

TECNOLÓGICO NACIONAL DE MÉXICO
Instituto Tecnológico de Apizaco

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

INGENIERÍA EN TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN

Proyecto Residencias Profesionales

NOMBRE DEL PROYECTO
**Implementación de un sistema de control de asistencia basado en IoT para
mejorar la eficiencia en las oficinas del PRD Tlaxcala**

PRESENTA:
Danna Paola Vargas Cambrón

ASESOR EXTERNO:
Lauro Carlos Payan Reyes

ASESOR EXTERNO:
Lic. Miguel Hernández Méndez

Agradecimientos

Quiero agradecer a mi familia y amigos que siempre me han apoyado y han estado a mi lado en cada momento de esta etapa en mi vida.

Quiero agradecerle a Sandra Cambron Soria, mi madre, y mis hermanos, quienes siempre confiaron en mí, en mi potencial y sobre todo por nunca soltarme. A mi madre, por ser mi mejor amiga cuando más lo necesite, limpiarme las lágrimas ante mis momentos de estrés y darme toda la fuerza. Quien también me acompañó en las largas madrugadas y claro, por siempre darme un golpe de realidad cada que fuera necesario.

De igual modo quiero agradecer a Manuel Cambron y Rosa Soria, mis abuelos, que siempre han sido mi apoyo más grande y siempre han estado sin importar el clima, las circunstancias, ni la hora. También quiero agradecer a mis padrinos y tios, que han sido mi voz de conciencia cuando más necesitaba un consejo o solo platicar de cosas importantes.

También quiero agradecer a Ismael Hernández, mi pareja, mi mejor amigo, mi todo, quien siempre me dio fuerza para no rendirme y desde que lo conozco, hemos caminado de la mano y hemos estado siempre el uno para el otro a pesar de las altas y baja que hemos tenido. Gracias amor por no soltarme y prometo no soltarte.

De igual modo, agradecer a esas amistades que hice durante esta etapa, que me apoyaron y viví muy buenas aventuras que sin duda guardaré en mi mente y corazón para toda la vida, que sin duda espero sigan formando parte de mi vida y seguir viviendo más aventuras juntos (AC, el mejor equipo). También quiero agradecer al Instituto Tecnológico de Apizaco por proporcionarme los recursos y el apoyo necesario para hoy poder realizar este proyecto y hoy poder ser egresada. Igualmente agradecer a mi asesor Lauro Carlos Payan Reyes, por el apoyo y comprensión para la realización de este proyecto.

Resumen

El presente proyecto tiene como propósito principal el diseño e implementación de un sistema de control de asistencia basado en IoT, específicamente para las oficinas del PRD Tlaxcala. A través del uso de tecnología como el microcontrolador ESP32 y un sensor biométrico de huella digital (AS608), el sistema permitirá registrar de manera automática y segura la asistencia del personal. La información recopilada será enviada y almacenada en una base de datos para su posterior consulta, análisis y generación de reportes a través de una interfaz web intuitiva.

Este sistema busca resolver las deficiencias del método tradicional de control de asistencia, reduciendo errores, optimizando el tiempo administrativo, y fortaleciendo la capacidad de gestión del partido mediante información precisa y actualizada. Adicionalmente, se contemplan medidas de protección de datos y privacidad, alineadas con las normativas vigentes.

El proyecto se desarrolla exclusivamente en las oficinas del PRD en Tlaxcala, con una duración determinada, y se enfoca en automatizar el registro de asistencia, mejorar la eficiencia operativa y proporcionar herramientas de análisis que respalden la toma de decisiones estratégicas en la organización.

Índice

Agradecimientos	2
Resumen	3
Generalidades del proyecto	5
Introducción	5
Delimitación	6
Lugar donde se realizó el proyecto	7
Problemas a Resolver	9
Objetivos	10
Justificación	11
Cronograma preliminar de actividades	12
Marco Teórico	14
Desarrollo	28
Materiales	28
Diagramas de conexión	29
Instalación de Requerimientos	30
Instalación de Controladores	35
Inclusión de Librerías	37
Pruebas de Funcionamiento	41
Pruebas y optimización	42
Conclusiones	51
Recomendaciones	52
Referencias	53

Generalidades del proyecto

Introducción

La eficiencia en la gestión del recurso humano es un elemento clave para el buen funcionamiento de cualquier institución, incluidas las organizaciones políticas. En las oficinas del Partido de la Revolución Democrática (PRD) en Tlaxcala, el control manual de asistencia ha representado históricamente un desafío, tanto por su vulnerabilidad a errores como por la dificultad para generar reportes oportunos y confiables. En respuesta a esta problemática, el presente proyecto tiene como objetivo desarrollar e implementar un sistema de control de asistencia basado en tecnologías del Internet de las Cosas (IoT), orientado a optimizar la administración del personal mediante la automatización del registro de entradas y salidas, así como el monitoreo en tiempo real de la asistencia.

Este sistema utilizará dispositivos electrónicos como microcontroladores, sensores biométricos y conectividad inalámbrica para digitalizar completamente el proceso de control de asistencia. La información recolectada será centralizada en una base de datos segura y accesible a través de una plataforma web, lo que permitirá a los responsables administrativos generar reportes detallados y tomar decisiones fundamentadas. Además, el sistema garantizará el cumplimiento de estándares de seguridad y protección de datos personales.

El desarrollo del proyecto se limita a las instalaciones físicas del PRD en Tlaxcala, donde se llevará a cabo su diseño, prueba e implementación, dentro de un periodo previamente establecido. El enfoque estará centrado en la automatización del proceso de asistencia, la generación de estadísticas, y la mejora de los tiempos de respuesta administrativos.

Delimitación

El proyecto se centrará en el desarrollo e implementación de un sistema de control de asistencia basado en IoT para las oficinas del PRD Tlaxcala, con un enfoque en la automatización del proceso de registro de asistencia y la generación de informes y análisis de datos. El proyecto tendrá un plazo de [especificar plazo] para su desarrollo e implementación, y se realizará exclusivamente en las oficinas del PRD Tlaxcala.

Lugar donde se realizó el proyecto

Ubicación

Nombre de la empresa: Partido de la Revolución Democrática en Tlaxcala

Giro, ramo o sector: Público

Domicilio: Blvd. Mariano Sánchez, #11. Col. Centro. Tlaxcala de Xicoténcatl Tlaxcala

C.P.: 901000

RFC: PRD241201KE5

Ciudad: Tlaxcala

Teléfono: 2464621199

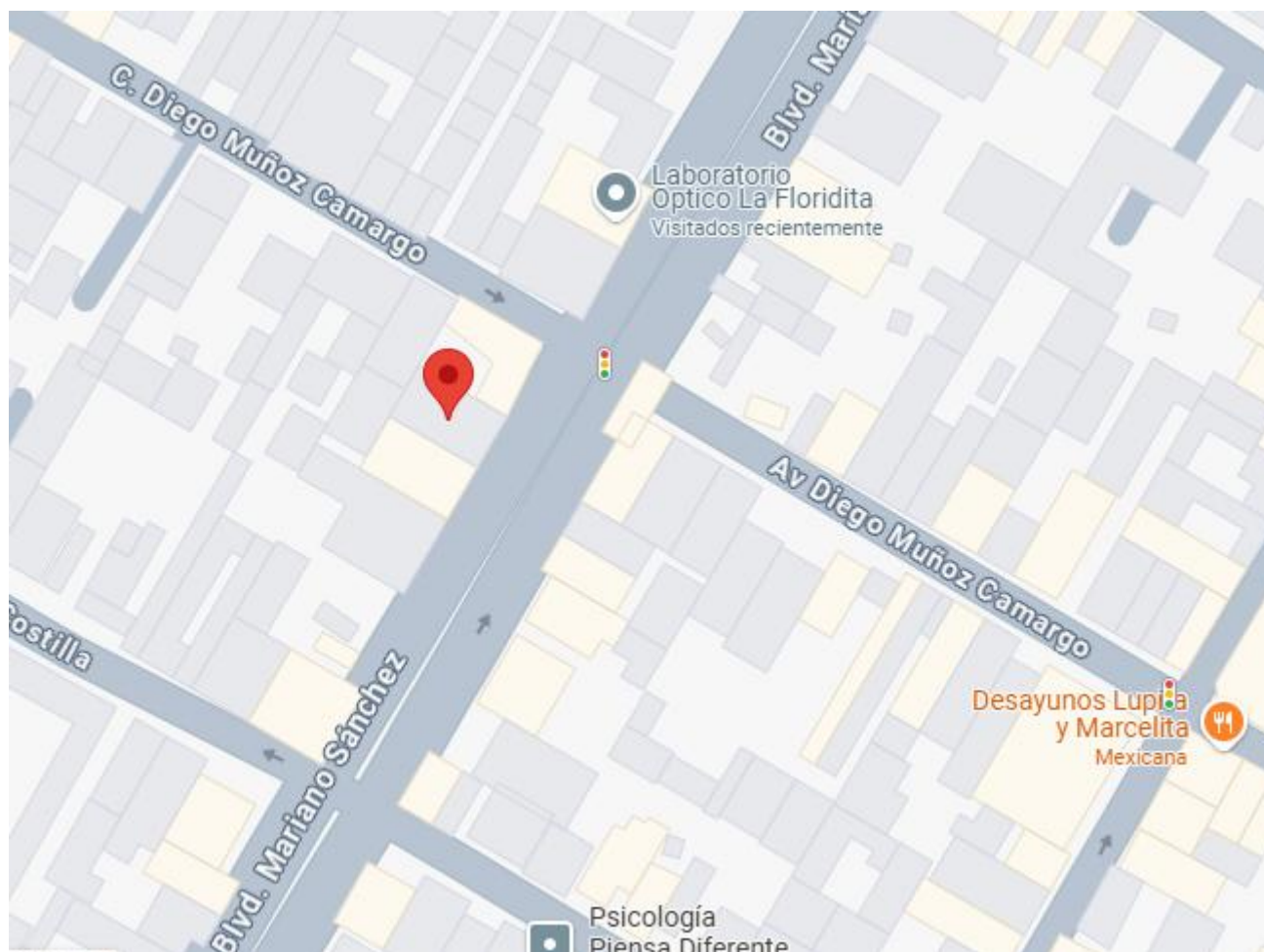


Figura 1. Ubicación del Partido

Logotipo de Empresa



Figura2. Logo del Partido

Misión

Somos un grupo de ciudadanos, mujeres y hombres simpatizantes y militantes de izquierda, organizados entorno al Partido de la Revolución Democrática Tlaxcala, dedicados a la práctica política y a la organización electoral, comprometidos con la defensa de la democracia, de las instituciones, de la república, de causas y derechos humanos que permitan el engrandecimiento de la sociedad tlaxcalteca; ejercemos la política con una visión moderna, profesional y estamos comprometidos con los valores y principios de la izquierda democrática, pretendemos ser un ente promotor e impulsor de la cultura política democrática en Tlaxcala.

Visión

Gobernar el Estado de Tlaxcala; consolidar al partido como opción política consistente con una posición de izquierda moderna, democrática y progresista.

Objetivo

Hacemos política. La política es un medio para la construcción de acuerdos que deriven en beneficios directos para la gente, la política debe privilegiar las coincidencias. Creemos que el PRD Tlaxcala es el partido político baluarte de la izquierda mexicana.

Promovemos el respeto sin menoscabo a las distintas opiniones, actitudes y aptitudes entre los militantes y con la sociedad en su conjunto.

Valores

- Honestidad
- Respeto
- Inclusión y pluralismo
- Solidaridad

Problemas a Resolver

En la actualidad, las oficinas del PRD Tlaxcala enfrentan limitaciones significativas en el control del registro de asistencia del personal, derivadas del uso de métodos manuales o sistemas no centralizados que dificultan la supervisión efectiva y oportuna. Esta situación genera múltiples consecuencias: pérdida de tiempo en procesos administrativos, errores en los registros, dificultad para generar reportes precisos, baja trazabilidad de los datos, y escasa capacidad para detectar patrones de ausentismo o incumplimiento de horarios laborales.

Además, la falta de automatización en este proceso limita la toma de decisiones informadas por parte de los responsables de recursos humanos y administración, quienes requieren datos fiables y en tiempo real para evaluar el rendimiento y cumplimiento del personal. Este problema se agrava ante la necesidad de implementar buenas prácticas de gobernanza, rendición de cuentas y uso eficiente de los recursos humanos, particularmente en una organización política con responsabilidades públicas como lo es el PRD.

Por tanto, se identifica la necesidad de contar con un sistema de control de asistencia que incorpore tecnología moderna, segura y automatizada, que permita mejorar la eficiencia operativa, reducir el margen de error, optimizar el tiempo y proporcionar datos útiles para la toma de decisiones estratégicas.

Objetivos

Objetivo General

Desarrollar e implementar un sistema de control de asistencia basado en IoT que permita mejorar la eficiencia y la productividad en las oficinas del PRD Tlaxcala, mediante la automatización y el monitoreo de la asistencia del personal, y proporcionar información precisa y oportuna para la toma de decisiones.

Objetivos Específicos

- Diseñar y desarrollar un sistema de control de asistencia que integre tecnologías IoT y permita la automatización del proceso de registro de asistencia.
- Implementar un sistema de monitoreo y seguimiento de la asistencia del personal, que permita identificar patrones y tendencias.
- Proporcionar información precisa y oportuna para la toma de decisiones, mediante la generación de informes y análisis de datos.
- Mejorar la eficiencia y la productividad en las oficinas del PRD Tlaxcala, mediante la reducción de errores y la optimización del tiempo.

Justificación

La implementación de un sistema de control de asistencia basado en IoT en las oficinas del PRD Tlaxcala se justifica por la necesidad de mejorar la eficiencia y productividad en la gestión de personal y recursos. La automatización del proceso de registro de asistencia permitirá reducir errores y optimizar el tiempo, proporcionando información precisa y oportuna para la toma de decisiones. Además, este sistema ayudará a cumplir con los requerimientos de seguridad y privacidad, protegiendo la información personal y confidencial de los empleados.

Cronograma preliminar de actividades

ACTIVIDAD	Enero		Febrero				Marzo				Abril				Mayo				Junio			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
Planificación y diseño	P																					
	R																					
Desarrollo del <u>cheCADOR</u> y <u>Backend</u>	P																					
	R																					
Desarrollo del <u>Frontend</u> e investigación de tecnologías	P																					
	R																					
Pruebas y optimización	P																					
	R																					
Implementación	P																					
	R																					
Cierre	P																					
	R																					
OBSERVACIONES																						
+																						
Entrega de reportes	Asesor interno																					
	Estudiante																					
	Jefe de Depto.																					

Descripción de actividades

ACTIVIDADES	DURACION	COMIENZO	FIN
Planeación y diseño	24 días	06 – Enero -- 2025	06 – Feb -- 2025
Asignación, investigación, diseño del sistema y selección de sensores	14 días	06 – Enero -- 2025	23 – Enero – 2025
Validación y ajustes del diseño	14 días	24 – Enero -- 2025	06 – Feb -- 2025
Desarrollo del <u>cheCADOR</u> y <u>Backend</u>	20 días	07 – Feb -- 2025	07 – Marzo – 2025
Creación de la base de datos, backend y Firmware	10 días	07 – Marzo – 2025	20 – Feb – 2025
Integración del backend con el <u>cheCADOR</u> y seguridad del sistema	10 días	21 – Feb – 2025	07 – Marzo – 2024

Desarrollo del Frontend e investigación con el sistema	20 días	10 – Marzo – 2025	04 – Abril – 2025
Reunión para la distribución de tareas	7 días	10 – Marzo – 2025	18 – Marzo – 2025
Realización del diseño	7 días	19 – Marzo – 2025	27 – Marzo – 2025
Realización del Modelado de datos	6 días	28 – Marzo – 2025	04 – Abril – 2025
Pruebas y optimización	20 días	07 – Abril – 2025	02 – Mayo -- 2025
Pruebas de funcionamiento, rendimiento y seguridad	10 días	07 – Abril – 2025	18 – Abril – 2025
Corrección de errores y documentación del sistema	10 días	21 – Abril – 2025	02 – Mayo -- 2025
Implementación	15 días	05 – Mayo -- 2025	30 – Mayo – 2025
Implementación en producción y capacitación de usuarios	3 días	02– Mayo – 2025	30 – Mayo – 2025
Cierre	5 días	02 – Junio – 2025	20 – Junio – 2025
Ajustes finales y cierra del proyecto	1 día	02 – Junio – 2025	20 – Junio – 2025

Marco Teórico

Internet de las Cosas (IoT)

El Internet de las Cosas, conocido por sus siglas en inglés como IoT (Internet of Things), representa una de las transformaciones tecnológicas más significativas del siglo XXI. Se trata de una red interconectada de objetos físicos equipados con sensores, software y tecnologías de comunicación que les permiten recopilar, enviar y recibir datos a través de internet. Estos objetos, que tradicionalmente eran considerados “tontos” o inertes, adquieren así la capacidad de interactuar entre sí, con sistemas informáticos complejos y con los seres humanos, facilitando procesos automatizados, monitoreo en tiempo real y toma de decisiones inteligente.

El concepto de IoT no es reciente. Su origen puede rastrearse hasta la década de 1990, pero ha sido en los últimos años cuando su implementación se ha expandido con mayor fuerza, impulsada por el abaratamiento de la tecnología, la miniaturización de los sensores, el avance de las telecomunicaciones y el crecimiento de la capacidad de procesamiento en la nube. Hoy en día, el IoT no es solo una promesa futura, sino una realidad que se manifiesta en una variedad de aplicaciones prácticas en sectores como la salud, la industria, el transporte, la agricultura y la gestión de ciudades.

El ecosistema del IoT está conformado por varios componentes interrelacionados. En primer lugar, están los dispositivos físicos o “cosas” que pueden ser desde electrodomésticos y vehículos hasta maquinaria industrial, drones, sistemas de riego o dispositivos médicos portátiles. Estos están equipados con sensores que capturan información del entorno, como temperatura, presión, movimiento, humedad, calidad del aire o niveles de luz. Los datos generados por estos sensores son transmitidos a través de tecnologías de conectividad como Wi-Fi, redes móviles 4G y 5G, Bluetooth, Zigbee o tecnologías de bajo consumo como LoRaWAN y NB-IoT, dependiendo del contexto y las necesidades de cada aplicación.

Una vez transmitidos, los datos son procesados, ya sea en la nube (cloud computing) o cerca del lugar donde se generan (edge computing), dependiendo del nivel de latencia requerido. En muchos casos, este procesamiento incluye el uso de algoritmos de inteligencia artificial y aprendizaje automático (machine learning), que permiten extraer patrones, realizar predicciones y tomar decisiones automatizadas. Por ejemplo, un sistema agrícola basado en IoT puede predecir las condiciones ideales para el riego y ejecutarlo automáticamente, mejorando la eficiencia en el uso del agua y aumentando el rendimiento de los cultivos.

Además de los sensores, otro componente clave del IoT son los actuadores, que permiten ejecutar acciones físicas como abrir una compuerta, activar una alarma o ajustar el termostato de una habitación. La interacción entre sensores, redes, algoritmos de análisis y actuadores da lugar a sistemas ciberfísicos que integran el mundo digital y físico de forma sinérgica.

Las aplicaciones del IoT son tan amplias como variadas. En el hogar, permite la creación de entornos inteligentes en los que se pueden controlar dispositivos como luces, cámaras, refrigeradores o asistentes virtuales a través de un smartphone o comandos de voz. En el ámbito industrial, conocido como IIoT (Industrial Internet of Things), se utiliza para monitorear en tiempo real la maquinaria, predecir fallas antes de que ocurran (mantenimiento predictivo), gestionar líneas de producción y optimizar recursos. En la salud, los dispositivos portátiles y sensores biométricos permiten un monitoreo continuo de pacientes, facilitando diagnósticos más tempranos y tratamientos personalizados. En el sector del transporte, los sistemas IoT permiten el seguimiento en tiempo real de flotas, la planificación de rutas eficientes, la prevención de accidentes y el desarrollo de vehículos autónomos. Por otro lado, en las ciudades inteligentes, se emplean para gestionar el tráfico, el consumo energético, los residuos sólidos y la seguridad pública.

A pesar de sus múltiples beneficios, el desarrollo y expansión del IoT enfrenta importantes desafíos. Uno de los más críticos es la ciberseguridad. Al estar constantemente conectados, los dispositivos IoT son vulnerables a ataques que pueden comprometer la integridad de los datos o el funcionamiento de sistemas críticos. Según datos recientes, se estima que más de la mitad de los dispositivos IoT en el mercado presentan vulnerabilidades graves, debido principalmente a la falta de estándares de seguridad y actualizaciones deficientes. Este problema ha llevado a la creación de iniciativas como el Cyber Trust Mark en EE.UU., que busca establecer certificaciones de seguridad obligatorias para dispositivos conectados.

Otro desafío es la privacidad. Muchos de estos dispositivos recopilan información sensible del usuario o del entorno, lo cual plantea interrogantes legales y éticos sobre el uso y almacenamiento de dichos datos. La falta de regulación clara y la existencia de marcos legales obsoletos dificultan la protección efectiva de la información personal en contextos donde el IoT se entrelaza con la vida cotidiana.

La interoperabilidad también constituye una barrera significativa. La mayoría de los dispositivos IoT son desarrollados por distintos fabricantes con protocolos de comunicación propietarios, lo cual dificulta su integración dentro de un ecosistema común. Para mitigar este problema, se han impulsado iniciativas como el estándar Matter, promovido por empresas como Apple, Google y Amazon, que busca unificar el lenguaje de comunicación entre dispositivos de distintos fabricantes, especialmente en el ámbito del hogar inteligente.

Por otro lado, el volumen de datos generado por el IoT plantea grandes exigencias en términos de almacenamiento, procesamiento y análisis. La cantidad de información que se produce de forma continua requiere infraestructuras robustas, así como el uso de tecnologías avanzadas como big data, edge computing y plataformas de inteligencia artificial escalables.

A nivel global, las cifras proyectadas para el crecimiento del IoT son impresionantes. Según estimaciones de Cisco, para el año 2030 habrá más de 50 mil millones de dispositivos conectados. Esto implicará no solo una transformación tecnológica

profunda, sino también un cambio cultural y organizacional en la forma en que las personas interactúan con la tecnología. El impacto económico de este ecosistema podría superar los 3 billones de euros anuales en sectores como la manufactura, el comercio, la salud y la energía.

Mirando hacia el futuro, se prevé que el Internet de las Cosas se integre aún más con otras tecnologías emergentes, como la inteligencia artificial generativa, la computación cuántica, el blockchain y las redes 6G. Esta convergencia tecnológica abrirá nuevas posibilidades para crear entornos aún más autónomos, resilientes e inteligentes. No obstante, también exigirá mayores esfuerzos en materia de gobernanza, estandarización, sostenibilidad y ética tecnológica.

Sistemas de Control de Asistencia

El control de asistencia constituye una práctica esencial dentro de la gestión organizacional, cuya finalidad es registrar, supervisar y verificar la presencia de individuos —ya sean trabajadores, estudiantes o visitantes— en determinados espacios y franjas horarias. Su aplicación resulta estratégica en distintos sectores como la educación, la industria, la administración pública y los entornos corporativos, donde el cumplimiento horario, la puntualidad y la trazabilidad del tiempo laboral o académico representan variables clave para la eficiencia operativa, la productividad y la rendición de cuentas. Con el desarrollo de las tecnologías de la información y la comunicación, los sistemas de control de asistencia han evolucionado drásticamente desde métodos manuales hacia soluciones automatizadas, precisas y muchas veces basadas en inteligencia artificial.

En sus orígenes, los sistemas de control de asistencia se limitaban al uso de métodos analógicos, como libros de firmas, tarjetas perforadas, listas en papel o relojes mecánicos. Si bien estos sistemas cumplían una función básica de registro, presentaban limitaciones importantes: eran vulnerables a errores humanos, manipulaciones o fraudes —por ejemplo, el marcaje por parte de compañeros ausentes (conocido como “fichaje por otro”)—, además de ser ineficientes en términos de consolidación, análisis y almacenamiento de información. Estas debilidades motivaron el desarrollo de soluciones digitales, más seguras, ágiles y capaces de integrarse con otras plataformas institucionales.

En la actualidad, un sistema moderno de control de asistencia está compuesto por al menos tres componentes principales: los dispositivos de captura, la infraestructura de comunicación y el sistema de gestión centralizado. El primero hace referencia a los mecanismos mediante los cuales se registra la presencia de una persona: desde relojes checadores tradicionales hasta sistemas más sofisticados como lectores de tarjetas de proximidad, dispositivos biométricos (huella digital, reconocimiento facial, iris, voz, etc.), códigos QR y aplicaciones móviles con geoposicionamiento (GPS). La elección del método depende de múltiples factores: número de usuarios, nivel de seguridad requerido, entorno físico, presupuesto y condiciones de conectividad.

El segundo componente es la infraestructura tecnológica que permite el transporte seguro de los datos, ya sea por medio de redes cableadas, inalámbricas o vía internet. Esta debe garantizar estabilidad, baja latencia, redundancia y medidas de ciberseguridad. El tercer componente, el software de gestión, es donde se almacenan, procesan y visualizan los datos. Este sistema permite la generación de reportes automáticos, gráficos estadísticos, alertas por ausencias reiteradas o incumplimientos, así como la integración con sistemas de nómina, recursos humanos o bases de datos académicas.

Uno de los avances más significativos en esta área es el uso de la biometría, entendida como el reconocimiento automático de individuos basado en sus características fisiológicas o conductuales. Entre las tecnologías biométricas más utilizadas en control de asistencia se encuentran la huella dactilar, el reconocimiento facial, el escaneo del iris, el reconocimiento de voz y la geometría de la palma o del rostro. Estas tecnologías tienen la ventaja de ofrecer un nivel de seguridad mucho más alto, dado que las características biométricas son únicas, no transferibles y muy difíciles de falsificar. Además, permiten agilizar el proceso de identificación, reducir el contacto físico (en el caso de los sistemas sin contacto) y garantizar mayor transparencia en el manejo de la información.

Dentro de las instituciones educativas, los sistemas de control de asistencia cumplen una función adicional: impactar positivamente en el desempeño académico, reducir los niveles de deserción escolar, facilitar la intervención temprana frente a ausencias sistemáticas y mejorar la comunicación con las familias. Por ejemplo, existen sistemas que registran la asistencia de los estudiantes y notifican automáticamente a los padres de familia mediante mensajes SMS o correo electrónico cuando un alumno falta a clase. En este contexto, la asistencia deja de ser un simple dato y se convierte en un indicador pedagógico relevante para la gestión educativa.

En las empresas y organizaciones del sector privado, el control de asistencia forma parte de las políticas de gestión del talento humano, dado que permite supervisar la jornada laboral, evaluar la puntualidad, detectar ausentismo injustificado y gestionar los tiempos de descanso, permisos y vacaciones. Los reportes generados por estos sistemas son útiles para la elaboración de nóminas, el cálculo de horas extras o la identificación de patrones de comportamiento que afectan la productividad. Asimismo, los sistemas de control de asistencia permiten cumplir con requisitos legales y contractuales sobre el control horario, cada vez más exigentes en países con leyes laborales rígidas o sujetos a auditorías externas.

En los entornos laborales modernos —donde proliferan modalidades como el teletrabajo, el trabajo híbrido y los equipos distribuidos—, los sistemas tradicionales de control presencial pierden eficacia. En respuesta, han surgido soluciones basadas en geolocalización móvil, tokens digitales, reconocimiento facial por videollamada, o monitoreo de actividad mediante IA, que permiten registrar la actividad de los trabajadores desde cualquier lugar, sin vulnerar su privacidad. Este tipo de sistemas suele incorporar funciones de inicio y cierre de jornada,

registro de pausas, georreferenciación, e incluso autoevaluaciones o informes de desempeño.

A pesar de su utilidad, la implementación de sistemas de control de asistencia plantea retos significativos desde el punto de vista técnico, ético y legal. Uno de los principales es la protección de los datos personales, especialmente en los sistemas biométricos. Las características fisiológicas de una persona no pueden ser modificadas si se ven comprometidas, lo que convierte su manejo en una responsabilidad crítica. La normativa vigente en muchas jurisdicciones —como el Reglamento General de Protección de Datos (GDPR) en la Unión Europea o leyes similares en América Latina— establece criterios estrictos sobre el consentimiento informado, la finalidad del tratamiento, la minimización de datos y la conservación limitada de la información.

Otro aspecto a considerar es la aceptación por parte de los usuarios. Los trabajadores o estudiantes pueden resistirse al uso de tecnologías que perciben como invasivas, discriminatorias o excesivamente controladoras. Esto hace indispensable acompañar cualquier proceso de implementación con estrategias de comunicación, transparencia, capacitación y diálogo, que generen confianza y permitan una apropiación adecuada de la tecnología. Asimismo, se requiere garantizar que estos sistemas no se utilicen como herramientas de vigilancia masiva o de control disciplinario injustificado.

Desde el punto de vista técnico, los desafíos incluyen la fiabilidad del sistema, la interoperabilidad con otras plataformas, la escalabilidad para instituciones de gran tamaño, y la resistencia a condiciones adversas como polvo, humedad o fallos de energía. Por ello, muchos proveedores de software y hardware para control de asistencia invierten en el desarrollo de sistemas modulares, con arquitectura en la nube, capacidades de inteligencia artificial, mantenimiento remoto y soporte multilingüe.

En cuanto a las tendencias futuras, se prevé una mayor integración de los sistemas de control de asistencia con tecnologías emergentes como el blockchain —para garantizar la integridad de los registros—, el machine learning —para detectar patrones de ausentismo o predecir rotación de personal—, y la realidad aumentada —en entornos educativos y de formación. Asimismo, la automatización de la gestión horaria y la personalización de alertas y reportes jugarán un papel central en los sistemas de nueva generación.

En definitiva, los sistemas de control de asistencia representan mucho más que una herramienta administrativa; constituyen un componente esencial de la gobernanza institucional y organizacional. Cuando son diseñados e implementados con enfoque ético, técnico y participativo, pueden contribuir significativamente a la mejora del rendimiento, la equidad, la disciplina, la eficiencia operativa y la calidad del ambiente laboral o académico.

Biometría: concepto y aplicaciones

La biometría es una disciplina tecnológica y científica que se encarga del estudio, reconocimiento y verificación de la identidad de las personas mediante sus características físicas o comportamentales únicas. El término proviene del griego *bios* (vida) y *metron* (medida), lo que refleja su propósito principal: medir y analizar atributos biológicos para identificar individuos. Esta tecnología se basa en el principio de que ciertos rasgos humanos son distintivos, persistentes a lo largo del tiempo y difíciles de replicar, lo que los convierte en herramientas efectivas para procesos de autenticación, control de acceso, seguridad informática y gestión de la identidad.

Los sistemas biométricos se componen, en general, de cuatro elementos principales: el sensor, encargado de capturar el rasgo biométrico; el procesador, que convierte esa información en una representación digital (patrón); el almacenamiento, donde se guarda dicha información; y el mecanismo de comparación, que evalúa la similitud entre el patrón registrado y uno nuevo, con el fin de verificar la identidad. El objetivo de un sistema biométrico puede ser la verificación (comprobar si una persona es quien dice ser) o la identificación (descubrir la identidad de una persona a partir de su rasgo biométrico).

Existen dos grandes categorías de biometría: fisiológica y conductual. La biometría fisiológica analiza características físicas del cuerpo humano, como la huella dactilar, el rostro, el iris, la retina, la geometría de la mano, el patrón venoso o incluso el ADN. Por otro lado, la biometría conductual estudia aspectos relacionados con el comportamiento, como la dinámica de tecleo, la firma manuscrita, la forma de caminar (marcha) o el patrón de uso del ratón y dispositivos móviles. Aunque las características fisiológicas tienden a ser más estables y precisas, las conductuales tienen la ventaja de poder capturarse de forma continua y sin contacto directo.

Uno de los métodos biométricos más extendidos es la huella dactilar, utilizada ampliamente en sistemas de control de asistencia, desbloqueo de dispositivos móviles, control de fronteras y bancos. Su adopción se debe a su bajo costo, facilidad de implementación y alta precisión, siempre que el lector y las condiciones sean adecuadas. Otro método muy difundido es el reconocimiento facial, que ha ganado popularidad gracias a los avances en visión por computadora, inteligencia artificial y machine learning. Este sistema permite la identificación sin contacto físico y se usa comúnmente en aeropuertos, smartphones, sistemas de videovigilancia y eventos masivos.

El escaneo de iris y retina, aunque menos utilizado por su alto costo, ofrece niveles de seguridad extremadamente altos debido a la unicidad y estabilidad de estas estructuras oculares. Estas tecnologías son comúnmente empleadas en sistemas de seguridad de alto nivel, como laboratorios, centrales nucleares y servicios de inteligencia. La geometría de la mano fue una de las primeras formas biométricas utilizadas comercialmente, aunque ha sido desplazada por tecnologías más precisas y menos intrusivas. Recientemente, también ha surgido el reconocimiento

de patrones venosos, que utiliza infrarrojos para mapear las venas del dorso de la mano o de los dedos, ofreciendo alta seguridad y resistencia al fraude.

Las aplicaciones de la biometría son diversas y abarcan tanto el ámbito público como privado. En el campo de la seguridad física, se utiliza para controlar el acceso a instalaciones restringidas, como edificios gubernamentales, laboratorios, bancos o centros de datos. En el entorno digital, la biometría permite una autenticación robusta en plataformas de banca en línea, sistemas de firma electrónica, redes corporativas y dispositivos personales. Con el crecimiento del comercio electrónico y las operaciones digitales, la autenticación biométrica se está convirtiendo en un componente clave para reforzar la ciberseguridad y proteger la identidad digital de los usuarios.

En el ámbito gubernamental, los sistemas biométricos son fundamentales para la gestión de documentos de identidad (pasaportes biométricos, cédulas electrónicas), el registro electoral, la emisión de licencias de conducir, el control migratorio y la administración de programas sociales. Por ejemplo, países como India, a través del programa Aadhaar, han creado bases de datos biométricas que permiten identificar a más de mil millones de ciudadanos mediante huellas dactilares e iris, lo que ha facilitado el acceso a servicios públicos y subsidios, aunque también ha generado debates sobre privacidad.

En el sector educativo, la biometría ha sido adoptada como herramienta para el control de asistencia de estudiantes y personal docente, así como para garantizar la integridad en exámenes presenciales o virtuales. En entornos corporativos, se utiliza para monitorear la entrada y salida de empleados, autorizar accesos a áreas restringidas y generar reportes automáticos sobre horarios y productividad.

Otra área emergente es la biometría móvil, implementada en smartphones, tabletas y dispositivos portátiles. Tecnologías como el reconocimiento facial, el escáner de huella dactilar y el reconocimiento de voz permiten desbloquear dispositivos, autorizar pagos, validar transacciones bancarias y acceder a información sensible, todo ello sin necesidad de contraseñas tradicionales. Esto no solo mejora la experiencia del usuario, sino que refuerza la seguridad al dificultar el robo o suplantación de identidad.

Sin embargo, a pesar de sus múltiples ventajas, la biometría también plantea importantes retos éticos, técnicos y legales. Uno de los principales desafíos es la protección de los datos biométricos, que por su naturaleza son permanentes, sensibles y únicos. A diferencia de una contraseña, una huella dactilar o un rostro no pueden modificarse si son robados o vulnerados, lo que eleva los riesgos asociados al uso indebido o filtración de datos biométricos. En este contexto, resulta fundamental cumplir con normas de seguridad como la ISO/IEC 24745, que establece directrices para la protección de información biométrica, y legislaciones de protección de datos personales, como el Reglamento General de Protección de Datos (RGPD) en la Unión Europea o leyes locales como la LOPDGDD en España y la Ley 1581 en Colombia.

Otro tema relevante es la exactitud y sesgo de los sistemas biométricos. Diversos

estudios han demostrado que algunos algoritmos de reconocimiento facial presentan mayores tasas de error en personas de determinadas razas, géneros o edades, lo que puede derivar en discriminación algorítmica. Por ello, el diseño y entrenamiento de estos sistemas debe ser inclusivo, transparente y sujeto a auditorías éticas y técnicas.

Asimismo, el uso masivo de la biometría en espacios públicos y sistemas de videovigilancia plantea interrogantes sobre la vigilancia social y el derecho a la privacidad. La posibilidad de identificar a cualquier persona en tiempo real sin su consentimiento puede vulnerar libertades fundamentales si no se establecen límites legales claros, mecanismos de supervisión y criterios de proporcionalidad.

Reconocimiento de huellas dactilares

El reconocimiento de huellas dactilares es una de las tecnologías biométricas más antiguas, confiables y ampliamente utilizadas para la identificación y verificación de la identidad de las personas. Su eficacia se basa en una característica fundamental del cuerpo humano: cada huella dactilar es única e irreplicable, incluso entre gemelos idénticos. Además, los patrones de las huellas se mantienen prácticamente inalterables a lo largo de la vida, lo que convierte a esta tecnología en una solución ideal para procesos de control de acceso, seguridad, identificación forense y sistemas de autenticación digital.

Las huellas dactilares están formadas por una serie de crestas y valles presentes en la superficie de los dedos. Estas estructuras forman patrones reconocibles, conocidos como minucias, que pueden clasificarse en diferentes tipos según su forma: arcos, lazos y espirales, entre otros. La identificación por huellas se basa en el análisis de estas minucias, su ubicación relativa y la dirección de las crestas. Esta forma de identificación ha sido utilizada desde la antigüedad con fines legales y administrativos, pero fue formalizada científicamente en el siglo XIX por Sir Francis Galton y Henry Faulds, quienes establecieron que no existen dos huellas dactilares idénticas.

Los sistemas de reconocimiento de huellas dactilares modernos operan mediante un proceso de cuatro etapas: captura, procesamiento, almacenamiento y comparación. En la fase de captura, se utiliza un sensor especializado para escanear la huella del usuario. Este sensor puede funcionar por diversos métodos: óptico, que utiliza luz para obtener una imagen; capacitivo, que detecta variaciones eléctricas causadas por las crestas y valles; ultrasónico, que emplea ondas de sonido para crear una imagen tridimensional; o térmico, que registra el calor emitido por los dedos. Cada uno de estos métodos tiene sus ventajas en términos de precisión, costo y resistencia a manipulaciones.

Una vez capturada la imagen, el sistema procesa la información para extraer los puntos característicos (minucias) y convertirlos en un patrón digital. Este patrón no almacena la imagen completa de la huella, sino un conjunto de datos matemáticos que representan los rasgos más significativos. Este enfoque permite una comparación rápida y eficiente, además de ofrecer cierto nivel de protección a la privacidad del usuario, ya que el patrón es difícilmente reversible.

En la fase de comparación, el patrón extraído se compara con los datos almacenados previamente en una base de datos. Dependiendo del objetivo del sistema, este puede operar en modo verificación (uno a uno, confirmando si el usuario es quien dice ser) o en modo identificación (uno a muchos, buscando coincidencias en toda la base). La precisión de este proceso se mide mediante tasas como el FAR (False Acceptance Rate), que indica cuántas veces un sistema acepta una huella incorrecta, y el FRR (False Rejection Rate), que mide cuántas veces el sistema rechaza erróneamente una huella válida. Un sistema ideal busca un equilibrio entre ambos valores para maximizar la seguridad y la usabilidad.

Las aplicaciones del reconocimiento de huellas dactilares son diversas y abarcan tanto el ámbito público como privado. En la identificación forense, las huellas se utilizan para resolver crímenes, identificando sospechosos o víctimas a partir de huellas latentes encontradas en escenas del delito. En el sector financiero, se emplean para autorizar transacciones bancarias y acceder a aplicaciones móviles. En el ámbito laboral, los dispositivos de control de asistencia y acceso a instalaciones suelen basarse en esta tecnología por su eficacia y bajo costo. También es habitual en sistemas gubernamentales de identificación, como pasaportes electrónicos, cédulas digitales, programas de seguridad social y registros electorales.

En dispositivos personales como teléfonos inteligentes, laptops y tablets, el reconocimiento de huellas ha reemplazado en muchos casos a las contraseñas tradicionales, facilitando el acceso seguro y rápido a los datos personales. Esta tendencia ha impulsado el desarrollo de sensores cada vez más pequeños, rápidos y precisos, integrados directamente en las pantallas o botones de los dispositivos móviles.

Sin embargo, a pesar de sus ventajas, esta tecnología también presenta algunos desafíos y limitaciones. Uno de ellos es la posibilidad de falsificación mediante huellas artificiales hechas con materiales como silicona o gelatina. Para contrarrestar esto, muchos sensores modernos incorporan tecnologías de detección de vida, que analizan características como la temperatura, la conductividad o el pulso del dedo para asegurar que se trata de una huella real. Otro reto es la calidad de la huella capturada, que puede verse afectada por factores como humedad, suciedad, lesiones en la piel o desgaste por trabajos manuales intensivos. En estos casos, el sistema puede tener dificultades para identificar correctamente al usuario, lo que afecta la experiencia y la fiabilidad.

Desde el punto de vista legal y ético, el uso de huellas dactilares para identificación plantea interrogantes relacionados con la protección de datos personales. La huella es un dato biométrico sensible, y como tal, debe ser gestionado bajo estrictas condiciones de seguridad y confidencialidad. Las legislaciones internacionales, como el Reglamento General de Protección de Datos (GDPR) de la Unión Europea, y normativas locales en América Latina y otras regiones, establecen que el tratamiento de datos biométricos debe hacerse con consentimiento explícito, bajo una finalidad legítima, y con mecanismos de control que garanticen el respeto a los derechos del titular de los datos.

Además, es fundamental que los sistemas biométricos implementen estándares de cifrado, autenticación y auditoría, tanto en la recolección como en el almacenamiento y transmisión de los datos. En este sentido, normas técnicas como ISO/IEC 19794-2 (formato de datos biométricos para huellas dactilares) y ISO/IEC 30107 (detección de presentación o "anti-spoofing") proporcionan lineamientos técnicos para asegurar la calidad y seguridad de los sistemas.

Arduino UNO y automatización

El desarrollo tecnológico de las últimas décadas ha permitido la creación de herramientas accesibles y versátiles que facilitan la implementación de soluciones automatizadas en múltiples áreas. En este contexto, Arduino UNO se ha consolidado como una de las plataformas de hardware más utilizadas en proyectos de automatización, tanto a nivel educativo como profesional. Su estructura abierta, bajo costo y facilidad de programación han democratizado el acceso a la electrónica y al desarrollo de sistemas automatizados, lo que ha estimulado el surgimiento de una cultura global orientada al "hazlo tú mismo" (DIY, por sus siglas en inglés) y la innovación tecnológica desde abajo.

Arduino UNO es una placa de desarrollo basada en un microcontrolador ATmega328P, diseñada para facilitar la interacción con sensores, actuadores y otros componentes electrónicos. Cuenta con 14 pines digitales de entrada/salida (de los cuales 6 pueden ser usados como salidas PWM), 6 entradas analógicas, un cristal de 16 MHz, una conexión USB, un conector de alimentación y un botón de reinicio. La placa puede ser alimentada a través de una fuente externa o mediante el cable USB, lo cual la hace muy adaptable a diferentes condiciones de trabajo. La programación se realiza mediante el entorno de desarrollo Arduino IDE, que utiliza una variante del lenguaje C/C++ simplificado, lo que permite a usuarios con poca experiencia incursionar en la programación de microcontroladores.

Uno de los aspectos más relevantes de Arduino UNO en el campo de la automatización es su capacidad para interactuar en tiempo real con el entorno a través de sensores y actuadores. Esta interacción se traduce en una amplia gama de aplicaciones prácticas: sistemas de riego automático, control de iluminación, apertura y cierre de puertas, regulación de temperatura, sistemas de alarma, monitoreo ambiental, robots móviles, entre muchos otros. De este modo, Arduino se convierte en un puente entre el mundo físico y el digital, posibilitando que eventos

del entorno (como la presencia de una persona, un cambio de temperatura o la luz solar) sean procesados por el microcontrolador y transformados en respuestas automáticas.

El concepto de automatización hace referencia a la sustitución parcial o total de la intervención humana en procesos repetitivos, mediante el uso de tecnologías capaces de ejecutar acciones predefinidas con precisión, eficiencia y autonomía. Tradicionalmente, la automatización ha estado asociada a sistemas industriales complejos y costosos, sin embargo, con plataformas como Arduino, se ha extendido su implementación a espacios domésticos, educativos y experimentales. Esta democratización ha promovido el aprendizaje práctico de conceptos de electrónica, programación y diseño de sistemas embebidos, permitiendo el desarrollo de soluciones personalizadas y de bajo costo.

Una de las características clave de Arduino UNO en automatización es su capacidad para trabajar en sistemas de control en lazo cerrado. En este tipo de sistemas, un sensor recoge información del entorno (por ejemplo, un sensor de temperatura), el microcontrolador procesa esa información según el código programado, y luego actúa en consecuencia mediante un actuador (por ejemplo, encendiendo un ventilador si la temperatura supera cierto umbral). Este principio de retroalimentación permite construir sistemas inteligentes que responden dinámicamente a los cambios del entorno.

La integración de Arduino UNO con otros dispositivos y tecnologías ha ampliado aún más su potencial en automatización. A través de módulos de comunicación como Bluetooth, Wi-Fi (ESP8266 o ESP32), Ethernet o LoRa, es posible diseñar sistemas que interactúan remotamente, lo que constituye la base para proyectos de automatización domótica o Internet de las Cosas (IoT). De este modo, un sistema de control de iluminación basado en Arduino puede ser activado desde un smartphone, o un sensor de humedad puede enviar datos a la nube para su análisis posterior. Además, la compatibilidad con servicios como ThingSpeak, Blynk o Firebase permite el desarrollo de aplicaciones más complejas que combinan automatización, análisis de datos y visualización en tiempo real.

Desde el punto de vista educativo, Arduino UNO ha transformado la enseñanza de las ciencias aplicadas, al permitir que estudiantes de niveles secundarios y universitarios experimenten directamente con conceptos abstractos como programación estructurada, lógica digital, circuitos eléctricos y control automático. Su amplia comunidad de usuarios, foros, tutoriales y bibliotecas compartidas facilita el aprendizaje colaborativo y acelera el desarrollo de proyectos funcionales.

No obstante, Arduino también presenta limitaciones técnicas que deben ser consideradas al momento de diseñar sistemas de automatización más exigentes. Su capacidad de procesamiento, memoria y almacenamiento es limitada frente a otras plataformas como Raspberry Pi o microcontroladores industriales más robustos. Asimismo, carece de un sistema operativo, lo que significa que solo puede ejecutar una tarea principal a la vez (aunque es posible simular multitarea con

técnicas como la temporización). Por ello, en sistemas críticos o de gran escala, Arduino puede utilizarse como nodo periférico o parte de una arquitectura distribuida donde se encargue de tareas específicas de bajo nivel.

A pesar de estas limitaciones, su bajo costo, facilidad de uso y flexibilidad lo convierten en una herramienta excepcional para prototipos, sistemas automatizados pequeños y soluciones personalizadas. Gracias al modelo de código abierto, tanto la placa como su entorno de desarrollo pueden ser modificados, lo que ha impulsado el surgimiento de múltiples versiones compatibles, adaptadas a diferentes necesidades (como Arduino Mega, Nano, Leonardo, entre otras).

Sensor biométrico AS608

El avance de la biometría como herramienta para la identificación y autenticación de personas ha impulsado el desarrollo de sensores especializados que integran funcionalidades avanzadas en formatos compactos y accesibles. Uno de los dispositivos más populares dentro de esta categoría es el sensor biométrico AS608, un módulo de reconocimiento de huellas dactilares ampliamente utilizado en proyectos de electrónica, automatización, seguridad y control de acceso, debido a su bajo costo, facilidad de integración con plataformas como Arduino y Raspberry Pi, y su rendimiento confiable en contextos no industriales.

El AS608 es un módulo biométrico basado en un sensor óptico que capta imágenes de la huella dactilar a través de una superficie acrílica. El sensor escanea la huella mediante una cámara interna con iluminación LED, que captura los patrones de crestas y valles característicos de cada dedo. Estos patrones se convierten en una imagen digital que es procesada por un chip DSP (Digital Signal Processor) integrado en el módulo. Este chip no solo se encarga del procesamiento de la imagen, sino también de la extracción de características, la creación del modelo biométrico y su almacenamiento en una memoria interna flash, que puede albergar hasta 162 registros de huellas en promedio.

Uno de los aspectos más destacables del AS608 es que funciona como un sistema biométrico autónomo: no requiere que un microcontrolador externo realice el procesamiento de los datos de huella, ya que el módulo incluye su propio sistema de reconocimiento. Esto simplifica enormemente su integración con otros sistemas, ya que el microcontrolador únicamente necesita enviar comandos a través de una interfaz serial UART (TTL) para registrar, borrar, verificar o identificar huellas.

El proceso básico de funcionamiento del AS608 puede dividirse en tres etapas principales:

1. Inscripción de huella: cuando una nueva huella es registrada, el sensor toma dos imágenes consecutivas del dedo y las compara para verificar que ambas pertenezcan al mismo patrón biométrico. Si ambas coinciden, se crea un modelo digital de la huella y se guarda en la memoria del sensor con un ID asignado.

2. Verificación (1:1): el sistema compara la huella escaneada en tiempo real con una huella específica previamente registrada. Si ambas coinciden, la verificación es exitosa. Este modo es útil cuando se conoce de antemano qué ID verificar.
3. Identificación (1:N): el sensor compara la huella escaneada con todas las almacenadas en su base de datos interna y determina si hay alguna coincidencia. Esta función es útil para identificar a un usuario sin requerir un ID previo.

El módulo AS608 opera con una tensión de alimentación de 3.3V y se comunica a través de UART con una velocidad de transmisión configurable (por defecto 57600 bps). Su consumo energético es bajo, lo cual lo hace ideal para sistemas portátiles o alimentados por batería. Además, su tamaño compacto (aproximadamente 35 mm x 20 mm) facilita su incorporación en carcasas o estructuras personalizadas.

En términos de aplicaciones, el AS608 ha demostrado ser altamente versátil. Se utiliza comúnmente en:

- Sistemas de control de acceso, como cerraduras electrónicas para puertas, casilleros, oficinas o cajas de seguridad.
- Proyectos de control de asistencia laboral o estudiantil, donde se requiere identificar rápidamente a los usuarios sin contacto físico.
- Dispositivos portátiles o IoT, donde el reconocimiento biométrico reemplaza al uso de contraseñas o llaves físicas.
- Sistemas embebidos de seguridad personalizada, como maletines inteligentes, armarios biométricos, o automóviles con arranque biométrico.

Una de las principales ventajas del AS608 es su compatibilidad con plataformas abiertas como Arduino, mediante bibliotecas de código ya desarrolladas (por ejemplo, “Adafruit Fingerprint Sensor Library” o librerías específicas en GitHub). Esta compatibilidad ha permitido su difusión masiva entre estudiantes, desarrolladores y makers, quienes lo utilizan en proyectos educativos, prototipos funcionales e incluso soluciones comerciales de pequeña escala.

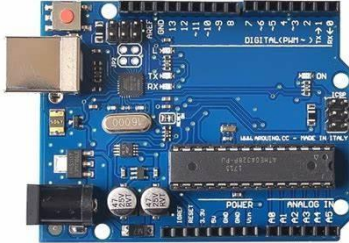
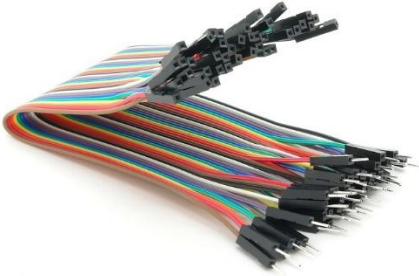
En el contexto académico y educativo, el AS608 permite ilustrar de forma práctica el funcionamiento de los sistemas biométricos, enseñando conceptos como el escaneo de huellas, el modelado de características, la autenticación y la gestión de una base de datos local. Su uso también estimula la reflexión crítica sobre temas como la privacidad, la seguridad de los datos biométricos y la ética en el reconocimiento de identidad.

No obstante, el sensor presenta algunas limitaciones técnicas que deben considerarse. Por ejemplo, la calidad de reconocimiento puede verse afectada por condiciones de luz, suciedad en el dedo o en la superficie del sensor, huellas dañadas, o manos húmedas o secas. Asimismo, la capacidad de almacenamiento está limitada a un número relativamente bajo de huellas, lo que puede no ser adecuado para aplicaciones con muchos usuarios. En términos de seguridad, aunque el AS608 cuenta con algoritmos internos para evitar falsificaciones básicas, no es inmune a ataques sofisticados como huellas artificiales o ataques de replay si no se implementan medidas de seguridad adicionales a nivel del sistema host.

Para mitigar estas limitaciones, algunos desarrollos combinan el AS608 con otras tecnologías como lectores RFID, códigos PIN, o sistemas de doble autenticación, mejorando así la robustez del sistema en entornos más exigentes. También es posible integrar el módulo a sistemas conectados a la nube, lo que permite almacenar registros de acceso, crear interfaces gráficas y realizar análisis de uso mediante plataformas IoT.

Desarrollo

Materiales

Arduino UNO		Microcontrolador con conectividad Wi-Fi y Bluetooth, ideal para proyectos IoT.
Sensor de huellas AS608		Sensor óptico para captura y verificación de huellas digitales.
Cables Dupont		Cables para realizar conexiones entre componentes en la protoboard.
Cable USB Micro USB		Cable para alimentar y programar el Arduino UNO desde una computadora.

Diagramas de conexión

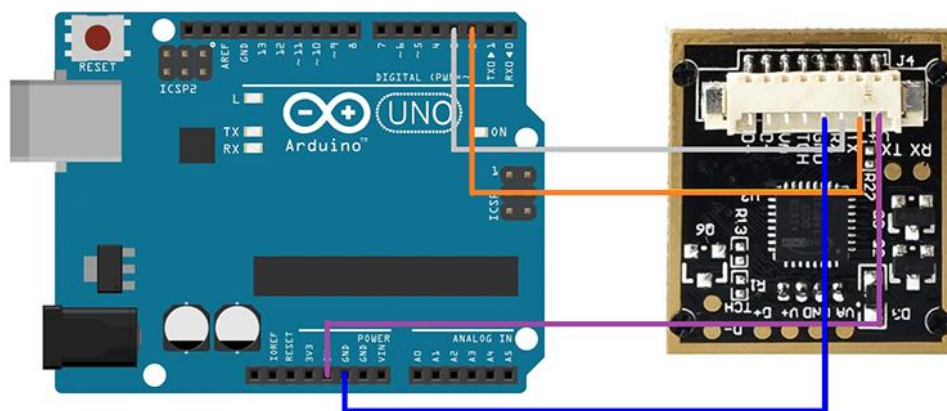


Figura3. Diagrama de conexión

Instalación de Requerimientos

Los frameworks de desarrollo constituyen plataformas integradas por herramientas y bibliotecas que ofrecen una estructura estandarizada para la creación de software. Su implementación requiere el cumplimiento de determinados requisitos técnicos que permiten atender adecuadamente las demandas del sistema en desarrollo. Entre los aspectos más relevantes se encuentran la disponibilidad de un sistema de gestión de bases de datos, el uso del framework establecido por la organización, y un gestor de dependencias compatible con el lenguaje de programación PHP, entre otros componentes fundamentales.

A continuación, mostraré la instalación de los frameworks usados para este proyecto:

- Arduino IDE

Primero se hace la descarga en la página oficial de Arduino:

<https://www.arduino.cc/en/software>

Y se instala la versión mas reciente, y adecuada para nuestro sistema.

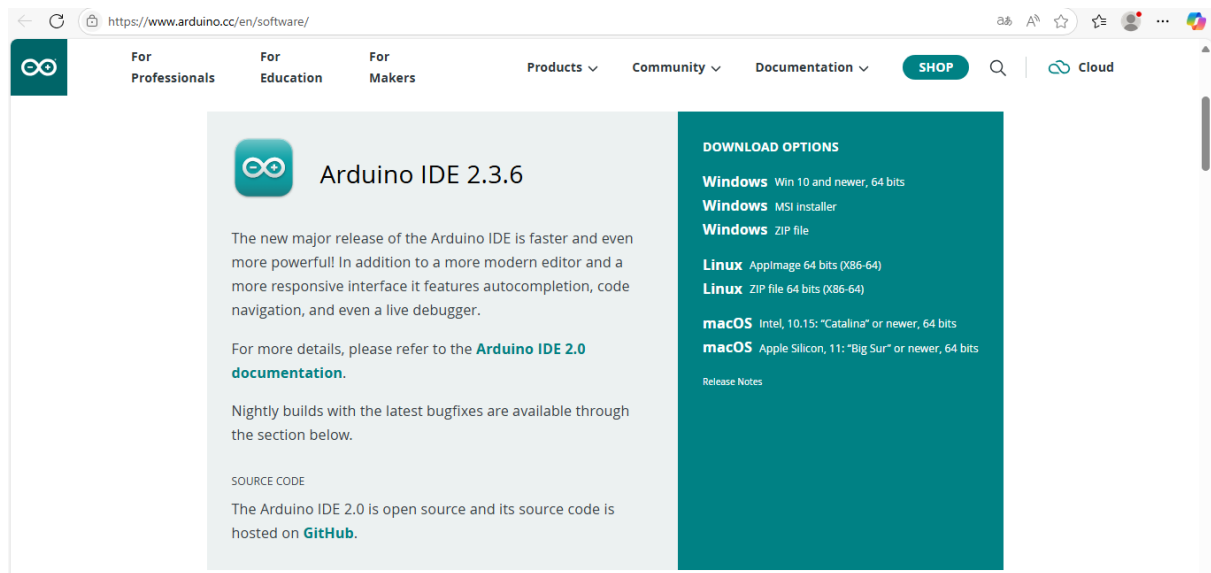


Figura4. Página oficial de Arduino

Una vez hecha la descarga, se ejecuta el archivo descargado, y debemos seguir los pasos que se van mostrando para que la instalación sea exitosa.

Una vez hecha la instalación, instalación el soporte para ESP32

Para esto, abrimos Arduino IDE, y de ahí nos dirigiremos a la opción de Archivo, la cual esta ubicada en la parte superior izquierda de la pantalla, y entonces seleccionamos Preferencias.

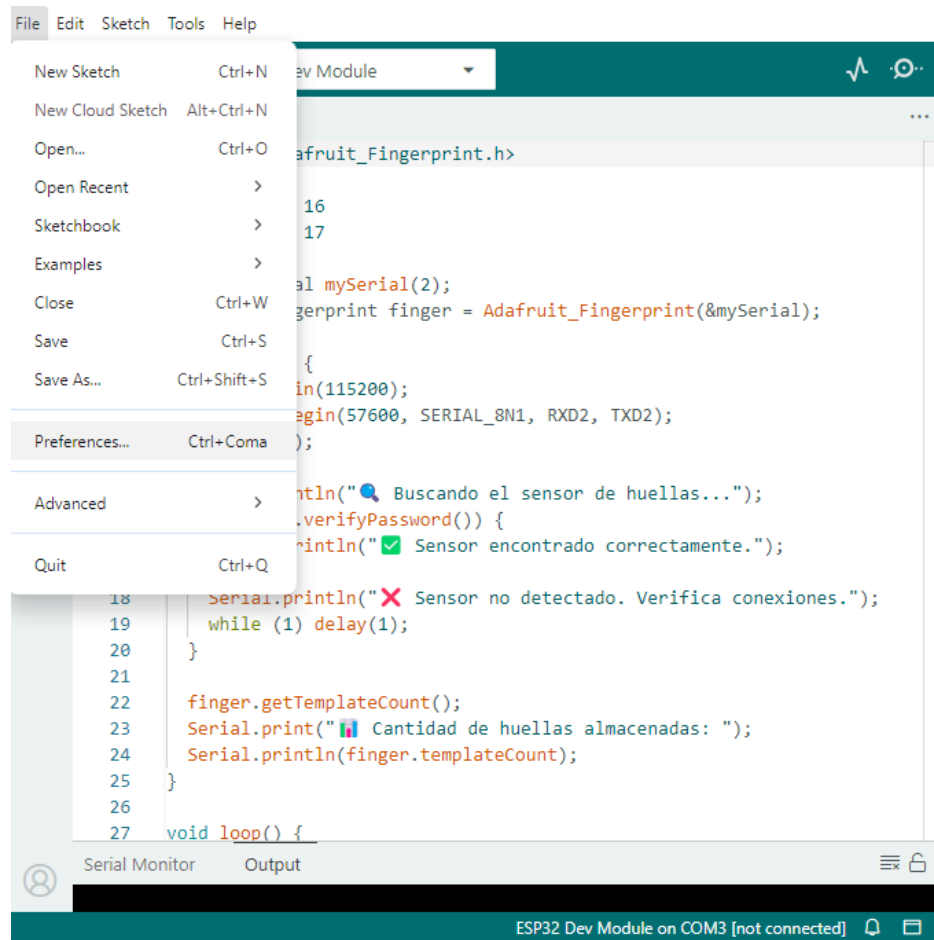


Figura4. Muestra de Archivo>Preferencias.

Después seleccionamos Preferencias, y nos abrirá una ventana y en donde dice “Gestor de URLs adicionales de tarjetas”, agregaremos este link https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

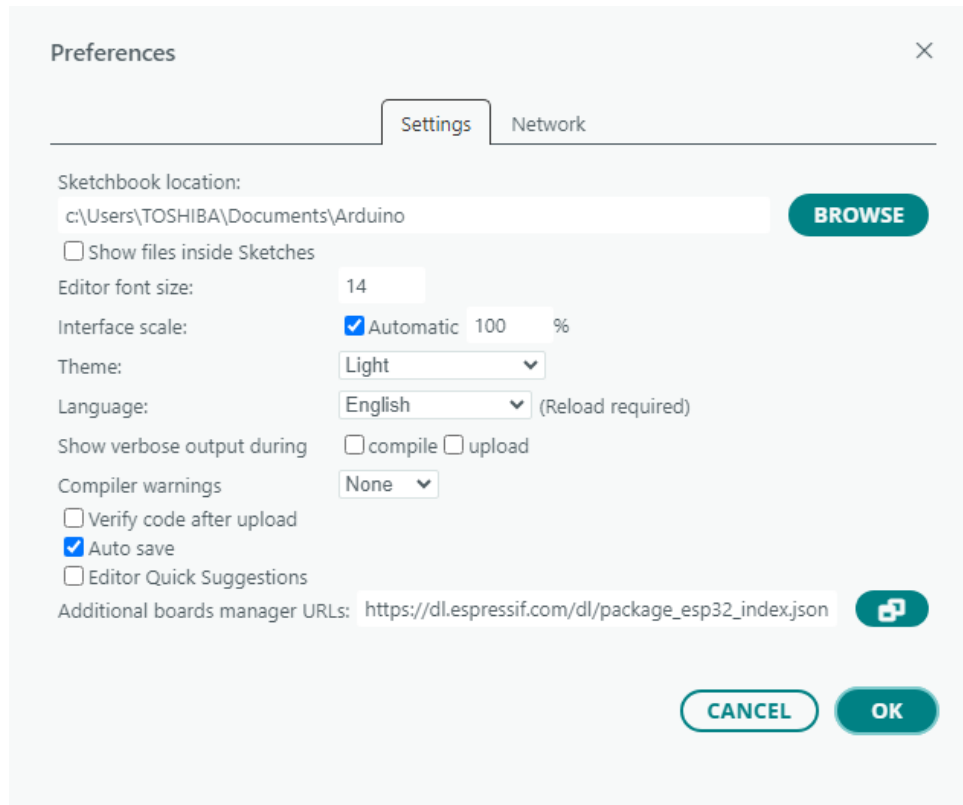


Figura5. Insertar el link

Después iremos al apartado de herramientas, donde seleccionaremos placa y el gestor de tarjetas.

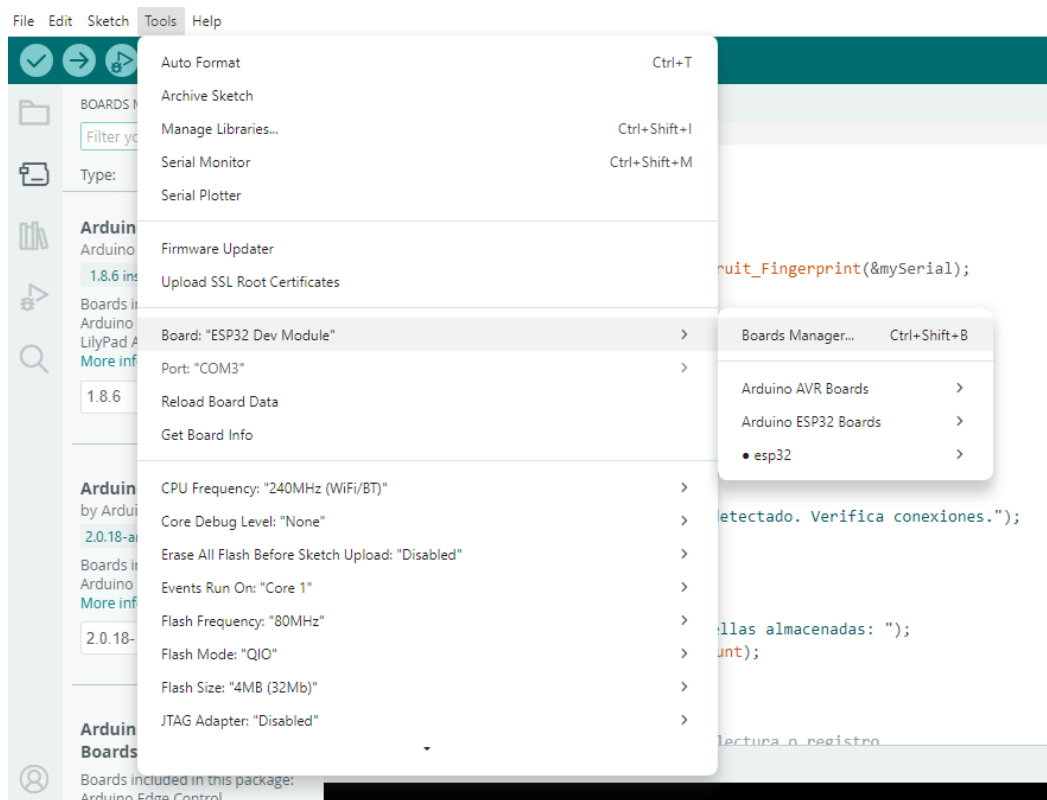


Figura6. Muestra de Herramientas

Una vez realizado esto, seleccionamos el gestor de placas y buscamos ESP32, y hacemos la instalación.

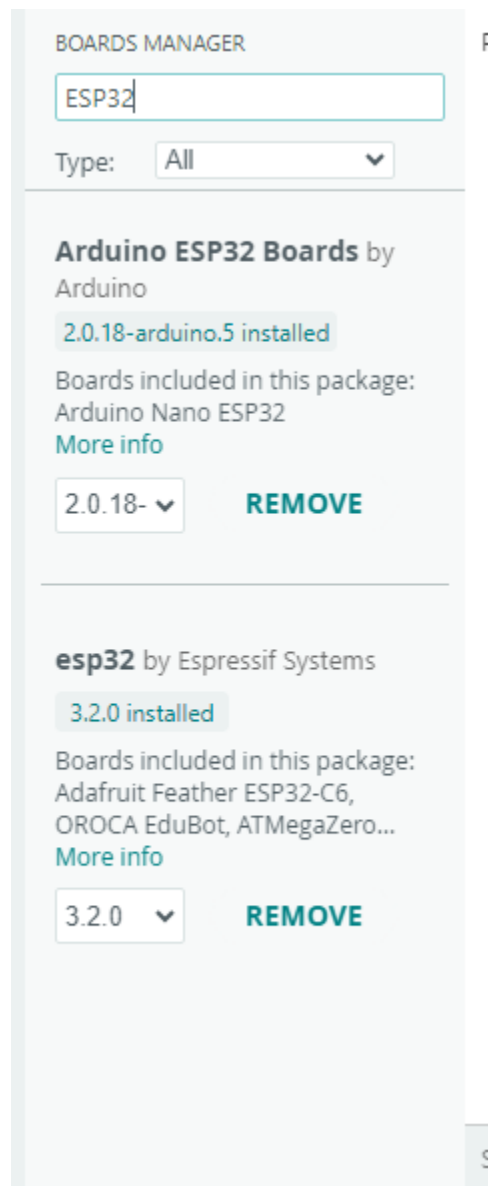


Figura7. Instalación de ESP32

Ya con esto, podemos ver que nuestra placa será encontrada.

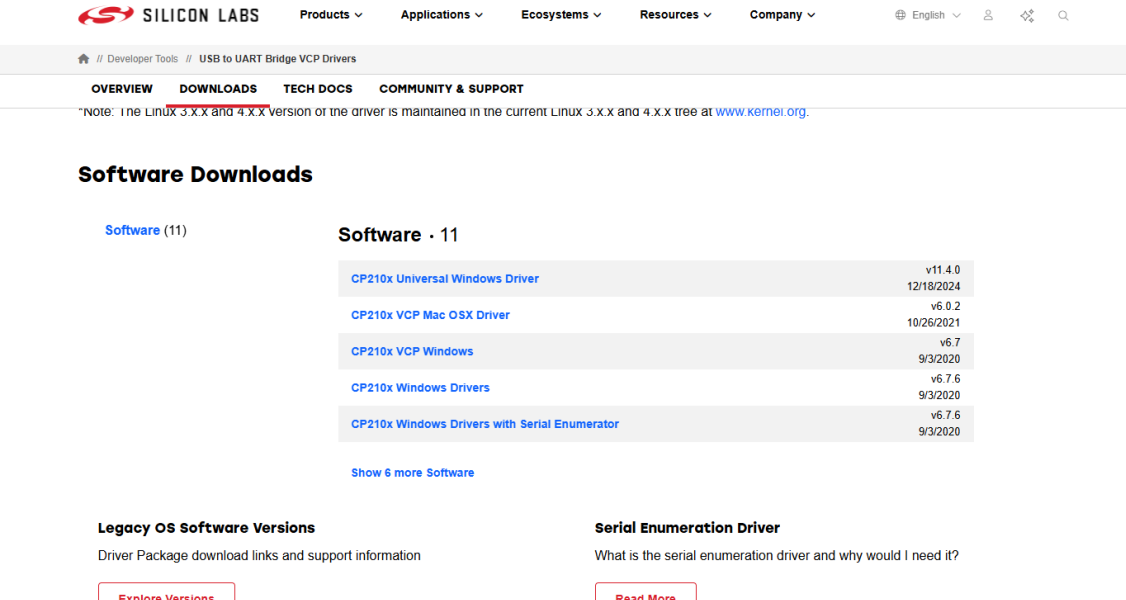
En caso de que no nos aparezca, debemos hacer la instalación de Controladores.

Instalación de Controladores

En esta sección, se hizo la instalación del controlador puesto que la mayoría de placas ESP32 no se conectan directamente al puerto USB de una computadora. En su lugar, se usa un chip puente (como el CP2102) que convierte la señal USB a UART (serial).

Para este proyecto, el controlador usado fue CP2102, pues es el que se vincula mejor con nuestro modelo de ESP32.

Primero, Ingresar a la página oficial de Silicon Labs:
<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>.



The screenshot displays the official Silicon Labs website for USB-to-UART Bridge VCP Drivers. The page includes a navigation bar with links to Products, Applications, Ecosystems, Resources, and Company. Below the navigation bar, there is a breadcrumb trail: Home // Developer Tools // USB to UART Bridge VCP Drivers. The main content area is titled "Software Downloads" and lists 11 software items. The first five items are displayed in a table format, showing the driver name, version, and release date. Below the table, there is a link to "Show 6 more Software". At the bottom of the page, there are two sections: "Legacy OS Software Versions" and "Serial Enumeration Driver".

Software	Version	Release Date
CP210x Universal Windows Driver	v11.4.0	12/18/2024
CP210x VCP Mac OSX Driver	v6.0.2	10/26/2021
CP210x VCP Windows	v6.7	9/3/2020
CP210x Windows Drivers	v6.7.6	9/3/2020
CP210x Windows Drivers with Serial Enumerator	v6.7.6	9/3/2020

[Show 6 more Software](#)

Legacy OS Software Versions
Driver Package download links and support information
[Explore Versions](#)

Serial Enumeration Driver
What is the serial enumeration driver and why would I need it?
[Read More](#)

Figura8. Muestra de la pagina oficial

Aquí seleccionaremos, el Universal Windows Driver, puesto esta es muy funcional. Una vez descargado el ZIP, se debe de extraer y ejecutamos CP210xVCPInstaller_x64.exe.

Una vez realizado está parte, volvemos a conectar el ESP32 por el puerto USB, debemos revisar que ya lo reconoce.

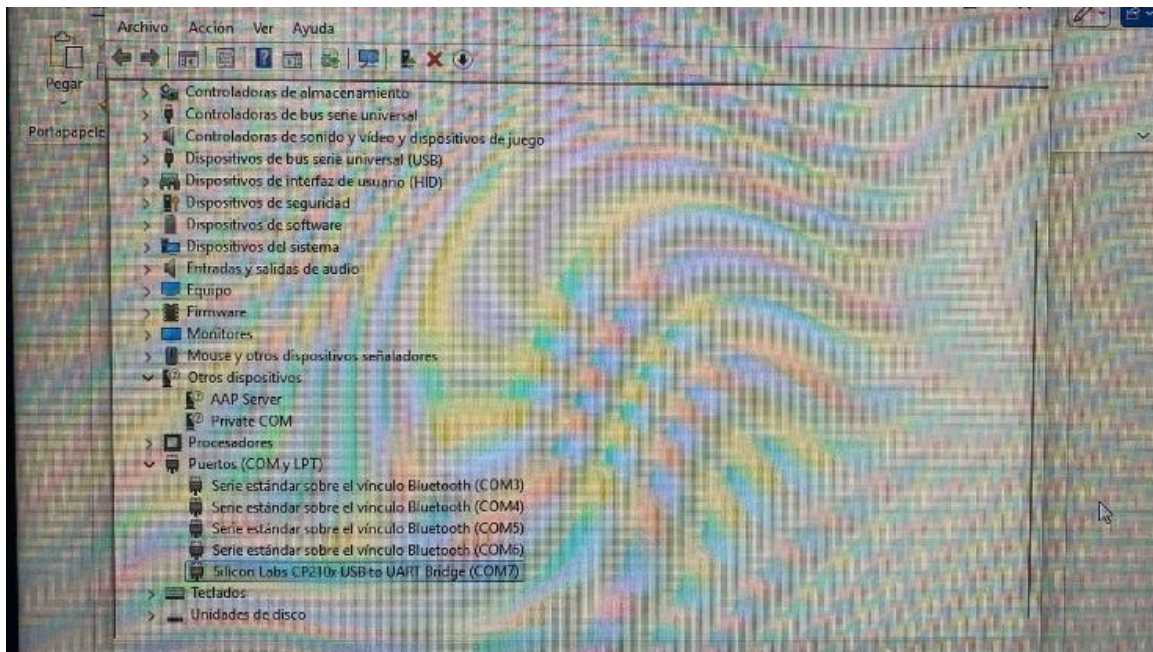


Figura9. Pantalla para ver controladores.

Ahora abrimos Arduino IDE y agregamos el COM como lo muestre en los puertos, en mi caso lo marca como COM7 y se seleccionó el ESP32 a usar.

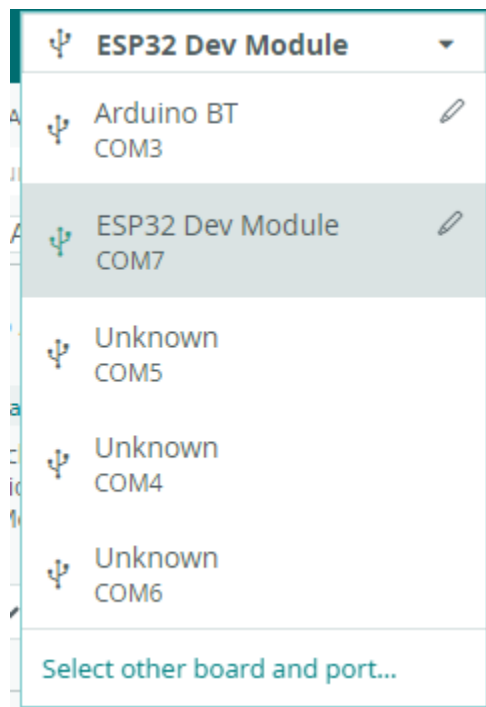


Figura10. Ya funciona correctamente.

Inclusión de Librerías

Primero, en el apartado de Arduino, seleccionamos programa, y luego el gestor de librerías, y se selecciona el buscar, esto nos redirigirá a una barra para hacer la búsqueda de la librería que se necesite, en este caso es Adafruit Fingerprint Sensor Library.

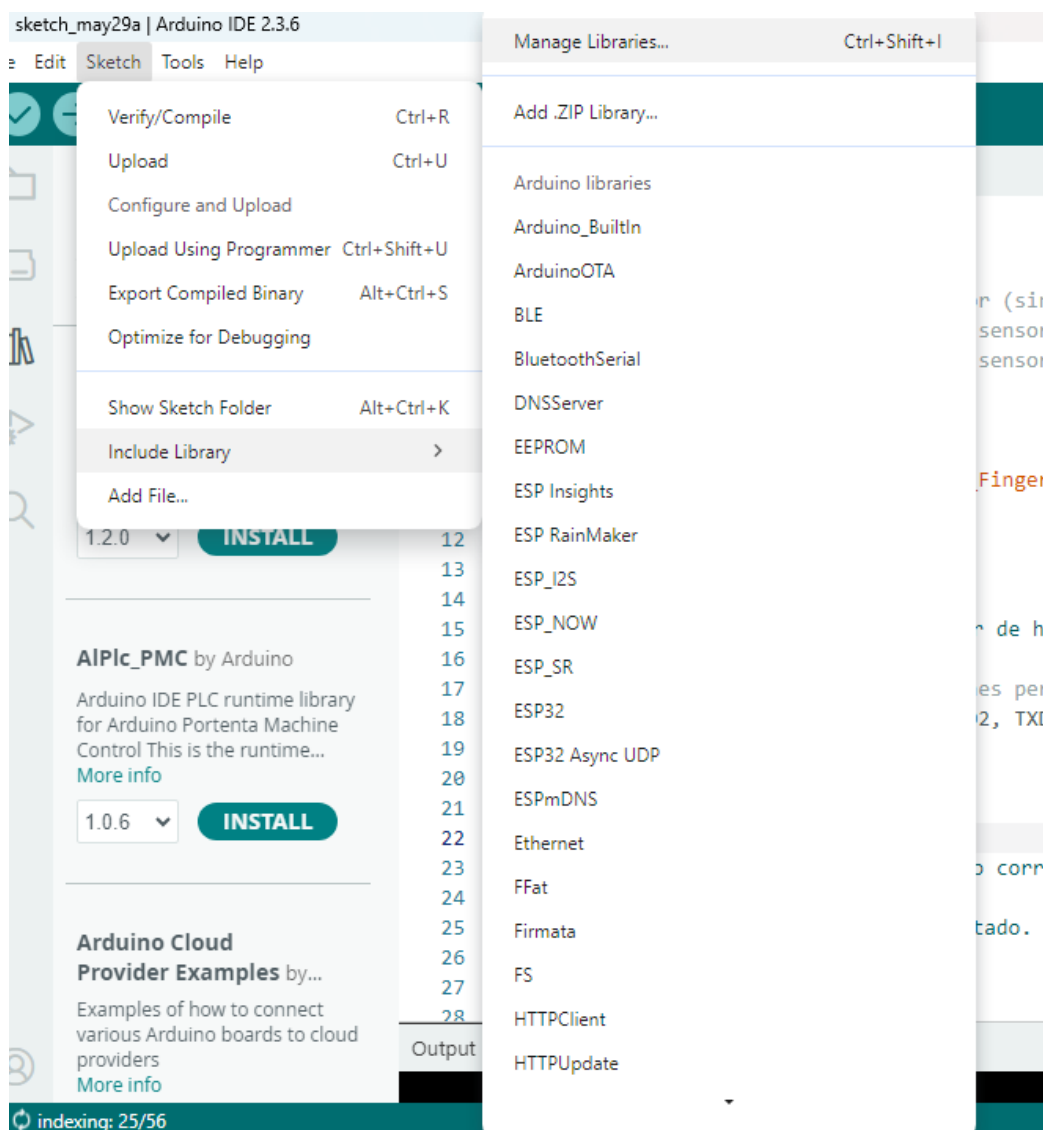


Figura12. Descripción para abrir el buscador de librerías.

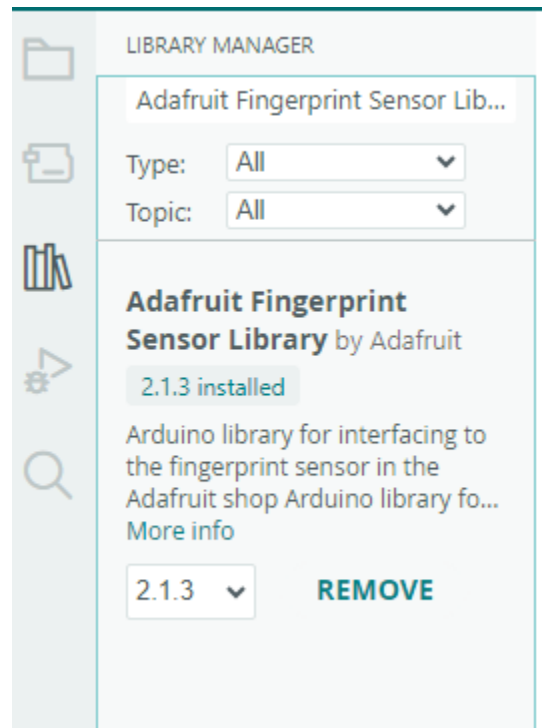


Figura13. Buscador e instalador de la librería

Una vez hecho esto, se verifica si está incluida correctamente. Primero seleccione la opción de archivo en Arduino IDE, después seleccionamos donde dice ejemplo, para buscar la librería que se usará y por ultimo seleccionamos el enroll.

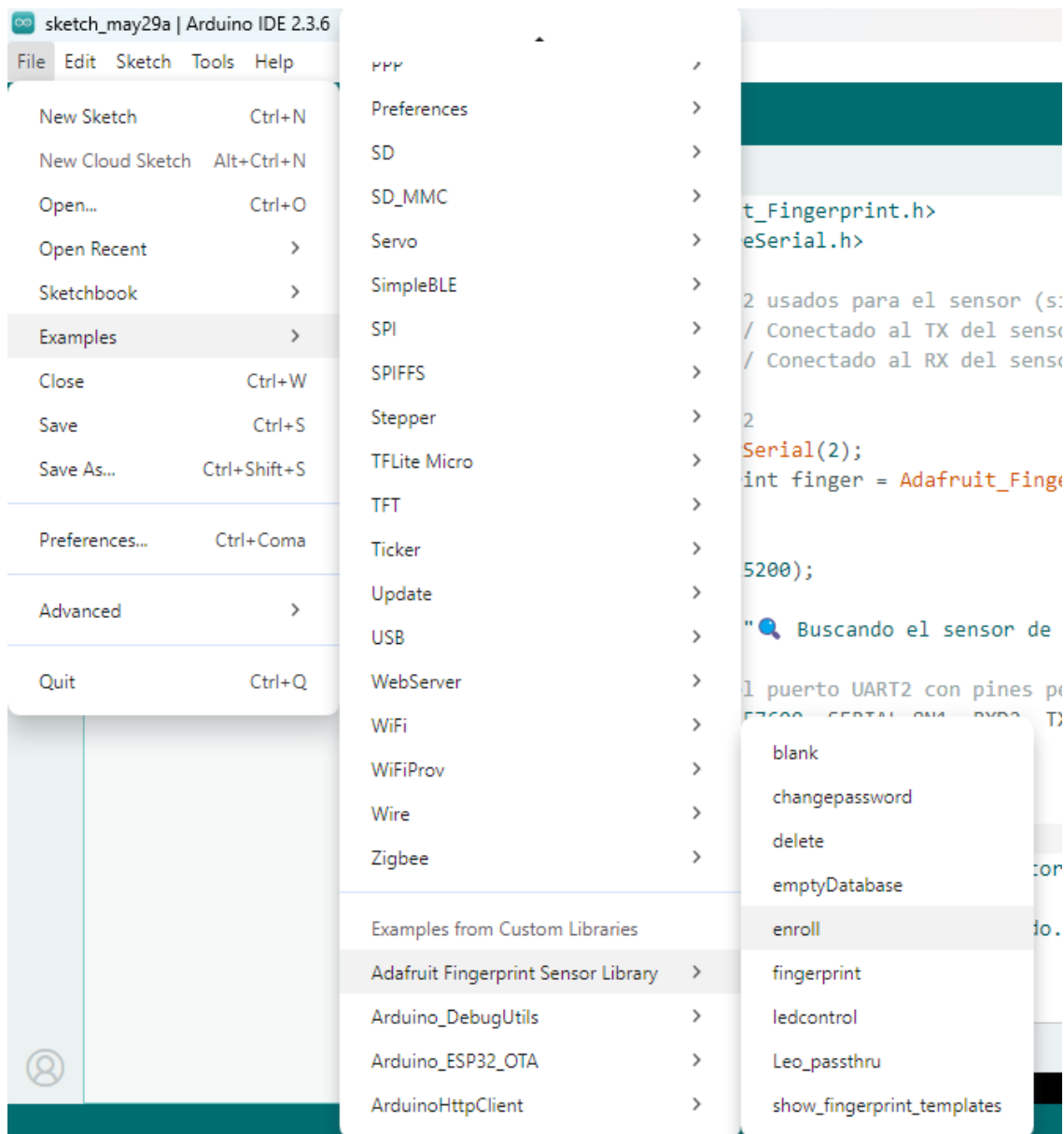


Figura14. Paso a paso de verificación exitosa de librería.

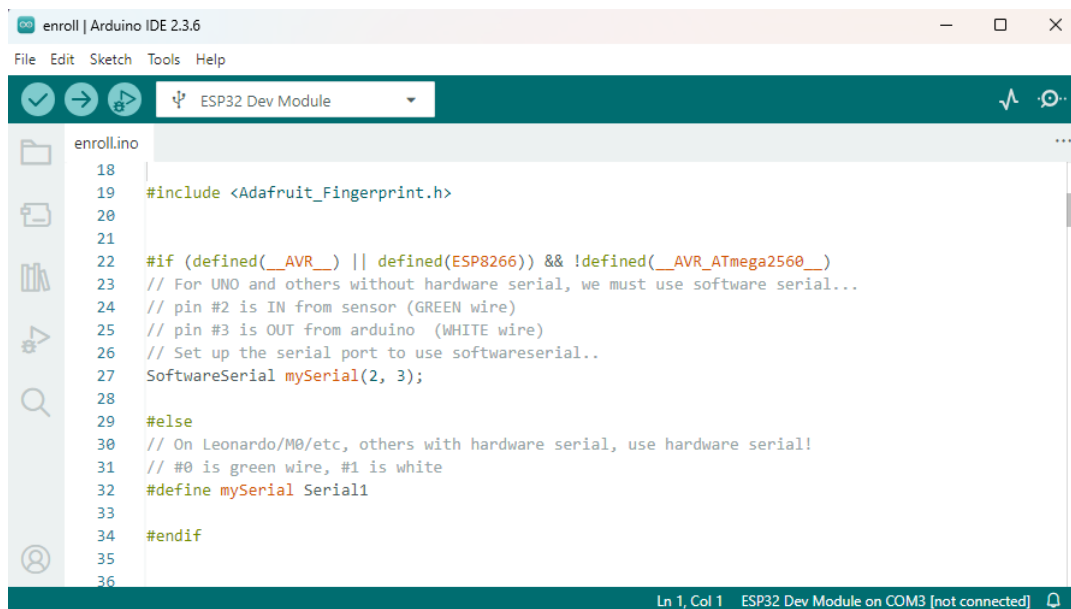


Figura15. Nuevo sketch

Aquí podemos observar que nos abre un nuevo sketch con el código de ejemplo para que el código funcione correctamente.

También se incluyó la librería de “RTCLib”

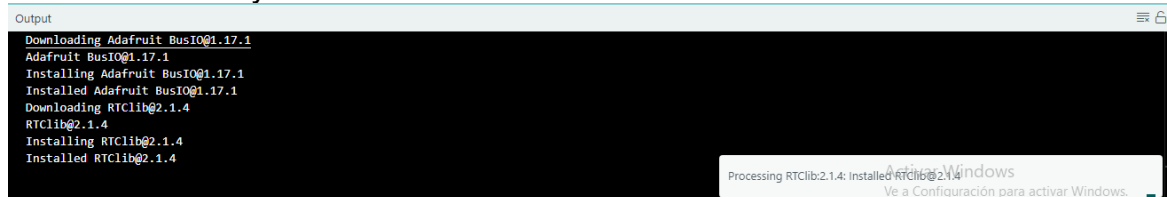


Figura16. Instalación de librería

Pruebas de Funcionamiento

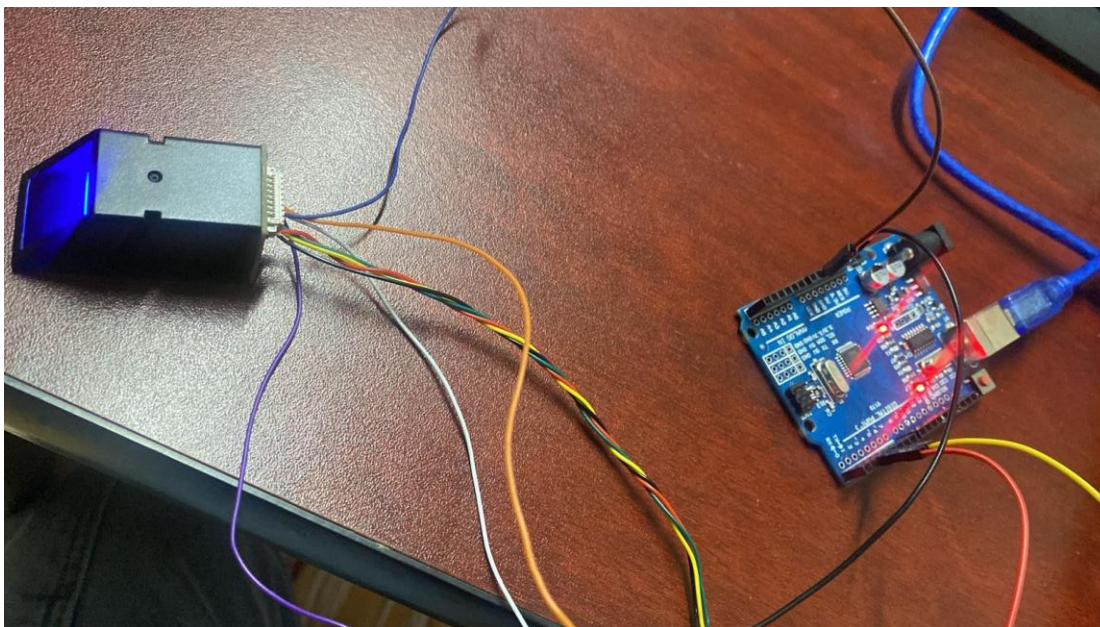


Figura17. Conexión física.

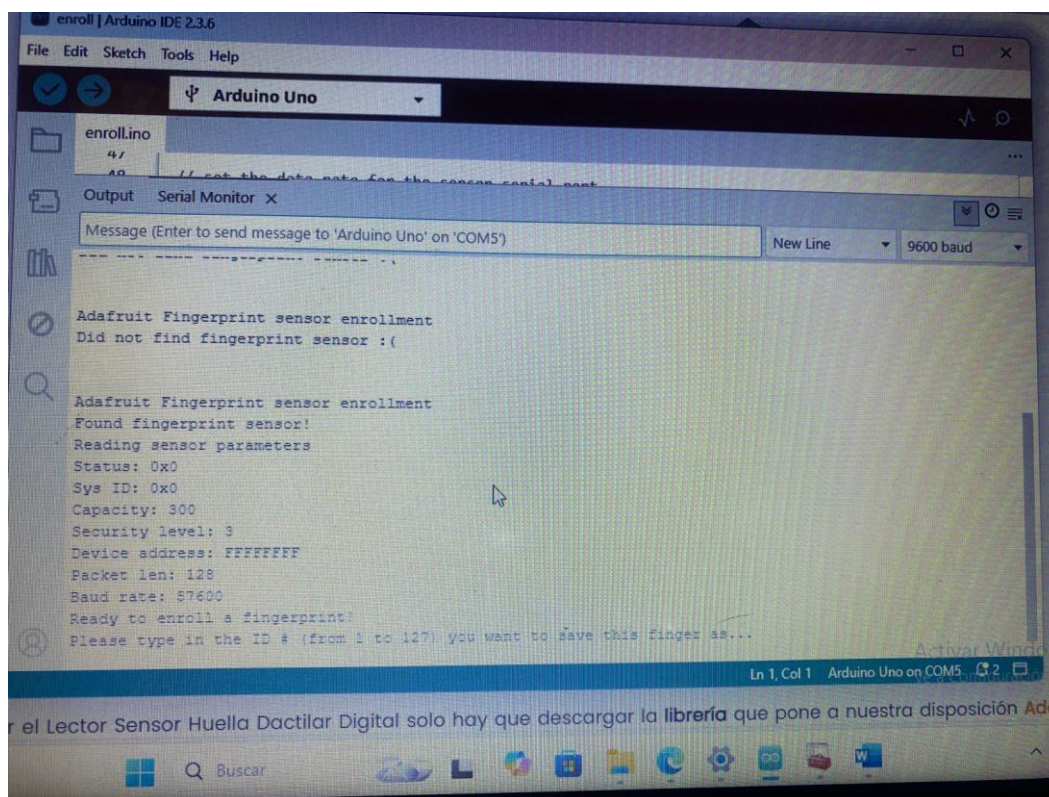


Figura18. El ejemplo explicado antes funciona correctamente

Pruebas y optimización

- Lee y registra huella

```
#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>

// Sensor de huellas en pines 2 (RX) y 3 (TX) del Arduino UNO
SoftwareSerial fingerSerial(2, 3);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&fingerSerial);

uint8_t id;

void setup() {
  Serial.begin(9600);
  while (!Serial); // Espera monitor serial
  delay(100);
  Serial.println("👋 Sistema de registro de huellas listo.");

  finger.begin(57600);

  if (finger.verifyPassword()) {
    Serial.println("✅ Sensor de huellas detectado.");
  } else {
    Serial.println("❌ Sensor no detectado. Revisa conexiones.");
    while (1);
  }

  finger.getTemplateCount();
  Serial.print("📁 Huellas almacenadas: ");
  Serial.println(finger.templateCount);
}

void loop() {
  Serial.println("🆔 Escribe el ID para la nueva huella (1-127):");
  id = readNumber();
  if (id == 0) return;

  Serial.print("🔗 Registrando huella con ID ");
  Serial.println(id);

  if (enrollFingerprint(id) == FINGERPRINT_OK) {
    Serial.println("✅ Huella guardada correctamente en el sensor.");
  }
}
```

```

    } else {
        Serial.println("❌ Fallo al guardar la huella.");
    }

    delay(2000);
}

uint8_t readNumber() {
    while (!Serial.available());
    return Serial.parseInt();
}

uint8_t enrollFingerprint(uint8_t id) {
    int p = -1;
    Serial.println("👉 Coloca el dedo en el sensor.");

    while (p != FINGERPRINT_OK) {
        p = finger.getImage();
        if (p == FINGERPRINT_NOFINGER) {
            Serial.print(".");
            delay(100);
        }
    }

    p = finger.image2Tz(1);
    if (p != FINGERPRINT_OK) return p;

    Serial.println("\n👈 Retira el dedo.");
    delay(2000);
    while (finger.getImage() != FINGERPRINT_NOFINGER);

    Serial.println("👉 Coloca el mismo dedo otra vez.");
    while (finger.getImage() != FINGERPRINT_OK);
    p = finger.image2Tz(2);
    if (p != FINGERPRINT_OK) return p;

    p = finger.createModel();
    if (p != FINGERPRINT_OK) return p;

    p = finger.storeModel(id);
    return p;
}

```

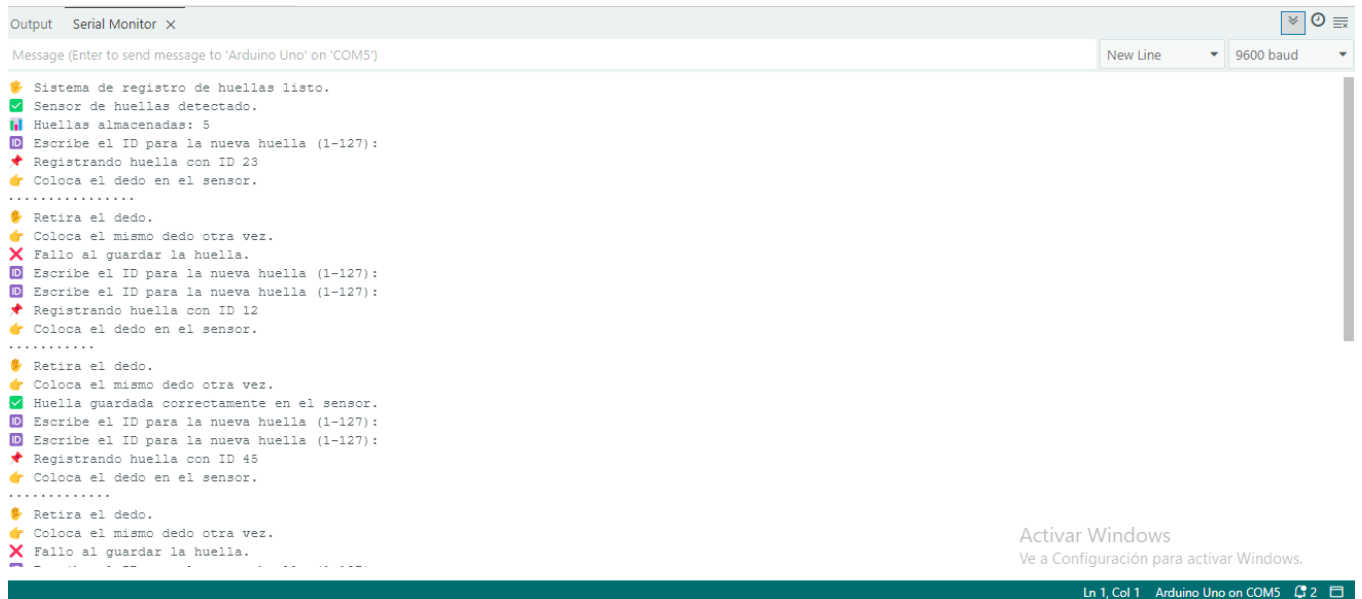


Figura19. Funcionamiento correcto

- Marca un menú, y nos dice si registramos huella o solo hace verificación

```
#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>
```

```
SoftwareSerial fingerSerial(2, 3); // RX, TX
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&fingerSerial);

void setup() {
  Serial.begin(9600);
  while (!Serial);
  delay(100);

  Serial.println("☰ Sistema de Registro y Asistencia con Huella");

  finger.begin(57600);
  if (finger.verifyPassword()) {
    Serial.println("☑ Sensor conectado correctamente.");
  } else {
    Serial.println("✗ Error: sensor no detectado.");
    while (1);
  }

  finger.getTemplateCount();
  Serial.print("☰ Huellas registradas: ");
  Serial.println(finger.templateCount);
}
```

```

    mostrarMenu();
}

void loop() {
    if (Serial.available()) {
        String opcion = Serial.readStringUntil('\n');
        opcion.trim();

        if (opcion == "1") {
            registrarHuella();
        } else if (opcion == "2") {
            verificarAsistencia();
        } else {
            Serial.println("❌ Opción inválida. Intenta de nuevo.");
        }

        delay(1000);
        mostrarMenu();
    }
}

void mostrarMenu() {
    Serial.println("\n📋 MENÚ:");
    Serial.println("1. Registrar nueva huella");
    Serial.println("2. Verificar asistencia");
    Serial.print("👉 Elige una opción: ");
}

void registrarHuella() {
    Serial.println("🆔 Ingresa un ID (1-127):");
    int id = leerNumero();

    if (id <= 0 || id > 127) {
        Serial.println("❌ ID inválido.");
        return;
    }

    Serial.print("👉 Registrando huella con ID ");
    Serial.println(id);

    if (enrollFingerprint(id) == FINGERPRINT_OK) {
        Serial.println("✅ Huella registrada correctamente.");
    } else {
        Serial.println("❌ Fallo al registrar la huella.");
    }
}

```

```

}

uint8_t enrollFingerprint(uint8_t id) {
    int p = -1;
    Serial.println("👉 Coloca el dedo en el sensor.");

    while (p != FINGERPRINT_OK) {
        p = finger.getImage();
        if (p == FINGERPRINT_NOFINGER) {
            Serial.print(".");
            delay(100);
        }
    }

    p = finger.image2Tz(1);
    if (p != FINGERPRINT_OK) return p;

    Serial.println("\n👉 Retira el dedo.");
    delay(2000);
    while (finger.getImage() != FINGERPRINT_NOFINGER);

    Serial.println("👉 Coloca el mismo dedo otra vez.");
    while (finger.getImage() != FINGERPRINT_OK);
    p = finger.image2Tz(2);
    if (p != FINGERPRINT_OK) return p;

    p = finger.createModel();
    if (p != FINGERPRINT_OK) return p;

    p = finger.storeModel(id);
    return p;
}

void verificarAsistencia() {
    Serial.println("👉 Coloca tu dedo para verificar...");

    while (finger.getImage() != FINGERPRINT_OK);

    if (finger.image2Tz() != FINGERPRINT_OK) {
        Serial.println("❌ Error al leer huella.");
        return;
    }

    int resultado = finger.fingerSearch();

```

```

if (resultado == FINGERPRINT_OK) {
    Serial.print("✅ Huella reconocida. ID: ");
    Serial.println(finger.fingerID);
    Serial.println("📋 Asistencia registrada.");
} else {
    Serial.println("❌ Huella no registrada.");
}
}

int leerNumero() {
    while (!Serial.available());
    return Serial.parseInt();
}

```

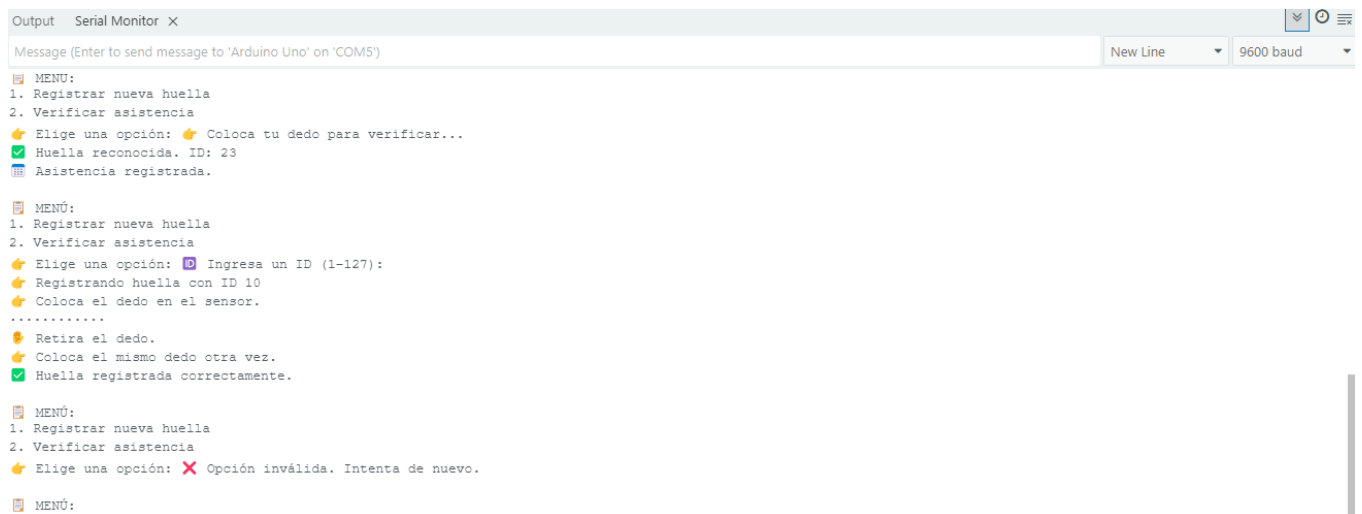


Figura20. Funciona correctamente

- Lectura de huella sin menú.

```

#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>

```

```

SoftwareSerial fingerSerial(2, 3); // RX, TX
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&fingerSerial);

```

```
int nextID = 1; // ID para nuevas huellas
```

```

void setup() {
    Serial.begin(9600);
    while (!Serial); delay(100);

```

```
Serial.println("👋 Sistema de Asistencia con Huella");
```

```

finger.begin(57600);
if (finger.verifyPassword()) {
  Serial.println("✅ Sensor conectado correctamente.");
} else {
  Serial.println("❌ Sensor no detectado.");
  while (1);
}

finger.getTemplateCount();
Serial.print("📁 Huellas registradas: ");
Serial.println(finger.templateCount);
}

void loop() {
  Serial.println("\n👉 Coloca el dedo para verificar asistencia...");
  while (finger.getImage() != FINGERPRINT_OK);

  if (finger.image2Tz() != FINGERPRINT_OK) {
    Serial.println("⚠️ Error al convertir imagen");
    return;
  }

  if (finger.fingerSearch() == FINGERPRINT_OK) {
    Serial.print("✅ Huella encontrada. ID: ");
    Serial.println(finger.fingerID);
    Serial.println("🔴 Asistencia registrada.");
    delay(2000);
  } else {
    Serial.println("❌ Huella no registrada.");
    Serial.println("¿Deseas registrar esta huella? (s/n)");

    unsigned long start = millis();
    while (!Serial.available() && (millis() - start < 10000)); // Espera hasta 10s
    if (Serial.available()) {
      char respuesta = Serial.read();
      if (respuesta == 's' || respuesta == 'S') {
        registrarHuella();
      } else {
        Serial.println("🛑 Registro cancelado.");
      }
    } else {
      Serial.println("⌚ Tiempo de respuesta agotado.");
    }
  }
}

```

```

}

void registrarHuella() {
  if (nextID >= 127) {
    Serial.println("🚫 Límite de huellas alcanzado.");
    return;
  }

  Serial.println("👉 Registrar nueva huella. Coloca el dedo...");
  while (finger.getImage() != FINGERPRINT_OK);
  if (finger.image2Tz(1) != FINGERPRINT_OK) {
    Serial.println("❌ Error en la primera lectura.");
    return;
  }

  Serial.println("👉 Retira el dedo.");
  delay(2000);
  Serial.println("👉 Coloca el mismo dedo nuevamente...");
  while (finger.getImage() != FINGERPRINT_OK);
  if (finger.image2Tz(2) != FINGERPRINT_OK) {
    Serial.println("❌ Error en la segunda lectura.");
    return;
  }

  if (finger.createModel() != FINGERPRINT_OK) {
    Serial.println("⚠️ No se pudo crear el modelo.");
    return;
  }

  if (finger.storeModel(nextID) == FINGERPRINT_OK) {
    Serial.print("✅ Huella registrada con ID ");
    Serial.println(nextID);
    nextID++;
  } else {
    Serial.println("❌ Fallo al guardar la huella.");
  }
}

```

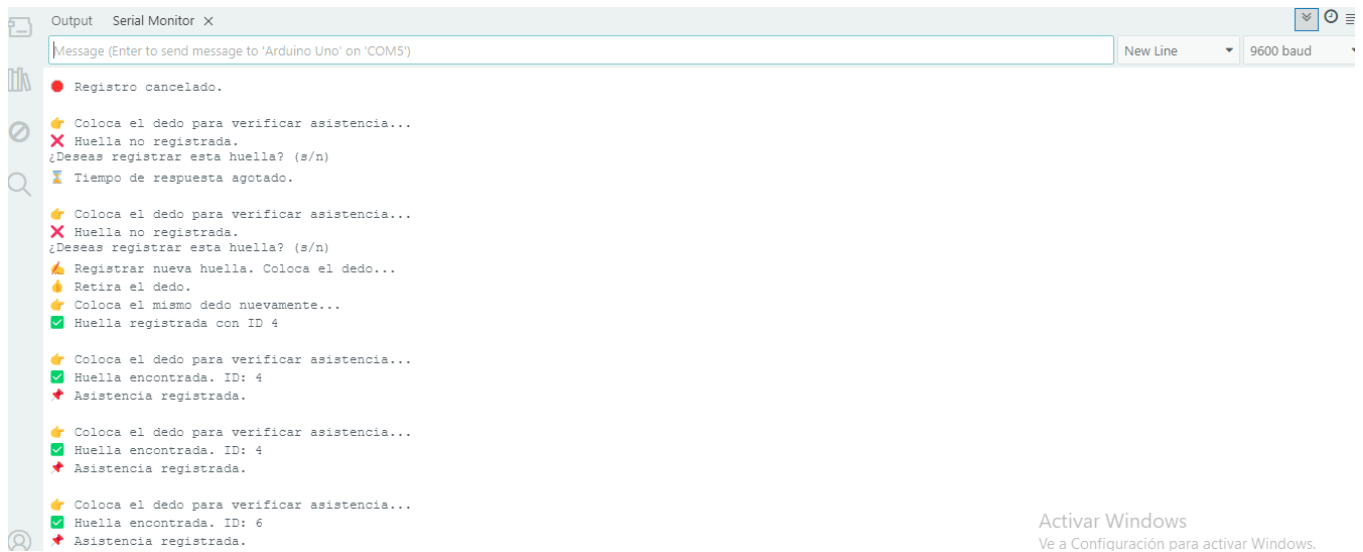


Figura21. Toma correctamente la asistencia de huellas antiguas, y también de nuevas, incluso si se acaban de registrar. En caso de no querer hacer el registro de la huella, te cancela el registro.

Conclusiones

El desarrollo e implementación del sistema de control de asistencia basado en IoT en las oficinas del PRD Tlaxcala demostró ser una solución eficiente y funcional frente a las limitaciones de los métodos tradicionales de registro manual. A través de la incorporación del sensor biométrico AS608 y la placa Arduino Uno, se logró automatizar el proceso de toma de asistencia, permitiendo identificar al personal de manera rápida, precisa y segura.

Este sistema no solo reduce el margen de error humano, sino que también mejora significativamente la eficiencia administrativa, al permitir una supervisión más confiable y oportuna del cumplimiento horario. La automatización del proceso contribuye a optimizar recursos, fortalecer la rendición de cuentas y respaldar la toma de decisiones institucionales basadas en datos reales.

Además, la experiencia adquirida en el diseño, configuración e integración de componentes electrónicos, así como en la programación de microcontroladores, representa una valiosa aportación académica y profesional para el estudiante, alineada con los objetivos del programa de residencias profesionales.

Recomendaciones

- Ampliar la capacidad del sensor biométrico, explorando versiones con mayor capacidad de almacenamiento de huellas digitales para contextos con más personal.
- Integrar una pantalla LCD o display OLED, que permita ofrecer retroalimentación visual al usuario al momento de colocar la huella, como mensajes de bienvenida, confirmación o error.
- Incluir un sistema de respaldo de energía (UPS o batería externa) para asegurar el funcionamiento continuo del dispositivo en caso de cortes eléctricos.
- Explorar la posibilidad de conexión inalámbrica mediante módulos como ESP8266 o ESP32, para enviar los registros a una base de datos o sistema centralizado en tiempo real.
- Fortalecer la seguridad física del dispositivo, asegurando que el hardware esté protegido contra manipulación no autorizada o daños accidentales.
- Documentar e informar al personal sobre el uso adecuado del sistema biométrico, asegurando la aceptación y el correcto uso de la tecnología instalada.

Referencias

- Atzori, L., Iera, A., & Morabito, G. (2022). The Internet of Things: A survey. *Computer Networks*, 54(15), 2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>
- CISCO. (2024). What is the Internet of Things (IoT)? Cisco Systems. <https://www.cisco.com/site/us/en/solutions/internet-of-things>
- IBM. (2024). IoT explained: Internet of Things overview. <https://www.ibm.com/topics/internet-of-things>
- López Research. (2023). An Introduction to the Internet of Things (IoT). *IEEE IoT Journal*, 10(1), 89–101.
- Chudley, R., & Greeno, R. (2023). *Building systems in modern workplaces*. Routledge.
- Rodríguez, J. M., & Pérez, L. A. (2023). Sistemas de control de asistencia basados en biometría para ambientes laborales. *Revista Electrónica de Tecnología Aplicada*, 11(2), 34–45.
- ISO/IEC 27001:2022. (2022). *Information Security Management Systems — Requirements*. International Organization for Standardization.
- Hernández, D. (2024). Implementación de sistemas automatizados de control de acceso en entornos educativos. Universidad Nacional Autónoma de México (Tesis de maestría).
- Jain, A. K., Ross, A., & Nandakumar, K. (2023). *Introduction to Biometrics* (2nd ed.). Springer. <https://doi.org/10.1007/978-3-319-69408-5>
- European Union Agency for Cybersecurity (ENISA). (2024). *Biometric recognition and data protection guidelines*. <https://www.enisa.europa.eu>
- IBM. (2024). What is biometrics? <https://www.ibm.com/topics/biometrics>
- ISO/IEC 24745:2023. *Biometric Information Protection*. International Organization for Standardization.
- Maltoni, D., Maio, D., Jain, A. K., & Prabhakar, S. (2023). *Handbook of Fingerprint Recognition* (2nd ed.). Springer. <https://doi.org/10.1007/978-1-84882-254-0>
- Jain, A. K., & Feng, J. (2022). *Latent Fingerprint Recognition: Challenges and*

Trends. IEEE Signal Processing Magazine, 39(1), 98–113.

- ISO/IEC 19794-2:2022. Biometric data interchange formats – Part 2: Finger minutiae data.
- NIST. (2023). Fingerprint Recognition Evaluation Report. National Institute of Standards and Technology. <https://www.nist.gov>
- Banzi, M., & Shiloh, M. (2024). Getting Started with Arduino (4th ed.). Make Media.
- Monk, S. (2024). Programming Arduino: Getting Started with Sketches (2nd ed.). McGraw-Hill Education.
- Ibrahim, D. (2023). Designing Embedded Systems with Arduino: A Fundamental Technology for Makers. Wiley.
- Arduino. (2025). Arduino UNO Product Page & Documentation. <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- IEEE Xplore. (2024). Applications of Arduino in automation and smart environments. IEEE Transactions on Instrumentation and Measurement, 73, 112–122.
- Adafruit. (2024). Fingerprint Sensor Module AS608 – Datasheet and Overview. <https://www.adafruit.com/product/4631>
- SparkFun. (2024). Biometric Sensor Integration Guide (AS608). <https://www.sparkfun.com>
- GitHub – Adafruit Industries. (2025). Adafruit Fingerprint Sensor Library. <https://github.com/adafruit/Adafruit-Fingerprint-Sensor-Library>
- Ghosh, A. (2022). Biometric Systems: Principles and Design. Springer.
- Kumar, V., & Ali, S. (2023). Lightweight biometric modules for embedded systems. Journal of Embedded Technology Research, 15(2), 55–69.