



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**Sistema de monitoreo de la Red Fría  
del INMEGEN**

**INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de

**Ingeniero en Sistemas Biomédicos**

**P R E S E N T A**

Fernando Salas Carmona

**ASESOR DE INFORME**

Dr. Didier Torres Guzmán



**Ciudad Universitaria, Cd. Mx., 2024**

## **Agradecimientos**

Este trabajo es la culminación de una etapa muy importante para mí, por ello en primer lugar agradezco a toda mi familia por el apoyo incondicional y la confianza recibida impulsándome con su cariño y consejos para lograr mis metas y sueños durante toda mi estancia académica.

Agradezco a todos mis profesores por transmitirme los conocimientos necesarios para desempeñarme en mi futuro profesional, pero en especial a la Dra. Michelin Álvarez y el Dr. Didier Torres por los consejos y la ayuda brindada desde que los conocí.

A todas las personas que traté en mi trayecto universitario, compañeros y amigos. De todos ustedes aprendí algo importante. Pero agradezco a mis mejores amigos Angelica, Edna y Emmanuel sin ustedes la universidad no hubiera sido lo mismo.

Agradezco al Instituto Nacional de Medicina Genómica de México (INMEGEN) por darme la oportunidad de trabajar en sus instalaciones en las que se desarrolló este proyecto. Especialmente a los ingenieros Jorge Alvarado y Raúl Corral pertenecientes a la subdirección de secuenciación y genotipificación quienes me orientaron para lograr y culminar el proyecto.

También agradezco el apoyo recibido por el proyecto DGAPA-PAPIIT TA100822 "Sistema para la evaluación psicofisiológica del estrés" del adquirí grandes herramientas para la elaboración de este trabajo.

Finalmente, agradezco a mi facultad y mi universidad por contar con muchas herramientas y recursos para la formación profesional de todos los alumnos que nos hemos esforzado para estar en sus instalaciones.

## **Contenido**

Introducción.....	3
Objetivo .....	3
Descripción del Instituto.....	4
Justificación .....	5
Antecedentes del sistema de monitoreo de la Red Fría en el INMEGEN .....	8
Planteamiento del problema .....	10
Materiales y método .....	12
Resultados.....	17
Conclusiones y trabajos futuros.....	25
Experiencia académica aplicada.....	26
Bibliografía.....	27
Anexos .....	28
Sección de código de programación en Arduino sin Bot de Telegram.....	28
Sección de código de programación en Arduino con Bot de Telegram.....	33

## **Introducción**

El objetivo principal de los sistemas de monitoreo para la cadena de frío es registrar la temperatura de operación de los equipos a lo largo del tiempo. Un aumento o disminución en esta variable, fuera de los rangos ideales, representa un riesgo para el material de trabajo y muestras biológicas, lo que puede generar pérdidas económicas y daños incuantificables en el desarrollo de investigación biomédica.

Aunque algunas compañías comercializan sistemas para el monitoreo de la temperatura de los equipos de la cadena de frío, dirigidos principalmente a la industria farmacéutica y de alimentos, sus precios pueden resultar incosteables para la mayoría de las dependencias de Salud Pública. Por lo tanto, es importante diseñar soluciones alternativas que optimicen los recursos disponibles.

El Instituto Nacional de Medicina Genómica de México ha desarrollado un sistema de monitoreo basado en herramientas de código abierto e infraestructura propia. Este sistema permite monitorear la temperatura, humedad relativa y el estado de la puerta (abierto/cerrado) de los equipos, visualizar en tiempo real el comportamiento de las variables, enviar alertas y realizar consultas vía remota. Esta tecnología tiene aplicaciones potenciales en los procesos de almacenamiento de biológicos y redes de Biobancos. Además, la representación gráfica y el registro de las variables en el tiempo pueden proporcionar información relevante para el análisis del comportamiento de los equipos y ser utilizados para tomar decisiones e implementar herramientas predictivas que mejoren la gestión.

Sin embargo, el sistema inicialmente desarrollado en el Instituto presentaba áreas de oportunidad, como la mejora en el monitoreo en tiempo real, la optimización de los códigos de adquisición y la implementación de sensores de estado de puerta y de ultrabajas temperaturas. En este trabajo, se proponen e implementan soluciones específicas para abordar cada uno de estos puntos de mejora utilizando los recursos disponibles

## **Objetivo**

Actualizar y mejorar el sistema de monitoreo de la Red Fría basado en herramientas de código abierto e infraestructura propia del Instituto Nacional de Medicina Genómica para la representación gráfica y el registro de las variables de temperatura, humedad y estado de la puerta (abierto/cerrado) de los equipos de la cadena de frío. Además, permitir la visualización en tiempo real del comportamiento de las variables, envío de alertas y consultas vía remota.

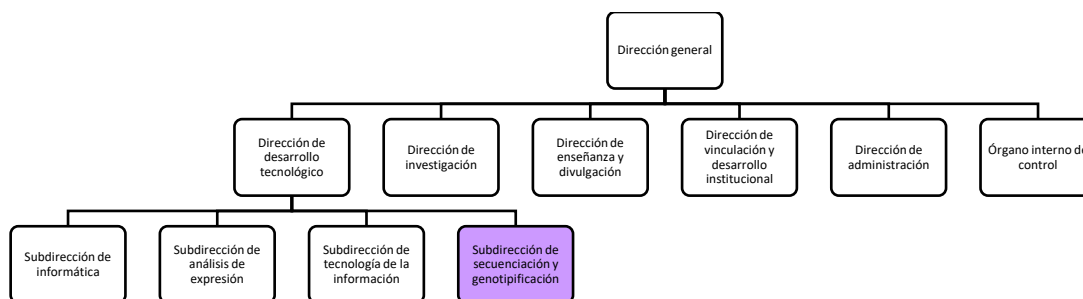
## Descripción del Instituto

Con dirección en Periférico Sur No. 4809, Col. Arenal Tepepan, C.P. 14610, Delegación Tlalpan, Ciudad de México, el Instituto Nacional de Medicina Genómica (INMEGEN), es un organismo público descentralizado y es parte de los Institutos Nacionales coordinados por la Comisión Coordinadora de Institutos Nacionales de Salud de la Secretaría de Salud, siendo el décimo primero de estos, fundado en el año 2004, como resultado del trabajo efectuado desde 2001 por el Consorcio Promotor del Instituto de Medicina Genómica, integrado por la Universidad Nacional Autónoma de México, el Consejo Nacional de Ciencia y Tecnología de la Secretaría de Salud, y la Fundación Mexicana para la Salud. La línea de investigación sobre la cual se trabaja el instituto es la del genoma humano [1].

El INMEGEN tiene como objetivo impulsar la medicina genómica desarrollando nuevos productos y servicios que podrán ser aplicados en beneficio de la población mexicana mediante una práctica médica más individualizada, predictiva, preventiva y participativa. Como todo Instituto Nacional de Salud, además de enfocar sus esfuerzos en el desarrollo e investigación de nuevas tecnologías y métodos, también tiene un carácter formativo con las nuevas generaciones, por esto mismo es que parte de sus recursos se destinan hacia la divulgación y la enseñanza, esto lo realiza mediante la impartición de cursos de posgrado, el desarrollo de programas de intercambio académico, de educación a distancia y de difusión sobre Medicina Genómica; con el fin de satisfacer las necesidades de formación de recursos humanos especializados para la salud y de divulgación del conocimiento generado en el Instituto [2].

El desarrollo del presente trabajo fue realizado dentro de la subdirección de secuenciación y genotipificación encargada de coordinar y mantener el adecuado funcionamiento de los equipos biomédicos, biotecnológicos y de laboratorio. Además, se contó con el apoyo de la Subdirección de tecnología de la información para establecer los puntos de conexión a red dentro del instituto.

*Organigrama del Instituto Nacional de Medicina Genómica de México*



## Justificación

El concepto de cadena de frío tiene varias definiciones, dependiendo de su aplicación. Una definición general, aplicable a nivel industrial y de investigación, sería:

“La cadena de frío es un conjunto de normas y procedimientos que aseguran la atmósfera ideal de temperatura fresca o fría, durante el almacenamiento y distribución de productos termolábiles o perecibles, a través de sistemas de pre enfriado, cámaras de refrigeración o transporte refrigerado, para su conservación a temperaturas recomendadas; con lo cual se mantiene el óptimo estado de las propiedades y características de tales productos o, en el caso de medicamentos, su efectividad y máxima potencia” [3].

Uno de los procedimientos definidos en la normatividad de la cadena de frío para asegurar las condiciones de temperatura es el monitoreo de los sistemas de enfriamiento. Con esta actividad se evalúa si las condiciones son adecuadas para los productos resguardados.

Para lograr el monitoreo, se debe de realizar la medición y registro de la temperatura de los sistemas o equipos de enfriamiento. Las mediciones de la temperatura se pueden realizar utilizando instrumentos analógicos o digitales y su registro de forma manual o automática. De hecho, algunos sistemas de enfriamiento cuentan con un display para la visualización del valor de temperatura y otros más sofisticados hasta permiten guardar un registro temporal del comportamiento de la temperatura y la configuración de alarmas cuando el equipo se encuentra fuera del rango de operación establecido, lo que facilita el monitoreo.

No obstante, la Organización Panamericana de la Salud (OPS) ha establecido que ni los más modernos y sofisticados equipos y transportadores disponibles para la cadena de frío serán efectivos si las personas responsables del programa no conocen con propiedad los principios de operación y funcionamiento de los equipos frigoríficos [4]. Por lo que un sistema de monitoreo que facilite la medición de la temperatura e informe oportunamente a los interesados vía remota sobre el estado de los equipos de enfriamiento resulta de gran utilidad para garantizar la correcta operación de estos sistemas.

La Organización Mundial de la Salud (OMS) junto con la Fondo Internacional de Emergencia de las Naciones Unidas para la Infancia (UNICEF) en 2016 buscó retroalimentación acerca del uso de sistemas de monitoreo remotos de temperatura. En los resultados, la aceptabilidad de un sistema de monitoreo fue muy alta y recomendaban tener uno para cada uno de los

sistemas de enfriamiento. Sin embargo, hubo muchas limitantes destacando que no son fáciles de usar y son sistemas muy costosos [5].

Los dispositivos para el monitoreo remoto de la temperatura en la cadena de frío en el mercado constan de dos elementos: un termógrafo, para el registro de temperatura y humedad y un receptor, dispositivo para realizar el registro de las señales en la nube. Las empresas que ofertan el servicio incluyen certificados de calibración de los equipos garantizando mediciones confiables en los rangos de operación, siendo estos de  $-30\text{ }^{\circ}\text{C}$  hasta  $80\text{ }^{\circ}\text{C}$  usualmente. Estos sistemas comerciales otorgan a su vez una vista general del funcionamiento y envío de notificaciones de las violaciones en los rangos de temperatura.

Aunque los sistemas comerciales cumplen con el objetivo principal de monitorear, su costo puede ser un obstáculo para su implementación cuando se requiere vigilar una gran cantidad de equipos. Esto se debe a que la adquisición de estos sistemas implica la compra de termógrafos, receptores, cuotas de servicio para almacenamiento en la nube y, en algunos casos, el acondicionamiento de los termógrafos para que cumplan con los rangos de operación requeridos por los equipos de enfriamiento. Porque de acuerdo en un análisis de mercado el costo de un termógrafo en *Avantar Delivered by VWR* se evalúa en \$508.10 USD para un equipo de ultracongelación [6] o en costos de \$307.00 a \$583.00 USD para los equipos *FisherBrand Ultra-Low Temperature* termógrafo de la marca *Thermo Fisher Scientific* [7].

En el Instituto Nacional de Medicina Genómica de México, existe una alta demanda de sistemas de enfriamiento para resguardar reactivos y muestras biológicas. Estos equipos se dividen en dos grandes áreas dentro de la cadena de frío del Instituto: los equipos ubicados en laboratorios y áreas comunes compartidas. En su mayoría, estos espacios utilizan equipos de refrigeración y congelación que operan a temperaturas promedio de  $5\text{ }^{\circ}\text{C}$  y  $-20\text{ }^{\circ}\text{C}$ , respectivamente. La segunda área es el Biobanco, donde se utilizan principalmente equipos de ultracongelación que operan a temperaturas promedio de  $-70\text{ }^{\circ}\text{C}$ .

Los equipos de enfriamiento del Instituto han sido monitoreados mediante métodos convencionales, que implican visitar las ubicaciones específicas de los equipos y registrar los valores de temperatura proporcionados por sus sensores. Sin embargo, este método presenta limitaciones, ya que no todos los equipos tienen sensores de temperatura y las mediciones se realizan en momentos específicos, lo que no refleja la temperatura promedio de la cámara.

Derivado de las áreas con elementos de la cadena de frío, dentro del Instituto el número de equipos de enfriamiento que deben de ser monitoreados para salvaguardar el contenido

asciende a 134. Sin embargo, el monitorear cada uno de estos de forma convencional resulta una tarea complicada, por lo cual, se buscó implementar un sistema de monitoreo remoto. Desafortunadamente, adquirir un sistema de monitoreo comercial para cada uno de los equipos resulta demasiado costoso, una situación que probablemente sea similar para la mayoría de las instituciones públicas prestadoras de servicios de salud en América Latina.

En consecuencia, el Instituto ha propuesto el desarrollo de un sistema de monitoreo aplicado a una cadena de frío, con el objetivo de optimizar el proceso de monitoreo, minimizar el riesgo de pérdida de productos y proporcionar una atención oportuna en caso de falla.

Es así que se establecieron las siguientes características con las que debe contar un buen sistema de monitoreo para cubrir las necesidades dentro del Instituto: ser fácil de interpretar; eliminar los errores de medición presentados al sensar la temperatura de forma puntual en determinados instantes, dado que la temperatura de los equipos oscila acorde al ciclo de enfriamiento; eliminar los errores dependientes del observador; concentrar la información de todos los sistemas de enfriamiento para facilitar la consulta; permitir diferenciar los cambios de temperatura por uso o mal funcionamiento del equipo; monitorear las condiciones ambientales para un buen funcionamiento de los equipos y notificar a los interesados cuando algún sistema se encuentre fuera del rango de temperatura.

Con base en lo especificado anteriormente, dentro del Instituto Nacional de Medicina Genómica de México se inició con el desarrollo de un sistema de monitoreo utilizando herramientas de código abierto, puntos de conexión a internet y componentes electrónicos para el sensado de las variables de interés.



## **Antecedentes del sistema de monitoreo de la Red Fría en el INMEGEN**

El sistema de monitoreo de red fría del INMEGEN ha sido un proyecto de gran interés desde su fundación. El proyecto comenzó con la instrumentación y la instalación de sensores de temperatura para monitorear los equipos de forma independiente. Sin embargo, debido al crecimiento del Instituto y al aumento en el número de equipos, se buscó una alternativa para mejorar el monitoreo. Pero por la dificultad para adquirir un sistema comercial, se decidió diseñarlo internamente. Después de un arduo trabajo, el sistema fue capaz de realizar funciones como el monitoreo de la temperatura de refrigeración y congelación, el envío de notificaciones a los usuarios y el monitoreo en tiempo real de la temperatura y humedad relativas en los laboratorios de investigación.

Para lograr el avance en el monitoreo de temperatura, se llevaron a cabo varias acciones. En primer lugar, se seleccionaron cuidadosamente los componentes apropiados para el módulo de medición de temperatura. Se optó por utilizar una placa de desarrollo ESP8266 debido a sus capacidades de procesamiento y conectividad, siendo esta última especialmente importante para lograr la conexión a redes Wi-Fi.

Para medir la temperatura se seleccionaron tres sensores, debido a los diferentes rangos de trabajo de los equipos. Para refrigeradores y congeladores cuyo funcionamiento adecuado es entre 0 a 8 °C y -15 a -25 °C respectivamente se implementó el sensor DS18B20. Este es un termómetro digital que opera en el rango de -55 a 125 °C con un error de precisión de  $\pm 0.5$  °C. Para los equipos de ultracongelación que operan con normalidad a temperaturas por debajo de -70 °C se seleccionó un sensor PT-100. Una termorresistencia que puede operar hasta en una temperatura de -200 °C con un error de precisión de  $\pm 0.3$  °C. Finalmente, para las condiciones ambientales (temperatura y humedad relativa) se implementó el sensor BM280 que opera en un rango de 0 a 60°C con un error de precisión de  $\pm 1$  % del valor de la medición.

La selección de los sensores se basó en los siguientes criterios: funcionamiento estable dentro de los rangos de temperatura de trabajo de los equipos de enfriamiento; facilidad de instalación en los equipos y áreas de trabajo; bajo error de precisión y de menor costo en el mercado.

Posteriormente, se realizó la conexión y caracterización de los sensores de forma independiente para garantizar que las mediciones fueran fiables, sin embargo, se detectaron irregularidades en los sensores PT-100 adquiridos para sensar los equipos de

ultracongelación. También, de forma paralela, se trabajó en un diseño para la placa de circuito impreso (PCB) que permitiera la conexión de todos los componentes en un mismo circuito.

Después, se unificaron los códigos de sensado de temperatura de los tres sensores para permitir la medición con cualquier sensor conectado a la placa. Además, se desarrollaron códigos para la conexión a la red y el envío de datos de temperatura. Paralelamente, se exploraron diferentes opciones para el envío, recepción, almacenamiento y visualización de los datos de temperatura. Finalmente, se seleccionó una conexión mediante una interfaz de programación de aplicaciones (API), el almacenamiento en una base de datos de Graphite Carbon y la visualización en Grafana.

Para implementar las herramientas de Graphite Carbon y Grafana, se utilizó un servidor con 100GB de almacenamiento y 16GB de RAM proporcionado por la Subdirección de Tecnología de la Información. Tras la instrumentación de algunos sensores y la configuración de las plataformas, se adecuó la infraestructura en red del instituto para montar el sistema de monitoreo, creando puntos de acceso a internet en cada uno de los laboratorios específicamente para la conexión de los módulos de medición de temperatura.

Finalmente se realiza una implementación del sistema de los equipos de refrigeración, congelación, cuartos fríos y las condiciones ambientales de laboratorios de alta especialidad. Sin embargo, aunque el sistema desarrollado cubría con las funciones más importantes, este presentaba errores y deficiencias para cubrir las necesidades del Instituto.

## Planteamiento del problema

El sistema implementado presentaba varias áreas de oportunidad. Una de ellas era el monitoreo en tiempo real, ya que la transmisión de datos se interrumpía por intervalos prolongados. Esto generaba un exceso de notificaciones a los usuarios, alertándolos sobre la pérdida de información de sus equipos. Estos problemas tenían dos causas principales: la primera era la mala colocación de los sensores dentro de las cámaras refrigeradas, ya que la apertura o cierre de la puerta podía desconectarlos. La segunda causa, más crítica, eran los múltiples errores en la programación del módulo de temperatura.

En cuanto a los errores en la programación se detectó un código desestructurado con líneas innecesarias para el correcto funcionamiento, lo que ocasionaba que el sistema permaneciera en *loops* sin funciones o si había pérdida de conexión a la red no se pudiera reestablecer la conexión interrumpiendo el envío de información. También, se encontraron inconsistencias en la secuencia del algoritmo en: Validación de conexión del sensor; Adquisición de datos; Envío de la información, esto porque el proceso no se realizaba en ese orden, sino que se adquiría, validaba el estado de conexión del sensor y finalmente el envío de los datos.

Además, dentro de la validación del estado de conexión del sensor, había un condicional que limitaba el rango de medición de los sensores a los límites de operación ideales de los equipos de refrigeración, en lugar de los límites reales de operación de los sensores. Por ejemplo, para el sensor DS18B20 no se establecían los límites de -55 a 80 °C en los que puede operar el sensor, sino que se delimitaba de -25 a 10 °C rango de operación en el que se encuentran los refrigeradores y congeladores. Esto causaba que, si la temperatura sensada estaba fuera del rango establecido, no se enviaba la información.

Otro error en la programación estaba relacionado con el periodo de muestreo de las señales de temperatura. Se buscaba establecer una frecuencia de una muestra por minuto, ya que el cambio de temperatura en los equipos de cadena de frío no es significativo en periodos más cortos. Por lo tanto, se configuró la herramienta Graphite Carbon para recibir un dato por minuto, con un margen de llegada de 10 segundos. Sin embargo, el envío de datos desde el módulo de temperatura no era consistente a la frecuencia deseada, lo que generaba un desfase en el envío-recepción de los datos interrumpiendo el monitoreo en tiempo real.

Otra de las áreas de oportunidad fue la falta de los sensores para la apertura y cierre de la puerta y los sensores para el monitoreo de los ultracongeladores, porque a pesar de que se contaban con los sensores WZP-PT100 y el módulo MAX31865 para su instrumentación, no se implementaron debido a los errores de medición que presentaban.

Finalmente, como otra área de oportunidad, se propuso implementar un sistema de códigos de errores para identificar las fallas principales en los módulos de temperatura y así tomar acciones correctivas adecuadas en menor tiempo. Además de implementar o configurar un sistema de notificaciones de alta prioridad.

## Materiales y método

Para ilustrar el progreso del proyecto, se propuso la siguiente estructura: dos líneas de trabajo divididas en etapas. La primera línea se enfoca en el diseño de hardware y en la programación del control en IDE de Arduino para los módulos de temperatura. La segunda línea, por su parte, se encarga de la configuración de Graphite Carbon y Grafana para la recepción, almacenamiento y despliegue de los datos. Además, las líneas de trabajo se dividieron en dos secciones principales. La primera sección de la tabla muestra los antecedentes del proyecto, los cuales se explican en detalle en la sección correspondiente de este informe. La segunda sección, llamada implementación, describe los pasos que se siguieron para resolver los problemas planteados previamente.

*Tabla 1. Estructura del progreso del proyecto dividida en la sección de antecedentes e implementación*

ANTECEDENTES					
	Etapa 1	Etapa 2	Etapa 3	Etapa 4	
Línea de trabajo de Arduino	Búsqueda y selección de componentes	Caracterización y conexión de componentes a placa	Conexión a red y envío de la información	Prueba piloto	
Línea de trabajo Graphite y Grafana			Modo de recepción, almacenamiento y visualización de datos		
	Etapa 1				
IMPLEMENTACIÓN					
	Etapa 5	Etapa 6	Etapa 7	Etapa 8	Etapa 9
Línea de trabajo de Arduino	Estructura del código	Codigos de error	Solución de errores	Sensor puerta	Bot de Telegram
Línea de trabajo Graphite y Grafana				Actualización para la recepción de información	
	Etapa 2				

Para iniciar la sección de implementación, en la quinta etapa, se abordó el problema del monitoreo en tiempo real. Como se mencionó anteriormente, el error tenía dos causas. La primera era la mala colocación de los sensores de temperatura, cuya solución consistió en revisar los equipos de refrigeración de los distintos laboratorios para colocar los sensores adecuadamente donde fuera necesario. La segunda causa eran los errores de programación previamente mencionados, cuyas acciones correctivas fueron las siguientes.

Se comenzó revisando el código de programación del módulo de temperatura. Se identificaron las partes funcionales, se eliminaron las líneas innecesarias y, a partir de ello, se reestructuró el código siguiendo el siguiente algoritmo simplificado:

1. Inicialización (Carga de librerías)
2. Configuración inicial de sensores y comunicación
3. Conexión a red WIFI
4. Búsqueda, selección y validación del sensor
5. Inicio del loop
  1. Medición de la temperatura y/o humedad
  2. Envío de la información por red
  3. Envío de mensajes por comunicación serial
6. Fin de loop

Una vez modificado el código, se realizó la actualización a los módulos de temperatura de los equipos del Biobanco y se dejó en evaluación por tres días. Se observó que el número de problemas había disminuido considerablemente. Sin embargo, el problema persistía de forma aleatoria y, al no haber una causa evidente del problema, se inició con la implementación de los códigos de error (etapa 6) para identificar la falla en la interrupción de envío de datos.

Durante la etapa 6, se evaluaron las posibles causas de fallo en el envío de datos y se encontraron cuatro: problemas de comunicación con el servidor, desconexión de la red, sensor no detectado y fallo en el sensor. A estas causas se les asignaron los códigos 20, 30, 40 y 50, respectivamente. La elección de estos códigos se basó en que son valores que las variables de temperatura de los equipos de refrigeración no alcanzan durante su correcto funcionamiento.

Una vez definidos los códigos de error, se realizaron las modificaciones necesarias en el código para que se mostraran en la interfaz de Grafana. Para los errores 40 y 50, como la comunicación con el servidor era estable, no hubo dificultades en el despliegue. Sin embargo, para los códigos de error 20 y 30, la comunicación estaba interrumpida, por lo que se creó un registro en memoria para almacenar el código de error y enviarlo como primer dato después de restablecer la comunicación con el servidor.

Es importante mencionar que la validación de la comunicación con el servidor se realizaba a través del código 200, código de respuesta de estado satisfactorio del API a la recepción del dato enviado. Si el código recibido no era 200, indicaba que la comunicación estaba

interrumpida. En ese caso, se evaluaba el estado de conexión a la red Wi-Fi para determinar si su desconexión era la causa de la interrupción.

Los códigos de error se activan sólo si la falla persiste de forma consecutiva durante tres minutos. Esto se debe a que, con las pruebas del sensor DS18B20, en periodos de tiempo prolongados y sin ninguna periodicidad, se producían mediciones de temperatura inadecuadas en un solo instante, lo que generaba falsos positivos en los códigos de error.

Implementados los códigos de error, se volvieron a actualizar los módulos de temperatura de los equipos del Biobanco y se dejaron en evaluación. En el transcurso del tiempo se identificó que la falla de envío de los datos era debido a alguna de estas cuatro causas sin ninguna en específico. No obstante, para solucionarlas se implementó lo siguiente: para los errores 40 y 50 la causa principal era en el hardware del módulo de temperatura, principalmente a problemas con las soldaduras de las terminales de los componentes. Por lo que la acción era la revisión de las PCB y la conexión adecuada de los sensores. Mientras que para los códigos 20 y 30 la solución era un reinicio del módulo de temperatura, sin embargo, realizarlo de forma manual no era la mejor solución, por lo que se inició con la etapa número siete (Solución de errores).

En la etapa 7, se comenzó a implementar un sistema de reinicio automático cuando se activara el código de error 20 o 30. Luego, se actualizaron nuevamente los módulos de temperatura de los equipos del Biobanco. Después de realizar estas acciones, el envío de datos ya no se interrumpía y se mantuvo un monitoreo en tiempo real estable.

Por otra parte, en esta etapa se comenzó a solucionar el problema del sensor de temperatura para los ultracongeladores. Como se mencionó en los antecedentes, se había seleccionado un sensor PT-100, que es una termorresistencia capaz de operar hasta  $-200\text{ }^{\circ}\text{C}$  con un error de precisión de  $\pm 0.3\text{ }^{\circ}\text{C}$ . Sin embargo, los sensores adquiridos eran del tipo WZP-PT100 y resultaron ser de muy mala calidad. Porque se realizó una caracterización para determinar su comportamiento y definir la ecuación de funcionamiento, pero cada uno de los sensores analizados presentaba un comportamiento diferente y difícil de modelar con precisión. Por lo tanto, se inició la búsqueda de alternativas para medir la temperatura de los ultracongeladores.

Se propusieron dos soluciones para la medición de temperatura en ultracongeladores. La primera consistió en cambiar el tipo de sensor por un termopar tipo K, ya que era el sensor utilizado por los instrumentos de medición para validar la temperatura de estos equipos. Se realizó una búsqueda y se seleccionó el sensor con las mejores características y el menor

costo en el mercado. El sensor seleccionado, de marca genérica, cubría un rango de  $-100\text{ }^{\circ}\text{C}$  a  $1250\text{ }^{\circ}\text{C}$  según las especificaciones del fabricante y tenía dimensiones adecuadas para cubrir las necesidades. Sin embargo, para su implementación era necesario contar con un módulo que consistía en un amplificador y convertidor digital, por lo que se seleccionó el módulo MAX 31855. Se presentó la propuesta y se autorizó para iniciar las pruebas de funcionalidad. Una vez adquirido el material, se realizó la instrumentación del sensor y su caracterización. Las pruebas iniciales mostraron que el sensor era estable hasta temperaturas de  $-50\text{ }^{\circ}\text{C}$ , pero a temperaturas más bajas no era estable y sufrió daño irreparable.

Dado que un nuevo sensor para los equipos de ultracongelación requería más elementos para su correcta instrumentación, se buscó la implementación de un nuevo sensor PT100 con el objetivo de reaprovechar el material y el diseño de PCB que ya se contaba. Se realizó una búsqueda en el mercado para identificar un sensor con las características adecuadas, pero no se encontró uno asequible debido a los altos costos para pruebas. Por ello, se buscó con empresas nacionales que se encargaban de la fabricación de sensores PT100 y se seleccionó INDUSTRIAS KEW S.A. de C.V., quien presentó una mejor propuesta en la fabricación de sensores de temperatura de acuerdo con las necesidades.

Después del trámite administrativo para la adquisición de un sensor de la empresa INDUSTRIAS KEW S.A. de C.V., se realizaron pruebas funcionales del sensor de temperatura. Con el sensor adquirido, se iniciaron las mediciones de temperaturas en equipos de refrigeración, congelación y finalmente en los equipos de ultracongelación. Con este sensor, las mediciones de temperatura fueron estables hasta el rango de temperatura de  $-85\text{ }^{\circ}\text{C}$ .

En la octava fase se buscó atender la necesidad de monitoreo de la puerta de los equipos de refrigeración. Para ello se analizó el sistema de cierre de puerta de los equipos más sofisticados y los intentos previos para el sensado del estado de la puerta. Algunos de los equipos como los congeladores REVCO cuentan con un interruptor que determina el cierre o la apertura de la puerta. Sin embargo, replicar este mecanismo en los equipos no era viable debido a las características físicas de los mismos. Por otra parte, previamente se había analizado la opción de realizar el monitoreo con sensores magnéticos comerciales para puertas, no obstante, se presentaron dificultades para la instalación de estos en los equipos, debido a que su colocación interfería con los usuarios durante la apertura o cierre de puerta con lo que estos terminarían despegándose o rompiéndose.

En consecuencia, se analizaron diferentes opciones para realizar el monitoreo de la puerta de los equipos de la red fría del instituto. Como una de las características de los sensores para el monitoreo de la puerta es que no debiera de intervenir con los usuarios y su manejo de diario



de los equipos. Es de ahí que surge la idea de un sensor de detección de obstáculos para el monitoreo de los equipos de refrigeración. El sensor utilizado es un TCRT5000L que es un tipo de sensor óptico reflectivo que detecta objetos mediante la reflexión de la luz en el mismo.

El sensor TCRT5000L presentó una gran facilidad en la implementación en el código y en la colocación física en los equipos para determinar la detección de la puerta en los equipos de la red fría. Una vez realizadas las pruebas de funcionalidad y obteniendo buenos resultados, se realizaron ajustes al código para realizar el sensado, el envío y el despliegue de la información. Para lograrlo se trabajó de forma paralela en la segunda etapa de la línea de trabajo de Graphite y Grafana al incluir la recepción de esta variable.

Finalmente, en la novena etapa se buscó implementar un medio de comunicación alternativo para la notificación de alertas críticas a los usuarios responsables de los equipos. La razón es que el sistema de notificación de Grafana se realiza a través de correo electrónico enviando alertas generales para todos los usuarios. Sin embargo, si un equipo alcanza temperaturas críticas la notificación puede pasar desapercibida ya que no hay diferencia entre una notificación de baja o alta prioridad, por lo que se buscó una forma específica en la que se envíen notificaciones únicamente de los casos considerados críticos en los equipos de la red fría. Estos son alta temperatura (temperatura por arriba de 15 °C de la media específica del equipo durante un periodo de tiempo de más de cinco minutos) y/o una apertura de puerta por un tiempo de más de dos minutos.

Como solución se buscó enviar las notificaciones mediante un sistema de mensajería instantánea, por lo que se propone la implementación de un Bot en Telegram. Con esta propuesta se logró un medio de comunicación bidireccional en el cual se reciben las notificaciones críticas emitidas por el módulo de medición y por parte del usuario realizar consulta del valor de la temperatura de forma remota aun estando fuera del INMEGEN, además de que el usuario puede ajustar los límites de alerta para las notificaciones por este medio de acuerdo con sus intereses.

## Resultados

El sistema de monitoreo de la red fría del INMEGEN es un conjunto de dos productos. El primero es módulo de medición para determinar la temperatura en los diferentes sistemas de enfriamiento de la cadena de frío (refrigeradores, congeladores, ultracongeladores, cuartos de trabajo o almacenamiento), condiciones ambientales del área (temperatura y humedad relativa de los laboratorios), el registro del uso del equipo (estado de la puerta abierta/cerrada) y finalmente la comunicación con el usuario para el envío de alertas prioritarias y consulta de información vía remota.

El segundo producto es el sistema de recepción, almacenamiento y la visualización gráfica de la información de la temperatura de los equipos de la red fría con el que se permite un monitoreo en tiempo real y envío de notificaciones acerca del funcionamiento de los equipos. En forma de resumen el funcionamiento del sistema desarrollado se muestra en la siguiente imagen:

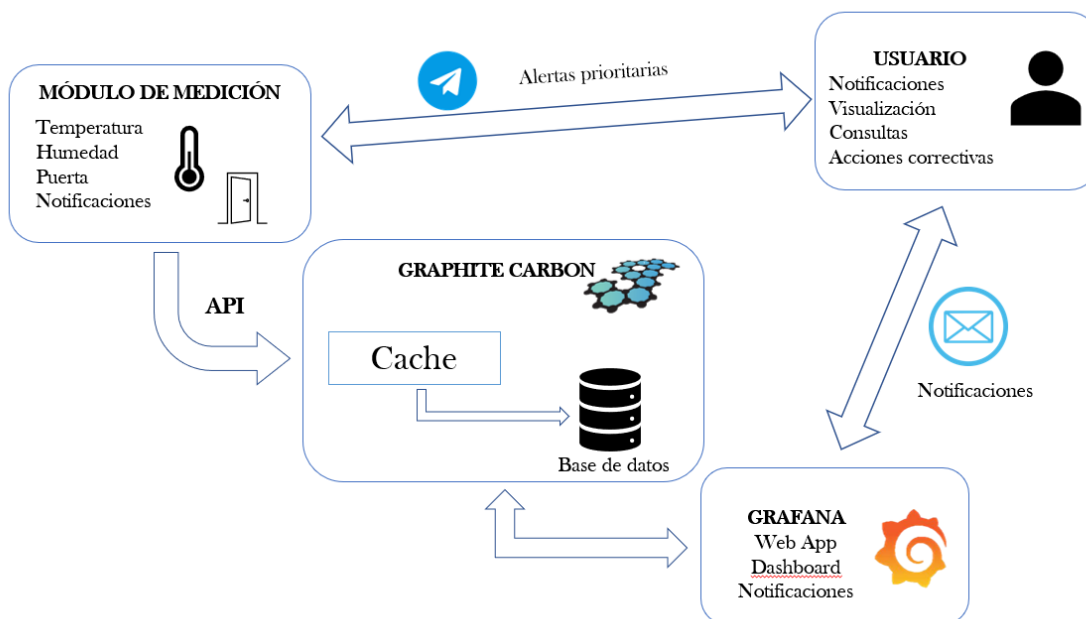


Imagen 1. Esquema de funcionamiento del sistema de monitoreo de la Red Fría del INMEGEN

El primer producto que es el módulo de medición se compone de lo siguiente: la selección de los sensores adecuados para realizar las mediciones de las variables, una placa PCB la cual permite la conexión con los diferentes sensores adquiridos y un código en el IDE de Arduino que permite realizar la conexión, mediciones y la comunicación para el envío con la base de datos.

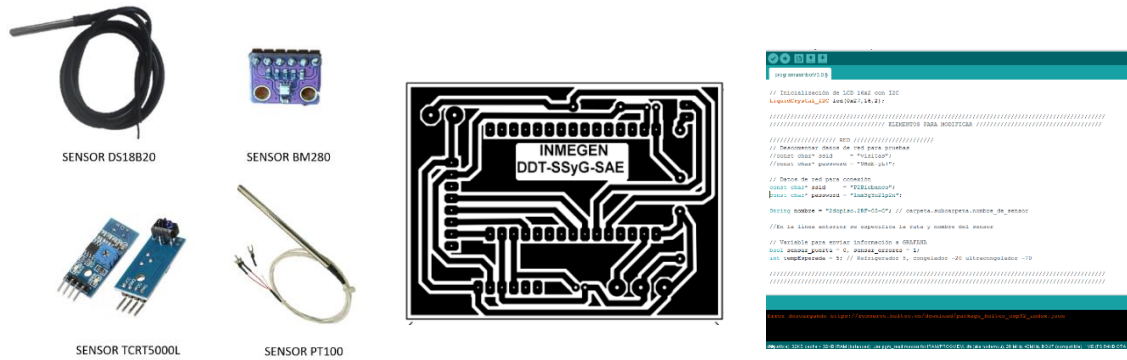


Imagen 2. Componentes del módulo de medición. Izquierda los cuatro sensores seleccionados; Centro diseño de placa PCB; Derecha Imagen del código de programación en Arduino

La placa PCB se diseñó de manera unificada para permitir la conexión de cualquier tipo de sensor. En consecuencia, el código desarrollado en la plataforma Arduino se adaptó para realizar un escaneo y comenzar las mediciones con cualquier sensor conectado. Además, el diseño de la placa y el código facilitó la adaptación de otros elementos necesarios para satisfacer las demandas del Instituto. Un ejemplo de esto fue la inclusión de una pantalla LCD para mostrar los valores de temperatura, lo que permitió enviar el dispositivo para calibrar los sensores de temperatura. También se pudo conectar tres sensores de temperatura DS18B20 para validar la temperatura de un equipo de refrigeración en tres puntos diferentes, tal como se solicitó en un protocolo de investigación.



Imagen 3. Funcionamiento del módulo de medición con LCD integrada

Además, todos los sensores utilizados para medir la temperatura se enviaron para su certificación, con el fin de garantizar la precisión de las mediciones. Para ello, se realizaron las adaptaciones necesarias tanto en el código como en la placa PCB para permitir que la información de los sensores se mostrara en una pantalla LCD, eliminando la dependencia de

la interfaz de Grafana para visualizar la información. Sin embargo, la certificación se llevó a cabo antes de la adquisición del nuevo sensor PT-100 y los resultados obtenidos fueron del sensor WZP-PT100 y a pesar de la calibración no se lograron obtener mediciones correctas ni caracterizables. No obstante, aunque la certificación del nuevo sensor PT-100 aún está pendiente, su funcionamiento se validó al obtener mediciones precisas en los puntos de medición evaluados.

TLABLA DE RESULTADOS BM280				
Temperatura Nominal °C	Temperatura Promedio Patrón °C	Temperatura Promedio I.B.C. °C	E.M °C	U °C
-10	-10	-9.8	0.2	0.42
-5	-4.99	-4.7	0.29	0.42
0	0.01	0	-0.01	0.42
10	10.01	10.1	0.09	0.45
25	25.01	25.2	0.19	0.45

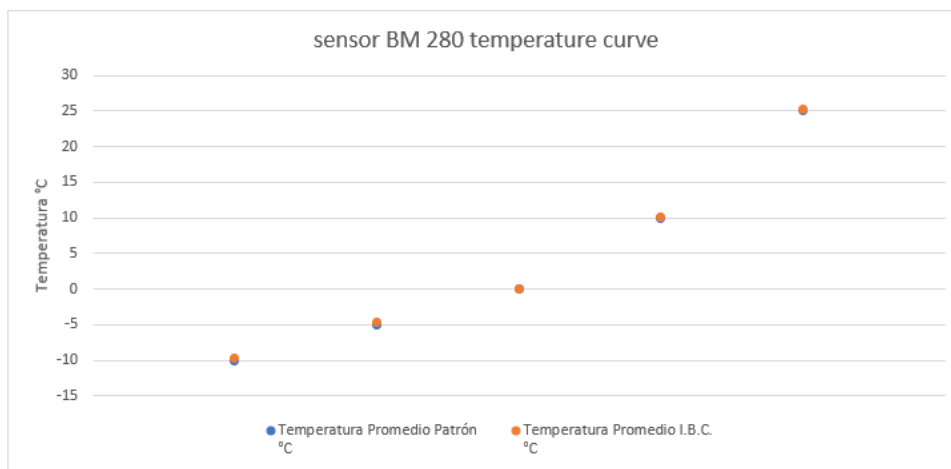
U = Insertidumbre de medida

E.M= Error de Medida

I.B.C = Instrumento de medida Bajo Calibración

Error Máximo = 0.29 °C

\*Los valores de temperatura son el resultado del promedio de 3 mediciones por cada valor nominal



TLABLA DE RESULTADOS DS18B20				
Temperatura Nominal °C	Temperatura Promedio Patrón °C	Temperatura Promedio I.B.C. °C	E.M °C	U °C
-10	-10	-9.6	0.4	0.16
-5	-5	-4.9	0.1	0.17
0	0	-0.1	-0.1	0.17
10	10	10.2	0.2	0.16
25	25	25.7	0.7	0.17

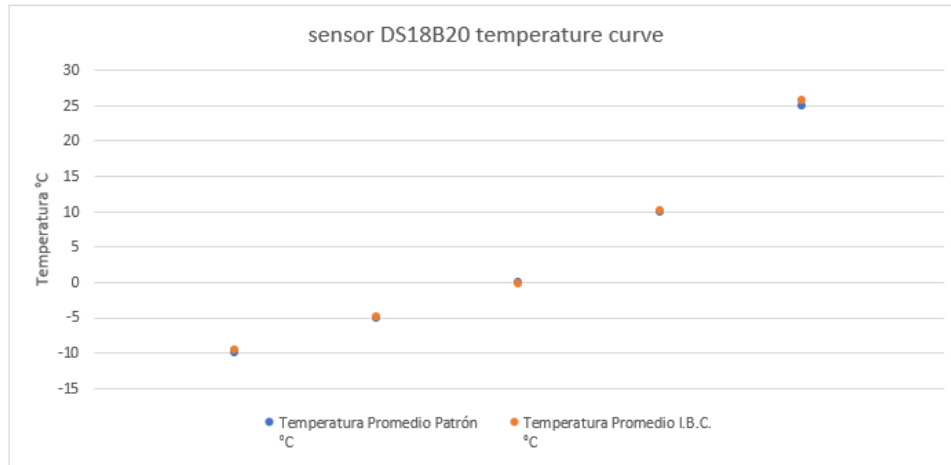
U = Insertidumbre de medida

E.M= Error de Medida

I.B.C = Instrumento de medida Bajo Calibración

Error Máximo = 0.70 °C

\*Los valores de temperatura son el resultado del promedio de 3 mediciones por cada valor nominal



TLABLA DE RESULTADOS PT100				
Temperatura Nominal °C	Temperatura Promedio Patrón °C	Temperatura Promedio I.B.C. °C	E.M °C	U °C
-70	-69.99	-59.8	10.19	0.11
-65	-64.99	-55.9	9.09	0.11
-50	-50	-43.8	6.25	0.12
-45	-45	-39.5	5.5	0.12
0	0	0.89	0.89	0.15

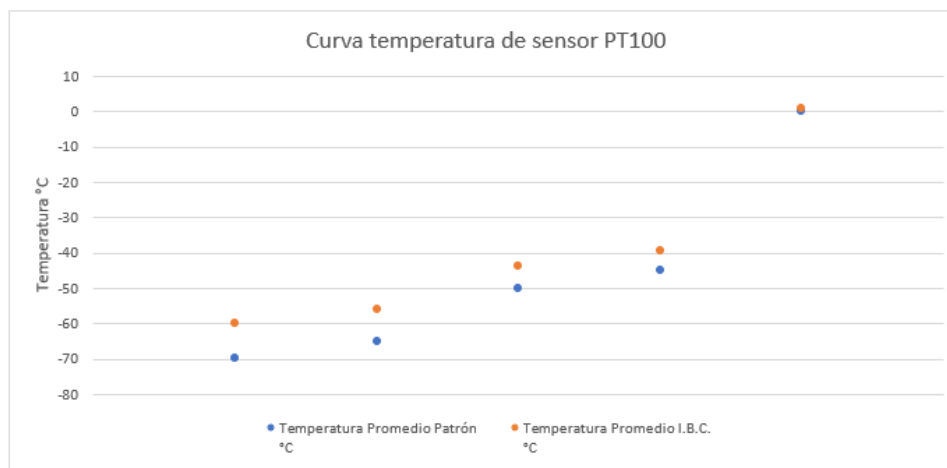
U = Insertidumbre de medida

E.M= Error de Medida

I.B.C = Instrumento de medida Bajo Calibración

Error Máximo = 10.19 °C

\*Los valores de temperatura son el resultado del promedio de 3 mediciones por cada valor nominal



*Imagen 4. Resultados de calibración de los sensores de temperatura*

El segundo producto, que es el sistema de recepción, almacenamiento y visualización gráfica de la información de temperatura de los equipos, se compone de los siguientes elementos: un servidor local donde se almacena la información; el código de recepción en la base de datos de Graphite Carbon; la infraestructura en red para la conexión de los módulos de medición; y la configuración en la interfaz gráfica de Grafana para mostrar la información en tiempo real.



*Imagen 5. Interfaz en Grafana. Se visualizan las señales de temperatura y los límites de temperatura del rango de operación*

La interfaz de Grafana facilita la visualización de las mediciones de temperatura. Dichas mediciones pueden organizarse según el área o laboratorio donde se ubique el sistema de refrigeración. Además, es posible configurar los paneles para que muestren los límites de temperatura dentro del rango operativo del equipo. Esto simplifica la tarea de buscar e interpretar las temperaturas de los sistemas de refrigeración.

La temperatura de los equipos se registra automáticamente cada minuto y se guarda en la base de datos. Esto posibilita la visualización del comportamiento histórico y la fluctuación de la temperatura a lo largo del ciclo de enfriamiento. Esto elimina los errores por mediciones de temperatura en intervalos aislados de tiempo y aquellos dependientes del observador. Además, este sistema facilita la verificación de que el equipo está operando dentro del rango permitido.



Imagen 6. Opciones de visualización. Se permite seleccionar el rango tiempo de interés

Por otra parte, la información proporcionada por el sensor de puerta permite validar que los cambios de temperatura del equipo se deben a la apertura de la puerta por parte de los usuarios y no a un mal funcionamiento de este. Además, la señal del estado de la puerta ayuda a identificar prácticas incorrectas con los sistemas de enfriamiento.



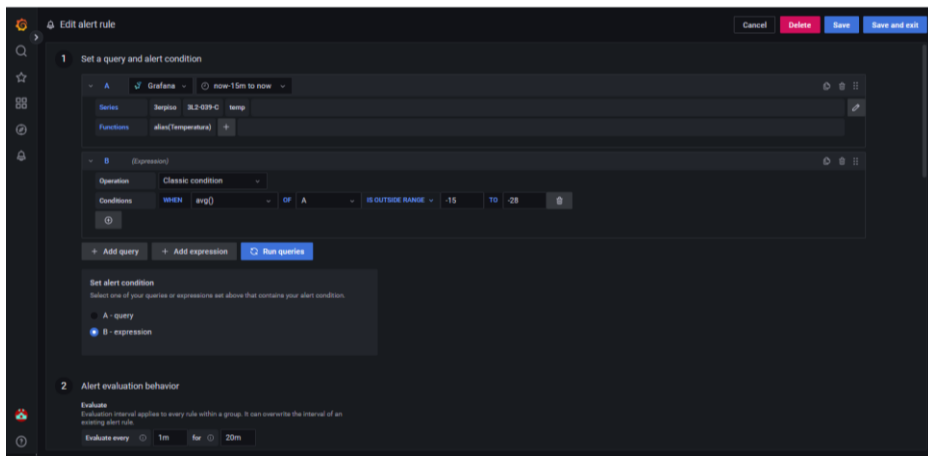
Imagen 7. Señal de puerta. Se aprecia un aumento de temperatura por uso del equipo.

Además, otras de las variables a medir son la temperatura y humedad relativa de las áreas en las que se encuentra el equipo de enfriamiento. Con ello se monitoriza que los equipos no se encuentren operando en condiciones ambientales extremas que generan mayor desgaste del equipo de trabajo.



*Imagen 8. Interfaz en Grafana. Se visualiza la medición de las condiciones ambientales*

La interfaz de Grafana permite configurar alertas para los equipos monitoreados. En la configuración, se especifican los límites operativos del equipo, el tiempo para analizar el valor promedio de la temperatura y la frecuencia de evaluación. Además, es posible establecer puntos de contacto que recibirán notificaciones cuando el sistema de enfriamiento se salga del rango de temperatura establecido. Esto proporciona una capa adicional de seguridad y eficiencia en el monitoreo de los sistemas de enfriamiento.

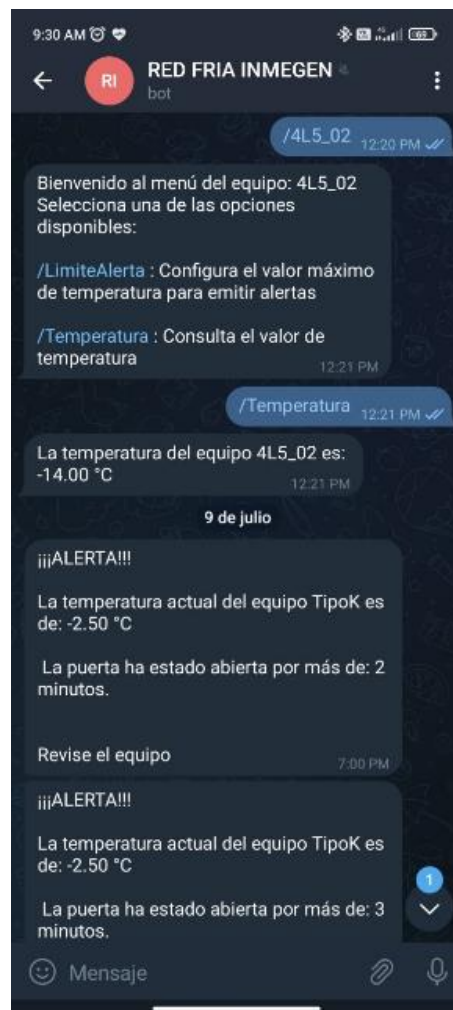


*Imagen 9. Interfaz para la configuración de alertas*



Además de las alertas en la interfaz de Grafana, se puede utilizar el Bot de Telegram como otro medio de alerta. A través de este canal, el usuario recibe notificaciones críticas, como registros de altas temperaturas o tiempos excesivos de apertura de la puerta. El Bot también permite al usuario consultar la temperatura de sus equipos de forma remota y configurar los límites de alerta para este medio de notificación. Esto proporciona una mayor flexibilidad y accesibilidad para el monitoreo de los sistemas de enfriamiento.

Finalmente, en un análisis económico, los costos del desarrollo se dividen en dos elementos principales que componen el sistema. Los elementos necesarios para el circuito de Arduino incluyen una tarjeta de desarrollo, un sensor de temperatura, un cable de conexión, componentes electrónicos y una placa de circuito. El costo del sistema es de \$40.00 USD para equipos de refrigeración y congelación, así como para el módulo de medición de variables ambientales. Por otro lado, los costos de los elementos para la medición de temperatura en equipos de ultracongelación ascienden a \$102.00 USD. La variación en los costos se debe al tipo de sensor utilizado para la medición de temperatura.



*Imagen 10. Interfaz de principal del Bot de Telegram*

Por otro lado, el costo del producto desarrollado en Graphite y Grafana se debe principalmente al servidor configurado para la recepción y almacenamiento de los datos, cuyo costo es de \$995.00 USD (donado por la Subdirección de Tecnologías de la Información). Sin embargo, se debe tener en cuenta un costo adicional que corresponde a los puntos de acceso a la red que deben instalarse para la conexión de los módulos de medición de temperatura. Este costo fue descartado para el Instituto ya que se aprovechó la infraestructura de red existente.

## Conclusiones y trabajos futuros

El sistema de monitoreo para equipos de cadena de frío desarrollado cumple con los criterios establecidos. La tecnología implementada permite integrar todos los elementos de una cadena en un solo lugar, facilitando la búsqueda e interpretación de las señales de temperatura de los equipos. Al ser un sistema automatizado, reduce los errores de medición. Además, permite realizar consultas del estado de los equipos de forma remota a través de Telegram, facilita la diferenciación entre los cambios de temperatura debido al uso o al mal funcionamiento del equipo, y permite identificar condiciones ambientales extremas que pueden afectar el funcionamiento de los equipos. También envía notificaciones de alerta en tiempo real, permitiendo tomar acciones preventivas o correctivas para salvaguardar el estado de medicamentos, muestras biológicas y reactivos utilizados en la investigación.

Además, el costo de implementación de un módulo de medición para un equipo de ultracongelación (el más costoso) es significativamente más asequible en comparación con un termógrafo comercial. El valor de estos últimos representa al menos el triple del valor del equipo desarrollado. Esto hace que esta tecnología sea viable para su implementación dentro del INMEGEN y, si es posible, para su expansión en otras instituciones de salud o investigación. Esto representa una solución costo-efectiva y eficiente para el monitoreo de la cadena de frío.

A lo largo del proyecto desarrollado se cubrió el objetivo que es actualizar y mejorar el sistema de monitoreo de la Red Fría basado en herramientas de código abierto e infraestructura propia del Instituto Nacional de Medicina Genómica para la representación gráfica y el registro de las variables de Temperatura, humedad y estado de la puerta (abierto/cerrado) de los equipos de la cadena de frío. Además de permitir la visualización en tiempo real del comportamiento de las variables, envío de alertas y consultas vía remota. No obstante, dentro del proyecto quedan muchos puntos de oportunidad de mejora e implementación. Hasta ahora, se han identificado las siguientes:

- Implementación de los sensores PT-100 en todos los equipos de ultracongelación.
- Revisión y reacomodo de los módulos implementados en los equipos de enfriamiento.
- Diseño de una carcasa para los módulos de medición para facilitar la colocación y conexión de sensores.
- Finalización del “Procedimiento para la instalación de nuevos módulos de medición y soporte del sistema de monitoreo de red fría”.

Estos puntos representan áreas clave para mejorar aún más el sistema y garantizar su eficacia y eficiencia.

## **Experiencia académica aplicada**

Para el desarrollo de este proyecto, se aplicaron muchos de los conocimientos adquiridos durante la formación académica. En particular, las asignaturas de análisis de circuitos, electrónica básica, amplificación y filtrado fueron fundamentales para analizar, comprender y mejorar el hardware del módulo de medición de las variables de interés. Además, para mejorar el código de adquisición de las variables y la implementación de los nuevos sensores, así como para emitir alertas por otros medios, se aplicaron conocimientos de fundamentos en programación y circuitos digitales.

Por otro lado, los conocimientos en termodinámica y las asignaturas del área médico-biológica, como bioquímica y biología celular y tisular, permitieron comprender la importancia del ciclo de los sistemas de enfriamiento en el área de investigación. Esto se debió a la comprensión del principio de funcionamiento de los sistemas de refrigeración, adquirido en termodinámica, y a la conciencia sobre la importancia y seguridad en el manejo del contenido de dichos sistemas, gracias a las asignaturas del área médico-biológica

Finalmente, para el desarrollo del sistema de visualización de la información, los conocimientos adquiridos durante mi participación en el proyecto DGAPA-PAPIIT TA100822 “Sistema para la evaluación psicofisiológica del estrés” fueron fundamentales. Estos conocimientos me permitieron plantear el algoritmo para el envío, recepción y despliegue de la información. Además, abordé técnicas de programación que optimizaron la comunicación entre el módulo de medición y la plataforma de despliegue. También diseñé algoritmos para emitir alertas cuando los valores de interés se encontrarán fuera de rango.

## Bibliografía

- [1] Instituto Nacional de Medicina Genómica, «El instituto,» INMEGEN, [En línea]. Available: <https://www.inmegen.gob.mx/el-instituto/>. [Último acceso: Marzo 2023].
- [2] Instituto Nacional de Medicina Genómica , «Misión, Visión y Valores,» INMEGEN. [En línea]. [Último acceso: Marzo 2023].
- [3] Biotempak, «Biotempak,» [En línea]. Available: <https://biotempak.com/cadena-de-frio/>.
- [4] OMS y OPS, «Taller sobre la planificación, administración y evaluación. Módulo 3 Cadena de frío,» Oficina Sanitaria Panamericana , 2013.
- [5] World Health Organization (WHO), «Perception of health workers on the use of remote temperature monitoring systems,» 2017.
- [6] VWR, «Avantor,» Traceable® Excursion-Trac™ USB Datalogging ULT Freezer Thermometers, [En línea]. Available: <https://us.vwr.com/store/product/21084730/vwr-traceable-excursion-tractm-usb-datalogging-ult-freezer-thermometers>. [Último acceso: Octubre 2022].
- [7] Thermo Fisher Scientific, «Fisher Scientific,» Fisherbrand™ TraceableLIVE™ Ultra-Low Temperature WiFi Datalogging Thermometer with Remote Notification, [En línea]. Available: <https://www.fishersci.ie/shop/products/traceablelive-wifi-datalogging-refrigerator-freezer-thermometer-remote-notification-1/p-6824015>. [Último acceso: Octubre 2022].

## Anexos

### Sección de código de programación en Arduino sin Bot de Telegram

```
/*PROGRAMA PARA SENSAR TEMPERATURA Y HUMEDAD, ASÍ COMO LA TRANSMISIÓN DE
INFORMACIÓN*/

/*INFORMACIÓN ACERCA DE LOS PINES DE CONEXIÓN EN LA TARJETA
I2C Pines SCL=D1, SDA=D2
DS18B20 Pin de datos en D1 resistencia de 4.7k entre V y D1
BME280 SCL=D1, SDA=D2
MAX31865 CS = D0, SD1 = D3, SD0 = D5, CLK = D6
LCD SCL=D1, SDA=D2
led en D7
puerta D8*/

/*//////////////////// CÓDIGOS DE ERROR //////////////////////
Error 20 problema de conexión con el servidor
Error 30 desconexión de red
Error 40 no se detectó ningún sensor
Error 50 el sensor funcionaba bien y se generó un problema
*/

//////////////////// BIBLIOTECAS PARA LOS SENSORES, PLACA Y CONEXIÓN //////////////////////
// Placa y conexión
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
WiFiClient wifiClient;
#include <ESP8266HTTPClient.h>
#include <WiFiUdp.h>
// Sensores
#include <OneWire.h> // 18b20
#include <DallasTemperature.h> // 18b20
#include <Adafruit_Sensor.h> // bme280
#include <Adafruit_BME280.h> // bme280
#include <Adafruit_MAX31865.h> // max31865
#include <LiquidCrystal_I2C.h> // lcd
#include <Wire.h> // lcd
#include <EEPROM.h> // Memoria

////////////////////CONFIGURACIÓN DE WIFI //////////////////////
// Link de conexión
String Link = "http://grafana.inmegen.gob.mx:8080/api_temp/";

//////////////////// CONFIGURACIÓN DE SENSORES Y CREACIÓN DE OBJETOS
////////////////////
```

```

// DS18B20 sensor temperatura refrigeradores y congeladores
#define ONE_WIRE_BUS 5
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature DS18B20(&oneWire);
DeviceAddress Ds18b20Thermometer;

// BME280 sensor de humedad
#define SEALEVELPRESSURE_HPA (1013.25)
Adafruit_BME280 bme; // I2C

// MAX31865 sensor para pt100
Adafruit_MAX31865 max1 = Adafruit_MAX31865(16, 0, 14, 12);
#define RREF 430.0

// Inicialización de LCD 16x2 con I2C
LiquidCrystal_I2C lcd(0x27,16,2);

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////// ELEMENTOS PARA MODIFICAR
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////// RED
// Descomentar datos de red para pruebas
//const char* ssid      = "visitas";
//const char* password = "9HdE-yL+";

// Datos de red para conexión
const char* ssid      = "Red-fria2";
const char* password = "1nm3g3n20p2";

String nombre = "esp.2LDG-035"; // carpeta.subcarpeta.nombre_de_sensor

//En la línea anterior se especifica la ruta y nombre del sensor

// Variable para enviar información a GRAFANA
bool sensar_puerta = 0, sensar_errores = 1;
int tempEsperada = 0; // Refrigerador 5, congelador -20 ultracongelador -70

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////// VARIABLES

```

```

// Variable puerta
const int PuertaPin = 15;
int PuertaState = 0, PuertaTiempo=0, p = tempEsperada;

// Variable sensores (Se inicializan en cero)
bool sensor_18b20 = 0, sensor_bme280 = 0, sensor_pt100 = 0;
bool sensor_error = 0;

// Variables temperatura y humedad
float humedad, temp;

// Variable de red
String Mac, ip;

// Variables para mostrar en LCD
String linea1 = "";
String linea2 = "";

// Variables control de errores
int i=0, error=0, tempUltima=0;

// Variables LED
const int LED=13;

// Servidor
int servidor=1, Cserv=0;

// Control de tiempos
unsigned long tiempoTranscurrido = 0, tiempoInicial=0, tiempoFinal=0;

////////////////////// CONFIGURACIÓN INICIAL ////////////////////////
void setup() {
  // Configuración de leds
  pinMode(PuertaPin, INPUT);
  pinMode(LED_BUILTIN, OUTPUT); // Configura como salida
  pinMode(LED, OUTPUT); // Configura como salida
  digitalWrite(LED_BUILTIN, LOW); // Apaga led de la placa

  // Inicia Comunicación Serial
  Serial.begin(115200); // Inicia comunicación serial
  delay(500); // Espera conexión Serial
  /*No se ocupa un while para validar la comunicación Serial porque no en
  todos los casos se podrá establecer*/
}

```

```

// Inicializa comunicación a LCD
lcd.init();           // Inicializa
lcd.backlight();     // Enciende el backlight

// Inicia la conexión a la red WIFI
digitalWrite(LED, LOW);           // Enciende
el LED (Indica conectividad a la red)
WiFi.mode(WIFI_STA);             //
Configuración del WIFI como estación (Dispositivo que se conecta a un AP)
WiFi.begin(ssid, password);      // Inicia
la conexión a la red configurada
Serial.println("");              // Mensaje
Serial.println("SISTEMA DE MONITOREO DE LA RED FRÍA INMEGEN"); // Mensaje
de inicio
LCD_text("MONITOREO DE LA", "RED FRIA INMEGEN"); // Mensaje
de inicio LCD
delay(3000);                     // Tiempo
de 3s para mostrar mensaje
Serial.print("Conectando a WiFi: "); // Mensaje
Inicio de conexión WIFI
Serial.println(ssid);
LCD_text("Conectando a", ssid);   // Mensaje
Inicio de conexión WIFI en LCD
int t=0;
while(WiFi.status() != WL_CONNECTED) { // Ciclo para
validar la conexión del WIFI
Serial.print(".");                // Mensaje de
espera a conexión de red
delay(500);                       // Retardo para
imprimir mensaje de conexión
t++;
if (t> 360){
Serial.println(".");              //
Mensaje
Serial.println("Tiempo máximo alcanzado. Reiniciando ESP..."); //
Mensaje
ESP.reset();                       //
Reinicio del código
}
}

// Mensaje de conexión de éxito e información de red
Serial.println("");                // Imprime un espacio para formato
Serial.println("WiFi conectado"); // Mensaje de conexión establecida

```



```

    Mac = WiFi.macAddress();           // Obtiene la información MAC de la
    PLACA
    ip = WiFi.localIP().toString();    // Obtiene la IP de la red
    Serial.print("IP address: ");      // Imprime mensaje de IP de conexión
    Serial.println(WiFi.localIP());    // Imprime dirección IP
    Serial.print("MAC: ");             // Imprime mensaje de MAC
    Serial.println(Mac);               // Imprime dirección MAC
    LCD_text(ip, Mac);                 // Imprime información de la conexión
    digitalWrite(LED, HIGH);          // Apaga el led indicando éxito en la
    conexión
    delay(5000);                       // Retardo de 5 segundos antes de
    iniciar la búsqueda de sensores

    //Búsqueda de sensores
    digitalWrite(LED, LOW);            // Enciende el led indicando la
    búsqueda de sensores

    // Configuración de memoria y lectura del último del estado del servidor
    EEPROM.begin(15);
    servidor = EEPROM.read(0);         // Lee el último
    registro del servidor
    Serial.print("Código del estado del servidor: "); // Mensaje de estado
    del servidor
    Serial.println(servidor);          // Código de error
    escaneo_sensores();               // Inicia escaneo de
    los sensores
}

////////// CICLO //////////
void loop() {
    tiempoTranscurrido = 0;           // Reinicia el contador de tiempo
    tiempoInicial = millis();         // Asigna el tiempo de inicio
    // Válida conexión de sensores y obtiene la información de los sensores
    Serial.println("=====
    ====="); // Se imprime línea de formato

```

El resto del código no se comparte por temas de confidencialidad del Instituto Nacional de Medicina Genómica de México

## Sección de código de programación en Arduino con Bot de Telegram

```
/*PROGRAMA PARA SENSAR TEMPERATURA Y HUMEDAD, ASÍ COMO LA TRANSMISIÓN DE
INFORMACIÓN*/

/*INFORMACIÓN ACERCA DE LOS PINES DE CONEXIÓN EN LA TARJETA
I2C Pines SCL=D1, SDA=D2
DS18B20 Pin de datos en D1 resistencia de 4.7k entre V y D1
BME280 SCL=D1, SDA=D2
MAX31865 CS = D0, SD1 = D3, SD0 = D5, CLK = D6
LCD SCL=D1, SDA=D2
led en D7
puerta D8*/

/*//////////////////// CÓDIGOS DE ERROR //////////////////////
Error 20 problema de conexión con el servidor
Error 30 desconexión de red
Error 40 no se detectó ningún sensor
Error 50 el sensor funcionaba bien y se generó un problema
*/

//////////////////// BIBLIOTECAS PARA LOS SENSORES, PLACA Y CONEXIÓN //////////////////////
// Placa y conexión
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
WiFiClient wifiClient;
#include <ESP8266HTTPClient.h>
#include <WiFiUdp.h>
// Telegram
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
WiFiClientSecure client;
// Sensores
#include <OneWire.h> // 18b20
#include <DallasTemperature.h> // 18b20
#include <Adafruit_Sensor.h> // bme280
#include <Adafruit_BME280.h> // bme280
#include <Adafruit_MAX31865.h> // max31865
#include <LiquidCrystal_I2C.h> // lcd
#include <Wire.h> // lcd
#include <EEPROM.h> // Memoria

////////////////////CONFIGURACIÓN DE WIFI //////////////////////
// Link de conexión
String Link = "http://grafana.inmegen.gob.mx:8080/api_temp/";
```

```

////////// CONFIGURACIÓN DE SENSORES Y CREACIÓN DE OBJETOS
//////////
// DS18B20 sensor temperatura refrigeradores y congeladores
#define ONE_WIRE_BUS 5
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature DS18B20(&oneWire);
DeviceAddress Ds18b20Thermometer;

// BME280 sensor de humedad
#define SEALEVELPRESSURE_HPA (1013.25)
Adafruit_BME280 bme; // I2C

// MAX31865 sensor para pt100
Adafruit_MAX31865 max1 = Adafruit_MAX31865(16, 0, 14, 12);
#define RREF 430.0

// Inicialización de LCD 16x2 con I2C
LiquidCrystal_I2C lcd(0x27,16,2);

////////////////////////////////////
////////////////////////////////////
//////////////////////////////////// ELEMENTOS PARA MODIFICAR
////////////////////////////////////

//////////////////////////////////// RED //////////////////////////////////////
// Descomentar datos de red para pruebas
//const char* ssid = "visitas";
//const char* password = "9HdE-yL+";

// Datos de red para conexión
const char* ssid = "IZZI-3324";
const char* password = "xueAKMuh";

String nombre = "esp.2LDG-035"; // carpeta.subcarpeta.nombre_de_sensor
//En la línea anterior se especifica la ruta y nombre del sensor

// Variable para enviar información a GRAFANA
bool sensar_puerta = 1, sensar_errores = 1;
int tempEsperada = 5;

//////////////////////////////////// TELEGRAM //////////////////////////////////////
// Los datos de telegram deben ser proporcionados por el encargado del
laboratorio

```

```

#define BOTtoken "5389070183:AAHA9CFCNWdV3dxU5tDWlAIouQuk0uzh4J0" //
BOTtoken obtenido de Telegram
String chat_id = "1565407662"; // ID chat obtenido de Telegram
String equipo = "pruebaN"; // Número de control de IB (todo en mayúsculas)
// TLimite es el valor máximo de temperatura que alcanzará el equipo para
enviar alertas
// TiempoLimiteT es el tiempo que debe de transcurrir fuera del rango de
temperatura para enviar las alertas
// TiempoLimiteP es el tiempo que debe de transcurrir con la puerta abierta
para enviar las alertas
float TLimite = 15, TiempoLimiteT= 10, TiempoLimiteP=2;
String text; // Variable para cargar información de los mensajes

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////

//////////////////////////////////// VARIABLES //////////////////////////////////
// Variable puerta
const int PuertaPin = 15;
int PuertaState = 0, p = tempEsperada, PuertaTiempo=0;

// Variable sensores (Se inicializan en cero)
bool sensor_18b20 = 0, sensor_bme280 = 0, sensor_pt100 = 0;
bool sensor_error = 0;

// Variables temperatura y humedad
float humedad, temp;

// Variable de red
String Mac, ip;

// Variables para mostrar en LCD
String linea1 = "";
String linea2 = "";

// Variables control de errores
int i=0, error=0, tempUltima=0;

// Variables LED
const int LED=13;

// Control de temperatura
int TiempoFR=0;

```

```

String TLimiteS;
char TLimiteC[6];

// Servidor
int servidor=1, Cserv=0;

// Control de tiempos
unsigned long tiempoTranscurrido = 0, tiempoInicial=0, tiempoFinal=0;

// Telegram
UniversalTelegramBot bot(BOTtoken, client);

////////// CONFIGURACIÓN INICIAL //////////
void setup() {
  // Configuración de leds
  pinMode(PuertaPin, INPUT);
  pinMode(LED_BUILTIN, OUTPUT);    // Configura como salida
  pinMode(LED, OUTPUT);           // Configura como salida
  digitalWrite(LED_BUILTIN, LOW);  // Apaga led de la placa

  // Inicia Comunicación Serial
  Serial.begin(115200);            // Inicia comunicación serial
  delay(500);                     // Espera conexión Serial
  /*No se ocupa un while para validar la comunicación Serial porque no en
  todos los casos se podrá establecer*/

  // Inicializa comunicación a LCD
  lcd.init();                     // Inicializa
  lcd.backlight();                // Enciende el backlight

  // Inicia la conexión a la red WIFI
  digitalWrite(LED, LOW);         // Enciende
  el LED (Indica conectividad a la red)
  WiFi.mode(WIFI_STA);           //
  Configuración del WIFI como estación (Dispositivo que se conecta a un AP)
  WiFi.begin(ssid, password);    // Inicia
  la conexión a la red configurada
  Serial.println("");             // Mensaje
  Serial.println("SISTEMA DE MONITOREO DE LA RED FRÍA INMEGEN"); // Mensaje
  de inicio
  LCD_text("MONITOREO DE LA", "RED FRIA INMEGEN"); // Mensaje
  de inicio LCD
  delay(3000);                    // Tiempo
  de 3s para mostrar mensaje

```

```

    Serial.print("Conectando a WiFi: ");           // Mensaje
Inicio de conexión WIFI
    Serial.println(ssid);
    LCD_text("Conectando a", ssid);                 // Mensaje
Inicio de conexión WIFI en LCD
    int t=0;
    while(WiFi.status() != WL_CONNECTED) {        // Ciclo para
validar la conexión del WIFI
        Serial.print(".");                         // Mensaje de
espera a conexión de red
        delay(500);                               // Retardo pra
imprimir mensaje de conexión
        t++;
        if (t> 360){
            Serial.println(".");                  //
Mensaje
            Serial.println("Tiempo máximo alcanzado. Reiniciando ESP..."); //
Mensaje
            ESP.reset();                          //
Reinicio del código
        }
    }

    // Mensaje de conexión de éxito e información de red
    Serial.println("");                           // Imprime un espacio para formato
    Serial.println("WiFi conectado");             // Mensaje de conexión establecida
    Mac = WiFi.macAddress();                      // Obtiene la información MAC de la
PLACA
    ip = WiFi.localIP().toString();              // Obtiene la IP de la red
    Serial.print("IP address: ");                // Imprime mensaje de IP de conexión
    Serial.println(WiFi.localIP());              // Imprime dirección IP
    Serial.print("MAC: ");                       // Imprime mensaje de MAC
    Serial.println(Mac);                         // Imprime dirección MAC
    LCD_text(ip, Mac);                           // Imprime información de la conexión
    digitalWrite(LED, HIGH);                    // Apaga el led indicando éxito en la
conexión
    client.setInsecure();                        // Para conexión con telegram
    delay(5000);                                 // Retardo de 5 segundos antes de
iniciar la búsqueda de sensores

    //Búsqueda de sensores
    digitalWrite(LED, LOW);                     // Enciende el led indicando la
búsqueda de sensores

    // Configuración de memoria y lectura del último del estado del servidor

```

```

EEPROM.begin(15);
servidor = EEPROM.read(0);           // Lee el último
registro del servidor
Serial.print("Código del estado del servidor: "); // Mensaje de estado
del servidor
Serial.println(servidor);           // Código de error
// Válida si ya hay cambio en memoria en los valores generado por el
usuario
byte value = EEPROM.read(1);        // Lee código en
dirección 7 de memoria
if (value == 79){                   // Válida el código
  TLimiteS="";                       // Limpia la variable
  for(int j=2; j<8; j++){           // Carga la información
    value = EEPROM.read(j);         // Lee valor de memoria
    TLimiteC[j-1]=value;           // Almacena la
información
    TLimiteS.concat(TLimiteC[j-1]); // Concatena el dato
  }
  float TLimite2 = TLimiteS.toFloat(); // Da formato al valor
de memoria
  if (TLimite2 < 100 and TLimite2 > -100) // Valida el rango
    TLimite=TLimite2;               // Configura el límite
}
escaneo_sensores();                // Inicia escaneo de
los sensores
}

///////////////////////////////// CICLO ///////////////////////////////////
void loop() {
  tiempoTranscurrido = 0;           // Reinicia el contador de tiempo
  tiempoInicial = millis();        // Asigna el tiempo de inicio
  // Analiza alertas

```

El resto del código no se comparte por temas de confidencialidad del Instituto Nacional de Medicina Genómica de México