

Pase de asistencia vía Bluetooth

Sistemas de comunicaciones

Presenta:

Ayala Hernández María Fernanda

Jiménez Ayala Yordi Josué

Portilla Hermenegildo Elizabeth

Profesor:

Dr. Mario Alfredo Ibarra Carrillo

Grupo:

05

Semestre:

2026-1

Entrega: 24 de noviembre del 2025

1. Objetivo

Diseñar e implementar una solución tecnológica que automatice el proceso de captura y registro manual de asistencia en tiempo real, utilizando comunicación Bluetooth SPP (Serial Port Profile) y un microcontrolador Arduino Nano, complementado con almacenamiento persistente mediante una tarjeta SD. El sistema permitirá sustituir el registro manual tradicional por un método más eficiente, preciso y confiable.

2. Limitación del estudio

Nuestro interés principal en el uso de la tecnología es resolver el problema de la ineficiencia y la susceptibilidad a errores en el proceso manual del pase de lista y alcanzar un conocimiento básico de las técnicas de modulación y cómo la tecnología puede ser utilizada para resolver el problema. Para ello, implementamos una solución de adquisición de datos mediante Bluetooth SPP (Serial Port Profile), ya que permite la transferencia de datos de identificación del usuario de forma inalámbrica y directa a una interfaz UART/Serial en el microcontrolador (Arduino nano), eliminando la transcripción manual de la información y logrando un registro automático y preciso del tiempo mediante el módulo RTC (DS3231).

3. Material

3.1. Hardware Principal y Control

- Arduino Nano
- Módulo Bluetooth HC-06
- Módulo RTC DS3231 (Reloj de Tiempo Real)
- Módulo Lector MicroSD Mini
- Micro SD

3.2. Componentes Electrónicos

- LEDs (Verde, Rojo y Azul)
- Resistencias de $1,1\text{ k}\Omega$ y $240\ \Omega$
- Cables Jumper
- Cables de conexión negro y rojo (0.5m es suficiente)
- Protoboard

3.3. Equipamiento y Software

- Laptop (Para reporte y programación de hardware, debe tener el interprete de python instalado)
- Smartphone Android
- App Asistencia (Para conexión del teléfono con el módulo, desarrollada por nosotros)
- Overleaf (Herramienta para el reporte)
- Arduino IDE (Entorno de programación de hardware)
- VS Code (Para programación)

4. Planteamiento del problema

4.1. Contexto general

El registro de asistencia es una actividad esencial en ambientes académicos, pues permite llevar control sobre la participación y presencia de los estudiantes en clase. Sin embargo, los métodos tradicionales (como el pase de lista verbal o el uso de hojas en papel) presentan múltiples desventajas que afectan tanto la eficiencia del proceso como el aprovechamiento del tiempo destinado a la sesión de clases.

En un entorno donde la tecnología avanza rápidamente y la automatización es parte fundamental de las actividades diarias, resulta necesario replantear la forma en que se realiza el control de asistencia para volverlo más preciso, rápido y confiable.

4.2. Problemas del método tradicional (pase de lista manual)

El método más común consiste en que el profesor anuncia el nombre de cada alumno y marca la asistencia en papel o en una lista impresa. Este procedimiento presenta las siguientes limitaciones:

4.2.1. Consumo excesivo de tiempo

- El pase de lista puede tomar varios minutos, especialmente en grupos numerosos.
- Estos minutos reducen el tiempo efectivo de clase y afectan el ritmo de la sesión.
- El profesor debe detener la dinámica del grupo para completar este proceso cada día.

4.2.2. Poca practicidad y alto margen de error

- Es posible que se omita un nombre o se registre mal una asistencia.
- Si el profesor olvida revisar adecuadamente la lista, los errores no se detectan hasta después.

4.3. Problemas de otras alternativas

Existen diversos métodos más “rápidos” que se han intentado implementar, pero de igual manera presentan ciertas fallas considerables.

4.3.1. Hoja de asistencia que circula entre los alumnos

Un alternativa común es pasar una hoja donde cada estudiante escribe su nombre completo. Sin embargo, este método presenta desventajas claras:

- Un alumno puede escribir el nombre de compañeros que no estén presentes.

- El profesor debe transcribir manualmente esta información al sistema oficial de asistencia.
- Con varias sesiones, se generan múltiples hojas que deben archivarse, capturarse o verificar.
- Alumnos pueden escribir con letra poco legible o datos incompletos, lo cual provoca errores al registrar la información.

4.3.2. Pagina web para el registro de asistencia

En el curso de Sistemas de Comunicaciones de la Facultad de Ingeniería se utiliza un sistema donde cada alumno ingresa a una pagina web donde al escribir su número de cuenta (identificador de cada alumno), materia y salón registra su propia asistencia.

Aunque esta solución es más viable, rápida, eficiente y tiene ciertas mejoras a alternativas antes mencionadas, aún presenta ciertos problemas:

- No garantiza que el alumno esté físicamente en el salón, ya que, un estudiante puede registrar su asistencia desde otro lugar sin haber ingresado realmente al aula.
- No existe un mecanismo de verificación presencial, el sistema depende completamente de la honestidad del usuario.
- Puede prestarse a suplantación de identidad, ya que cualquier persona con el número de cuenta podría registrar la asistencia.

4.4. Necesidad de una solución automática y verificable

Debido a estas limitaciones, se hace evidente la necesidad de un sistema que:

- No consuma tiempo de clase.
- Evite la intervención manual del profesor y del alumno.
- Imposibilite que alguien registre asistencia sin estar físicamente presente.
- Genere un registro digital ordenado, seguro y verificable.

4.5. Registro de asistencia por Bluetooth

Para resolver la problemática mencionada, se propone implementar un sistema basado en Arduino que utilice Bluetooth para realizar el registro de la asistencia mediante el celular de cada alumno.

4.5.1. ¿Cómo ayuda esta propuesta?

- Cada alumno porta un dispositivo Bluetooth (comúnmente teléfono celular).
- El sistema detecta la asistencia del alumno a través de la vinculación del teléfono con un modulo Bluetooth almacenando el número de cuenta, fecha y hora exacta de ingreso del alumno.

4.5.2. Ventajas de esta solución

- Reduce posibilidades de fraude, ya que solamente se registra a quien realmente entra en el salón.
- Ahorra tiempo a profesores y alumnos.
- Genera un almacenamiento de datos confiable, útil para consultas posteriores.
- Automatiza el proceso, integrándolo a un sistema más moderno y eficiente.

5. Requisitos funcionales clave del hardware

5.1. Rango

Por el contexto de nuestro problema, el rango de comunicación requerido es corto, pues todo debe ocurrir en el salón de clases.

No existe una única medida para un salón de clases en México, ya que depende de la capacidad y el nivel educativo. Sin embargo, las normas establecen que un salón de entre 1 y 15 alumnos requiere un mínimo de $24\ m^2$, mientras que uno para entre 16 y 30 alumnos necesita $48\ m^2$, y para 31 a 40 alumnos se requieren $64\ m^2$ [DOF, s.f.].

Dado que nuestro problema a resolver implica que el estudiante debe estar físicamente presente e ingresar al área inmediata del aula no se requiere de un rango mayor a los 10 metros.

5.2. Velocidad de datos

La velocidad de datos requerida es baja y no intensiva pues solo implica el envío de una pequeña cantidad de datos (Número de cuenta) entre el Smartphone y el Arduino Nano. Esta información se transmite en un solo "paquete" por estudiante y el proceso se repite solo cuando un nuevo alumno registra su asistencia. Las velocidades típicas del Bluetooth 2.0 (que usa el HC-06) son de cientos de kilobits por segundo (kbps), en este caso, el modulo tiene una velocidad de transferencia de datos asíncrona máxima de 2 Mbps o una velocidad constante de 160 kbps. En cuanto a la velocidad de transferencia sincrónica, ofrece 1 Mbps tanto de subida como de bajada [OKYSTAR, 2024]. Esto es extremadamente superior a lo necesario para enviar una cadena de texto de unos pocos bytes.

5.3. Latencia crítica

La respuesta debe ser **baja** o casi **en tiempo real**. La experiencia del usuario es clave: el sistema debe registrar y confirmar la asistencia rápidamente para evitar congestiones en la entrada del aula (esto en caso de que el profesor desee hacerlo de esa manera). Una latencia aceptable debe ser de menos de 1 segundo (idealmente 200-500 ms) para que el estudiante perciba la interacción como instantánea.

5.4. Topología

El proyecto cuenta con una topología del tipo **estrella** es una de las configuraciones de red más comunes. En esta configuración, cada nodo se conecta a un dispositivo de red central, como un concentrador o un commutador. [Satoshi, 2020]

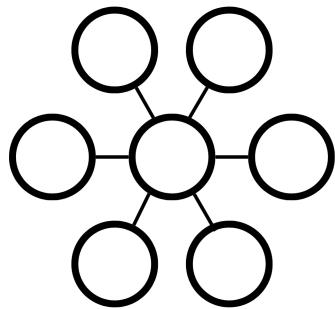


Figura 1: Topología estrella

Para este caso, la red está compuesta por un dispositivo central (que consta del Arduino Nano que se apoya del HC-06 para la función Bluetooth) que actúa como el punto de registro. De ahí, múltiples dispositivos periféricos (Smartphones) se conectan a este nodo central.

6. Hardware

6.1. Bluetooth

Bluetooth es una tecnología inalámbrica que utiliza radiofrecuencia para transferir y sincronizar datos entre dispositivos a corta distancia, como una computadora portátil y auriculares inalámbricos. A medida que evoluciona la tecnología Bluetooth, las versiones más nuevas de Bluetooth pueden transferir datos entre dispositivos a mayor velocidad y alcance [Canda C.S.E, 2024].

Esta radiofrecuencia puede ser de 1,2 GHz en las primeras conexiones a los 2.4 GHz actuales y con este sistema se logra la comunicación enviando señales débiles de menos de 1 [mW]. Así podemos transmitir tanto audio como imagen y datos.

La distancia máxima entre dos dispositivos Bluetooth está limitada a una cierta cantidad de metros para evitar interrupciones o interferencias en la conexión.

Esta distancia efectiva, o cobertura, depende directamente de la clase a la que pertenece el dispositivo Bluetooth. Cada clase indica una potencia de transmisión específica y, por lo tanto, determina el alcance que puede soportar la conexión.

	Alcance	Consumo medio	Potencia	dBm
Bluetooth Clase 1	100 metros	100 mW	Alta	20
Bluetooth Clase 2	20 metros	2,5 mW	Media/Alta	4
Bluetooth Clase 3	1 metro	1 mW	Media	0
Bluetooth Clase 4	0,5 metros	0,5 mW	Baja	-3

6.1.1. Perfiles Bluetooth

Los perfiles Bluetooth son protocolos adicionales que se basan en el estándar Bluetooth básico para definir más claramente qué tipo de datos transmite un módulo Bluetooth. Mientras que las especificaciones de Bluetooth definen cómo funciona la tecnología obras, los perfiles definen cómo es usado.

6.2. Bluetooth SSP (Serial Port Profile)

Bluetooth SSP es un perfil Bluetooth que emula una conexión de cable de puerto serie tradicional, permitiendo a dos dispositivos comunicarse mediante el envío y recepción de datos como si tuvieran líneas de transmisión (TX) y recepción (RX) conectadas. Este perfil es fundamental y se utiliza en una variedad de dispositivos como receptores GPS, lectores de códigos de barras y sensores, y es común en dispositivos que necesitan transferir datos de manera constante.

- **Frecuencia.** 2.4 GHz
- **Alcance Típico.** Corto (~ 10 metros)
- **Consumo de Energía.** Medio
- **Tasa de Datos.** ~ 2,1 Mbps (con EDR)
- **Casos de Uso Apropriados.** Conexión punto a punto serial, registro de asistencia secuencial.
- **Hardware.** Módulo HC-06 (o HC-05 en modo Esclavo)
- **Tasa de transferencia.** Media/Alta

Para el proyecto se optó por usar este perfil de Bluetooth con ayuda de un módulo HC-06.

6.3. Otras opciones de comunicación inalámbrica

6.3.1. Wi-Fi

- **Frecuencia.** 2.4 GHz
- **Alcance Típico.** Medio (hasta 100m)
- **Consumo de Energía.** Alto
- **Tasa de Datos.** Alto
- **Casos de Uso Apropriados.** Transferencia de archivos, Streaming de datos a la nube
- **Hardware.** Módulo ESP8266, ESP32
- **Tasa de transferencia.** Alta

6.3.2. Bluetooth (BLE)

- **Frecuencia.** 2.4 GHz
- **Alcance Típico.** Corto (hasta 30m)
- **Consumo de Energía.** Bajo
- **Tasa de Datos.** Bajo/Medio
- **Casos de Uso Apropriados.** Conexión y control de dispositivos cercanos.
- **Hardware.** Módulo HM-10, nRF52
- **Tasa de transferencia.** Baja/Media

6.3.3. LoRa (con módulo)

- **Frecuencia.** Sub-GHz
- **Alcance Típico.** Largo (km)
- **Consumo de Energía.** Muy Bajo
- **Tasa de Datos.** Muy Bajo
- **Casos de Uso Apropriados.** Monitoreo a distancia, datos esporádicos.
- **Hardware.** Módulo LoRa RA-01, RA-02, Semtech SX1276
- **Tasa de transferencia.** Muy baja

6.3.4. Otros (Zigbee/NRF)

- **Frecuencia.** Varios
- **Alcance Típico.** Medio/Corto
- **Consumo de Energía.** Bajo
- **Tasa de Datos.** Medio
- **Casos de Uso Apropriados.** Redes de sensores en malla (requiere hardware adicional).
- **Hardware.** Módulo NRF24L01, XBee, chipsets CC2530
- **Tasa de transferencia.** Media

6.4. Justificación

6.4.1. Cumplimiento de Requisitos de Alcance y Topología

El módulo HC-06 (que entra dentro de la clase 2) cumple perfectamente con el requisito de ser un dispositivo de corto alcance (~ 10 metros) para una operación en un aula o zona específica. El alcance superior del Wi-Fi y la cobertura extrema del LoRa son innecesarios y representan un consumo de energía excesivo y un sobrecoste en hardware que no se justifica.

6.4.2. Rendimiento (Tasa de Transferencia)

La tasa de transferencia del Bluetooth SPP ($\sim 2,1$ Mbps) se clasifica como Media/Alta. Esta velocidad ofrece una latencia extremadamente baja para el tipo de datos que maneja el proyecto (número de cuenta del alumno).

En base al análisis de otras opciones de comunicación tomamos en cuenta que

- **Wi-Fi** tiene una velocidad excesiva que es redundante, ya que el proyecto no requiere transferencia de video o archivos grandes.
- **LoRa** y otros protocolos de bajo consumo tienen tasas de transferencia muy bajas, lo que podría introducir latencia y ralentizar el registro secuencial de alumnos.

6.4.3. Simplicidad, Costo y Compatibilidad

Utilizar el módulo HC-06 (Bluetooth SPP) es la más eficiente por

- **Costo:** El módulo HC-06 es una de las soluciones inalámbricas más económicas disponibles para microcontroladores [Costo comparativo de módulos HC-06 vs ESP8266].
- **Simplicidad:** Su implementación requiere únicamente el uso de la comunicación serial (UART) de bajo nivel, lo que simplifica drásticamente el código y no exige la complejidad de una pila TCP/IP (Wi-Fi) o el manejo de paquetes GATT (BLE).

- **Topología (SPP):** Al operar en modo Esclavo y usar SPP, el HC-06 está perfectamente alineado con la necesidad de que el Smartphone (Maestro) inicie una conexión simple y secuencial para registrar la asistencia.

Con fundamento en estos 3 puntos, el Bluetooth SPP ofrece el mejor equilibrio entre rendimiento, bajo costo y simplicidad de integración, cumpliendo de forma óptima los requisitos funcionales del sistema de registro de asistencia sin introducir complejidad o consumo innecesario.

7. Requisitos funcionales clave del software

7.1. Arduino IDE

El Integrated Development Environment, conocido en español como Entorno de Desarrollo Integrado, es el software que se utiliza para programar la placa de Arduino.

Contiene un editor de texto para escribir código, un área de mensajes, una consola de texto, una barra de herramientas con botones para funciones comunes y una serie de menús. Se conecta al hardware de Arduino para cargar programas y comunicarse con ellos [Arduino, 2024].

7.1.1. Utilidad en el proyecto

Usar el IDE de Arduino es una necesidad fundamental para la elaboración del proyecto establecida por la propia arquitectura del sistema y la función del módulo Bluetooth.

El módulo HC-06 es un transceptor pasivo. Por sí solo, solo puede transmitir y recibir datos seriales (SPP). No tiene la capacidad de procesamiento para ejecutar la lógica del sistema (comparar el número de cuenta, consultar el RTC para obtener la hora ni gestionar el Módulo MicroSD para el almacenamiento de registros). Por lo tanto, requiere ser controlado por una unidad central.

De ahí que el Arduino Nano actúa como el cerebro que debe tomar decisiones, gestionar el RTC, y manejar la comunicación con la SD. Sin embargo, para que el Nano realice estas tareas, necesita ser inyectado con un firmware (el código de control).

Aquí es donde entra el Arduino IDE pues es el entorno que nos va a permitir escribir el código para el Arduino y de ahí que se necesite para el proyecto.

7.2. App Asistencia (Desarrollada por nosotros)

App Asistencia es una aplicación de consola / terminal con una interfaz serial / UART conectada con un convertidor bluetooth al dispositivo Android. Existen alternativas en la Play Store sin embargo queríamos una aplicación dedicada para solo pasar asistencia.

7.2.1. Utilidad en el proyecto

La elección de la aplicación es obligatoria porque proporciona el componente Maestro esencial que el hardware pasivo requiere para iniciar la transacción de asistencia.

El módulo HC-06 está configurado para operar como Esclavo. Esto significa que solo puede esperar y recibir conexiones; no tiene la capacidad de buscar, iniciar o establecer el enlace Bluetooth por sí mismo. Por lo tanto, requiere que un dispositivo Maestro externo inicie la comunicación.

El Smartphone Android debe asumir el rol de Maestro e iniciar la conexión a través del perfil SPP. Esta acción no ocurre de forma automática dentro del sistema operativo, sino que debe ser invocada por un software cliente específico.

Aquí es donde entra la aplicación pues esta es el software que sirve como el puente fundamental para la interacción y la inyección de datos. Tiene 2 funciones fundamentales para el proyecto:

1. **Activa la conexión SPP:** Convierte la capacidad del teléfono en una acción de conexión.
2. **Proporciona la interfaz:** Permite al usuario (o a un tester) introducir manualmente el número de cuenta del alumno y enviar esa cadena de datos de forma controlada al Arduino.

La decisión de realizar una aplicación cliente personalizada se basa en la necesidad de garantizar la fiabilidad del protocolo y optimizar la experiencia del usuario (UX) para el único propósito del sistema: el registro de asistencia.

7.3. VS Code

Visual Studio Code (VS Code) es un editor de código fuente gratuito, potente y ligero que funciona en Windows, macOS y Linux. Permite a los desarrolladores escribir y editar código para una amplia variedad de lenguajes de programación, como JavaScript, TypeScript, Python, C++ y más, y es personalizable mediante extensiones. Sus características principales incluyen resaltado de sintaxis, depuración integrada, control de versiones con Git y un ecosistema de extensiones para añadir funcionalidades.

7.3.1. Utilidad en el proyecto

La arquitectura del proyecto se divide entre el control en tiempo real (el Arduino Nano) y el análisis de datos (offline), lo que hace que el script de Python (lenguaje seleccionado por comodidad) sea indispensable.

El Arduino Nano está limitado a **2 KB** de memoria RAM. Esta limitación impone la necesidad del script de Python, ya que el Nano no puede:

- Cargar la lista maestra de alumnos y el registro del día al mismo tiempo.
- Ejecutar la lógica compleja de cruce de datos, comparación y ordenamiento necesaria para determinar quién faltó.

El Nano solo se encarga de la recolección de datos (ID, RTC) y el registro secuencial en la SD. El script de Python realiza la tarea de análisis al calcular las faltas y generar una lista final ordenada, incluyendo las faltas del día.

Aquí es donde entra VS Code pues simplifica el flujo de trabajo al ofrecer una terminal integrada y soporte para la sintaxis de Python, permitiendo que el script se ejecute y depure de manera rápida en el mismo lugar donde se escribe el código.

7.3.2. Alternativas a VS Code

Es crucial entender que VS Code no es un requisito funcional. El script de Python se ejecuta en cualquier terminal o IDE que tenga el intérprete de Python instalado. Algunas alternativas a VS Code son:

1. PyCharm (Community Edition)
2. Sublime Text / Atom
3. Terminal (CMD/Bash con editor Nano/Vim)

8. Software

8.1. Software de última generación

A continuación, y en el marco del Análisis del Estado del Arte del Software, se presenta una comparativa de las plataformas de Internet de las Cosas (IoT) líderes en la nube. Esta tabla permite evaluar las diferentes arquitecturas de servicio disponibles, su enfoque, seguridad e integración de datos.

8.1.1. Núcleo de IoT de AWS

- **Enfoque Central.** Amplitud y Escalabilidad
- **Servicio Principal.** Núcleo de IoT de AWS
- **Gestión de Dispositivos.** Registro, Sombras de dispositivos, Gestión de dispositivos. Ofrece "Device Shadow" para el estado sin conexión.
- **Computación de borde.** AWS Greengrass (Extiende el cómputo de AWS al edge).
- **Integración de Datos.** Rules Engine con integración nativa a Lambda, S3, DynamoDB y el amplio ecosistema de AWS.
- **Seguridad.** Autenticación robusta basada en certificados X.509, políticas IAM.
- **Para ideal.** Proyectos que buscan la máxima cantidad de servicios y necesitan escalar horizontalmente masivamente.

8.1.2. Centro de IoT de Azure

- **Enfoque Central.** Integración Empresarial y Edge
- **Servicio Principal.** Centro de IoT de Azure
- **Gestión de Dispositivos.** Gemelos de dispositivos, Servicio de aprovisionamiento de dispositivos (DPS). Fuerte integración con Microsoft Defender para IoT.
- **Computación de borde.** Azure IoT Edge (Extiende servicios de Azure, como AI, al edge mediante módulos Docker).
- **Integración de Datos.** Enrutamiento de mensajes con integración a Azure Storage, Cosmos DB, Power BI y Machine Learning.
- **Seguridad.** Autenticación de alto nivel, empresa recientemente con Microsoft Defender for IoT para protección avanzada contra amenazas.
- **Para ideal.** Organizaciones que ya utilizan el ecosistema Microsoft (Windows, Office 365, Azure AD).

8.1.3. Plataforma en la nube de Google (GCP) IoT

- **Enfoque Central.** Análisis de Datos y AI/ML
- **Servicio Principal.** Arquitectura Basada en Servicios (Pub/Sub, BigQuery, Edge: Anthos/Edge)
- **Gestión de Dispositivos.** Pub/Sub y servicios adyacentes; fuerte en la configuración remota y descubrimiento (antes con IoT Core).

- **Computación de borde.** Anthos / Google Cloud for the Edge (Centrado en contadores y IA en el edge).
- **Integración de Datos.** Pub/Sub (ingesta) directamente a BigQuery y Dataflow para proceso y análisis masivo.
- **Seguridad.** Seguridad a nivel de la plataforma GCP; el enfoque en Big Data garantiza el manejo de grandes volúmenes de credenciales.
- **Para ideal.** Proyectos donde la inteligencia de datos, el análisis avanzado (Big Data) y el Machine Learning son la prioridad máxima.

8.2. Justificación

Para el pase de asistencia Bluetooth, no se requiere de ninguna de las opciones anteriores pues es una solución local, de bajo consumo y de baja escala. Por lo tanto, no utiliza ninguna de las plataformas de IoT en la nube (AWS IoT, Azure IoT, GCP IoT) porque estas soluciones representan un nivel de complejidad, costo y escala que excede ampliamente los requisitos funcionales del sistema.

8.2.1. Costo y Escalabilidad (Overkill)

El objetivo de las plataformas en la nube (como las listadas anteriormente) es gestionar millones de dispositivos y permitir el escalamiento horizontal masivo, cobrando por el uso de servicios, mensajes y cómputo.

Para nuestro proyecto, esto resulta innecesario pues el sistema es un dispositivo único (el Arduino Nano) con un registro de datos secuencial y limitado. No requiere la amplitud y escalabilidad del Núcleo de IoT de AWS ni la arquitectura de servicios de GCP IoT.

La solución implementada utiliza el Módulo Lector MicroSD para el almacenamiento local. Esto resulta en un costo operacional casi nulo después de la inversión inicial en hardware. En contraste, integrar la nube requeriría costos recurrentes de mensajes (Pub/Sub), almacenamiento (Azure Storage o DynamoDB) y potencialmente el uso de funciones de cómputo (AWS Lambda), volviendo el proyecto innecesariamente caro.

8.2.2. Accesibilidad y Complejidad de Implementación

La integración con plataformas de nube requiere recursos de cómputo y conectividad que el hardware actual no posee de forma nativa.

Los módulos HC-06 (Bluetooth Clásico) no están diseñados para comunicarse mediante los protocolos de red complejos que exige la nube (como MQTT o HTTP sobre una pila TC-P/IP). Requeriría un módulo Wi-Fi (como el ESP32 y como justificamos en la sección de

hardware no es la opción de comunicación que usamos), lo que aumentaría la complejidad del hardware y la programación. En cuanto a simplicidad, la solución que proponemos se mantiene en la capa de comunicación más sencilla (SPP Serial), lo que facilita enormemente la programación con el Arduino IDE.

8.2.3. Seguridad

La infraestructura de seguridad de la nube está diseñada para defenderse de ataques a gran escala a través de Internet, lo cual no es el vector de riesgo principal de este proyecto.

- **Enfoque de la Nube:** Las soluciones como Microsoft Defender for IoT o la seguridad a nivel de la plataforma GCP están destinadas a proteger grandes volúmenes de datos y conexiones contra amenazas externas en la red.
- **Enfoque del Proyecto:** La principal preocupación de seguridad del proyecto es la integridad de los datos locales y la seguridad física del dispositivo (evitar que la tarjeta MicroSD sea manipulada).

A partir de lo anterior, integrar una capa de seguridad criptográfica y de autenticación de nivel empresarial sería una complejidad de desarrollo innecesaria, ya que no ofrece una ventaja práctica sobre la seguridad de los datos almacenados localmente y físicamente protegidos.

9. Arquitectura del sistema

9.1. Diagrama de bloques

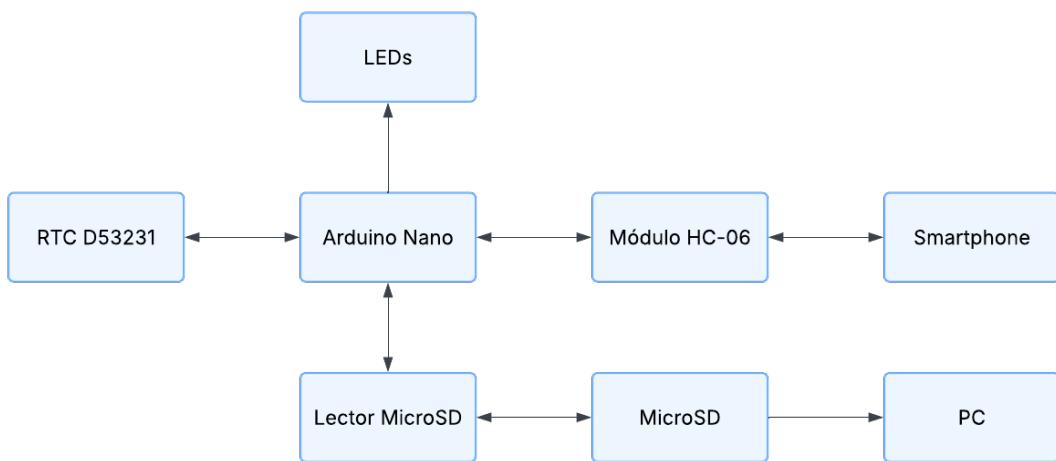


Figura 2: Diagrama de bloques

9.1.1. Conexiones Unidireccionales

- **Micro SD → PC:** Es unidireccional porque el Script de Python solo tiene un propósito: leer, procesar y reportar los datos CSV grabados. Es un proceso de post-análisis que no envía información de vuelta al sistema.
- **Arduino Nano → LEDs:** Es unidireccional porque los LEDs son actuadores que solo reciben una señal de control (HIGH/LOW) para indicar el estado del sistema.

9.1.2. Conexiones Bidireccionales

- **Arduino Nano ⇄ HC-06:** Se conectan de forma bidireccional porque el HC-06 debe recibir el número de cuenta del alumno y el Nano debe enviar comandos de control y la confirmación de registro de vuelta al smartphone.
- **Arduino Nano ⇄ Lector MicroSD:** Es bidireccional porque el sistema debe tanto leer la lista de alumnos (Entrada) como escribir el registro de asistencia del día (Salida). El protocolo SPI permite esta alta velocidad de transferencia de datos en ambos sentidos.
- **Arduino Nano ⇄ RTC DS3231:** Es bidireccional para que el Nano pueda leer la hora constantemente y, ocasionalmente, escribir y ajustar la hora del reloj.
- **Smartphone ⇄ HC-06:** Es bidireccional porque el smartphone envía el número de cuenta del alumno (Entrada) y necesita recibir la confirmación de registro de vuelta (Salida).

9.1.3. Arduino Nano

Este bloque es el centro de procesamiento y control del sistema. Funciona gracias al Arduino Nano (microcontrolador ATmega328P), que sirve como la unidad central de procesamiento encargada de ejecutar el firmware, orquestar todos los protocolos de comunicación y decidir la lógica de validación de la asistencia.

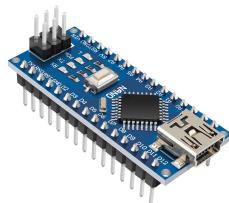


Figura 3: Arduino Nano

9.1.4. HC-06

Este bloque realiza la función de comunicación inalámbrica. Funciona gracias al módulo HC – 06 que sirve para establecer el enlace SPP (Serial Port Profile), permitiendo al sistema

recibir el número de cuenta del alumno y enviar mensajes de confirmación de vuelta al smartphone.



Figura 4: Módulo HC-06

9.1.5. RTC DS3231

Este bloque proporciona la fuente de tiempo precisa y persistente. Funciona gracias al Módulo RTC DS3231 (Reloj de Tiempo Real) que sirve para mantener un registro exacto de la fecha y hora (mediante su propia batería de respaldo), asegurando que cada entrada en el registro de asistencia sea inalterable. Se comunica por I²C.

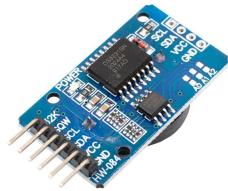


Figura 5: RTC DS3231

9.1.6. Lector Micro SD

Este bloque realiza la gestión de la interfaz de almacenamiento. Funciona gracias al Módulo Lector MicroSD mini que sirve como el driver de hardware para adaptar la tarjeta SD al protocolo SPI de alta velocidad del Nano, permitiendo tanto la lectura de la lista maestra como la escritura del registro del día.



Figura 6: Mini Módulo Lector Micro SD

9.1.7. Micro SD

Este bloque es el medio de almacenamiento de datos primario. Funciona gracias a la Tarjeta MicroSD que sirve para contener físicamente la Lista de alumnos y los archivos de Registro Diario, actuando como el punto de transferencia física de los datos a la computadora.



Figura 7: Micro SD

9.1.8. Smartphone

Este bloque actúa como el dispositivo maestro de entrada. Funciona gracias al Smartphone Android (y su capacidad de cliente Bluetooth) que sirve para ejecutar la aplicación de terminal y enviar el número de cuenta del alumno, iniciando la transacción de asistencia.

9.1.9. PC

Este bloque realiza el análisis de datos offline y la generación de reportes. Funciona gracias a una computadora con el intérprete de Python que sirve para tomar los datos brutos de la SD y procesarlos para calcular las faltas y generar un reporte final ordenado a como venia en la lisa de alumnos original.

9.1.10. LEDs

Este bloque realiza la función de retroalimentación de estado y alerta al usuario. Funciona gracias a los LEDs (Rojo, Verde, Azul) que sirven como actuadores controlados digitalmente para mostrar el estado operativo del sistema (Azul: disponible; Verde: registro exitoso; Rojo:cualquier error que se presente durante la ejecución del sistema).



Figura 8: LED

9.2. Diagrama de flujo

El sistema completo se basa en dos entornos principales. El primero implica todos los módulos que están conectados al nano, que es la parte del sistema que opera en tiempo real y

contiene dos subprocessos esenciales conectados. El segundo entorno es el Sistema de Post-Procesamiento (El Script de Python), que es un proceso externo y separado (Diagrama C). Este flujo es necesario porque el Nano no tiene la RAM para hacer análisis complejos.

9.2.1. Diagrama de Inicialización (A)

El Diagrama A (Flujo de Inicialización) tiene un propósito de verificación y estabilidad; se ejecuta al encender el sistema para asegurar que el hardware vital (como el reloj RTC y la tarjeta SD) esté presente y que la lista maestra de alumnos se cargue correctamente. Su finalización exitosa es una condición obligatoria para iniciar el siguiente flujo.

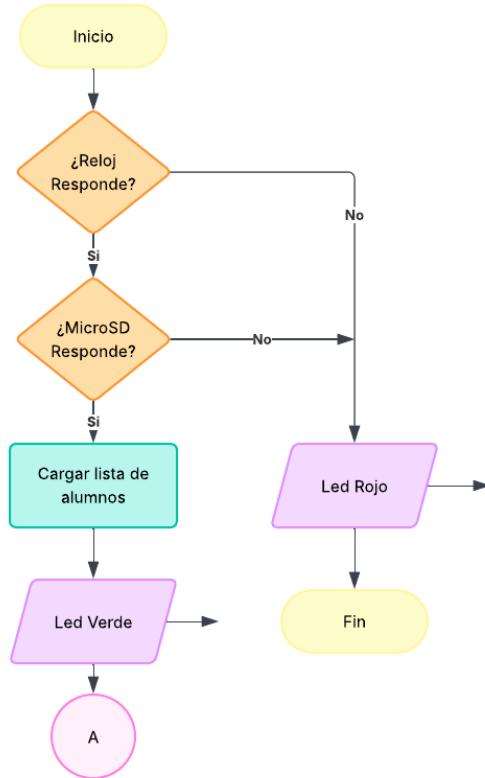


Figura 9: Diagrama A

9.2.2. Diagrama de Ciclo de Asistencia (B)

El Diagrama B (Ciclo de Asistencia) representa la lógica operativa continua del diagrama A para el registro en tiempo real. Su función es gestionar la interacción continua e infinita del sistema: escuchar la conexión BT, clasificar el mensaje entrante (STATUS o ID), ejecutar la doble validación, y finalmente, loguear la asistencia con la hora exacta en la SD. Este ciclo no puede detenerse, por lo que su foco es el procesamiento rápido y secuencial.

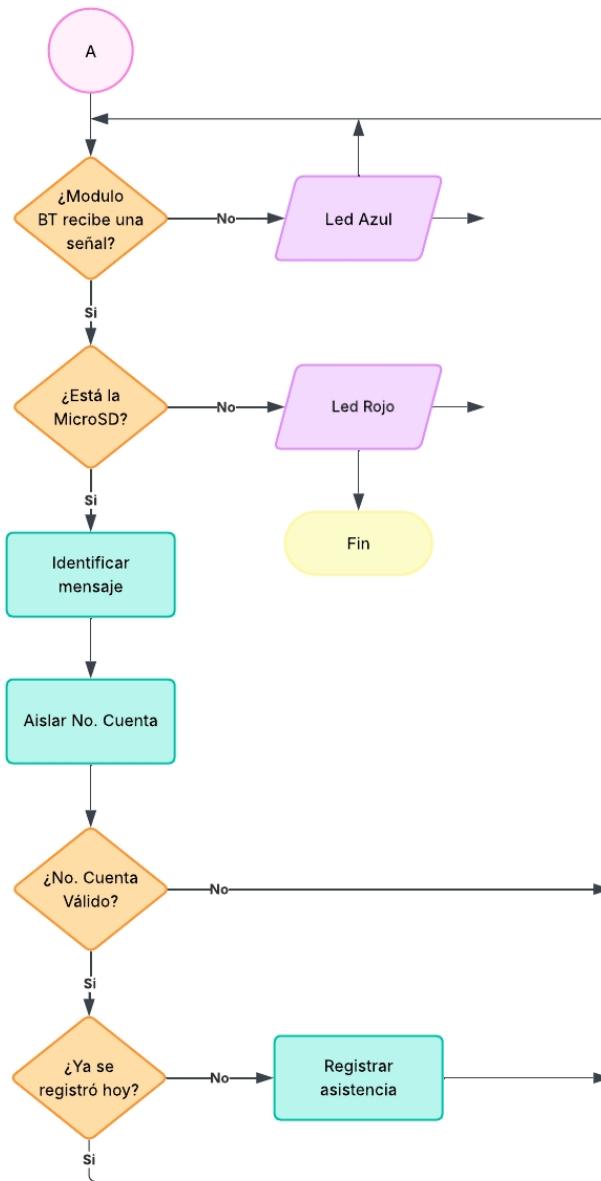


Figura 10: Diagrama B

9.2.3. Diagrama de Reporte Final (C)

Su propósito es completar la tarea que el Arduino no puede manejar debido a su limitación de RAM (**2 KB**). Este flujo toma los datos brutos de la SD, realiza la comparación masiva entre los presentes y la Lista Maestra, y genera el reporte final ordenado, calculando explícitamente las Faltas.

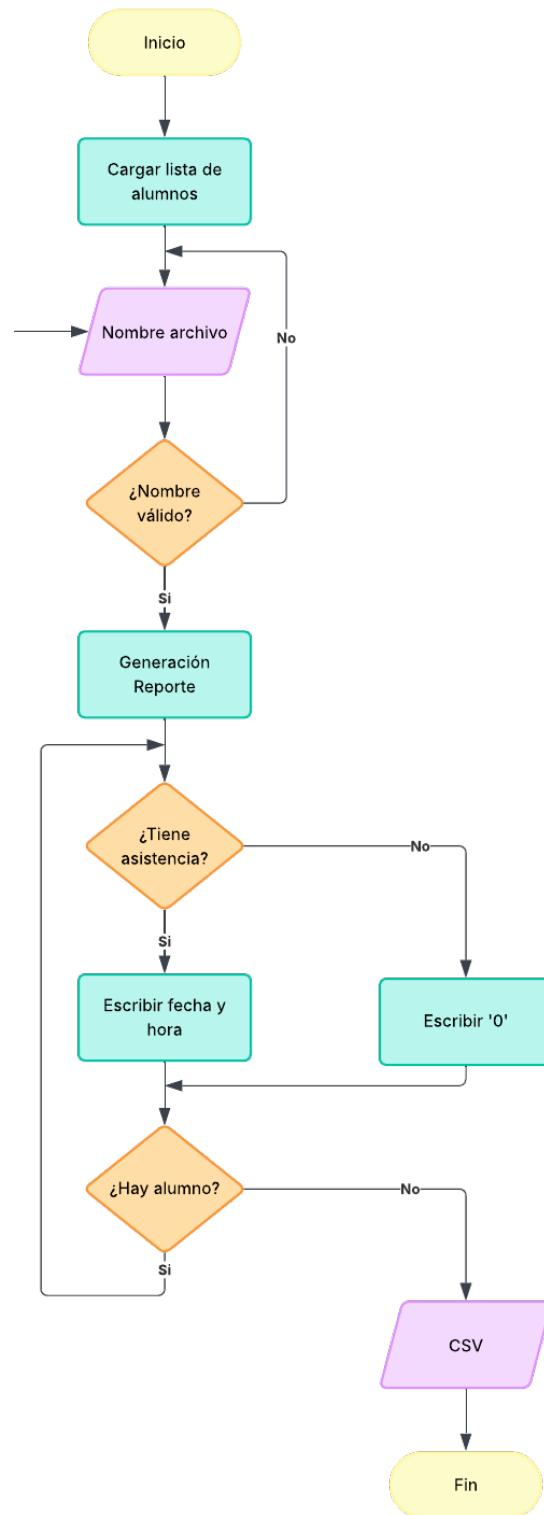


Figura 11: Diagrama C

10. Pruebas y validación

10.1. Metricas y criterios de éxito

1. **Latencia de Registro:** El tiempo máximo desde que el smartphone envía el número de cuenta hasta que se enciende el LED Verde (confirmación de registro) debe ser inferior a **2** segundos.
2. **Integridad de Datos:** **100 %** de las entradas exitosas deben escribirse correctamente en el archivo CSV del día, incluyendo el número de cuenta, la Fecha y la Hora del RTC.
3. **Fiabilidad de la Validación:** El sistema debe tener una tasa de **0 %** de registros de números de cuenta inválidos o duplicados en el archivo CSV.
4. **Disponibilidad del Sistema:** El firmware debe poder correr de forma continua durante al menos **2** horas sin fallos críticos de SD o RTC que es el promedio que dura una clase, sin embargo puede funcionar más tiempo.

10.2. Monitor Serie

El Monitor Serie (o Serial Monitor) es una herramienta fundamental integrada en el Arduino IDE que actúa como un canal de comunicación bidireccional entre la computadora y el microcontrolador (el Arduino Nano).

10.3. Pruebas Unitarias

10.3.1. HC-06

Se verifica la comunicación UART (puente serial) asegurando que el módulo envíe el echo de datos al Monitor Serie.

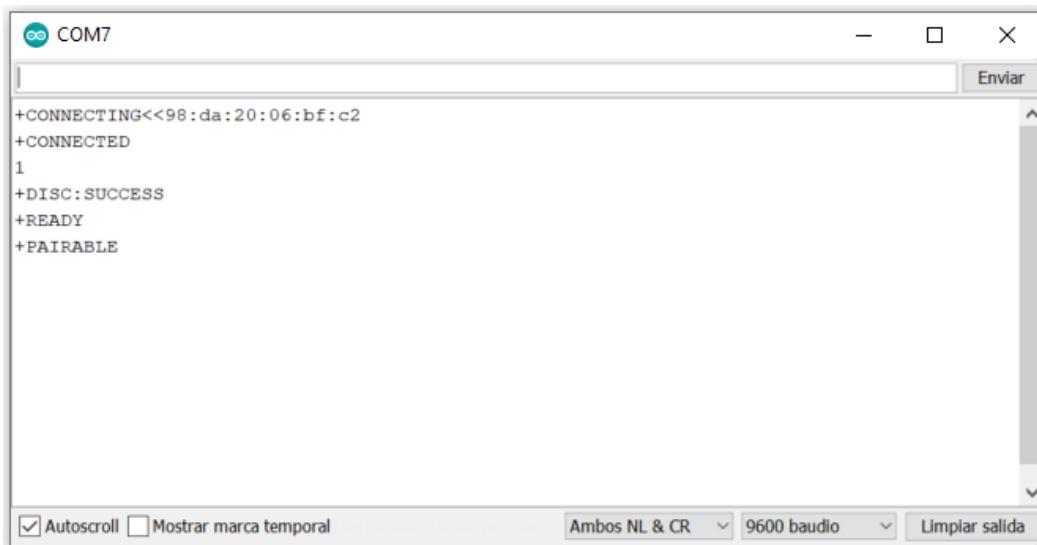


Figura 12: Monitor serie

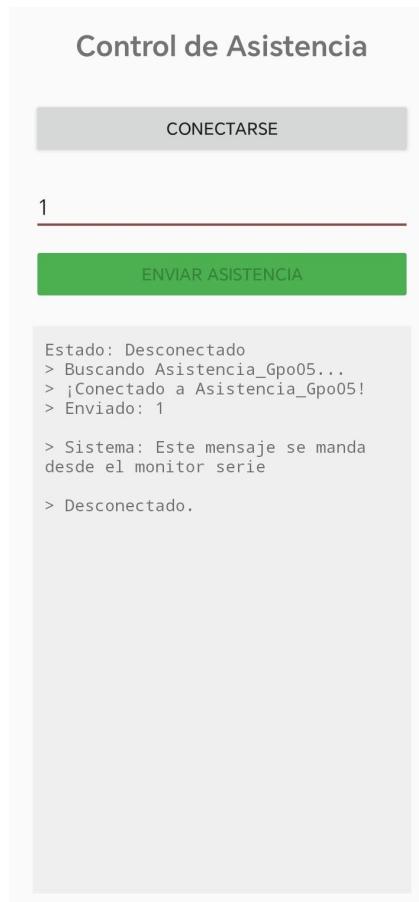


Figura 13: Pantalla smartphone

10.3.2. RTC DS3231

Se verifica la conexión I²C asegurando que la hora y fecha se lean y actualicen correctamente en el Monitor Serie. (La prueba se realizó el 19 de Noviembre a las 15:12 hrs.)

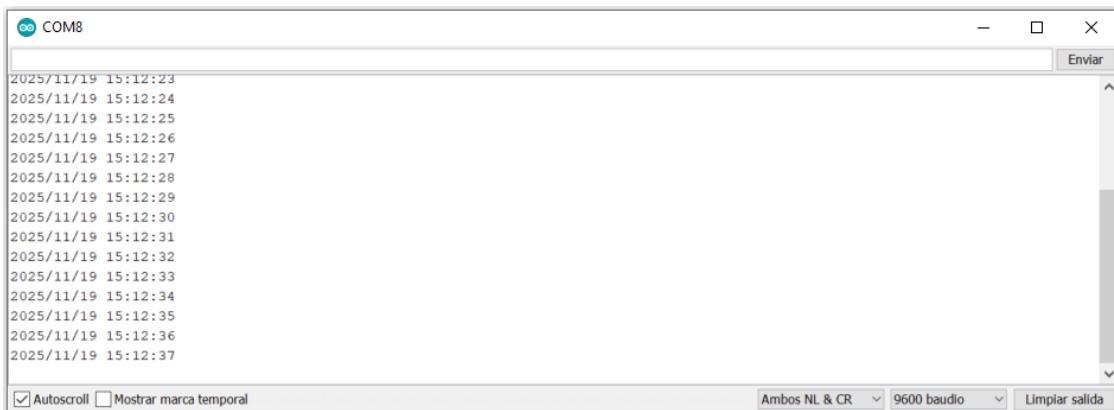


Figura 14: Monitor serie

10.3.3. Lector Micro SD

Se verifica la inicialización SPI y se confirma la capacidad de crear y escribir una línea en un archivo de prueba.

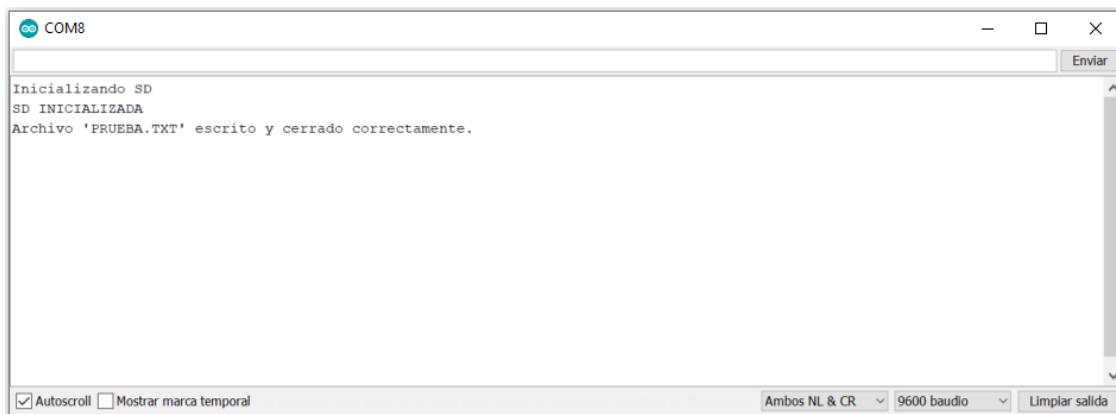


Figura 15: Monitor serie

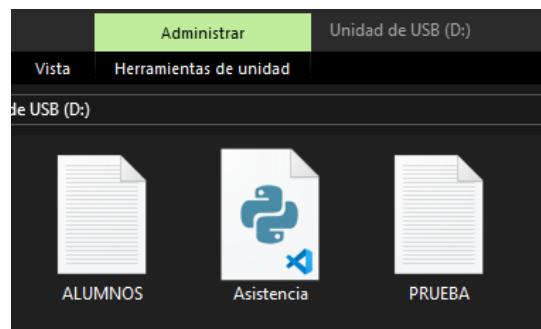


Figura 16: Archivo PRUEBA.TXT en la Micro SD

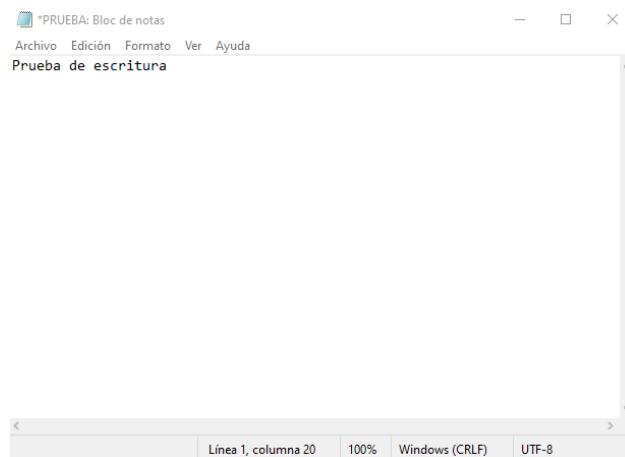


Figura 17: Contenido del archivo

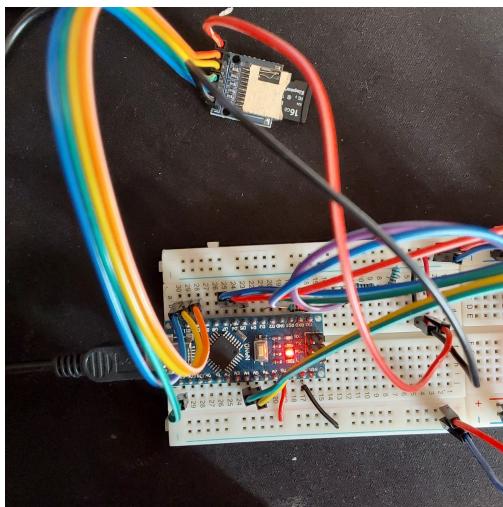


Figura 18: Micro SD conectada

10.3.4. LEDs

LEDs: Se verifica el control de I/O digital asegurando que cada LED se encienda y apague según la secuencia de prueba.

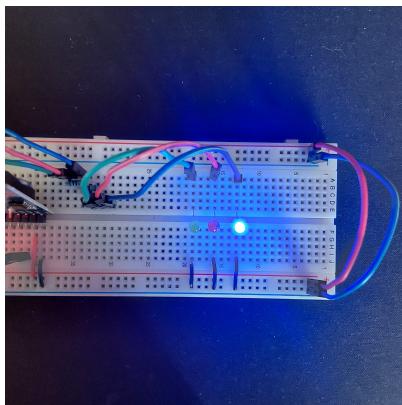


Figura 19: Led azul

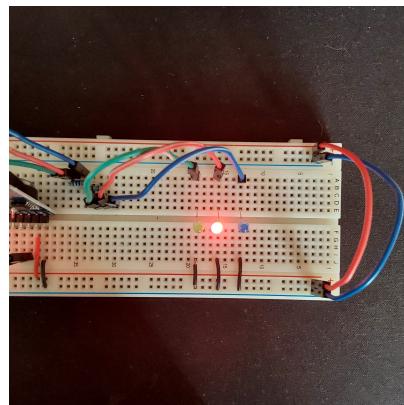


Figura 20: Led rojo

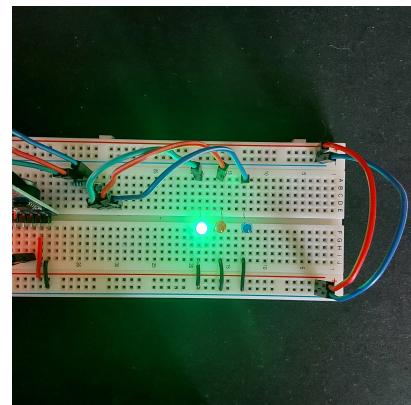


Figura 21: Led verde

10.4. Pruebas de Integración

10.4.1. Nano + Micro SD + Validación + LEDs

Se verifica que se carguen los números de cuenta del archivo ALUMNOS.TXT correctamente a la RAM sin errores de memoria.

Para corroborar esto se muestra un mensaje en el monitor serie así como también el led verde enciende en caso de que se haya cargado correctamente todo esto.



Figura 22: Monitor serie

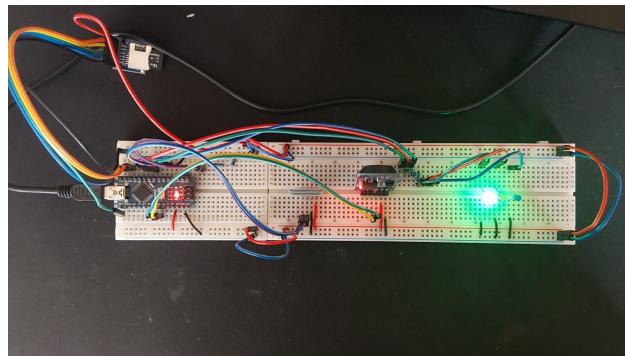


Figura 23: Led confirmando lectura exitosa

10.4.2. Nano + HC – 06 + LEDs

Se verifica que el LED Azul se apague cuando un dispositivo se conecte y reanude el parpadeo al detectar cuando se desconecte.

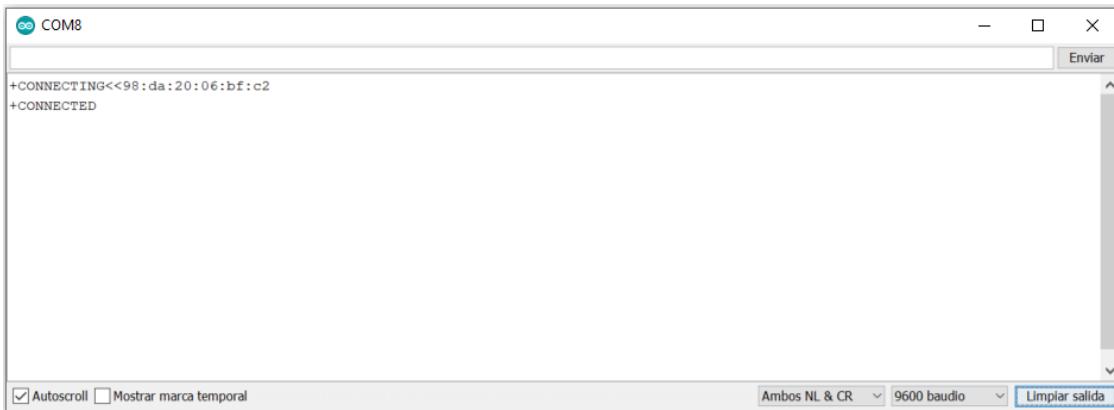


Figura 24: Monitor serie cuando se conecta un dispositivo

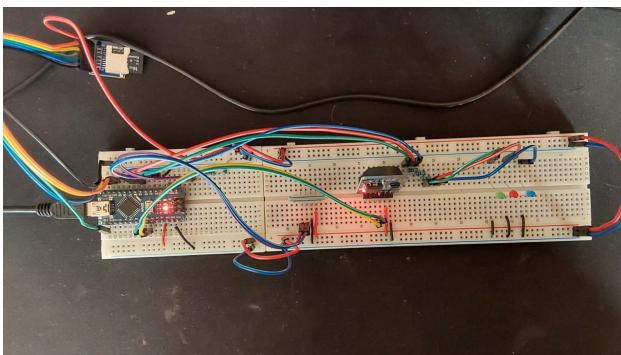


Figura 25: Led azul apagado cuando se conecta un dispositivo

```
+CONNECTING<<98:da:20:06:bf:c2
+CONNECTED

+DISC:SUCCESS
+READY
+PAIRABLE
```

The screenshot shows a serial monitor window titled "COM8". The text area displays the following log entries:
+CONNECTING<<98:da:20:06:bf:c2
+CONNECTED

+DISC:SUCCESS
+READY
+PAIRABLE

Figura 26: Monitor serie cuando se desconecta un dispositivo

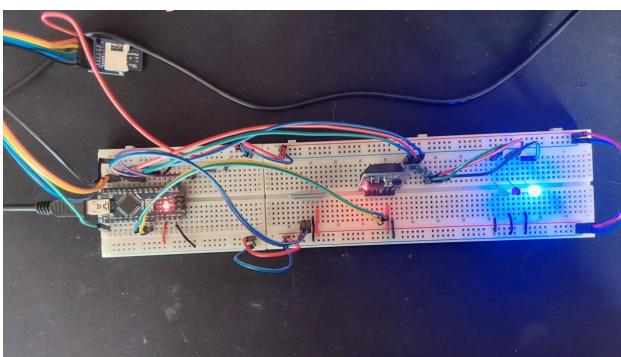


Figura 27: Led azul parpadeando cuando se desconecta un dispositivo

10.4.3. Nano + RTC + Micro SD (Registro)

Se verifica que se pueda leer la hora del RTC y escribir la línea completa en el archivo sin fallos de escritura. (La prueba se realizó el 19 de Noviembre a las 16:38 hrs. usando el número de cuenta 409123870 de prueba)

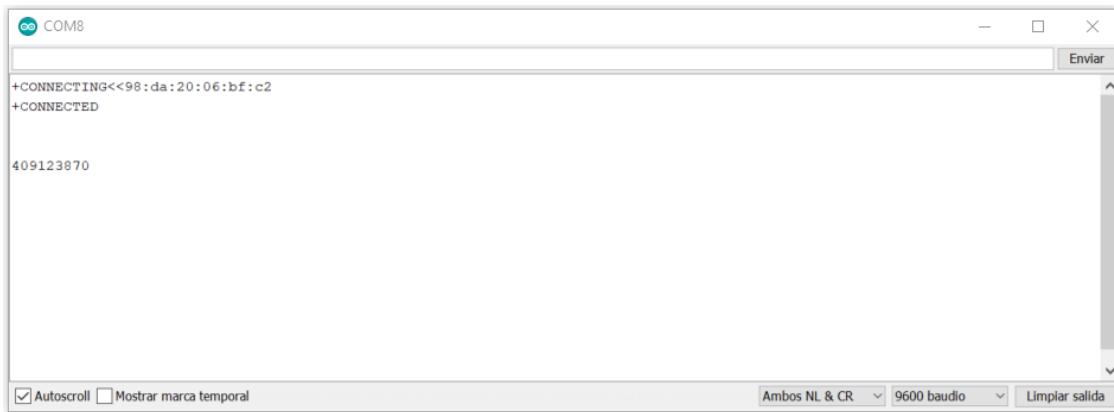


Figura 28: Monitor serie

	A	B	C	D
1	409123870	11/19/2025	16:38:05	
2				

Figura 29: Contenido del archivo

10.5. Pruebas de Sistema

10.5.1. Pase de lista completa (Éxito)

Para la prueba se tomo como escenario un grupo de 10 personas los cuales se encuentran en un archivo dentro de la Micro SD.

ALUMNOS: Bloc de notas	
Archivo	Edición
320068234	
320239126	
403719856	
320987654	
320257599	
317852103	
409123870	
329876541	
320278107	
117002029	

Figura 30: Números de cuenta

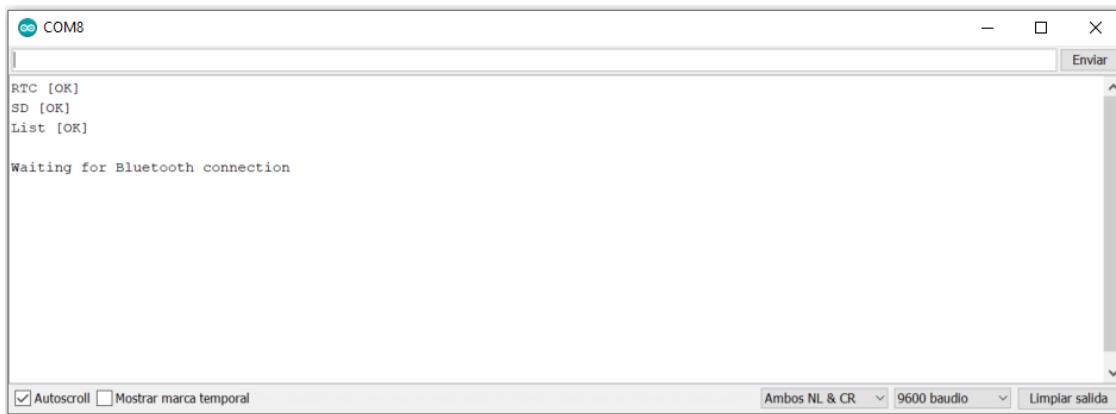


Figura 31: Monitor serie, validación de los módulos

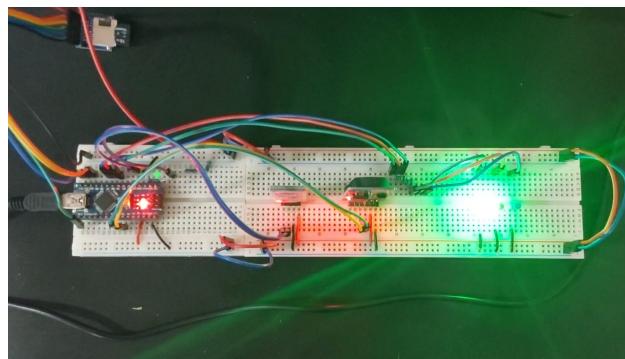


Figura 32: Led que indica que no hubo errores al iniciar el sistema

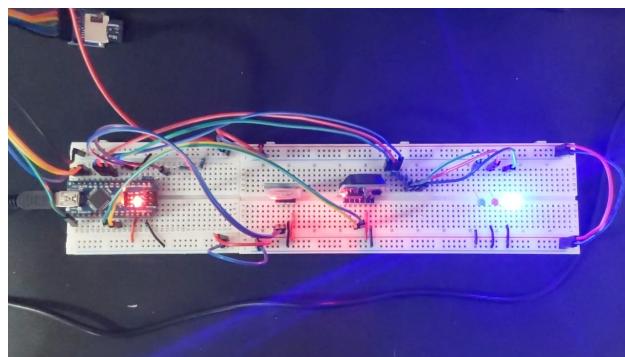


Figura 33: Led indicando que el módulo está disponible para emparejar

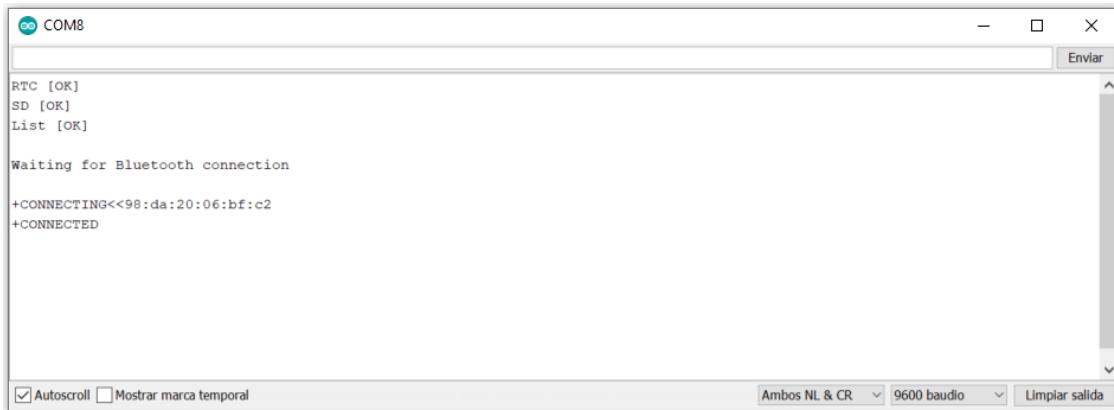


Figura 34: Monitor serie, conexión de dispositivo

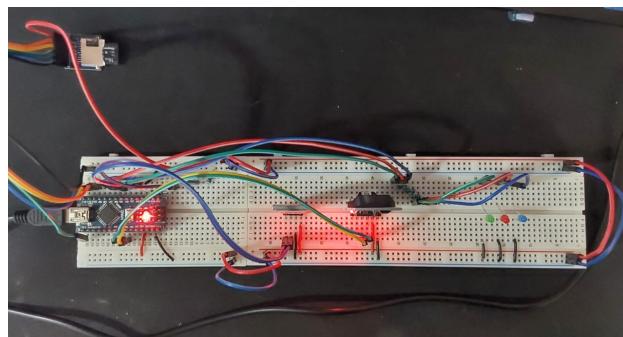


Figura 35: Led azul apagado porque ya hay un dispositivo conectado

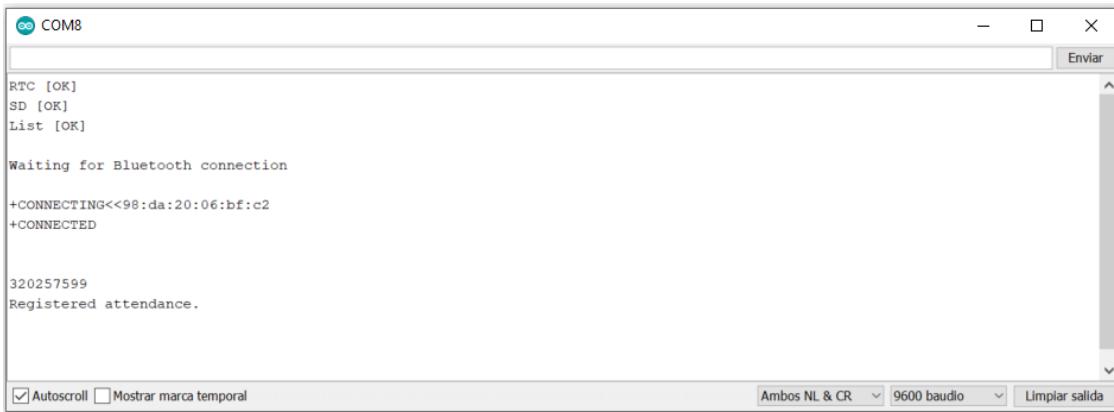


Figura 36: Monitor serie, registro de asistencia

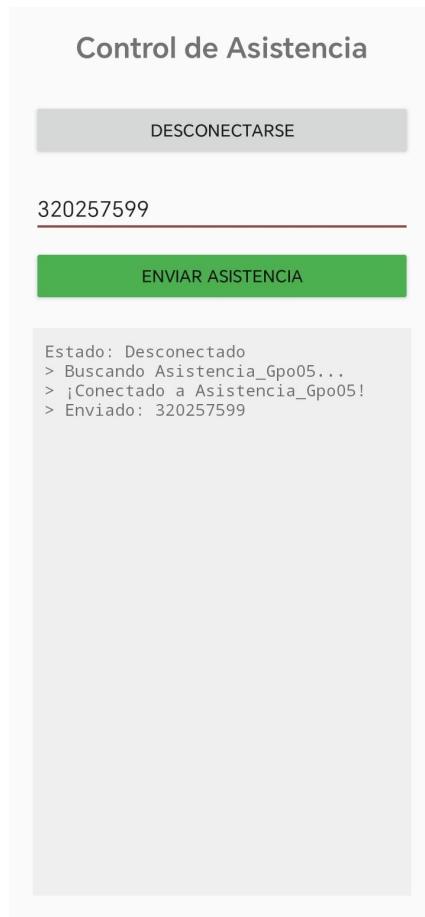


Figura 37: Enviando número de cuenta desde el smartphone

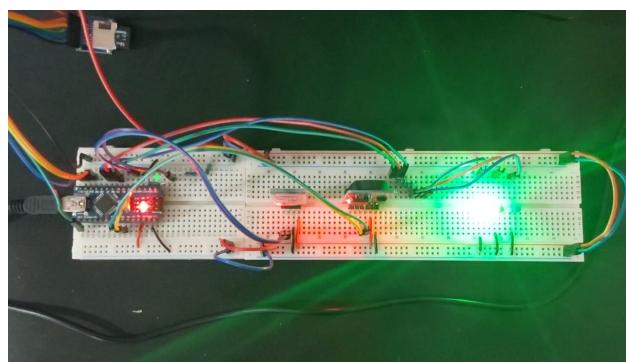


Figura 38: Led que indica el registro de asistencia

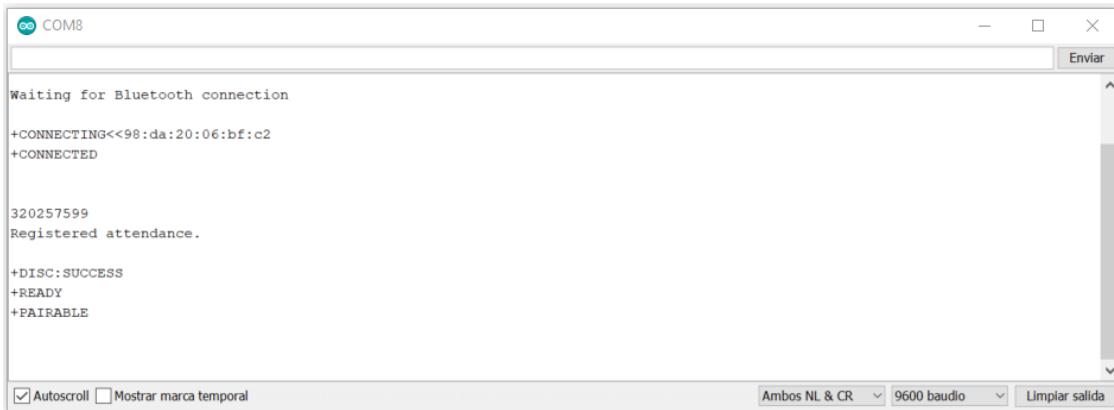


Figura 39: Monitor serie, desconexión de dispositivo

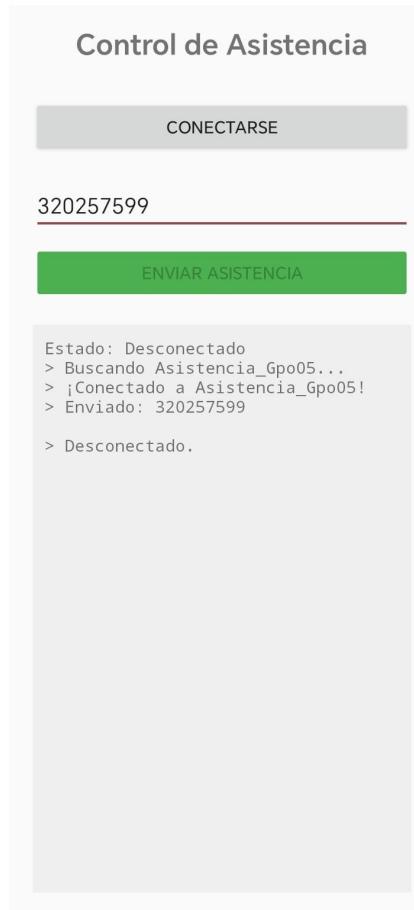


Figura 40: Desconexión del dispositivo desde el smartphone

Repetimos este proceso con los estudiantes que asistieron (conectar dispositivo, enviar número de cuenta, validar registro y desconectar dispositivo), una vez que esto se termina se retira la Micro SD y se inserta en la computadora.

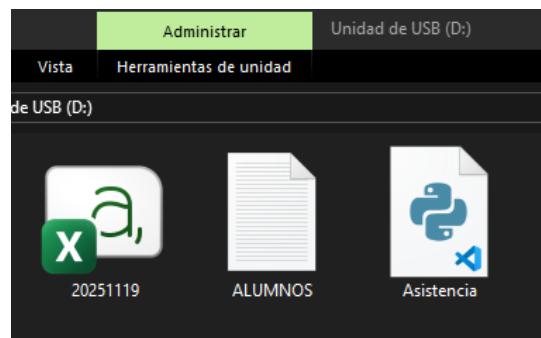


Figura 41: Archivo de asistencia que se genero en la micro SD

	A	B	C
1	320257599	11/19/2025	17:23:29
2	320068234	11/19/2025	17:29:05
3	117002029	11/19/2025	17:29:44
4	329876541	11/19/2025	17:30:03
5	320278107	11/19/2025	17:30:23
6	320239126	11/19/2025	17:30:38
7	403719856	11/19/2025	17:31:39
8			

Figura 42: Contenido del archivo

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\> & C:/Users/Yordi/AppData/Local/Microsoft/WindowsApps/python3.13.exe d:/Asistencia.py
Date of the attendance log (YYYYMMDD): 20251119
```

Figura 43: Ejecución del script de python

	A	B	C
1	No. Cuenta	Asistencia	Hora
2	320068234	1	17:29:05
3	320239126	1	17:30:38
4	403719856	1	17:31:39
5	32987654	0	0
6	320257599	1	17:23:29
7	317852103	0	0
8	409123870	0	0
9	329876541	1	17:30:03
10	320278107	1	17:30:23
11	117002029	1	17:29:44

Figura 44: Contenido del archivo generado con el script

10.5.2. Fallo de Validación (Número que no está en la lista)

The screenshot shows a terminal window titled "COM8". The text output is as follows:

```
RTC [OK]
SD [OK]
List [OK]

Waiting for Bluetooth connection

+CONNECTING<<98:da:20:06:bf:c2
+CONNECTED

1234567891
Invalid ID or not in the student list.
```

At the bottom of the window, there are checkboxes for "Autoscroll" and "Mostrar marca temporal", and a dropdown for "9600 baudio" with a "Limpiar salida" button.

Figura 45: Monitor serie, mensaje de error

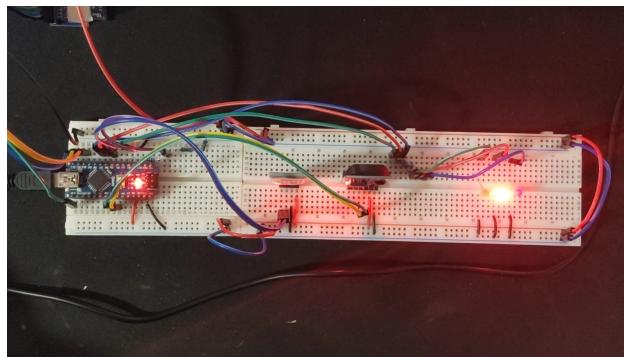


Figura 46: Led rojo que indica entrada invalida

10.5.3. Doble registro de asistencia

The screenshot shows a terminal window titled "COM8". The text output is as follows:

```
RTC [OK]
SD [OK]
List [OK]

Waiting for Bluetooth connection

+CONNECTING<<98:da:20:06:bf:c2
+CONNECTED

320257599
Student already registered today.
```

At the bottom of the window, there are checkboxes for "Autoscroll" and "Mostrar marca temporal", and a dropdown for "9600 baudio" with a "Limpiar salida" button.

Figura 47: Monitor serie, mensaje de error

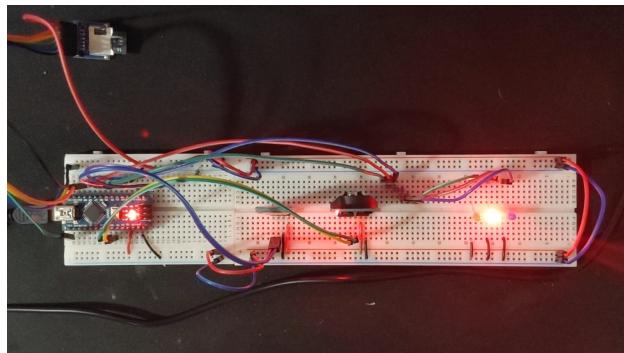
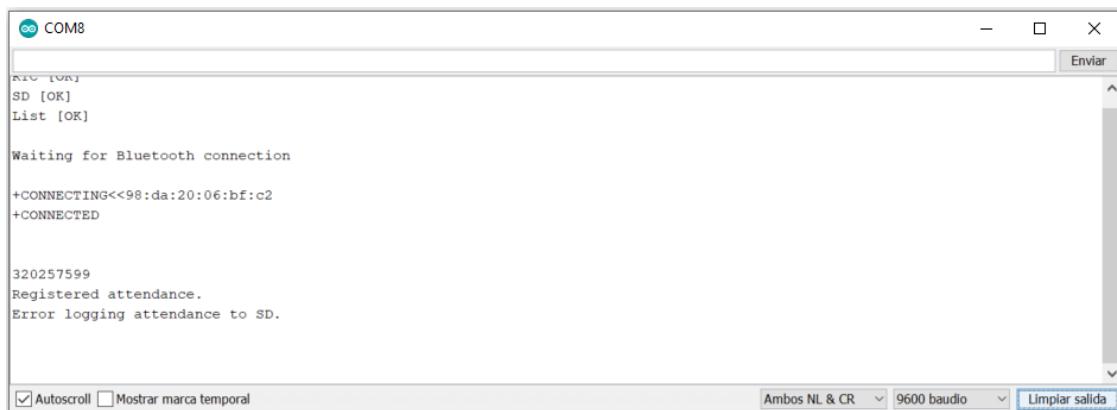


Figura 48: Led rojo que indica que no se registró la asistencia porque ya había un registro previo de ese día

10.5.4. Fallo Asíncrono de Hardware (Retirar la micro SD)



A screenshot of a serial monitor window titled "COM8". The window shows the following text:

```
HiV [OK]
SD [OK]
List [OK]

Waiting for Bluetooth connection

+CONNECTING<<98:da:20:06:bf:c2
+CONNECTED

320257599
Registered attendance.
Error logging attendance to SD.

[Autoscroll] [Mostrar marca temporal] Ambos NL & CR 9600 baudio Limpiar salida
```

Figura 49: Monitor serie, mensaje de error

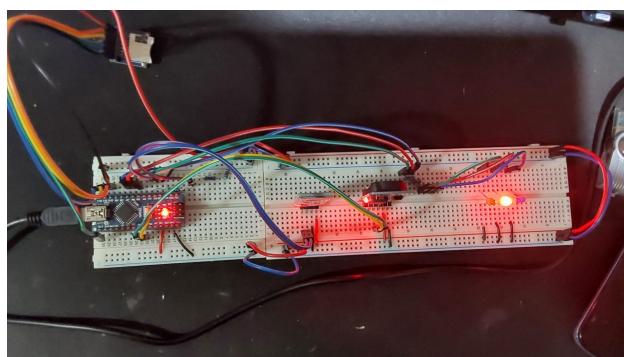


Figura 50: Led rojo que indica que no se registró la asistencia por un fallo en la SD

11. Resultados y análisis

11.1. Resultados cuantitativos

11.1.1. Latencia v.s. Distancia

Se realizaron 10 pruebas completas a diferentes distancias para validar la latencia del sistema. La distancia esta en [m] mientras que la latencia en [ms]

Cuenta	0.9	1.8	2.7	3.6	4.5	5.4	6.3	7.2	8.1	9.0
117002029	40 ms	45 ms	49 ms	56 ms	68 ms	73 ms	81 ms	85 ms	89 ms	100 ms
320068234	37 ms	43 ms	47 ms	56 ms	65 ms	72 ms	81 ms	87 ms	89 ms	95 ms
320239126	39 ms	45 ms	50 ms	55 ms	69 ms	74 ms	82 ms	89 ms	88 ms	95 ms
403719856	38 ms	44 ms	50 ms	55 ms	65 ms	74 ms	82 ms	88 ms	90 ms	98 ms
320257599	38 ms	46 ms	48 ms	58 ms	67 ms	73 ms	82 ms	88 ms	90 ms	98 ms
320278107	30 ms	45 ms	51 ms	57 ms	67 ms	75 ms	85 ms	88 ms	91 ms	99 ms
320987654	37 ms	43 ms	49 ms	54 ms	67 ms	73 ms	83 ms	86 ms	92 ms	95 ms
317852103	42 ms	45 ms	49 ms	55 ms	68 ms	74 ms	80 ms	86 ms	92 ms	94 ms
409123870	39 ms	43 ms	48 ms	55 ms	66 ms	73 ms	79 ms	86 ms	92 ms	94 ms
329876541	39 ms	46 ms	51 ms	56 ms	66 ms	73 ms	81 ms	85 ms	89 ms	95ms
Promedio	37.9	44.5	49.2	55.7	66.8	73.4	81.6	86.8	90.2	96.3

Latencia v.s. Distancia

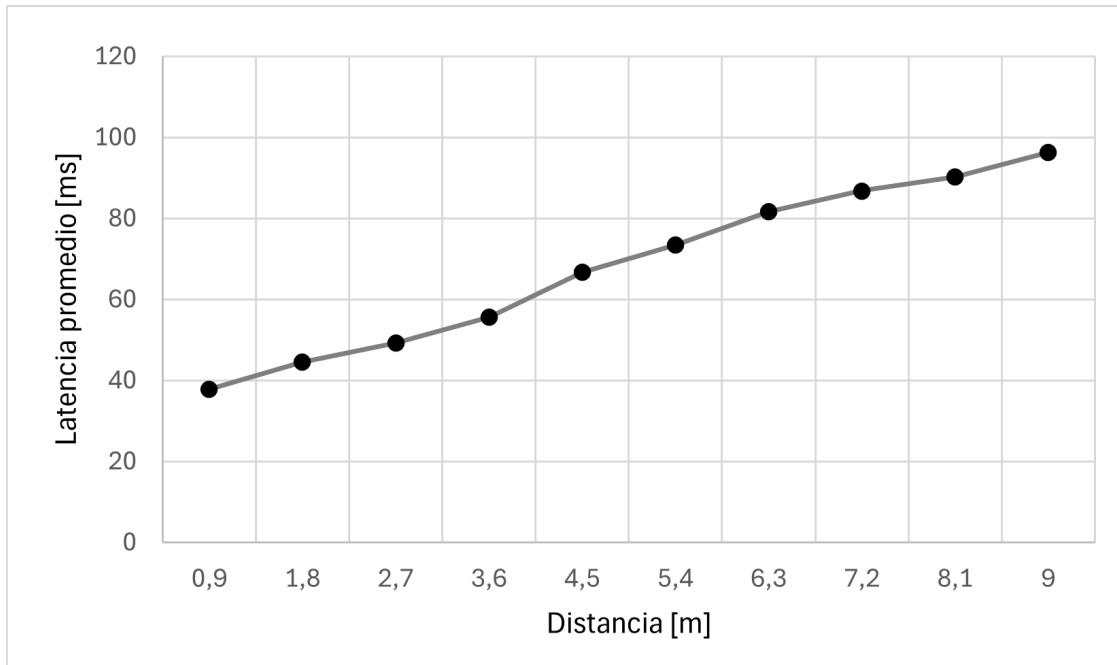


Figura 51: Gráfica Latencia v.s. Distancia

Mientras se media la latencia también se obtuvo la tasa de fiabilidad, en la que se corroboró que el registro exitoso se dio porque no hubo una perdida de datos.

Cuenta	0.9	1.8	2.7	3.6	4.5	5.4	6.3	7.2	8.1	9.0
117002029	1	1	1	1	1	1	1	1	1	1
320068234	1	1	1	1	1	1	1	1	1	1
320239126	1	1	1	1	1	1	1	1	1	1
403719856	1	1	1	1	1	1	1	1	1	1
320257599	1	1	1	1	1	1	1	1	1	1
320278107	1	1	1	1	1	1	1	1	1	1
320987654	1	1	1	1	1	1	1	1	1	1
317852103	1	1	1	1	1	1	1	1	1	1
409123870	1	1	1	1	1	1	1	1	1	1
329876541	1	1	1	1	1	1	1	1	1	1
Exitos	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10	10/10
% Fiabilidad	100 %	100 %	100 %	100 %	100 %	100 %	100 %	100 %	100 %	100 %

Registros exitoso. (1) Éxito (0) Fallo

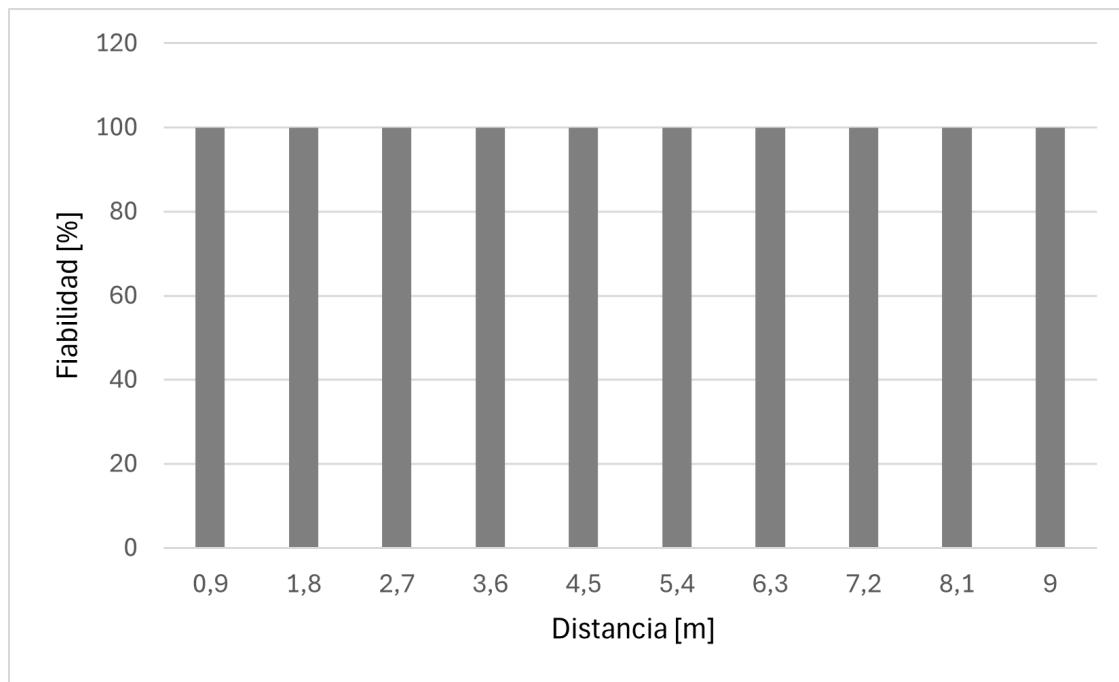


Figura 52: % Fiabilidad a diferentes distancias

11.2. Comparación con los requisitos

11.2.1. Latencia Máxima

El objetivo de la Latencia Máxima era mantener el tiempo de respuesta total (desde el envío del no. cuenta hasta la confirmación de registro) por debajo de **2** segundos (2000 ms). Las pruebas de sistema demostraron que el proyecto **Sí** cumple con este requisito, pues incluso en la distancia mas larga en la que se hizo a prueba (9 [m]) no representó mayor problema al ser en promedio de 96.3 ms, algo imperceptible para nuestros ojos, por lo que a cualquier distancia se obtuvo una respuesta inmediata.

11.2.2. Integridad de Datos

El objetivo de Integridad de Datos era asegurar que el registro CSV fuera auditabile y libre de corrupción. Este requisito se cumplió exitosamente porque el sistema logró confirmar la funcionalidad tanto del RTC como del módulo SD. El RTC DS3231 proporcionó una fuente de fecha y hora inalterable para cada registro, mientras que la SD demostró una fiabilidad del **100 %** en la escritura en modo append (FILE_WRITE), garantizando que no se perdieran datos y que la hora fuera precisa en todas las entradas del archivo CSV del día.

11.2.3. Fiabilidad

El requisito de Fiabilidad (0 % de registros duplicados o inválidos) fue completamente satisfecho gracias a la implementación de la doble capa de validación. La prueba del sistema confirmó que el algoritmo Sí puede prevenir registros incorrectos: primero, al asegurar que la ID entrante existe en la Lista Maestra cargada en RAM, y segundo, al ejecutar el chequeo anti-doble registro que lee el archivo CSV del día y detiene el proceso si detecta que la ID ya fue procesada, manteniendo el log libre de entradas duplicadas.

11.3. Análisis crítico

El rendimiento del sistema de asistencia superó significativamente los criterios de diseño, demostrando la alta eficiencia del protocolo Bluetooth Clásico (HC-06) para esta aplicación. El objetivo de latencia máxima de **2000** ms (2 segundos) fue ampliamente excedido, ya que la latencia promedio más alta registrada fue de solo **96,3** ms a **9** metros, algo que no notaríamos considerando la aplicación que tenemos que es el del pase de lista. Esta estabilidad en el rendimiento valida que el diseño es robusto y que el Nano pudo procesar el no. cuenta sin fallos, ya que la latencia se mantuvo en un rango bajo (menos de **100** ms), cumpliendo con creces el requisito de UX.

11.3.1. Discrepancias y Limitaciones

A pesar del éxito en el rendimiento, nuestro análisis revela una limitación fundamental en la implementación real del proyecto. La elección del Módulo HC – 06 implica que el sistema se basa únicamente en el estándar Bluetooth Clásico (BT Classic), lo cual genera una restricción severa para los usuarios de iPhone, que requieren la tecnología Bluetooth Low Energy (BLE 4.0 o superior) para interactuar con dispositivos de terceros. Esta limitación

de compatibilidad nos obligaría en una implementación real a restringir el acceso a usuarios de Android o requerirles el uso de un dispositivo externo. Esta brecha de usabilidad en el mercado móvil es un aspecto clave a mejorar.

12. Impacto social

1. ¿Cuáles son los aspectos de su proyecto que pueden resultar beneficiosos para las personas?

Los aspectos más beneficiosos del proyecto son la automatización del proceso y la verificación de la presencia física, lo cual se traduce directamente en la eficiencia académica:

- **Ahorro de Tiempo y Optimización del Recurso Educativo.** El sistema elimina el pase de lista verbal y el uso de hojas de papel , reduciendo el tiempo que el profesor debe invertir en la captura. Estos minutos se reintegran al tiempo efectivo de la sesión de clases.
- **Reducción de Fraude y Mayor Veracidad del Registro.** Al requerir que el teléfono del alumno se vincule al módulo Bluetooth en el salón, el sistema imposibilita el registro sin estar físicamente presente. Esto reduce significativamente la posibilidad de suplantación o falsificación de asistencia , generando un almacenamiento de datos confiable y seguro. Sin embargo tenemos que considerar la situación de los usuarios de iPhone que es algo que se debe de tratar.
- **Eliminación de Error Humano.** El sistema automatiza el registro de la ID, la fecha y la hora exacta 9999, eliminando la transcripción manual y el alto margen de error asociado a la ilegibilidad o la omisión de datos.

2. ¿Qué tipo de personas pueden ser beneficiarias de su proyecto?

El proyecto beneficia principalmente a dos grupos de usuarios, mejorando su productividad y la calidad del control de datos:

- **Profesores y Docentes.** Son los principales beneficiarios porque el sistema evita su intervención manual , permitiéndoles iniciar la clase de forma inmediata y dedicar el tiempo recuperado a la enseñanza. El registro digital y ordenado simplifica su labor administrativa.
- **Estudiantes.** Se benefician de un proceso más rápido y eficiente, percibiendo la interacción como instantánea. Además, la eliminación de la interrupción en el ritmo de la clase favorece un mejor aprovechamiento del tiempo académico.
- **Administradores Escolares.** Se benefician de la generación de un registro digital,

seguro y verificable, lo cual es indispensable para llevar un control estricto sobre la participación y asistencia de los estudiantes en la institución.

13. Conclusiones y trabajo futuro

13.1. Conclusión

Con la realización de este proyecto fue posible cumplir satisfactoriamente el objetivo planteado al inicio del diseño: desarrollar e implementar una solución tecnológica capaz de automatizar el proceso de captura y registro de asistencia en tiempo real, utilizando comunicación Bluetooth mediante Serial Port Profile (SPP), un microcontrolador Arduino Nano y un sistema de almacenamiento basado en tarjeta SD.

Los resultados obtenidos demostraron que:

- El sistema detecta correctamente la presencia de los dispositivos Bluetooth
- La asistencia se registra de manera inmediata, ordenada y confiable
- Se almacena de manera correcta la asistencia, registrando número de cuenta, fecha y hora de registro.
- Se reduce significativamente la posibilidad de suplantación o falsificación de asistencia, incrementando la precisión y veracidad del registro.

13.2. Trabajo a futuro

Si bien el sistema desarrollado cumple con los objetivos planteados, existen múltiples oportunidades de mejora y expansión que podrían incrementar su funcionalidad, robustez y alcance:

13.2.1. Migración a Bluetooth 4.0

El diseño actual utiliza un módulo Bluetooth que opera con Bluetooth Classic y el perfil Serial Port Profile (SPP). Aunque SPP es sencillo y funciona bien con muchos dispositivos Android y PCs, iOS (iPhone/iPad) no ofrece soporte general para SPP a menos que el accesorio esté aprobado por Apple mediante programas propietarios (MFi) o use perfiles autorizados, esto se debe a que apple eliminó la compatibilidad con Bluetooth SPP de iOS, optando en su lugar por un método de autenticación propietario.

Apple fomenta el uso de Bluetooth Low Energy (BLE) para la mayoría de accesorios modernos y permite el acceso a BLE desde aplicaciones iOS mediante la API Core Bluetooth. Es por ello que, los perfiles clásicos como SPP no están disponibles libremente para todas las aplicaciones iOS por razones de seguridad y control del ecosistema; por tanto, un módulo que exponga únicamente SPP no será detectable/conectable por muchos iPhone.

De tal manera que esto provocaría que en nuestro sistema de asistencia, se limite la compatibilidad del sistema solo a usuarios Android, excluyendo a usuarios de iPhone.

Para resolver esta limitación se recomienda migrar a un módulo con Bluetooth 4.0 (BLE) (por ejemplo, HM-10 o módulos basados en Nordic nRF52).

Esta migración aporta compatibilidad nativa con iPhone (a partir de iPhone 4S y modelos posteriores que soportan BLE), reduce consumo energético y facilita la implementación de sistemas multiplataforma (iOS/Android) que use Core Bluetooth y las APIs BLE nativas.

13.2.2. Aplicación móvil

La aplicación actual cumple con el objetivo básico, registrar la asistencia. Sin embargo, sería posible desarrollar una serie de mejoras, lo cual ampliaría significativamente las capacidades del proyecto sin necesidad de modificar el hardware central y así tener una plataforma más completa para la gestión y consulta del sistema de asistencia, aprovechando una interfaz gráfica intuitiva y accesible. Algunas funciones que podrían incorporarse son:

- Visualizar asistencias en tiempo real: Cada vez que un alumno envíe su número de cuenta, el Arduino valida el dato, registra la fecha y hora en la MicroSD, envía un mensaje de confirmación por Bluetooth y la aplicación lo muestra en tiempo real en una tabla dinámica (Actualmente solo puede reflejarse la asistencia con la luz verde).
- Generar reportes automáticos: La aplicación podría descargar el archivo CSV desde la MicroSD, procesarlo mediante librerías internas y generar reportes de puntualidad, alumnos cerca del límite de faltas y alumnos que ya han sobrepasado dicho límite.
- Exportar registros en formatos como CSV o PDF: La app permitiría exportar los registros como archivos CSV para Excel, o como PDF con tablas, gráficas y totales.
- Consulta de historial individual: La aplicación facilitaría al docente consultar el historial de cada estudiante: días asistidos, horarios de entrada, faltas acumuladas, porcentaje de asistencia, entre otros indicadores relevantes.

13.2.3. Registro automático

Actualmente, el sistema requiere que el alumno envíe sus datos manualmente. No obstante, sería posible automatizar completamente este proceso para que cada estudiante transmita su identificación sin intervención humana. Para lograrlo, sería necesario reemplazar el HC-06 por un módulo BLE moderno, ya que solo BLE permite el escaneo de advertising sin conexión previa.

Se propone que cada alumno instale una aplicación móvil que active el modo BLE advertiser, enviando paquetes que incluyan el número de cuenta o un token único de identificación. Al ingresar al salón, el teléfono comenzaría a emitir automáticamente los paquetes BLE, y el

módulo central (BLE scanner) detectaría estos paquetes, obteniendo la dirección BLE y el ID correspondiente. De esta forma, el sistema podría registrar la asistencia sin que el alumno tenga que conectarse o escribir sus datos.

Esta propuesta es viable, ya que los teléfonos modernos (Android y iOS) soportan BLE advertising, lo que permite emitir los paquetes de identificación de forma continua y eficiente. Además sería de gran utilidad pues, no requiere hardware adicional para los estudiantes, es compatible con grupos numerosos y reduce el tiempo de pase de lista al ofrecer un registro completamente automático.

13.2.4. Integración con un servidor en la nube

Aunque actualmente el Arduino almacena toda la información localmente en la MicroSD, sería altamente beneficioso migrar el almacenamiento a un servidor en la nube. Esto podría lograrse mediante una aplicación móvil que funcione como intermediaria: el profesor sincroniza los datos de la MicroSD con la app, y esta envía los registros a un servidor a través de una API. En el servidor, los registros se almacenarían en una base de datos que posteriormente podría consultarse desde distintos dispositivos.

Esta integración permitiría contar con un respaldo seguro de toda la información, acceso remoto desde cualquier ubicación, y la posibilidad de escalar el sistema a mayor capacidad (varios salones, edificios o incluso diferentes escuelas).

14. Video

En el siguiente enlace se encuentra el video del proyecto.

<https://www.youtube.com/watch?v=YCkDmbGQmos>

15. Referencias

- [1] Communications project. (n.d.). <https://catedraldeif.ngrok.io/SistemasComElectro/projects/project.html>
- [2] DOF - Diario Oficial de la Federación. (n.d.). https://www.dof.gob.mx/nota_detalle_popup.php?codigo=4998084
- [3] Secretaría de Educación Pública. (2018). Lineamientos para la organización y funcionamiento de las escuelas telesecundarias. <https://www.sep.gob.mx/work/models/sep1/Resource/42b00ee7-33da-4bff-85e3-ef45b0f75255/a255.pdf>
- [4] Satoshi. (2020, June 15). ¿Qué es una topología de red? - Opiron. Opiron. <https://www.opiron.com/que-es-una-topologia-de-red/>
- [5] International IT. (2022, July 5). Topología de Red: conozca los principales tipos. International IT. <https://goo.su/sb3r>
- [6] SANDOROBOTICS. (2025, April 16). Módulo WIFI ESP8266 ESP-01 - SANDOROBOTICS. <https://sandorobotics.com.mx/producto/hr0307/>
- [7] NodeMCU ESP8266 Development Board — Wi-Fi IoT Module —. (2021, August 22). Soluciones Totales En Electrónica, S.A. <https://www.steren.com.gt/placa-de-desarrollo-nodemcu-esp8266.html>
- [8] ESP32 Wi-Fi & Bluetooth SOC | Espressif Systems. (n.d.). <https://www.espressif.com/en/products/socs/esp32>
- [9] SoC serie nRF52. (n.d.). Nordic — Mouser. <https://www.mouser.es/new/nordic-semiconductor/nrf52-series-soc>
- [10] SX1276. (n.d.). <https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>
- [11] UNIT Electronics. (2025, November 8). Inalámbrico NRF24L01 2.4GHz - UNIT Electronics. <https://uelectronics.com/producto/inalambrico-24l01-nrf24l01-2-4ghz/>
- [12] ¿Qué es XBee? XBee.cl - Comunicación Inalámbrica para Tus Proyectos. (2019, November 29). XBee.cl - Comunicación Inalámbrica Para Tus Proyectos. <https://xbee.cl/que-es-xbee/>
- [13] ZIGBEE CC2530 TI MODULO TRANSECTOR RF 2.4GHZ UART SERIE. (n.d.). SSDIELECT ELECTRONICA SAS. <https://ssdiselect.com/xbee-1/4753-zigbee-cc2530.html>
- [14] Canada, C. S. E. (2024, April 25). Using Bluetooth technology (ITSAP.00.011) - Canadian Centre for Cyber Security. Canadian Centre for Cyber Security. <https://www.cyber.gc.ca/en/guidance/using-bluetooth-technology-itsap00011>
- [15] Carmona, J. A. (2021, October 11). Bluetooth y su evolución: estas son las diferencias entre las distintas clases y versiones. Xataka Smart Home. <https://www.xatakahome>

- .com/curiosidades/bluetooth-su-evolucion-estas-diferencias-distintas-versiones
- [16] Conceptos básicos de Bluetooth - SparkFun Learn. (n.d.). <https://learn.sparkfun.com/tutorials/bluetooth-basics/bluetooth-profiles>
 - [17] Bluetooth Basics - SparkFun Learn. (n.d.). <https://learn.sparkfun.com/tutorials/bluetooth-basics/bluetooth-profiles>
 - [18] Kg, R. & S. G. & C. (n.d.). Bluetooth Classic testing. Rohde & Schwarz. https://www.rohde-schwarz.com/fi/solutions/wireless-communications-testing/wireless-standards/bluetooth/bluetooth-classic-br-edr/bluetooth-classic-testing_251248.html
 - [19] Módulo Bluetooth HC06. (n.d.). Naylamp Mechatronics - Perú. <https://naylampaechtronics.com/inalambrico/24-modulo-bluetooth-hc06.html>
 - [20] Interfacing the HC-06 Bluetooth module with Arduino. (n.d.). Arduino Project Hub. <https://projecthub.arduino.cc/RucksikaaR/interfacing-the-hc-06-bluetooth-module-with-arduino-94aab>
 - [21] Arduino Integrated Development Environment (IDE) v1 — Arduino Documentation. (n.d.). <https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics/>
 - [22] Yañez, C. (2024, September 26). Qué es IDE de Arduino y qué se puede hacer con él. CEAC. <https://www.ceac.es/blog/que-es-el-ide-de-arduino-en-robotica>
 - [23] Dicsan Technology. (2025, July 22). Cloud Storage for Security Cameras vs Local Storage. <https://dicsan.com/cloud-storage-for-security-cameras-vs-local-storage/>
 - [24] Cloud Storage vs Local Storage: Which One Is Better for Security Cameras? (2025, June 24). aosu Deutschland. <https://de-aosulife.com/blogs/smart-home/cloud-storage-vs-local-storage-which-one-is-better-for-security-cameras>
 - [25] Emq. (n.d.). IoT in the Cloud: 8 key benefits and how to get started. www.emqx.com. <https://www.emqx.com/en/blog/iot-in-the-cloud-8-key-benefits-and-how-to-get-started>
 - [26] Los costos ocultos del IoT celular revelados. (n.d.). <https://www.hologram.io/es/revelando-los-gastos-ocultos-del-iot-cellular/>
 - [27] Microsoft Defender para IoT — Seguridad de Microsoft. (n.d.). <https://www.microsoft.com/es-es/security/business/endpoint-security/microsoft-defender-iot>
 - [28] Administrator, & Administrator. (2022, December 20). TCP/IP Stack: Network Layers and Protocols - 101 Computing. 101 Computing - Boost Your Programming Skills! <https://www.101computing.net/tcp-ip-stack-network-layers-and-protocols/>
 - [29] FERMIN FLORES. (2024, July 20). ¡Domina el Tiempo! Usando RTC DS3231 con Arduino NANO! [Video]. YouTube. https://www.youtube.com/watch?v=G_Q56qL5gEA

- [30] Nano | Arduino Documentation. (n.d.). <https://docs.arduino.cc/hardware/nano/>
- [31] González Teyssier, J. (s. f.). Pasar lista: un recurso disciplinario-pedagógico subestimado. Recuperado 17 de noviembre de 2025, de http://memoria.cch.unam.mx/tmp/pdfarticulo/31/JORGE_GONZALEZ_TEYSSIER_1381958307.pdf
- [32] DS3231 | Arduino Documentation. (n.d.). <https://docs.arduino.cc/libraries/ds3231>
- [33] Visual Studio: IDE y editor de código para el desarrollo de software. (2025, November 11). Visual Studio. <https://visualstudio.microsoft.com/es/>
- [34] JUAN JOSE. (2019, October 14). Cómo conectar Arduino NANO al computador [Video]. YouTube. <https://www.youtube.com/watch?v=sKMKPbbWfbI>
- [35] Rucksikaa Raajkumar. (2021, March 1). Interfacing HC-06 Bluetooth Module with Arduino [Full Explanation] | Pinout, Connections and Codes [Video]. YouTube. <https://www.youtube.com/watch?v=fCLlu7R2jYQ>
- [36] Martyn. (2024, June 14). Arduino and HC-06 (ZS-040). <https://www.martyncurrey.com/arduino-and-hc-06-zs-040/>
- [37] Interfacing the HC-06 Bluetooth module with Arduino. (n.d.). Arduino Project Hub. <https://projecthub.arduino.cc/RucksikaaR/interfacing-the-hc-06-bluetooth-module-with-arduino-94aab>
- [38] Descargar PyCharm: el IDE de Python para la ciencia de datos y el desarrollo web de JetBrains. (2021, June 2). JetBrains. <https://www.jetbrains.com/es-es/pycharm/download/?section=windows>
- [39] Download - Sublime Text. (n.d.). <https://www.sublimetext.com/download>
- [40] Why don't Apple devices recognise the B952 / B954 over Bluetooth? (2025). Em-trak. https://productsupport.em-trak.com/hc/en-gb/articles/28856277332765-Why-don-t-Apple-devices-recognise-the-B952-B954-over-Bluetooth?utm_source=chatgpt.com
- [41] iOS bluetooth development: mimicking the SPP profile.(s. f.). Software Engineering Stack Exchange. https://softwareengineering.stackexchange.com/questions/284990/ios-bluetooth-development-mimicking-the-spp-profile?utm_source=chatgpt.com
- [42] Wikipedia contributors. (2025, 17 octubre). Bluetooth Low energy. Wikipedia. https://en.wikipedia.org/wiki/Bluetooth_Low_Energy?utm_source=chatgpt.com
- [43] Bluetooth Low Energy 4.0 Module - HM-10. (s. f.). Arduino Official Store. https://store.arduino.cc/products/bluetooth-low-energy-4-0-module-hm-10?srsltid=AfmB0opWPrFNwV3dEpMyrEKqzEMCpBHYxu7LTPeKC9TOWLXKQus5V9Jq&utm_source=chatgpt.com
- [44] Using the Serial Monitor tool — Arduino Documentation. (n.d.). <https://docs.arduino.cc/software/ide-v2/tutorials/ide-v2-serial-monitor/>