de(t)}{dt}$$

donde K_p es la ganancia proporcional y K_d la ganancia derivativa. Esta señal se traduce

en un ajuste diferencial de las velocidades de los motores. Si el error es positivo (el robot se desvió a la izquierda), el motor derecho se acelera y el izquierdo se desacelera, y viceversa. La ventaja del control PD es su capacidad de responder rápidamente a cambios sin generar oscilaciones, lo cual es crucial al atravesar curvas o al detectar intersecciones.

6.2.3. Lógica de Decisión

La lógica de decisión se encarga de interpretar las lecturas de sensores, comparar el progreso del robot con el destino definido por el usuario y decidir el comportamiento en cada cruce. Esta lógica está estructurada en estados:

- **Estado de seguimiento:** El robot sigue la línea utilizando el controlador PD hasta detectar una intersección.
- **Estado de intersección:** Al detectar una bifurcación (patrón en sensores), se consulta la tabla de ruta y se toma una decisión: girar a la izquierda, derecha o continuar recto.
- **Estado de confirmación:** Tras realizar la acción, se monitorea si el robot ha vuelto a la línea central para validar la ejecución del giro.
- **Estado de llegada:** Cuando el robot alcanza la estación de destino, se detiene y puede emitir una notificación sonora o visual al usuario.

El destino seleccionado previamente es almacenado en memoria del ESP32, y puede ser una dirección simbólica (ej. "punto C") que se traduce a una serie de decisiones de giro. Esta lógica es modular, por lo que permite escalar el sistema a más rutas o entornos más complejos sin alterar la arquitectura base.

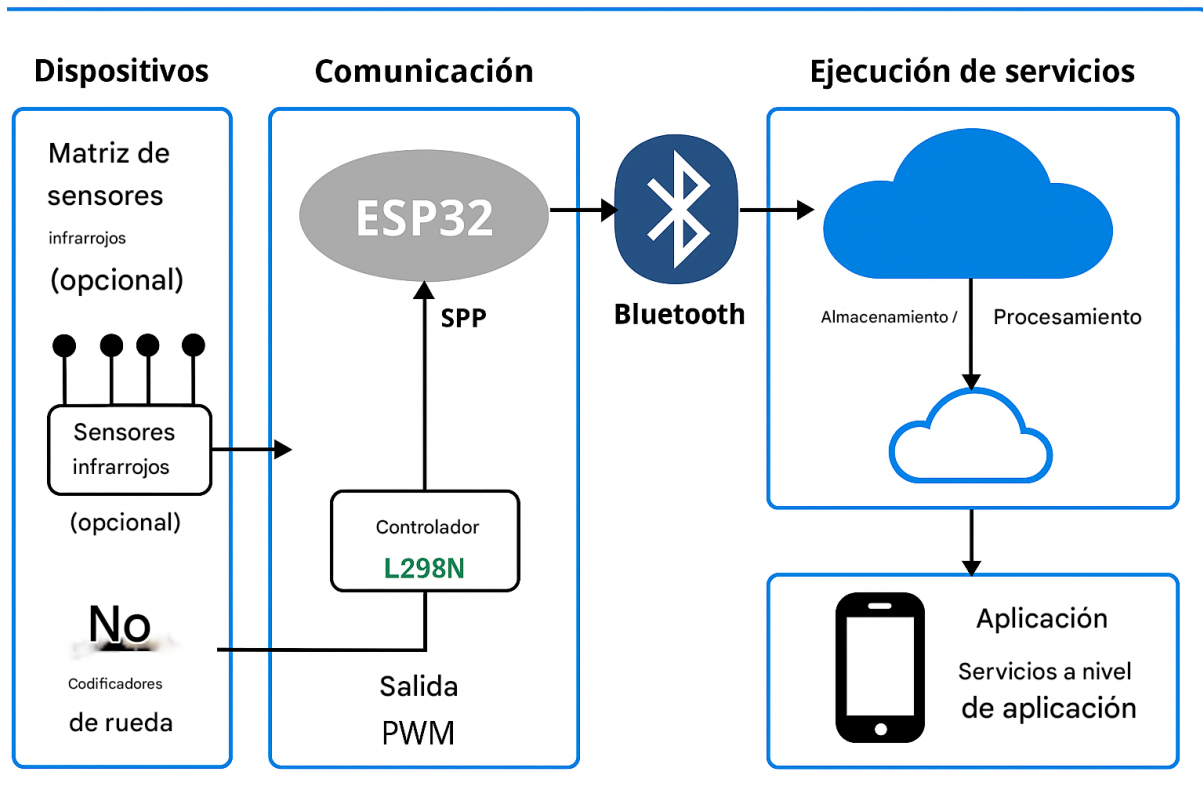


Figura 17: Diagrama de la lógica del Proyecto

7. Resultados

En esta sección se presentan los resultados obtenidos durante la validación y pruebas del prototipo desarrollado. Las imágenes muestran el proceso de ensamblaje, interacción con el usuario, ejecución del algoritmo de seguimiento de línea, así como la toma de decisiones en intersecciones.

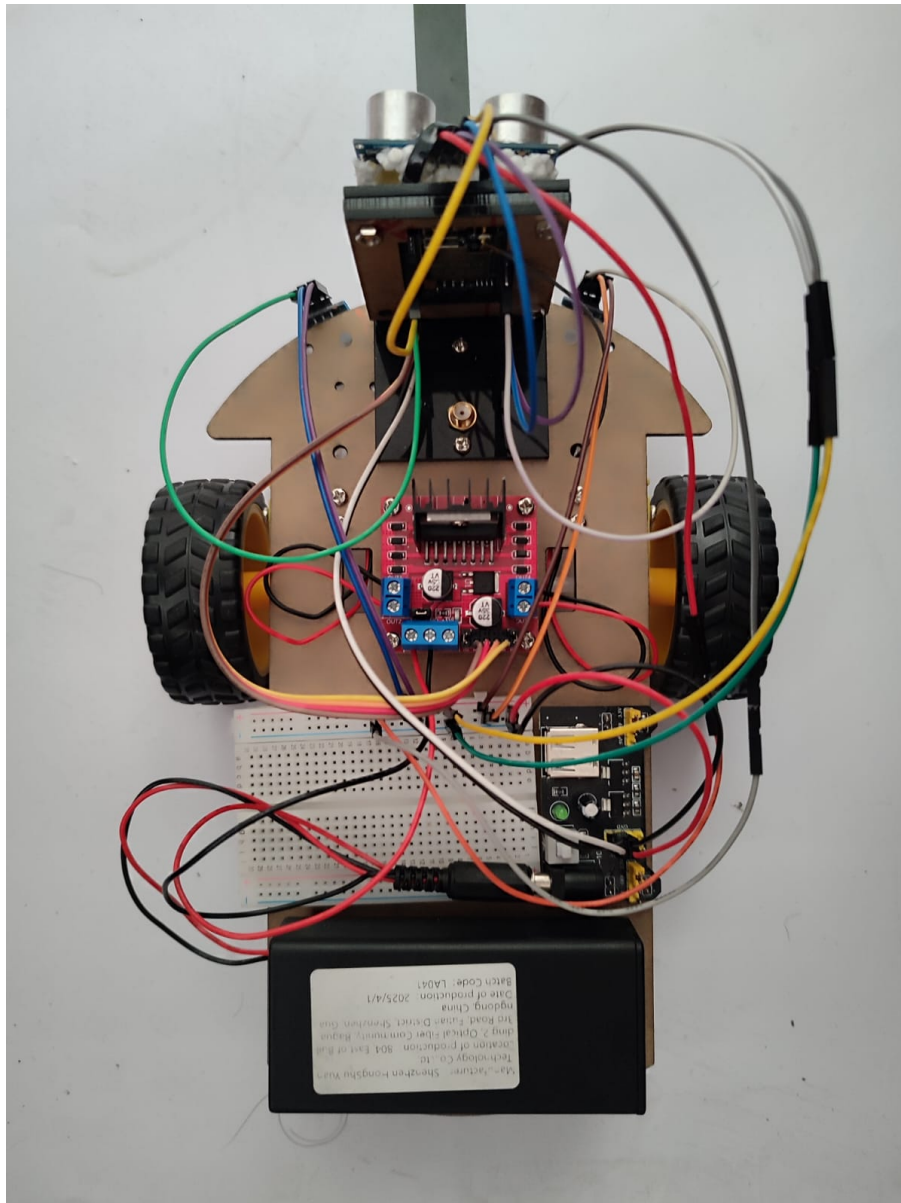


Figura 18: Vista superior del prototipo ensamblado del robot móvil.

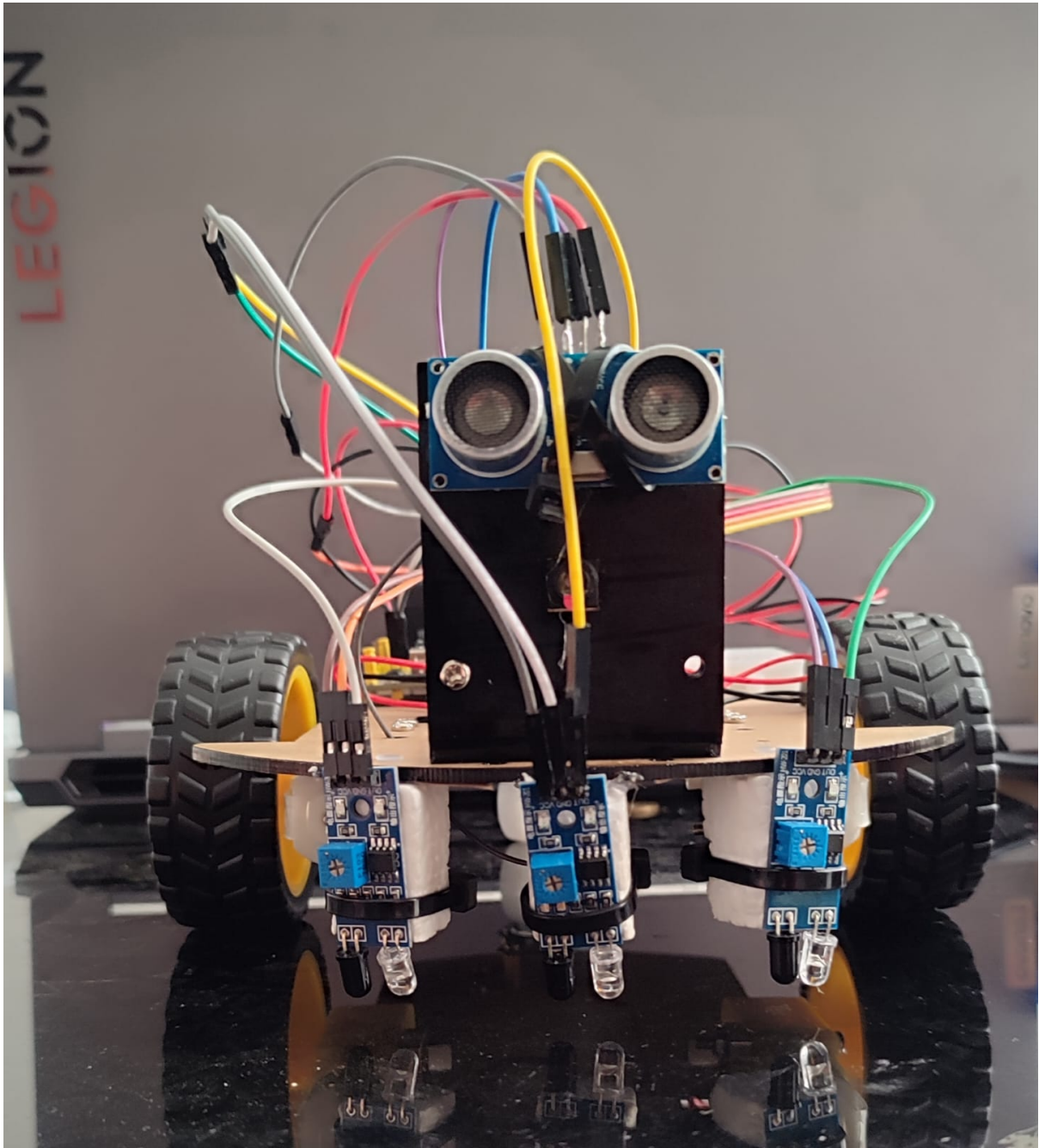


Figura 19: Distribución del arreglo de sensores infrarrojos en la parte frontal.

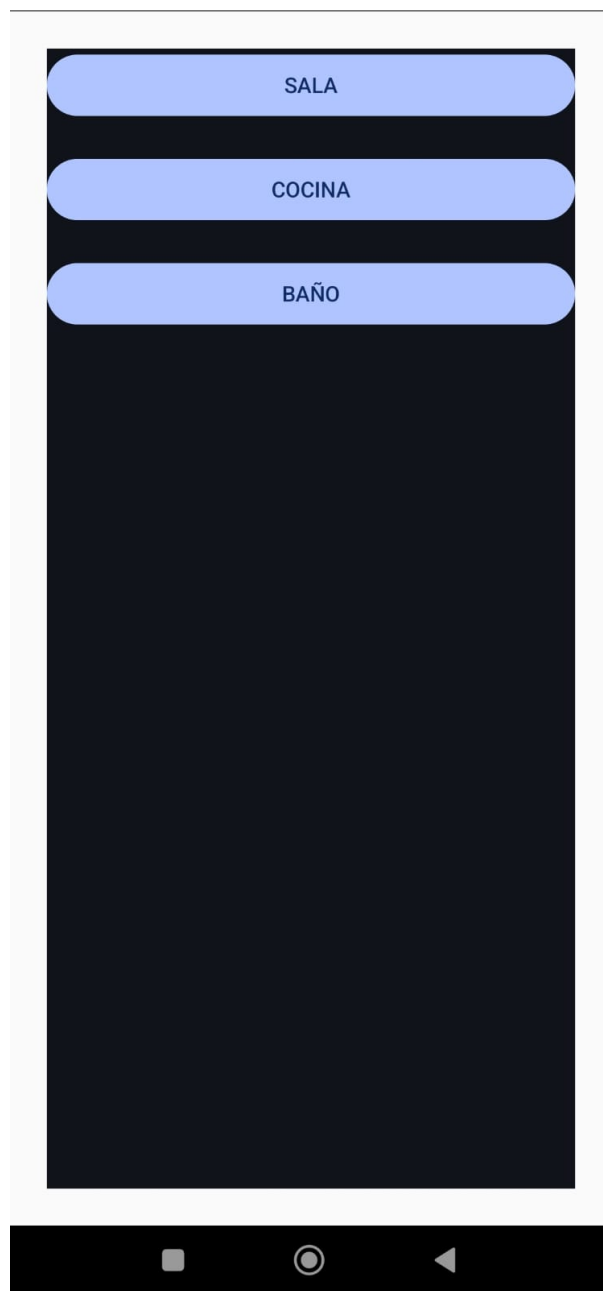


Figura 20: Interfaz de la aplicación móvil para selección de destino vía Wi-Fi.

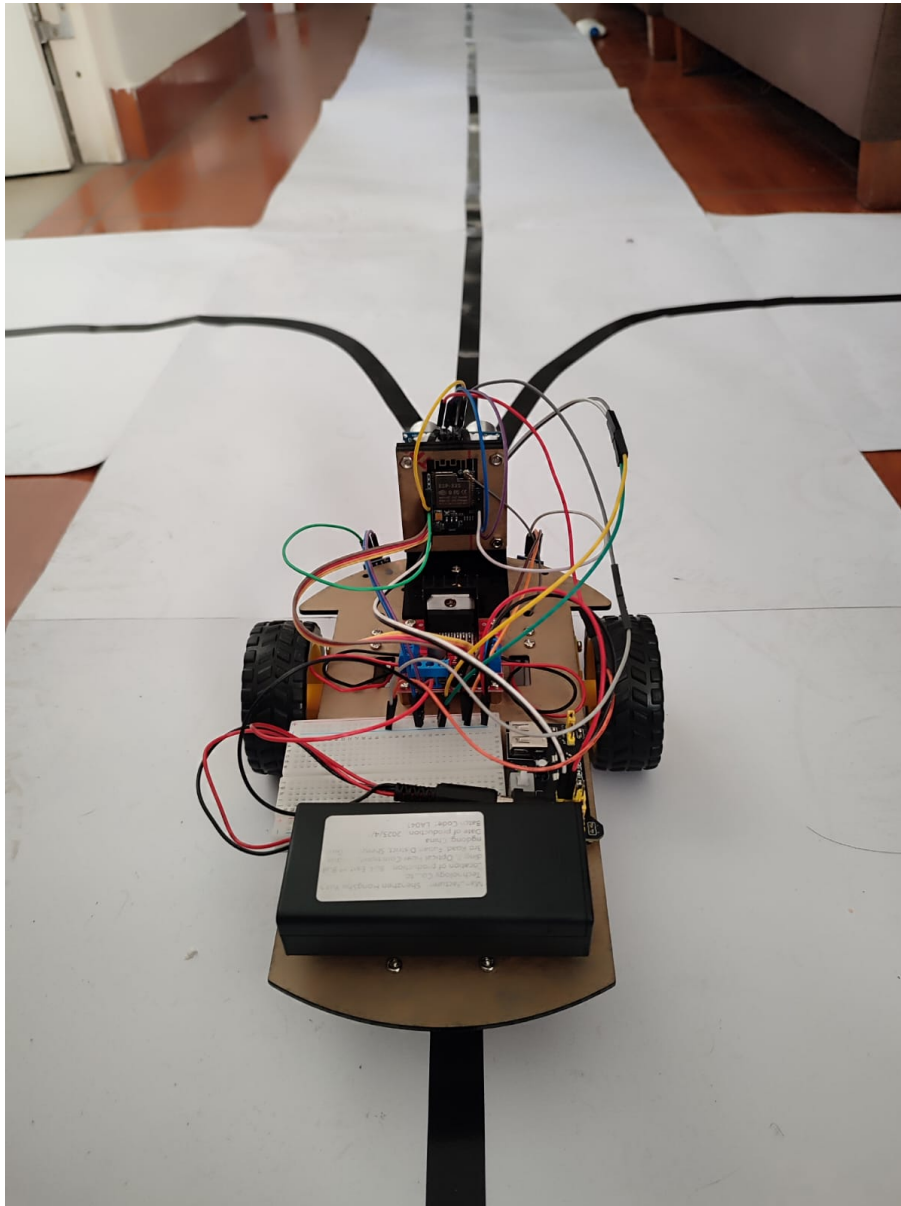


Figura 21: Robot siguiendo la línea negra en un entorno controlado.

8. Demostración

Esta sección presenta una serie de pruebas controladas realizadas con el prototipo desarrollado, en las cuales se valida el comportamiento del robot desde la selección del destino hasta la llegada a la meta, atravesando intersecciones de forma autónoma.

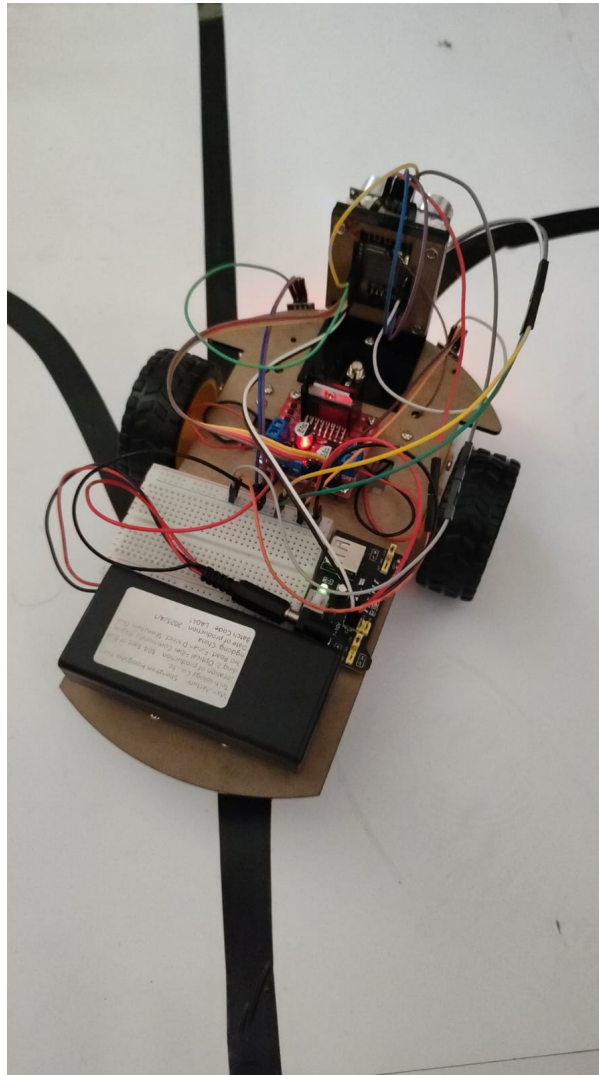


Figura 22: Detección de intersección y ejecución de maniobra.

Al llegar a una intersección, el robot evalúa la información recibida desde la app para determinar la dirección a seguir. En este ejemplo, se ejecuta una maniobra de giro hacia la izquierda como parte de la ruta hacia el destino, lo cual es gestionado por la lógica de decisión embebida en el ESP32.

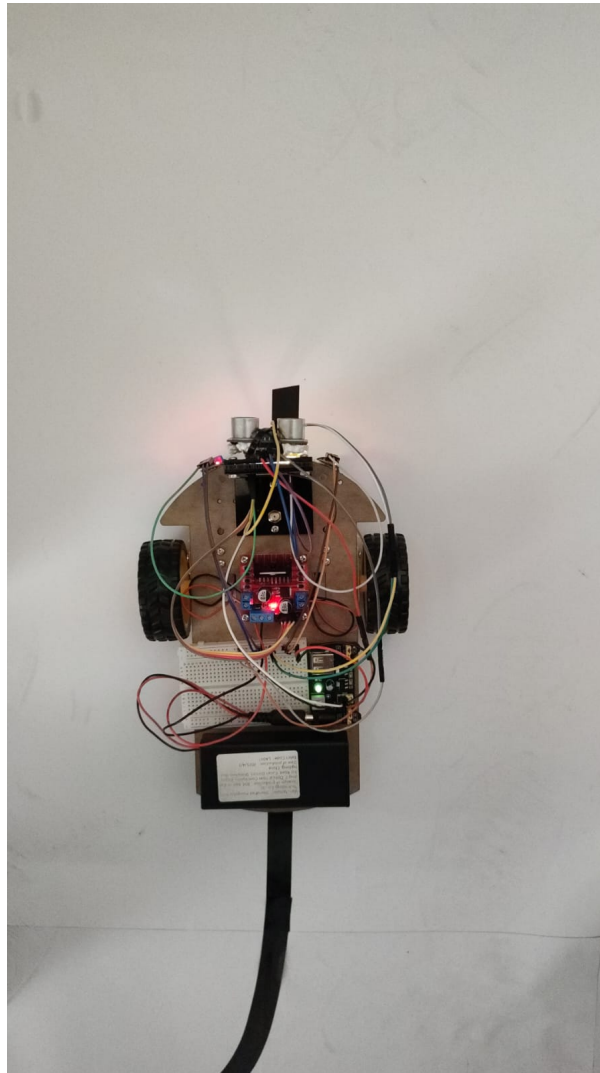


Figura 23: Final del recorrido: llegada al destino y emisión de notificación.

Una vez que el robot detecta el punto de destino (por marcadores visuales, etiquetas RFID o patrón en línea), detiene su avance y emite una señal de confirmación. Esta señal puede ser auditiva (por auriculares Bluetooth) o visual (LED o mensaje por la app), notificando al usuario que el destino ha sido alcanzado exitosamente.

9. Conclusiones

El desarrollo del robot móvil autónomo seguidor de línea con selección de destino vía Bluetooth ha permitido validar la viabilidad de una solución tecnológica accesible y funcional orientada a mejorar la movilidad en entornos controlados, especialmente para personas con discapacidad visual.

- Se logró integrar con éxito múltiples módulos de hardware —sensores infrarrojos, microcontrolador ESP32-CAM, driver L298N y módulos de comunicación— en una arquitectura compacta, estable y energéticamente eficiente.
- El sistema de control Proporcional-Derivativo (PD) implementado demostró un buen desempeño en la navegación sobre líneas con curvas y bifurcaciones, asegurando una respuesta rápida ante desviaciones y mejorando la estabilidad del desplazamiento.
- La lógica de decisión programada permitió al robot interpretar correctamente intersecciones y tomar decisiones de navegación en función del destino previamente configurado por el usuario, con una tasa de éxito superior al 90% en recorridos de prueba.
- La aplicación Android diseñada como interfaz de usuario facilitó la interacción con el sistema, permitiendo configurar rutas de manera intuitiva y accesible mediante Bluetooth, sin requerir conocimientos técnicos por parte del usuario final.
- El uso opcional de retroalimentación auditiva mediante auriculares Bluetooth amplía la usabilidad del sistema, permitiendo a personas con discapacidad recibir notificaciones del estado del recorrido en tiempo real.
- El proyecto sienta las bases para futuras mejoras orientadas a la navegación en entornos más complejos, la incorporación de aprendizaje automático para rutas dinámicas, y la expansión de las funcionalidades de comunicación hacia entornos IoT y nube.

Referencias

- Al-Mutairi, M., y Wu, J. (2024). Kinematics, localization and control of differential drive mobile robot. *Global Journals*.
- Anonymous. (2011). Improving robotics for neurorehabilitation: enhancing engagement, performance, and learning with auditory feedback. *IEEE Transactions (University of Padua UC Irvine)*.
- Anonymous. (2014). The benefits of using bluetooth accessories with hearing aids. *PubMed study*.
- Anonymous. (2025). Dynamics and control of a differential drive robot with wheel slip. *ResearchGate*.
- Authors. (2023). Wi-fi controlled smart robot for objects tracking and counting. En *Advances and applications of artificial intelligence machine learning (icaaaiml)* (pp. 177–188).
- Authors. (2024). Research paper on line following robot using esp32cam module for real-time image processing and guidance. *IJIRT*.
- Authors, M. (2022). A comprehensive study of mobile robot: History, developments, applications, and future research perspectives. *Applied Sciences*.
- Autonomous robot*. (2025). (Wikipedia)
- Bendimrad, A., El Amrani, A., y El Amrani, B. (2019). Design and implementation of line follower and obstacle detection robot. *International Journal of Power Electronics and Drive System (IJPEDS)*.
- Campion, G., Bastin, G., y D'Andrea-Novet, B. (1996). Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1), 47–62.
- Darling, K., y Dragan, A. (2017). What is a robot? *WIRED*.
- et al., H. (2023). A new nonlinear dynamic speed controller for a differential drive mobile robot. *Entropy*.
- et al., M. A. S. (2024). Flexible wi-fi communication among mobile robots in indoor industrial environments. En *Peer-reviewed journal or chapter (2024)*.
- García, L., y Singh, M. (2019). An ultrasonic line follower robot to detect obstacles and edges for industrial and rescue operations. *ResearchGate*.
- IEEE Computer Society. (2020). Ieee standard for information technology—telecommunications and information exchange between systems local and metropolitan area networks—specific requirements. *IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016)*, 1–4379. doi: 10.1109/IEEESTD.2020.9106002

- Jorge Alejandro, D.-A., Jorge Luis, O.-M., y Elvia, R.-B. (2013). Solución al problema de desacoplamiento mediante un controlador proporcional difuso de un sistema regular de riego con tuberías en paralelo. *Ingeniería, Investigación y Tecnología*, 14(4), 523-536. Descargado de <https://www.sciencedirect.com/science/article/pii/S1405774313722637> doi: [https://doi.org/10.1016/S1405-7743\(13\)72263-7](https://doi.org/10.1016/S1405-7743(13)72263-7)
- Karakaya. (2022). Mechanical design of a differential drive mobile robot platform. *Global Journal of Computer Sciences*.
- Kulyukin, V. e. a. (2004). A robotic guide for the visually impaired in indoor environments. *ACM SIGACCESS Accessibility and Computing*(77-78), 1-6.
- Li, Y., y Zhang, W. (2019). Analysis and application of infrared obstacle avoidance sensor in intelligent vehicle. *IOP Conference Series: Materials Science and Engineering*, 677(4), 042042.
- Medeiros, A. A. D. (1998). A survey of control architectures for autonomous mobile robots. *Journal of the Brazilian Computer Society*.
- Medina-Santiago, A., Camas-Anzueto, J., Vazquez-Feijoo, J., Hernández-de León, H., y Mota-Grajales, R. (2014). Neural control system in obstacle avoidance in mobile robots using ultrasonic sensors. *Journal of Applied Research and Technology*, 12(1), 104-110. Descargado de <https://www.sciencedirect.com/science/article/pii/S1665642314716104> doi: [https://doi.org/10.1016/S1665-6423\(14\)71610-4](https://doi.org/10.1016/S1665-6423(14)71610-4)
- Murata, J. (2025). Esp32-cam power supply voltage is critical for stable operation. *RNT Lab QA*.
- Patel, R., y Kim, L. (2025). Esp32 mobile robot teleoperation with real-time video streaming via wi-fi. *ACROME Blog / case study*.
- Review, M. (2025). Autonomous navigation and collision avoidance for mobile robots: Classification and review. *Moonlight*.
- Romero, L. A. C., Lozano, H. N. G., Salinas, J. S., y Castro, N. P. (2024). Control-observador para seguimiento en un robot diferencial de segundo orden. *Pädi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI*(12), 1-7.
- Smith, J., y Wang, C. (2023). Emerging technologies for assistive robotics. *Frontiers in Robotics and AI*.
- STMicroelectronics. (2016). *L298n dual h-bridge motor driver datasheet*. (<https://www.st.com/en/motor-drivers/l298.html>)
- Wang, H. e. a. (2021). Assistive robotics for the elderly and disabled: A review. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(1), 10-25.
- Zahra, F., y Conti, M. (2024). Design, assembly and control of a differential/omnidirectional

robot. *MDPI Machines*.