



# Tecnológico de Monterrey

## **Entrega 3: Implementación del prototipo** **TC1004B: Implementación de Internet de las Cosas**

Alan Estiel Aguirre Mohar  
Tecnológico de Monterrey, campus Ciudad de México  
A01657644@itesm.mx

Alejandro Hernández Ramos  
Tecnológico de Monterrey, campus Ciudad de México  
A01658969@itesm.mx

Diego Reyna Reyes  
Tecnológico de Monterrey, campus Ciudad de México  
A01657387@itesm.mx

Jonatan Hernández García  
Tecnológico de Monterrey, campus Ciudad de México  
A01653004@itesm.mx

Manuel Hernández Bravo  
Tecnológico de Monterrey, campus Ciudad de México  
A01652985@itesm.mx

Noviembre 2020

## 1. Índice

1.	<b>Índice</b>	2
2.	<b>Resumen ejecutivo</b>	3
3.	<b>Introducción</b>	4
	3.1. Análisis descriptivo	4
	3.2. Implementación del diseño conceptual	5
	3.3. Sensores de MATLAB y Packet Tracer	7
	3.4. Implementación de la solución	8
4.	<b>Desarrollo</b>	10
	4.1. Sistemas Digitales	10
	4.2. Bases de Datos	23
	4.3. Recursos de un Sistema computacional	26
	4.4. Diseño Interactivo	28
	4.5. Administración de proyectos	30
	4.6. Internet de las Cosas	35
5.	<b>Conclusiones</b>	38
	5.1. Conclusiones por módulo	38
	5.2. Conclusiones individuales	38
6.	<b>Bibliografía</b>	40
7.	<b>Anexos</b>	41

## **2. Resumen ejecutivo**

### **Problema:**

En la actualidad es necesario que los ciudadanos y ciudadanas tengan acceso a la información que respecta a su salud. En el caso de la zona capitalina, uno de los elementos más importante es la calidad del aire. Esta no cumple con los estándares para obtener una buena calidad de vida. Igualmente, el acceso a los datos es un proceso que toma tiempo y que desalienta a la población interesada en obtenerlos. Esta falta de información ocasiona pérdidas económicas y humanas al no permitir que se tomen medidas de prevención inmediatas para evitar entrar en contingencia ambiental, que tiene un efecto en la actividad económica de la urbe.

### **Solución:**

Es por esto por lo que es importante permitir el acceso a la información a las personas, con el objetivo de que cada individuo pueda tomar decisiones basadas en datos sobre la calidad de aire. Esto se obtendría a través de una página web debido a la gran cantidad de gente que usa dispositivos tecnológicos en la actualidad.

### **Propuesta:**

Por esto se plantea realizar una simulación y una base de datos que permitan el acceso rápido a los datos, fáciles de entender. Igualmente se ofrecerán los datos históricos para poder detectar patrones y realizar las medidas preventivas.

### **Beneficios:**

Aplicar nuestra solución nos permitirá incluir publicidad en la página web, lo que significa un ingreso asegurado debido a la gran cantidad de gente afectada por la calidad del aire. Igualmente nos permitirá mejorar nuestra imagen como institución al realizar esta actividad de interés público.

### 3. **Introducción**

#### 3.1. **Análisis descriptivo**

##### **Objetivos y alcances:**

Con el desarrollo del reto se pretende diseñar, elaborar e implementar una base de datos para poder simular los comportamientos del estado del aire en la CDMX para así poder advertir sobre posibles contingencias, logrando con ello la oportunidad para optimizar la calidad de vida de los ciudadanos.

Por otra parte, se simularán sensores que puedan hacer un trabajo similar a los que se usaron originalmente para la obtención de datos.

Para la función de los objetivos anteriores, se necesita de una red con una arquitectura IoT que nos permita acceder y hacer uso de nuestros datos con un flujo de trabajo optimizado.

##### **Antecedentes**

El auge de muchas tecnologías a principios del siglo XX trajo consigo grandes cambios en el estilo de vida de las personas y muchas comodidades para su día a día, lo cual favoreció a que la población creciera y por ende las demandas en cuanto a recursos naturales y por ende la producción de contaminantes nocivos para la salud. Fue hasta 1987 cuando la OMS publicó directrices sobre calidad del aire y desde entonces más investigaciones se han hecho al respecto. Actualmente la revisión que nos rige es la revisión del 2005 en la cual se han establecido los parámetros para determinar la calidad del aire. El primer instrumento de monitoreo de la calidad del aire se instaló entre 1967 y 1973 y desde entonces se han ido mejorando e implementando parámetros señalados por la OMS y adoptados por las normas mexicanas.

##### **Descripción del problema**

La salud de las personas está estrechamente relacionada con la calidad del aire que respiran, es por ello que la presencia de contaminantes en el aire representa un riesgo para el bienestar de la sociedad, al ser un factor de riesgo para desarrollar enfermedades respiratorias y cardiovasculares; de acuerdo con las Directrices de la OMS si se reduce la contaminación con partículas de 70 a 20 microgramos por metro cúbico sería posible reducir en un 15% el número de muertes provocadas por contaminación.

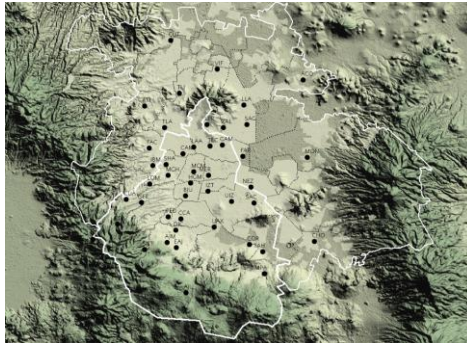
Debido a la estrecha relación que existe entre la calidad del aire que se respira y la salud de las personas es importante la implementación de un sistema de monitoreo de la calidad del aire que compare las mediciones obtenidas en tiempo real con los parámetros establecidos por las Normas Oficiales Mexicanas de Salud Ambiental como adecuados, para así saber si es necesario implementar medidas más estrictas por parte de autoridades para la reducción de contaminantes.

Esto es una respuesta a un interés internacional por la calidad del aire, ya que no solo afecta la salud de las personas, sino que también afecta al calentamiento global, ya que muchas de las partículas son de efecto invernadero. Igualmente se ha detectado que la calidad del aire está estrechamente relacionada con la posición geográfica en la que se encuentra, especialmente si uno se encuentra en zonas urbanas, como la Ciudad de México.

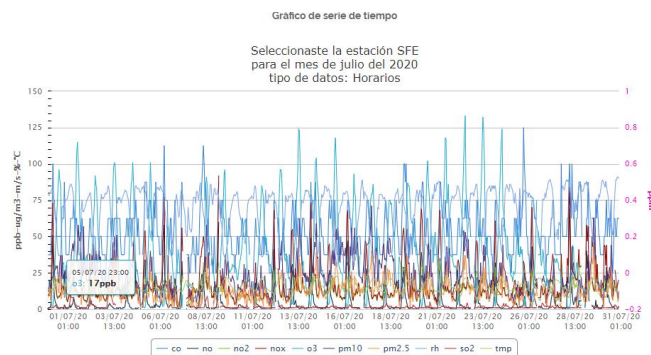
##### **Modelo actual**

Hay 44 estaciones que conforman el Sistema de Monitoreo Atmosférico, 28 en la Ciudad de México y 16 en el Estado de México, que miden diferentes contaminantes como

SO<sub>2</sub>, CO, NO<sub>2</sub>, O<sub>3</sub>. Cumplen con normas de la EPA y la NOM para cada contaminante. Se pueden visualizar datos especificando el contaminante, la estación y el intervalo de tiempo.



Se puede acceder a los datos a través de la página web: <http://www.aire.cdmx.gob.mx/default.php> Para poder descargar los datos de las estaciones de monitoreo es sencillo. Pero, al querer generar gráficas de los datos, es necesario pasar por muchos pasos. Esta gran cantidad de pasos desanima al usuario debido al tiempo que se debe dedicar, al igual a la poca claridad de los parámetros a graficar. A continuación, se muestra un ejemplo de una de las gráficas que se pueden generar en el sitio.



### 3.2. Implementación del diseño conceptual

#### 3.2.1. Requerimientos Funcionales y No Funcionales

##### Requisito funcional

El sistema en cuestión creará reportes con base en la información recogida por sensores de monóxido de carbono que serán instalados en diversos puntos de la Ciudad de México, para determinar la calidad de aire en un periodo de tiempo determinado.

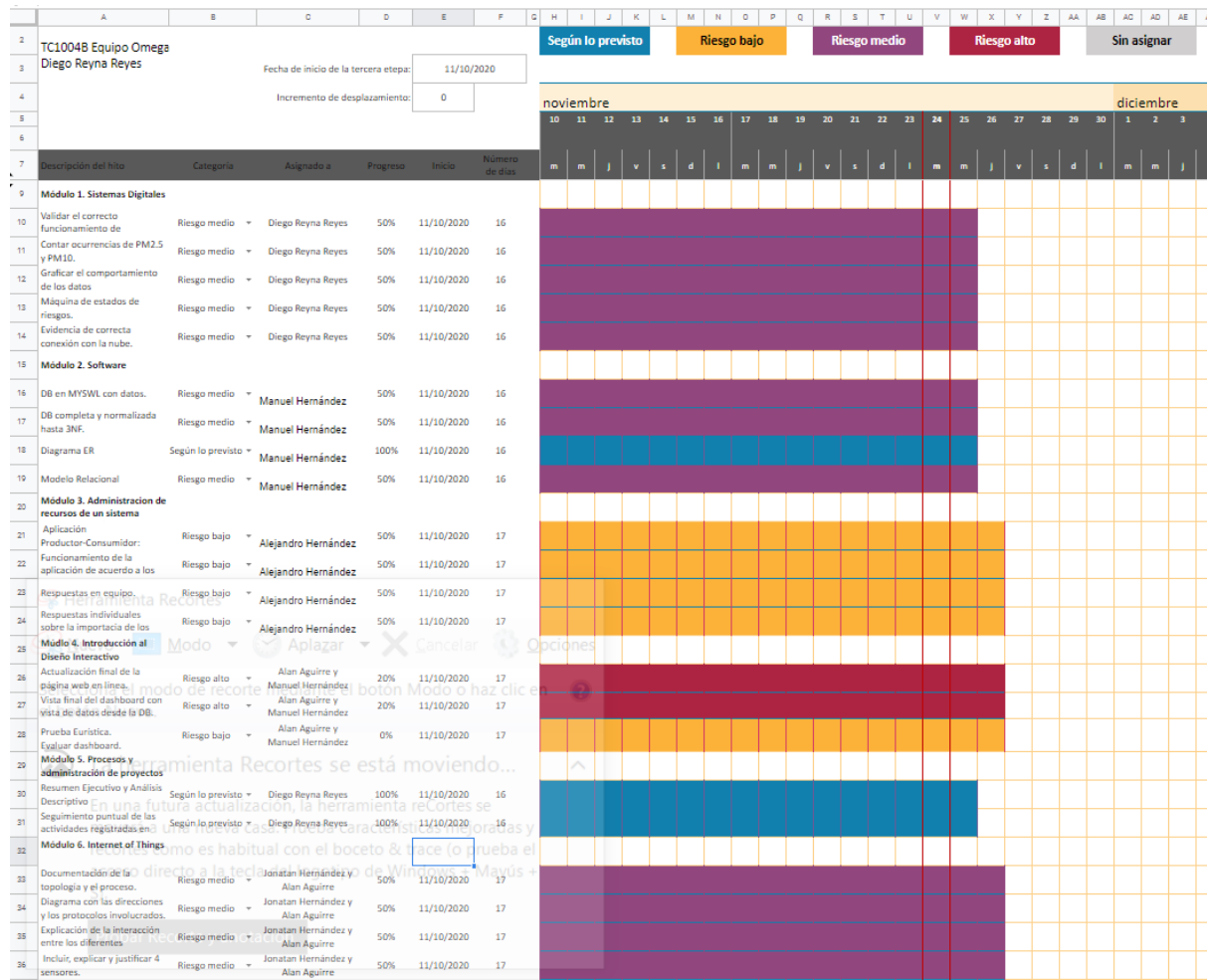
##### Requisito no funcional

El backend del sistema en cuestión se programará en PHP 7 de manera segura para evitar ataques e intrusiones maliciosas.

En cuanto al frontend, se utilizará la versión de HTML 5, acompañada del framework Bootstrap, el cual nos permite realizar un sitio limpio, amigable con el usuario y responsivo para que pueda ser visualizado en un mayor número de dispositivos.

##### Administración de las dimensiones

Para poder acoplar y trabajar los diferentes procesos de cada sección con las diferentes dimensiones (Alcance, tiempo y presupuesto), se implementó un diagrama de Gantt en donde se llevó un registro de las diferentes tareas a realizar con su debido encargado, el riesgo pertinente, su progreso y fecha inicial y final estimada. En este caso, por cuestiones académicas no tomamos en cuenta el presupuesto (en su valor monetario).



Gracias a este diagrama el equipo pudo visualizar de forma más ilustrada los objetivos, su importancia y el tiempo que se debería tomar cada uno. En consecuencia, fue más fácil notar si un retraso de alguna tarea se presentaba y fácilmente cambiar, aseverar y aplazar fechas. La comunicación entre el equipo sobre el progreso en sus respectivas tareas es un factor sumamente importante para poder notar estos retrasos.

### 3.2.2. Flujo de los datos

#### Envío de datos dentro de PT

Módulo 6: Uso y envío de datos dentro de PT, usando SCB a través de conexión alámbrica

#### Envío de datos de PT a base de Datos

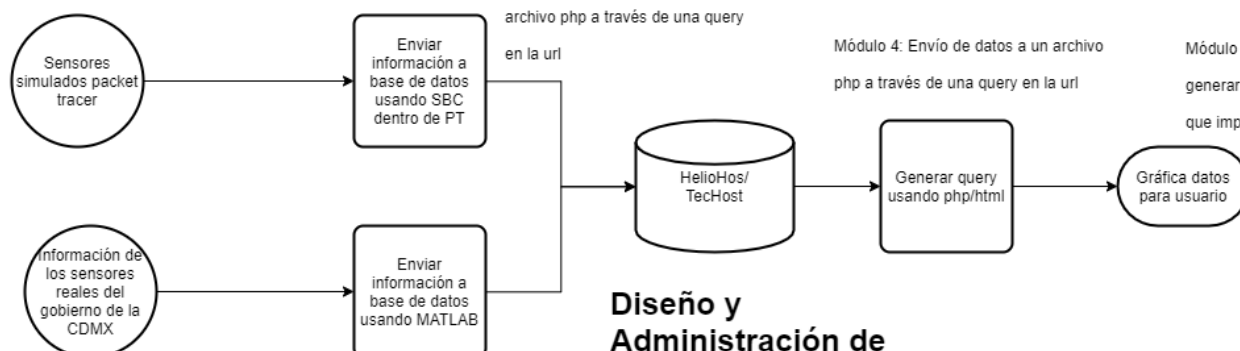
Módulo 4: Envío de datos a un archivo php a través de una query en la url

#### Envío de datos de base de Datos a Usuario final

Módulo 4: Envío de datos a un archivo php a través de una query en la url

#### Impresión de los datos

Módulo 4: Uso de php para generar un archivo html que imprima los datos



#### Envío de datos en MATLAB y SIMULINK

Módulo 1: Lectura de los datos en archivo usando bloques de SIMULINK  
Módulo 1: Diseño e implementación de un check sum, usando tablas de verdad para generar paridad

#### Diseño y Administración de la base de Datos

Módulo 2: Generación del modelo Relacional, a partir de un modelo Entidad Relación normalizado a la 3FN  
Módulo 6: Conexión a una dirección IP(TechHost) dentro de la red del campus usando una VPN

#### Administración del proyecto

Módulo 5: Coordinación de los diferentes miembros del equipo para dividir el trabajo y permitir generar la implementación  
Módulo 5: Generación del Resumen Ejecutivo y Análisis descriptivo para permitir generar el objetivo y el diseño de la implementación del proyecto

#### Diseño de la implementación.

Módulo 3: Usando como base una arquitectura von Neuman, se realizó el diseño presente, donde el envío de los datos serían los buses de entrada y salida. Al igual que el servidor en la nube (Donde se guarda la base de datos y los php) sería la memoria donde se almacenan los datos e instrucciones. Y finalmente, el administrador de Base de datos y los php actuarían de controlador

### 3.3. Sensores de MATLAB y Packet Tracer Sensores MATLAB

Los sensores de MATLAB se desarrollaron en SIMULINK, usando datos recuperados y procesados del sitio del Índice Aire y Salud de la ZMVM. Este proceso fue para eliminar los datos NULL de los archivos descargados.

Posteriormente, estos datos fueron cargados a SIMULINK, donde se hizo el promedio de los datos. Este promedio es la Lectura que genera nuestro sensor. Estos sensores usan los datos sobre PM10, PM2.5, Temperatura y Humedad Relativa. Finalmente, estos datos se enviaron a la base de datos usando una función de MATLAB dentro de SIMULINK, que se las envía a un archivo php en el servidor.

Igualmente se subieron los datos originales recuperados del sitio del Índice Aire y Salud de la ZMVM. Esto fue usando funciones de MATLAB, que envían los datos a un archivo php dentro del servidor. Estos datos están fuera del alcance del proyecto, debido a su volumen y la cantidad de estaciones que los generan.

#### Sensores PT

Por otra parte, los sensores en Packet Tracer, a diferencia de los de MATLAB, no procesaron datos reales. Se elaboró una simulación en un entorno de Packet Tracer que considera las diferentes variables necesarias para poder elaborar una simulación



cercana a la realidad. Como variables adicionales implementamos 2 coches viejos que simulan desprender CO<sub>2</sub> y Co en nuestro entorno simulado. Esto permitió que los cuatro diferentes sensores (CO, CO<sub>2</sub>, humedad y temperatura) recopilaran y mandaran datos a cuatro dispositivos SBC que se encargan de mandar los datos recibidos a Heliohost. Cada sensor trae consigo un Id para su fácil identificación en la base de datos.

### 3.4. Implementación de la solución

Con la finalidad de visualizar de una forma más intuitiva y amigable con el usuario los datos obtenidos por los sensores en diferentes estaciones se hizo una página web en la cual se puedan realizar búsquedas por medio de fecha de inicio de lectura y fecha final para ciertos sensores de diferentes fuentes, ya sea Simulink o Packet Tracer. Todo esto fue hecho en HTML, JavaScript y PHP. El principio en el que está basada la página es emplear el menor número de clics para navegar entre las páginas y en tratar de cumplir con los 10 principios de heurística para el diseño de interfaz propuestos por Jakob Nielsen.

Para subir los datos se usaron dos plataformas:

- **Packet Tracer**

Los sensores se conectan a dispositivos SBC, los cuales fueron programados en lenguaje JavaScript para ejecutar un Script para enviar los datos recibidos a nuestra base de datos en Helio Host bajo protocolo HTTP. Dicho Script es el siguiente:

```
Save Monoxide Log (JavaScript) - main.js
Open New Delete Rename Import

1* function loop() {
2  var sensorValue;
3  sensorValue = digitalRead(0) / 4;
4
5  var d = new Date();
6  var n = Math.floor(d.getTime() / 1000);
7  var date= d.getFullYear().toString()+"-"+d.getMonth().toString()+"-"+d.getDate().toString()+" "+d.getHours().toString()+" "+d.getMinutes().toString()+" "+d.getSeconds().toString();
8
9  var file = FileSystem.open('/log.txt',
10   File.WRITE | File.READ | File.APPEND);
11
12  file.print("19, " + sensorValue + ", " + date + "\n");
13  file.close();
14
15  Serial.print("19, " + sensorValue + ", " + date + "\n");
16
17  var http = new XMLHttpRequest();
18  http.open("GET", "http://rrdie.heliohost.org/insertMatlab.php?query=INSERT INTO 'Lectura' ('fecha', 'valor', 'sensor_id') VALUES ('"+date+"', '"+sensorValue+"', 19)");
19
20  delay(1000);
21 }
```

- **MATLAB y SIMULINK**

Para subir los datos se usó una función de MATLAB que se ejecutó dentro de SIMULINK. Esta función regresaba un valor numérico dependiendo del estado del envío de datos. Se decidió no implementar un algoritmo que intentara subir los datos nuevamente en caso de fallo. Esto con el objetivo de simular una red IoT más apegada a la realidad, debido a errores en las redes celulares que pueden presentarse incluso dentro de la misma Ciudad de México. El algoritmo para subir los datos es el siguiente:

```
%Sube los datos de Simulink a HelioHost
function s=subirSimulink(entrada)
    %Estructura de la entrada=pm10,pm25,temp,hum,mod,paridad,mes,numLec
    options = weboptions('ContentType','text','Timeout',15);%Configuración de la conexión
    nom_mat=strcat('PM25_2020_',num2str(entrada(7)),'.mat');%Generar nombre del archivo.mat con los datos
    data_file= load(nom_mat);%Cargar los datos del PM10 del mes en entrada(7)
    fechas=getabstime(data_file.ts);%Encontrar las fechas de todos los datos del mes
    s=2;%Caso de no paridad
    if(checkSum(entrada) && checkParidad(entrada) && entrada(8)+1<=length(fechas))%Solo entrar si hay paridad, el checkSum e
        dateIndex=entrada(8)+1;%Obtener el número de entrada final. +1 porque empieza en 0
        fecha=fechas(dateIndex);%Recuperar la fecha como una cell
        fecha=changeDate(fecha(1));%Formatear la fecha para estar en formato de datetime
        url=sprintf("http://rrdie.heliohost.org/archivosMod/insertMatlab.php?type=%s&pm10=%f&pm25=%f&temp=%f&hum=%f&date=%s",
            %Usar try en el caso que el servidor este caído
            try
                disp('Enviando a base de datos');
                messageBack = webread(url, options);%Obtener el html generado por la url
                disp(url); %Imprimir la url
                disp (messageBack);
                s=1;%Mostrar caso de subida exitoso
            catch %Si falla la subida de datos
                disp('Error de envío');
                s=0;%Mostrar caso de error
            end
        end
    end
end
```



Igualmente se subieron los datos de los sensores del Gobierno de la Ciudad de México sin ser procesados por SIMULINK. Esto para simular el gran flujo de datos que hay dentro de una red IoT grande. En este caso si se implementa una función que evita la pérdida de muchos datos. Esto con el objetivo de demostrar que se puede implementar un sistema que evite la pérdida masiva de datos. En este caso se desarrollaron 2 alternativas. La primera al detectar 3 errores de subida consecutivos espera 10 minutos para subir los siguientes datos, debido a que el servidor HelioHost suele tener errores internos que duran alrededor de 10 minutos. La segunda opción sube datos, al contrario, recupera datos de la plataforma, pero su lógica puede ser implementada para subir datos. Esta va a intentar acceder a la base de datos hasta que sea un intento exitoso.

Imagen de la primera opción:

```
% Finalmente, enviar el INSERT:
try
    messageBack = webread(url, options);%Entrar a la url
    disp(url);%Si hubo éxito se imprime la url
    num_error=0;%Reinicializar el número de errores
catch
    fprintf('Error en: %s, sensor: %i\n',fecha, sensor_id);%Imprimir error
    num_error=num_error+1;%Agregar al num_error
    if (num_error==3)%Si hay 3 errores consecutivos dejar de subir
        disp('3 errores consecutivos, esperando 10 minutos');%Imprimir mensaje de espera
        num_error=0;%Reinicializar errores
        pause(600);%Esperar
    end
    pause(10);%Esperar 10 segundo después de un error
end
% Esperar para enviar el siguiente INSERT a la BD:
pause(2);
```

Imagen de la segunda opción:

```
while (true)
    try
        mensaje=webread(url, options);%Obtener lo que regresa la página php
        break;%Salir del ciclo al tener conexión exitoso
    catch
        continue;%No salir del ciclo
    end
end
```

## 4. Desarrollo

### 4.1. Sistemas Digitales

Para cumplir todos los puntos de esta entrega se obtuvo el siguiente resultado:

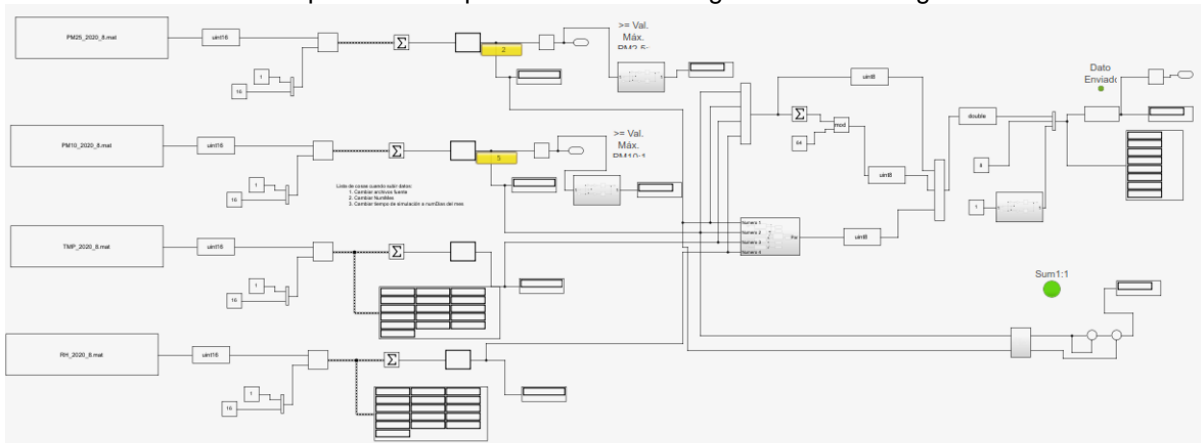


Imagen 1: Sistema Completo

#### 4.1.1. Validar el correcto funcionamiento de Humedad Relativa y Temperatura.

Para mostrar el funcionamiento de Temperatura y Humedad se decidió comprobar que el promedio calculado fuera correcto. Por lo que usando un Display (1) se mostraron los 16 datos usados para el promedio, que es mostrado en un segundo Display (2).

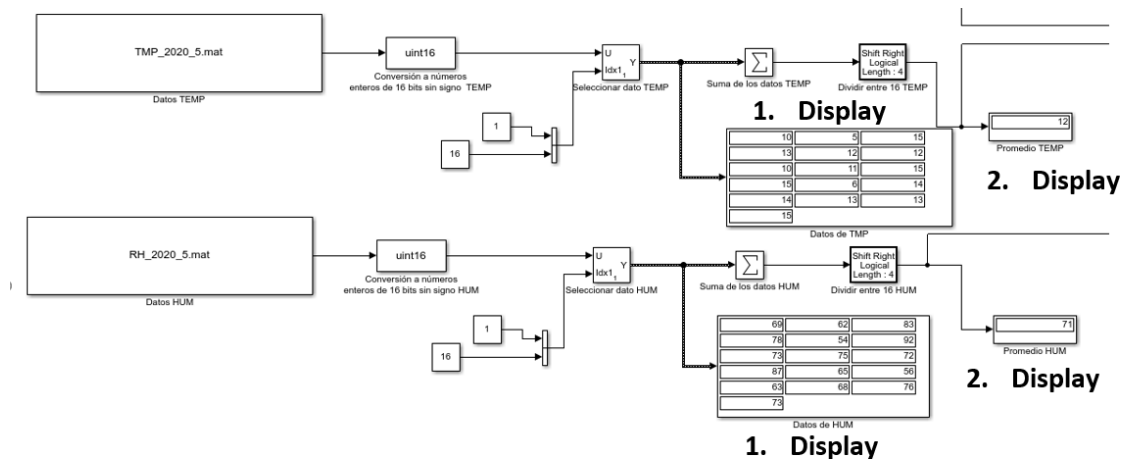


Imagen 2. Comprobación de Humedad y Temperatura

Se puede ver que los datos de TMP son 10, 13, 10, 15, 14, 15, 5, 12, 11, 6, 13, 15, 12, 15, 14 y 13. Si se saca el promedio de estos se obtiene 12.0625. En el Display se ve 12 debido a que al hacer el corrimiento de bits se pierde la parte decimal de la división.

Para HUM se tienen los datos 69, 62, 83, 78, 54, 92, 73, 75, 72, 87, 65, 56, 63, 68, 76 y 73. Si se saca el promedio de estos se obtiene 71.625. En el Display se ve 71, debido a la pérdida de precisión al usar el corrimiento de bits.

#### 4.1.2. Contar ocurrencias de PM2.5 y PM10.

Para contar las ocurrencias de PM2.5 y PM10 se usó un subsistema personalizado, sumarBool (3). Este subsistema recibe la entrada booleana de la comparación del bloque  $\geq \text{Val. Máx.}$  (4) y su salida será el número de veces que la entrada booleana haya sido 1. Esta salida se ve en el Display (5).

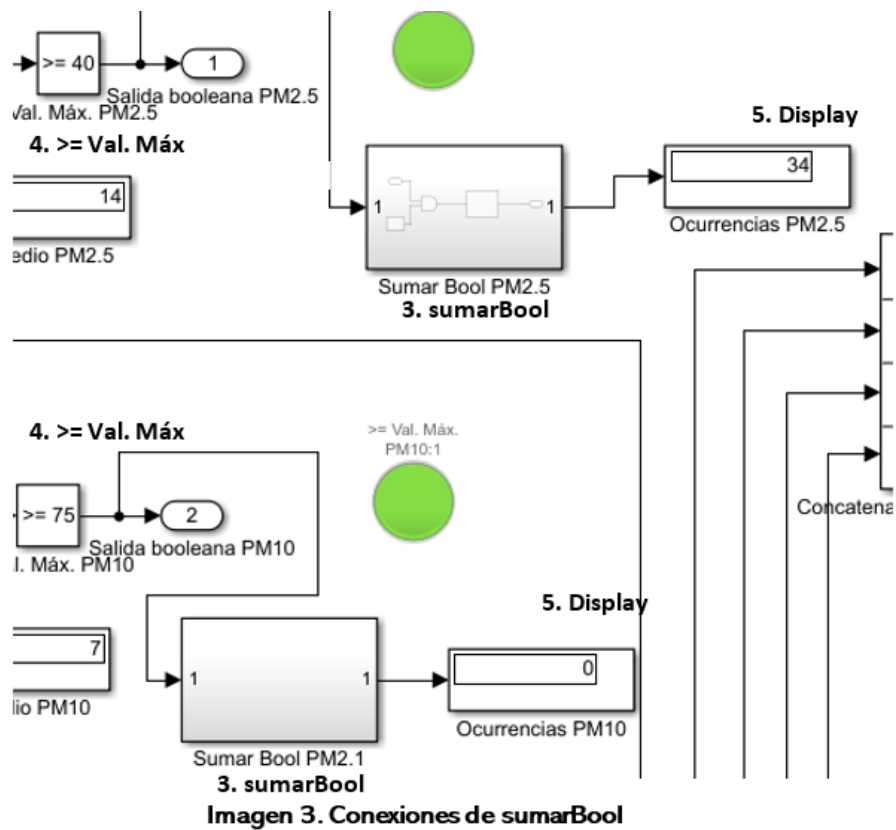


Imagen 3. Conexiones de sumarBool

Dentro del subsistema se tiene un reloj (6) que está conectado a un contador (7) por medio de una compuerta AND (8). Esta compuerta tiene el objetivo de solo enviar la señal Inc. Al contador cuando sea un flanco positivo y la entrada booleana sea 0. El reloj tiene un periodo de 1/48. Esto es ya que el reloj mantiene un flanco positivo durante la mitad de este periodo, por lo que si se ponía con un periodo de 1/24 (El periodo de entrada de datos) Este iba a contar la mitad de los datos solamente, ya que al llegar el segundo dato estaría en flanco de bajada, en vez de flanco de subida que es lo que acciona el contador. La salida del subsistema es el Cnt del contador (7).

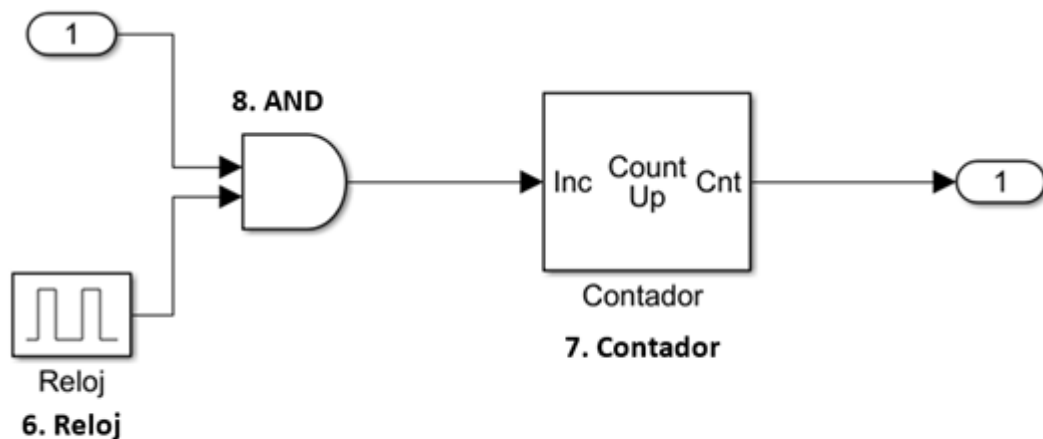
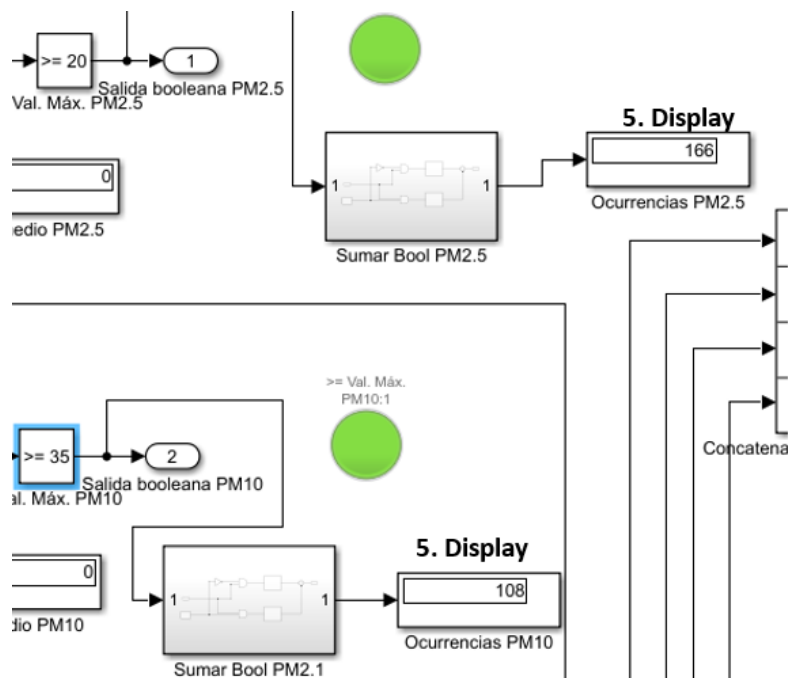


Imagen 4. Funcionamiento de sumarBool

Aquí se puede observar los Display (5) después de subir los datos del mes de agosto.

Con fines demostrativos, se bajó el límite de la comparación para este ejemplo, debido a que no había ocurrencias sobre los límites establecidos.



**Imagen 5. Resultados de las ocurrencias de Agosto**

Como se puede observar, hubo 166 ocurrencias de PM2.5 sobre 20 y 108 ocurrencias de PM10 sobre 35

#### 4.1.3. Máquina de estados de riesgos.

Para realizar la máquina de estados nos basamos en esta tabla:

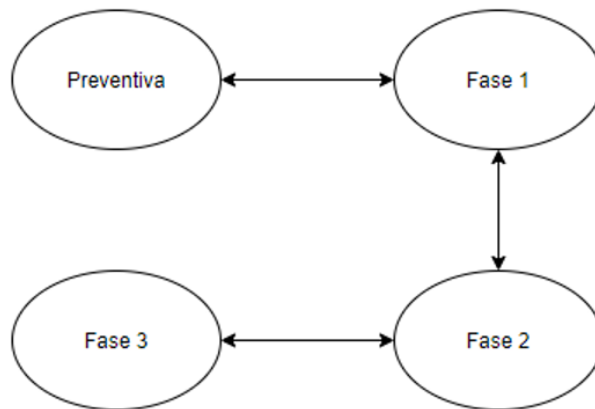
FASE	PM2.5	PM10
Preventiva	<25	<15
Fase I	<35	<25
Fase II	<45	<35
Fase III	<55	<45

**Tabla 1: Fases de la Contingencia**

Posteriormente se identificaron los estados posibles:

- Preventiva
- Fase 1
- Fase 2
- Fase 3

Luego se realizó un diagrama de estados:



**Imagen: Diagrama de Estados**

La bidireccionalidad de la flecha significa que se puede mover en ambas direcciones.

Tras esto se buscó implementar este diagrama usando FlipFlops tipo JK. Al tratarse de 4 estados, se decidió usar 2 Flip Flops Tipo JK, debido que dos bits pueden representar hasta 4 estados diferentes. Tras esto se actualizó el diagrama:



**Imagen: Diagrama de Estados Actualizado**

Luego se determinaron las entradas de nuestra máquina de estados.

Las entradas evalúan los niveles de PM10 y PM2.5. Deben evaluar el rango en el que se encuentran. Basándose en la tabla 1, se pudo determinar que para encontrar el rango se puede usar una compuerta AND, al igual que se pueden encontrar si se debe aplicar la contingencia usando una compuerta OR de los rangos. De esta forma quedaron las siguientes expresiones para cada uno de los estados.

- 00

$$C = (PM2.5 < 25) + (PM10 < 15)$$

- 01

$$D = (PM2.5 \geq 25)(PM2.5 < 35) + (PM10 \geq 15)(PM10 < 25)$$

- 10

$$E = (PM2.5 \geq 35)(PM2.5 < 45) + (PM10 \geq 25)(PM10 < 35)$$

$$F = (PM2.5 \geq 45)(PM2.5 < 55) + (PM10 \geq 35)(PM10 < 45)$$

Obtenidas estas expresiones, se redefinieron a las entradas como C, D, E y F. Habiendo obtenido esto, se realizó la siguiente tabla donde se muestran los estados en los que cambiaría el estado.

Numero de caso	Estado Inicial		Entradas				Siguiete estado	
#	A	B	C	D	E	F	A	B
1	0	0	0	1	0	0	0	1
2	0	1	0	0	1	0	1	0
3	1	0	0	0	0	1	1	1
4	1	1	0	0	1	0	1	0
5	1	0	0	1	0	0	0	1
6	0	1	1	0	0	0	0	0

**Tabla 2: Estados principales**

En la tabla los casos 1 al 3 representan un cambio de estado ascendente (Fase Preventiva hacia Fase 3) y los casos 4 al 6 representan el cambio de estado descendente (Fase 3 hacia Fase Preventiva). Para cada caso se cumple que las entradas coincidan con los parámetros de la siguiente fase, con esto se puede generar un cambio en el estado.

Después se realizó la misma tabla, pero ahora mostrando los valores que deben tener Ja, Ka, Jb y Kb para generar el cambio de estado buscado

Numero de caso	Estado Inicial		Entradas				Siguiete estado		FF JK A		FF JK B	
#	A	B	C	D	E	F	A	B	Ja	Ka	Jb	Kb
1	0	0	0	1	0	0	0	1	0	x	1	X
2	0	1	0	0	1	0	1	0	1	x	x	1
3	1	0	0	0	0	1	1	1	x	0	1	X
4	1	1	0	0	1	0	1	0	x	0	x	1
5	1	0	0	1	0	0	0	1	x	1	1	X
6	0	1	1	0	0	0	0	0	0	x	x	1

**Tabla 3: Cambios de estados**

Habiendo obtenido esto, se realizó la tabla de verdad completa tomando en cuenta que en los estados no mostrados en la tabla no se espera un cambio de estado, por lo que Ja, Ka, Jb, y Kb deben ser 0.

Número de caso	Edo. Actual		Entradas				FF JK A		FF JK B	
	A	B	C	D	E	F	Ja	Ka	Jb	Kb
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0	0
2	0	0	0	0	1	0	0	0	0	0
3	0	0	0	0	1	1	0	0	0	0
4	0	0	0	1	0	0	0	x	1	x
5	0	0	0	1	0	1	0	0	0	0
6	0	0	0	1	1	0	0	0	0	0
7	0	0	0	1	1	1	0	0	0	0
8	0	0	1	0	0	0	0	0	0	0
9	0	0	1	0	0	1	0	0	0	0
10	0	0	1	0	1	0	0	0	0	0
11	0	0	1	0	1	1	0	0	0	0

12	0	0	1	1	0	0	0	0	0	0
13	0	0	1	1	0	1	0	0	0	0
14	0	0	1	1	1	0	0	0	0	0
15	0	0	1	1	1	1	0	0	0	0
16	0	1	0	0	0	0	0	0	0	0
17	0	1	0	0	0	1	0	0	0	0
18	0	1	0	0	1	0	1	x	x	1
19	0	1	0	0	1	1	0	0	0	0
20	0	1	0	1	0	0	0	0	0	0
21	0	1	0	1	0	1	0	0	0	0
22	0	1	0	1	1	0	0	0	0	0
23	0	1	0	1	1	1	0	0	0	0
24	0	1	1	0	0	0	0	x	x	1
25	0	1	1	0	0	1	0	0	0	0
26	0	1	1	0	1	0	0	0	0	0
27	0	1	1	0	1	1	0	0	0	0
28	0	1	1	1	0	0	0	0	0	0
29	0	1	1	1	0	1	0	0	0	0
30	0	1	1	1	1	0	0	0	0	0
31	0	1	1	1	1	1	0	0	0	0
32	1	0	0	0	0	0	0	0	0	0
33	1	0	0	0	0	1	x	0	1	x
34	1	0	0	0	1	0	0	0	0	0
35	1	0	0	0	1	1	0	0	0	0
36	1	0	0	1	0	0	x	1	1	x
37	1	0	0	1	0	1	0	0	0	0
38	1	0	0	1	1	0	0	0	0	0
39	1	0	0	1	1	1	0	0	0	0
40	1	0	1	0	0	0	0	0	0	0
41	1	0	1	0	0	1	0	0	0	0
42	1	0	1	0	1	0	0	0	0	0
43	1	0	1	0	1	1	0	0	0	0
44	1	0	1	1	0	0	0	0	0	0
45	1	0	1	1	0	1	0	0	0	0
46	1	0	1	1	1	0	0	0	0	0
47	1	0	1	1	1	1	0	0	0	0
48	1	1	0	0	0	0	0	0	0	0
49	1	1	0	0	0	1	0	0	0	0
50	1	1	0	0	1	0	x	0	x	1
51	1	1	0	0	1	1	0	0	0	0
52	1	1	0	1	0	0	0	0	0	0
53	1	1	0	1	0	1	0	0	0	0
54	1	1	0	1	1	0	0	0	0	0
55	1	1	0	1	1	1	0	0	0	0
56	1	1	1	0	0	0	0	0	0	0



57	1	1	1	0	0	1	0	0	0	0
58	1	1	1	0	1	0	0	0	0	0
59	1	1	1	0	1	1	0	0	0	0
60	1	1	1	1	0	0	0	0	0	0
61	1	1	1	1	0	1	0	0	0	0
62	1	1	1	1	1	0	0	0	0	0
63	1	1	1	1	1	1	0	0	0	0

**Tabla 4: Tabla de verdad**

Los estados donde hay un cambio de estado están resaltados de color morado.

Tras esto se buscó la expresión que representara cada una de las variables Ja, Ka, Jb y Kb usando una herramienta en línea (<http://www.32x8.com/index.html>) . Se obtuvieron los siguientes mapas de Karnaugh y las siguientes expresiones:

- Ja

Mapa de Karnaugh:

	$\overline{D.E.F}$	$\overline{D.E}.F$	$\overline{D.E}.F$	$\overline{D.E}.F$	$D.E.\overline{F}$	$D.E.F$	$D.E.F$	$D.E.F$
$\overline{A.B.C}$	0	0	0	0	0	0	0	0
$\overline{A.B.C}$	0	0	0	0	0	0	0	0
$\overline{A.B.C}$	0	0	0	0	0	0	0	0
$\overline{A.B.C}$	0	0	0	1	0	0	0	0
$A.B.C$	0	x	0	0	x	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	x	0	0	0	0

$$Ja = B C' D' E F'$$

- Ka

Mapa de Karnaugh:

	$\overline{D.E.F}$	$\overline{D.E}.F$	$\overline{D.E}.F$	$\overline{D.E}.F$	$D.E.\overline{F}$	$D.E.F$	$D.E.F$	$D.E.F$
$\overline{A.B.C}$	0	0	0	0	x	0	0	0
$\overline{A.B.C}$	0	0	0	0	0	0	0	0
$\overline{A.B.C}$	x	0	0	0	0	0	0	0
$\overline{A.B.C}$	0	0	0	x	0	0	0	0
$A.B.C$	0	0	0	0	1	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0

$$Ka = B' C' D E' F'$$

- Jb

Mapa de Karnaugh:

	$\overline{D.E.F}$	$\overline{D.E.F}$	$\overline{D.E.F}$	$\overline{D.E.F}$	$D.E.F$	$D.E.F$	$D.E.F$	$D.E.F$
$\overline{A.B.C}$	0	0	0	0	1	0	0	0
$\overline{A.B.C}$	0	0	0	0	0	0	0	0
$\overline{A.B.C}$	x	0	0	0	0	0	0	0
$\overline{A.B.C}$	0	0	0	x	0	0	0	0
$A.B.C$	0	1	0	0	1	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	x	0	0	0	0

$$Jb = B' C' D E' F' + A B' C' D' E' F'$$

- Kb

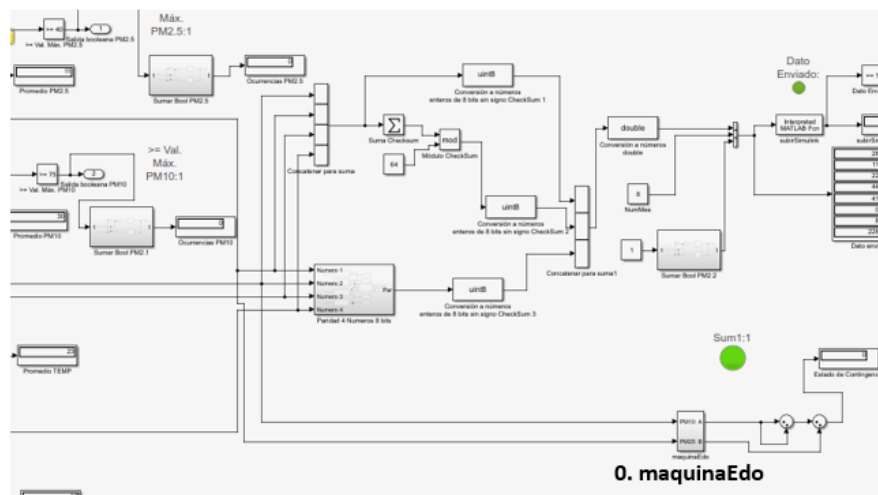
Mapa de Karnaugh:

**Map**

	$\overline{D.E.F}$	$\overline{D.E.F}$	$\overline{D.E.F}$	$\overline{D.E.F}$	$D.E.F$	$D.E.F$	$D.E.F$	$D.E.F$
$\overline{A.B.C}$	0	0	0	0	x	0	0	0
$\overline{A.B.C}$	0	0	0	0	0	0	0	0
$\overline{A.B.C}$	1	0	0	0	0	0	0	0
$\overline{A.B.C}$	0	0	0	1	0	0	0	0
$A.B.C$	0	x	0	0	x	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	0	0	1	0	0	0	0

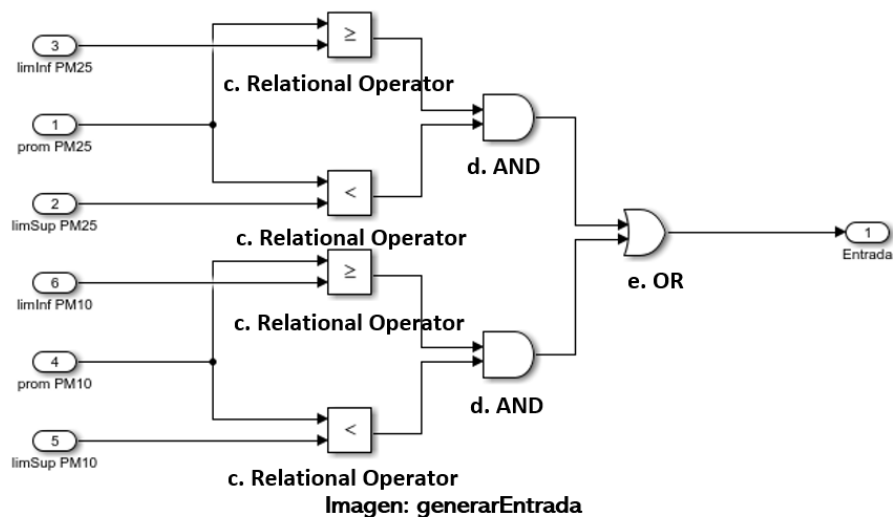
$$Kb = B C' D' E F' + A' B C D' E' F'$$

Tras obtener las expresiones se transcribió el modelo a SIMULINK. Se creó un subsistema maquinaEstados (0), cuyas entradas son el promedio de PM10 y el promedio de PM5 y sus salidas son A y B.

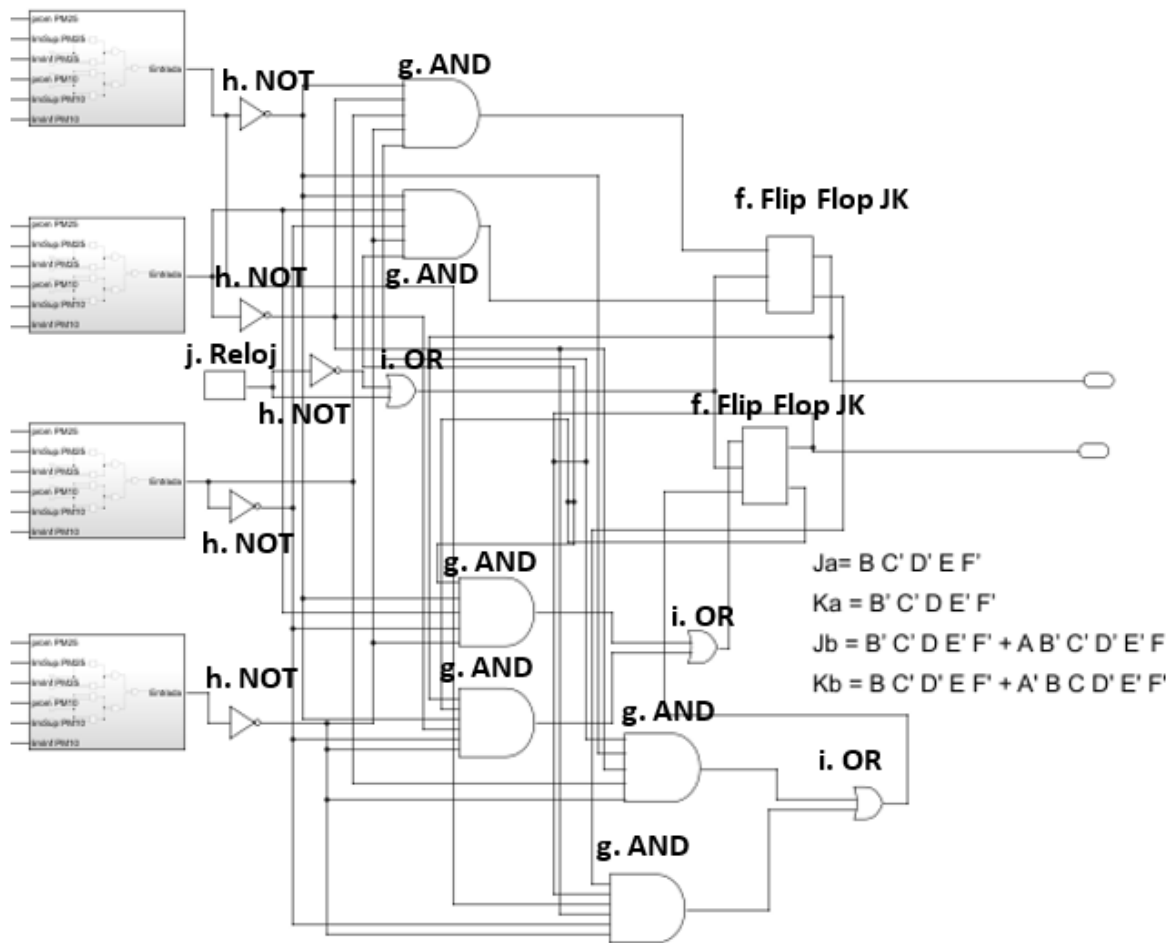


**Imagen: Máquina de estados en la simulación**

Este subsistema consiste en 4 generarEntrada (a), que recibe los promedios y los 4 límites del rango que se busca obtener por medio de un Constant (b). Compara que el valor esté dentro del rango usando Relational operators (c), al igual que usa compuertas AND (d) y OR (e) para completar la expresión.



Estas entradas podemos enviarlas a los Flip Flop JK (f) usando compuertas AND (g), NOT (h) y OR (i) conectadas como las expresiones obtenidas para Ja, Ka, Jb y Kb. Los Flip Flop JK funcionan con un Reloj (j) con periodo de 1/24, conectado a una compuerta OR (i) cuyas entradas son el Reloj (j) y su inversa, para evitar pérdida de datos. Este punto será más explicado en la sección 4.1.4.



Tras obtener los valores de A y B se transforman a un número base 10. Esto se logra usando el bloque Sum (k) para formar A+A+B que es la conversión de binario a decimal. Esto se conectará a un Display (l) y a una Lamp (m) que prenderá de color diferente dependiendo del estado.

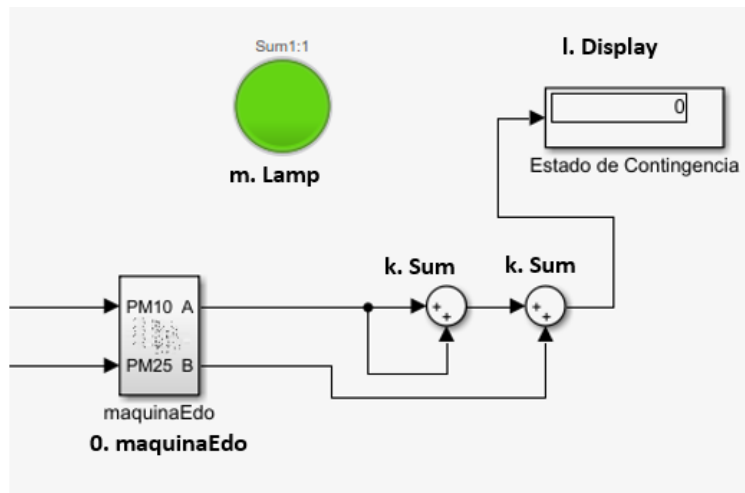


Imagen: Máquina de estados conectada

Este diseño tiene sus desventajas, principalmente, no cambiará de estado a menos que ambos parámetros estén dentro de un mismo límite de contingencia. Este problema se puede solucionar agregando caso a la tabla 2 y a la tabla 3. Sin embargo, debido a la fecha en la que se explicó el tema, no es posible arreglar estos errores en esta etapa. Estos problemas serán solucionados para la cuarta etapa.

#### 4.1.4. Evidencia de correcta conexión con la nube.

Para hacer la conexión con la nube se modificó el vector que se envió al final de la entrega pasada a partir de un vector concatenate (9). Este consistía en: Promedio de PM2.5, Promedio de PM10, Promedio de Temperatura, Promedio de Humedad Relativa, MOD 64 de la suma de los 4 datos anteriores y bit de paridad.

A este se le añaden número de mes con la Constant (10), número de entrada usando un contador (11) y un reloj (12) con periodo de 1/48. Estos datos se añaden al vector anterior usando un vector concatenate (13). Para convertir el vector anterior de uint8 a double se usa el bloque Data Type Conversion (14). Este nuevo vector se envía a un Display (15) y al bloque Integrated MATLAB Function (16). Este bloque ejecuta el script de Matlab con nombre de "subirSimulink.m". Este bloque tiene una salida que puede tener 3 valores (0,1 y 2). Esta salida se envía a un Display (17) y al bloque Compare to Constant (18) que está conectado a una Lamp (19) que nos indica si el dato fue enviado exitosamente a la base de datos.

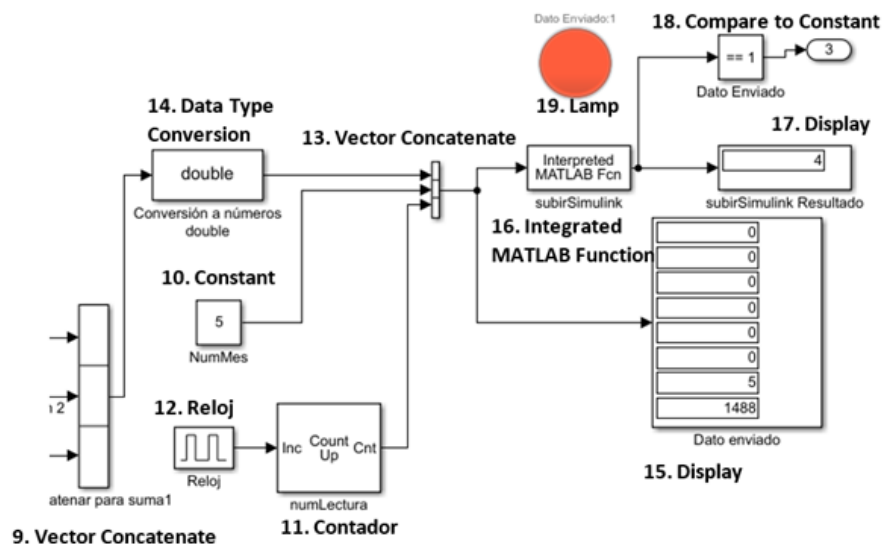
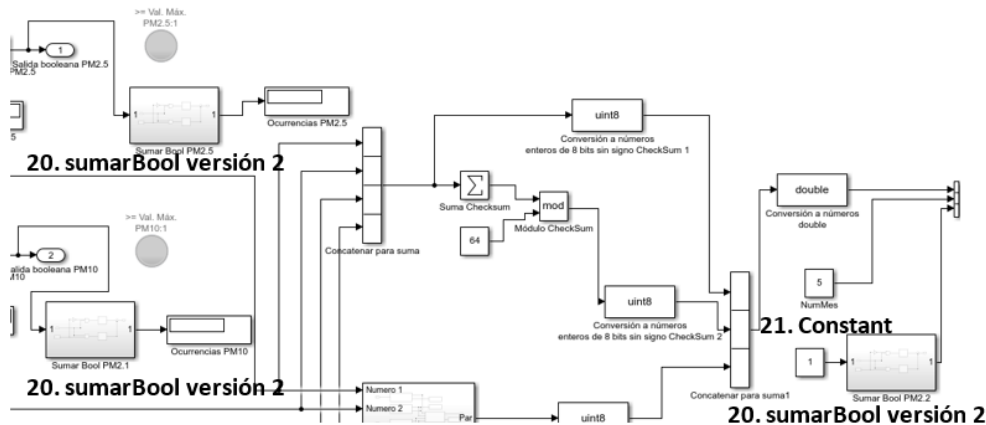


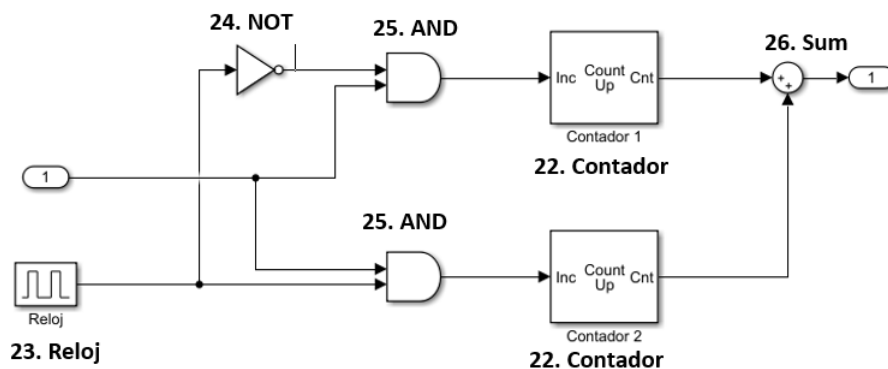
Imagen 6. Conexión con la nube

Cabe recalcar que, debido a los periodos de reloj, la función subirSimulink se ejecuta 2 veces con los mismos datos. Esto será manejado por el php dentro del servidor, para no afectar el diseño de los sensores. Por eso se propone cambiar el periodo de reloj para que coincida con los del envío de nuevos datos. Para esto se diseñó un nuevo sumarBool (20) que remplazaría a las 3 instancias donde se usarían relojes. Para generar el número de entrada, se usa un Constant (21) con una salida booleana de 1, ya que se busca que se aumente el valor en cada ciclo de reloj.



**Imagen 7. Posición del segundo diseño de sumarBool**

Este nuevo subsistema funciona con dos contadores (22) y un reloj (23). Este reloj tiene un periodo de 1/24. Ambos contadores reciben la señal de reloj, solo que uno la recibe negado con una compuerta NOT (24). Esto soluciona el problema de sincronización que se mencionó al explicar sumarBool. Al igual que en el sumarBool original, la señal de reloj está unido con una compuerta AND (25) a la señal booleana de entrada. Finalmente, ambas entradas se suman usando el bloque sum (26), lo que genera que la salida sea igual al sumarBool original.



**Imagen 8. Segundo diseño de sumarBool**

Ambas soluciones se incluyen en los anexos. El escritor prefiere la segunda solución, ya que reduce la complejidad del archivo php, sin embargo, ambos ejecutan el mismo archivo php, por lo que el tiempo de procesamiento es igual, pero la segunda opción nos genera un menor número de url usadas, y al usarse un servidor lento, esta es una ventaja.

La función subirSimulink es la siguiente:

```
%Sube los datos de Simulink a HelioHost
function s=subirSimulink(entrada)
    %Estructura de la entrada=pm10,pm25,temp,hum,mod,paridad,mes,numLec
    options = weboptions('ContentType','text','Timeout',15);%Configuración de la conexión
    nom_mat=strcat('PM25_2020_',num2str(entrada(7)),'.mat');%Generar nombre del archivo.mat
    con los datos
    data_file= load(nom_mat);%Cargar los datos del PM10 del mes en entrada(7)
    fechas=getabstime(data_file.ts);%Encontrar las fechas de todos los datos del mes
    s=2;%Caso de no paridad
```

```

        if(checkSum(entrada) && checarParidad(entrada) && entrada(8)+1<=length(fechas))%Solo
entrar si hay paridad, el checkSum es correcto y el número de entrada es válido
        dateIndex=entrada(8)+1;%Obtener el número de entrada final. +1 porque empieza en 0
        fecha=fechas(dateIndex);%Recuperar la fecha como una cell
        fecha=changeDate(fecha{1});%Formatear la fecha para estar en formato de datetime

        url=sprintf("http://rrdie.heliohost.org/archivosMod/insertMatlab.php?type=%s&pml0=%f&p
m25=%f&temp=%f&hum=%f&date=%s", 'simulink', entrada(1), entrada(2), entrada(3), entrada(4), fecha);%
Generar la url con los datos
        %Usar try en el caso que el servidor este caído
        try
            disp('Enviando a base de datos');
            messageBack = webread(url, options);%Obtener el html generado por la url
            disp(url); %Imprimir la url
            s=1;%Mostrar caso de subido exitoso
        catch %Si falla la subida de datos
            disp('Error de envio');
            s=0;%Mostrar caso de error
        end
    end
end
%Formatea la fecha de DD-MM-AAAA HH:MM:SS a AAAA-MM-DD HH:MM:SS
function nueva_fecha=changeDate(fecha)
    mes=['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'];%Array
con todos los mese
    numMeses=12;%Número de meses usados
    for i = 0:numMeses-1%Para cada uno de los meses en el array
        if(fecha(4:6)==mes(i*3+1:i*3+3))%Si el mes coincide
            break%Romper ciclo, se guarda el valor de i
        end
    end
    nueva_fecha=strcat(fecha(8:11), '-', num2str(i+1), '-
', fecha(1:2), fecha(12:size(fecha,2)));%Crear valor regresado
end
%Checa si se cumple la paridad
function paridad=checarParidad(entrada)
    %Obtener datos
    dato1=entrada(1);
    dato2=entrada(2);
    dato3=entrada(3);
    dato4=entrada(4);
    %Obtener el número de 1 al convertir el valor a binario

numUno=obtenerNumUno(dato1)+obtenerNumUno(dato2)+obtenerNumUno(dato3)+obtenerNumUno(dato4);
    numUno=numUno+entrada(6);%Sumar el bit de paridad
    if (mod(numUno,2)==0)%Si es un número par
        paridad=true;%Enviar true
    else
        paridad=false;%Enviar false
    end

end
%Obtiene el número de 1 en un número no binario al convertirlo en binario
function numUno=obtenerNumUno(dato)
    divisor=2;%Divisor dos para convertir a binario
    numUno=0;%Número de 1s en el dato
    while (dato>1)%Hasta que el dato sea 1
        temp=mod(dato,divisor);%Obtener el módulo
        if (temp==1)
            numUno=numUno+1;%Agregar a la cuenta si es uno
        end
        dato=(dato-temp)/2;%Actualizar el número
    end
    numUno=numUno+1;

end
%Checa el checkSum
function check=checkSum(entrada)
    %Recuperar datos
    dato1=entrada(1);
    dato2=entrada(2);
    dato3=entrada(3);
    dato4=entrada(4);
    %Sumar datos
    num=dato1+dato2+dato3+dato4;
    num=mod(num,64);%obtener módulo 64
    if (num==entrada(5))%Si es igual al número checkSum

```

```

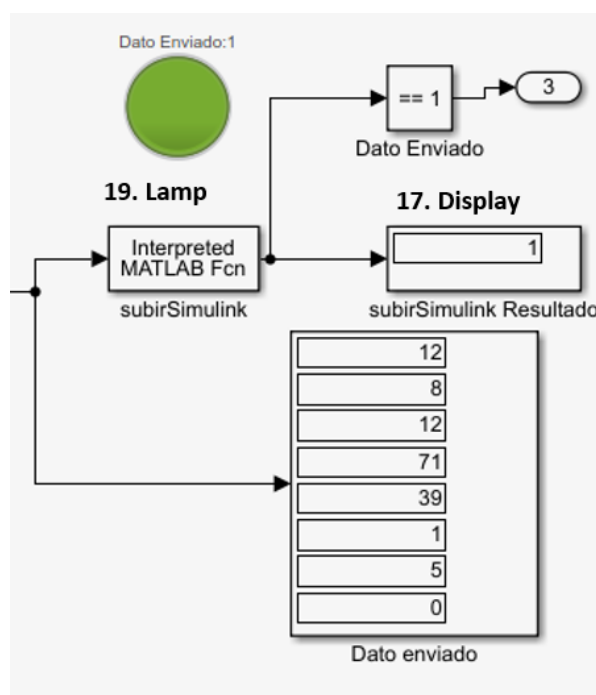
check=true;%Enviar true
else
check=false;%Enviar falso
end
end

```

Esta función tiene 3 salidas posibles. s=1 significa que el envío fue exitoso, si s=2, significa que hubo un error en el envío, si s=3 significa que hubo un error en la paridad o checksum.

En la imagen 6 se puede ver que no se pudo conectar con la nube, debido a un error de paridad/checkSum. Este fue un error generado a propósito, ya que, por el diseño, el dato tiene muy poca posibilidad de corromperse al ser enviado a la función.

A continuación, se muestra un caso donde el envío fue exitoso, como se puede apreciar en la Lamp (19) y en el display (17).



**Imagen 9. Conexión exitosa con la nube**



## 4.2. Bases de Datos

### 4.2.1. DB en MySQL con datos (imagen).

- Base de datos Tabla Lectura

Mostrando filas 0 - 17 (total de 18, La consulta tardó 0.0006 segundos)

SELECT \* FROM 'Lectura'

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

Opciones	id	fecha	valor	sensor_id
[Editar] [Copiar] [Borrar]	1	2020-04-01 00:59:59	27	1
[Editar] [Copiar] [Borrar]	2	2020-04-01 00:59:59	43	2
[Editar] [Copiar] [Borrar]	3	2020-04-01 00:59:59	26	3
[Editar] [Copiar] [Borrar]	4	2020-04-01 00:59:59	38	4
[Editar] [Copiar] [Borrar]	5	2020-04-01 00:59:59	42	5
[Editar] [Copiar] [Borrar]	6	2020-04-01 00:59:59	42	6
[Editar] [Copiar] [Borrar]	7	2020-04-01 00:59:59	35	7
[Editar] [Copiar] [Borrar]	8	2020-04-01 00:59:59	44	8
[Editar] [Copiar] [Borrar]	9	2020-04-01 00:59:59	34	9
[Editar] [Copiar] [Borrar]	10	2020-04-01 00:59:59	46	10
[Editar] [Copiar] [Borrar]	11	2020-04-01 00:59:59	45	11
[Editar] [Copiar] [Borrar]	12	2020-04-01 00:59:59	38	12
[Editar] [Copiar] [Borrar]	13	2020-04-01 00:59:59	47	13

- Base de datos Tabla Estacion

Mostrando filas 0 - 24 (total de 71, La consulta tardó 0.0004 segundos)

SELECT \* FROM 'Estacion'

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

Opciones	id	clave	nombre	ubicacion
[Editar] [Copiar] [Borrar]	1	ACO	Acolman	-98.912003, 19.635601
[Editar] [Copiar] [Borrar]	2	AJU	Ajusco	-99.162611, 19.154286
[Editar] [Copiar] [Borrar]	3	AJM	Ajusco Medio	-99.207744, 19.272161
[Editar] [Copiar] [Borrar]	4	ARA	Aragon	-99.074549, 19.470218
[Editar] [Copiar] [Borrar]	5	ATI	Atizapan	-99.254133, 19.576963
[Editar] [Copiar] [Borrar]	6	AZC	Azcapotzalco	-99.190657, 19.487728
[Editar] [Copiar] [Borrar]	7	BJJ	Benito Juarez	-99.159596, 19.370464
[Editar] [Copiar] [Borrar]	8	CAM	Camarones	-99.169794, 19.468404
[Editar] [Copiar] [Borrar]	9	CCA	Centro de Ciencias de la Atmosfera	-99.176111, 19.326111
[Editar] [Copiar] [Borrar]	10	CES	Cerro de la Estrella	-99.074678, 19.334731
[Editar] [Copiar] [Borrar]	11	CFE	Museo Tecnológico de la CFE	-99.194279, 19.414393
[Editar] [Copiar] [Borrar]	12	CHO	Chalco	-98.886088, 19.266948
[Editar] [Copiar] [Borrar]	13	COR	CORENA	-99.026040, 19.265346

- Base de datos Tabla Sensor

Mostrando filas 0 - 24 (total de 101, La consulta tardó 0.0078 segundos)

SELECT \* FROM 'Sensor'

Perfilando [Editar en línea] [Editar] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

Opciones	id	tipo	estacion_id
[Editar] [Copiar] [Borrar]	1	PM10	1
[Editar] [Copiar] [Borrar]	2	PM10	3
[Editar] [Copiar] [Borrar]	3	PM10	5
[Editar] [Copiar] [Borrar]	4	PM10	7
[Editar] [Copiar] [Borrar]	5	PM10	8
[Editar] [Copiar] [Borrar]	6	PM10	15
[Editar] [Copiar] [Borrar]	7	PM10	17
[Editar] [Copiar] [Borrar]	8	PM10	21
[Editar] [Copiar] [Borrar]	9	PM10	28
[Editar] [Copiar] [Borrar]	10	PM10	29
[Editar] [Copiar] [Borrar]	11	PM10	37
[Editar] [Copiar] [Borrar]	12	PM10	44
[Editar] [Copiar] [Borrar]	13	PM10	48

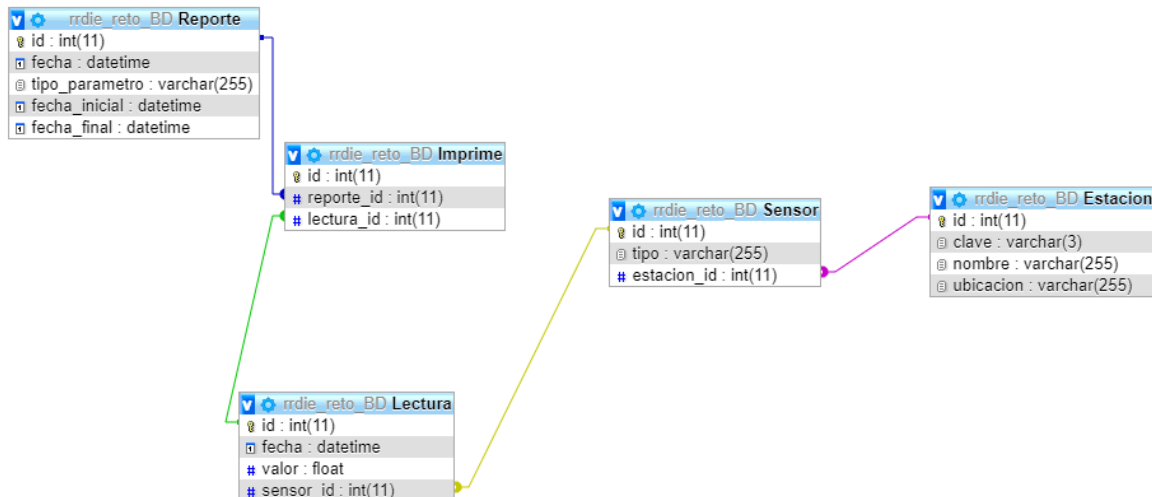
- Base de datos Tabla Reporte

id	fecha	tipo_parametro	fecha_inicial	fecha_final
1	2020-11-27 22:30:32	Humedad	2020-06-01 00:00:00	2020-11-27 23:59:59
2	2020-11-27 22:34:37	PM 10	2020-01-01 00:00:00	2020-11-27 23:59:59
3	2020-11-28 02:09:07	CO	2020-10-25 00:00:00	2020-10-25 23:59:59
4	2020-11-28 02:09:26	CO	2020-10-25 00:00:00	2020-11-27 23:59:59
5	2020-11-28 02:09:38	CO	2020-10-25 00:00:00	2020-11-27 23:59:59
6	2020-11-28 02:16:13	PM 10	2020-07-01 00:00:00	2020-11-27 23:59:59
7	2020-11-28 02:16:38	PM 10	2020-01-08 00:00:00	2020-11-05 23:59:59
8	2020-11-28 02:30:29	PM 2.5	2020-01-07 00:00:00	2020-09-09 23:59:59
9	2020-11-28 02:30:41	Humedad	2020-02-05 00:00:00	2020-11-03 23:59:59
10	2020-11-28 02:31:11	CO2	2020-01-08 00:00:00	2020-02-05 23:59:59
11	2020-11-28 02:31:35	CO	2020-07-01 00:00:00	2020-11-10 23:59:59
12	2020-11-28 02:31:49	CO	2020-07-25 00:00:00	2020-12-18 23:59:59
13	2020-11-28 02:32:03	CO	2020-09-22 00:00:00	2020-11-04 23:59:59

• Base de datos Tabla Imprime

id	reporte_id	lectura_id
1	1	37104
2	1	37108
3	1	37112
4	1	37116
5	1	37120
6	1	37124
7	1	37128
8	1	37132
9	1	37136
10	1	37140
11	1	37144
12	1	37148
13	1	37152

4.2.2. DB completa y normalizada hasta 3NF (imagen del modelo relacional desde PHPMyAdmin / tab Designer). (Mostrar por qué la BD está en 3NF)



Para cumplir la primera forma normal se debe evaluar si todos los elementos son atómicos. En nuestra base de datos todos lo son.

Para la segunda forma normal debemos comprobar que todos los elementos dependan de las super llaves.

Para Estacion

Superllaves: {id}

{id} -> {clave, nombre, ubicacion}

Para Sensor

Superllaves: {id}  
{id} -> {tipo, estacion\_id}

Para Lectura:  
Superllaves: {id}  
{id} -> {valor, fecha, sensor\_id}

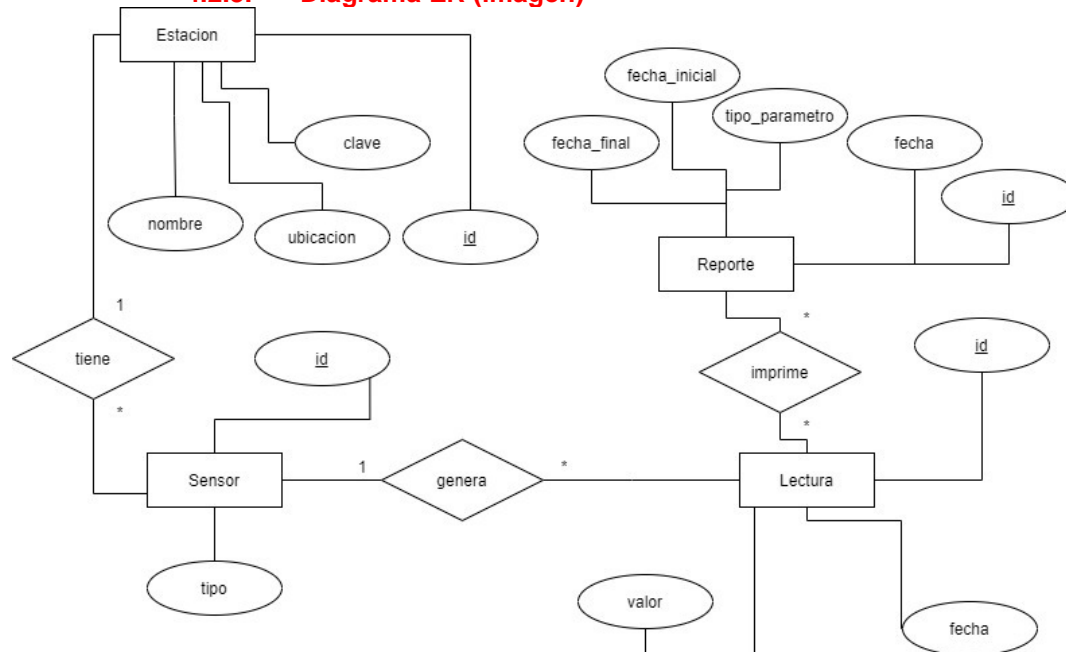
Para Imprime:  
Superllaves: {id}  
{id} -> {lectura\_id, reporte\_id}

Para Reporte:  
Superllaves: {id}  
{id} -> {fecha\_inicial, fecha\_final, tipo\_parametro, fecha}

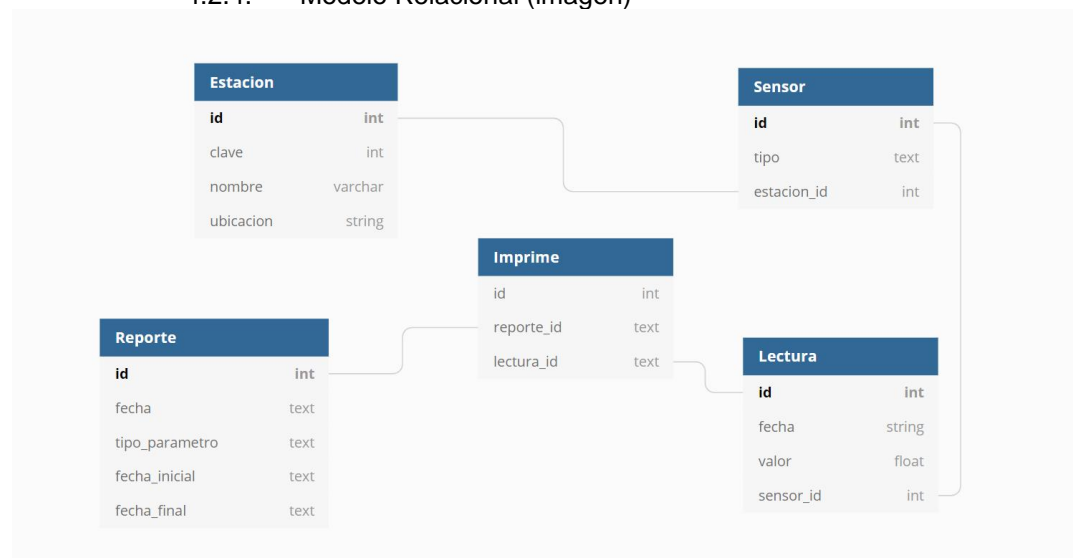
Todas las dependencias mostradas son verdaderas, por lo que se está en segunda forma normal.  
Para la tercera forma normal se busca que no haya transitividad.

En ninguno de las tuplas hay esto, ya que solo se usan id de otras tuplas para referenciar los datos, no los datos en sí. Por ejemplo, se tiene Lectura(id,valor, fecha, sensor\_id), en vez de tener Lectura(id,valor, fecha, sensor\_tipo, sensor\_estacion, estacion\_nombre). Este se trata de un ejemplo extremo, pero se usa para que sea más notable como se cumple la tercera forma normal.

#### 4.2.3. Diagrama ER (imagen)



#### 4.2.4. Modelo Relacional (imagen)



#### 4.3. Recursos de un Sistema computacional

##### 4.3.1. Aplicación Productor-Consumidor (Base de Datos y Sensores Simulados)

###### 4.3.1.1. Funcionamiento de la aplicación de acuerdo a los requerimientos.

Se utilizan seis procesos productores que simulan lecturas realizadas por un grupo de sensores que generan valores aleatorios de forma asíncrona y son almacenadas en un consumidor que almacena la información y obtiene estadísticas de los datos recolectados.

###### 4.3.1.2. Respuesta en equipo: ¿Qué temas del Módulo 3 se utilizan en su aplicación y de qué manera?

Los temas principales abordados en esta actividad son los procesos y el flujo de datos, basando todo esto en la arquitectura von Neumann que utilizan las computadoras. Los procesos son ejecutados como threads y controlados por un ThreadPoolExecutor que administra tanto los procesos del productor y consumidor para poder ejecutarlos de manera asíncrona. Algunas operaciones necesitan un lock para permitir el término de las operaciones y asegurar el funcionamiento normal de la salida de datos. Al almacenar los datos, se implementó una queue para no saturar al consumidor y permitir la entrada y almacenamiento correcto para realizar el procesamiento una vez que se dejarán de recibir datos y todos fueran guardados en las listas correspondientes.

###### 4.3.1.3. De manera individual, responder: Menciona un elemento técnico relevante que justifica la importancia de aprender cada tema del Módulo (lista en Sesión 1), en relación con su aplicación en el Reto.

- [Alan Estiel Aguirre Mohar](#)

Aprender sobre la arquitectura de Von Neumann fue de suma importancia para este proyecto. Las estaciones se rigen bajo este principio ya que como dispositivos de entrada tenemos sensores de diferentes tipos, los cuales son procesados y almacenados. El tema de Sincronización de Procesos es empleado en el reto para evitar bugs y junto a la administración de procesos evitamos que ciertos procesos se realicen de forma asíncrona y así los datos no son enviados si no han sido procesados. El concepto de entrada y salida de datos lo empleamos para poder realizar búsquedas mientras se siguen recibiendo datos.

- [Alejandro Hernández Ramos](#)

Entender la arquitectura de Von Neumann nos ayuda a comprender cómo funciona la lectura y almacenamiento de datos mediante sensores que pueden ser implementados en diversas aplicaciones, tal como en este caso, en el Internet de las cosas. En este reto se utilizaron procesos, como los que se encargaban de leer de los sensores, insertar en la base de datos y mostrar la salida en el sitio web, los cuales funcionaban de manera individual y de manera asíncrona.

- [Diego Reyna Reyes](#)

###### 1. Arquitectura de un sistema computacional

La arquitectura von Neumann es la base para el esquema del proyecto, debido al uso de la nube como almacenador de datos e instrucciones

###### 2. Concepto de proceso

###### a. Recursos de un proceso

Permite limitar el número de datos que se pueden subir a la base de datos en un tiempo dado

###### b. Administración de procesos

Permite subir datos de forma "simultánea" a la base de datos. Esto dentro de las funciones que usan los archivos php

###### c. Sincronización de procesos

Permite que al usar los sensores de SIMULINK no se envíen datos sin haber procesado los anteriores

### 3. Entrada y salida de datos

#### a. Programación para entrada y salida de datos

Permite usar la aplicación mientras se reciben datos, ya que el envío de datos no afecta su procesamiento.

- [Jonatan Hernández García](#)

#### 1. Arquitectura de un sistema computacional.

Diseñar o seguir una arquitectura es de suma importancia para poder seguir un proceso controlado de intercambio de datos, en nuestro caso Von Neuman nos permitió trabajar con los datos de los sensores y el almacenamiento de estos.

#### 2. Concepto de un proceso.

Gracias a el tema de conceptos de un proceso permitió que supiéramos nuestras limitantes, con los recursos de un proceso, la administración, para evitar malas subidas de datos o bugs, y finalmente la sincronización de datos para que estos mismos se modifiquen de tal forma los receptores de los datos siempre los puedan leer con el formato adecuado.

#### 3. Entrada y salida de datos.

Además de que los datos vayan adecuados para la lectura, los dispositivos que envían los mismo, deben poder tanto recibir o enviar datos de forma asíncrona dependiendo de su función.

- [Manuel Hernández Bravo](#)

Con la arquitectura von Neumann, entendimos el concepto básico de una computadora y su funcionamiento para abordar temas más complicados. Después, entendimos lo que realiza un proceso, que es un conjunto de instrucciones y vimos como los programas ejecutan esas operaciones. Finalmente, entendimos el procesamiento de los datos que puede ser realizado de manera asíncrona, optimizando el tiempo que se necesita, sin embargo, hay ciertas operaciones que no pueden ser realizadas de esta forma al depender de otros procesos.

#### 4.4. Diseño Interactivo

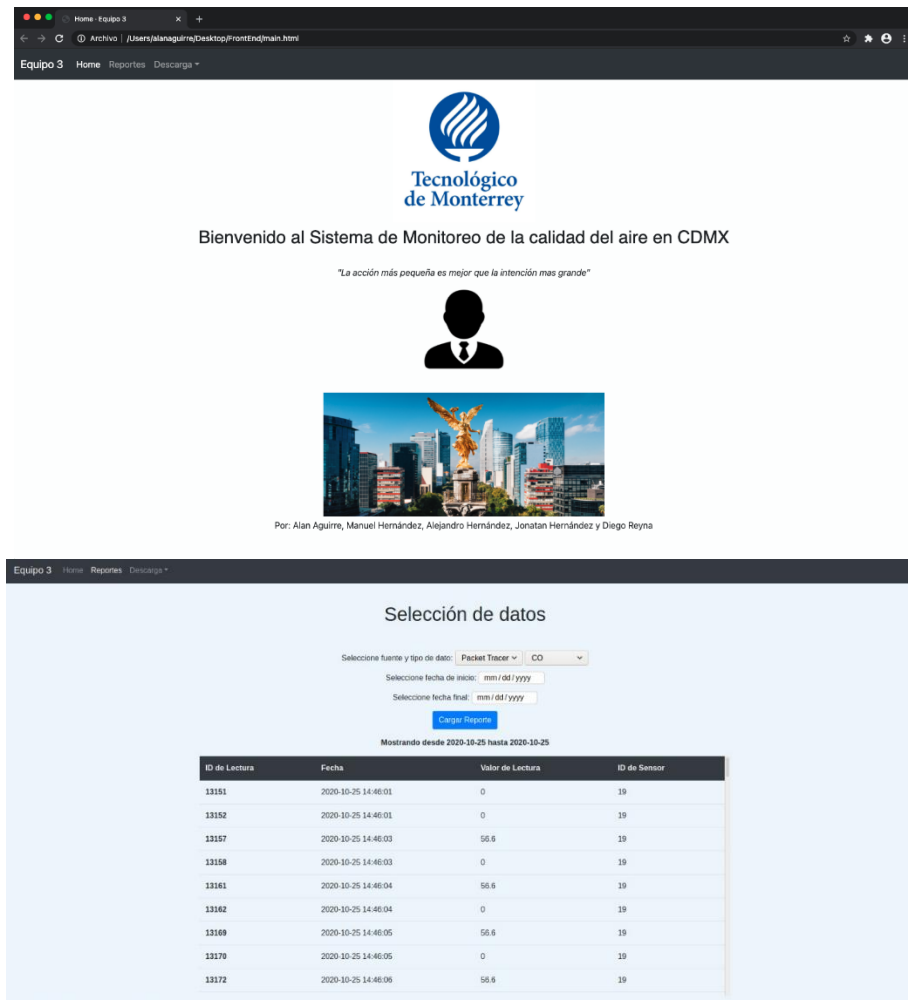
##### 4.4.1. Actualización final de la página web del equipo en línea.

##### 4.4.1.1. Vista final del dashboard con vista de datos desde la DB.

Se generó una página web en la cual el usuario es capaz de hacer búsquedas a partir del tipo de sensor y de la fuente de la información, ya sea Packet Tracer o Simulink. Además, se cuenta con una guía de usuario y la prueba heurística.

El enlace a la página es la siguiente y se recomienda abrirlo en Google Chrome: <http://rrdie.heliohost.org/>

A continuación, se adjuntan capturas de pantalla del sitio:



##### 4.4.1.2. Prueba heurística. Evaluar su dashboard utilizando las 10 heurísticas de Nielsen. Incluir en la página web una página con los resultados de esta evaluación.

- Visibilidad del estado del sistema.  
Se indica la página en la que se encuentra el usuario en el momento en la barra de navegación, facilitando al usuario moverse a través de la página.
- Correspondencia entre el sistema y la realidad.  
Se usa el ID de la fuente y el tipo de dato, sin embargo, lo que el usuario ve son los nombres para que sea más fácil de entender y seleccionar lo que necesita de forma sencilla.
- Control y libertad del usuario.  
Se agregó con la barra de navegación una forma sencilla de navegar a través de la página. El usuario decide cuándo enviar la búsqueda después de seleccionar los datos.
- Consistencia y estándares.

Mismo estilo en todos los componentes de la página web para, barra de navegación en la parte superior de la página web y el botón localizado en abajo de los campos para hacer la consulta.

- Prevención de errores.

Minimizamos las opciones para que el usuario no pueda elegir una fuente o tipo de dato invalido y mandamos un mensaje cuando la consulta realizada no arroja ningún resultado dependiendo de las fechas elegidas.

- Reconocer en lugar de recordar.

El usuario no debe recordar ningún tipo de dato de una sección de la página a la otra y todos los campos que deben ser rellenos para llevar a cabo una búsqueda despliegan un menú de opciones relacionadas a dicho campo, por ejemplo, para llenar un formulario de fecha se despliega un calendario. Facilitando así el llenado de los datos y hacer más intuitivo el uso de la aplicación.

- Flexibilidad y eficiencia de uso.

Al hacer una consulta de los datos no se abre una nueva página, los resultados son desplegados abajo de los parámetros de la búsqueda, para así hacer más eficiente el uso de las páginas y evitar tener que regresar al menú de consulta cada vez que se quiere hacer una búsqueda.

- Estética y diseño minimalista.

En el diseño de la página web solo se agregaron cosas funcionales para evitar saturar la página y asegurar que la información sea fácil de ver y accesible.

Ayudar a usuarios a reconocer, diagnosticar y recuperarse de errores.

El php muestra si hay algún problema con el servidor o la conexión.

- Ayuda y documentación.

Actualmente la página web no cuenta con un botón de documentación y a pesar de que el diseño es intuitivo será necesario agregarlo para expresar de forma concreta los pasos que el usuario debe de seguir.



#### 4.5. Administración de proyectos

##### 4.5.1. Seguimiento puntual de las actividades registradas en Trello de la 3ta etapa y la entrega final con las fechas y responsables de cada equipo.

La división de trabajo en esta entrega se realizó usando Trello.  
Tarjeta principal de la Entrega:



Para la Entrega se dividió el trabajo de dos formas, cada una con sus responsables (Las iniciales son visibles del lado izquierdo de la tarjeta y están marcadas con una palomita en la lista de miembros a la derecha).

La primera forma fue la división dentro de la Introducción y el primer punto de la conclusión. Esta se dividió en las siguientes partes:

##### Resumen Ejecutivo y Análisis Descriptivo:



##### Requerimientos Funcionales y No funcionales:

Etapa 3: Requerimientos Funcionales y No funcionales

en la lista IN PROGRESS

MIEMBROS

DR

+

ETIQUETAS

IN PROGRESS

+

VENCIMIENTO

☐

26 de nov. a las 23:59

▼

Descripción

Editar

Actualizar los Requerimientos funcionales de la Entrega 2

Actividad

Mostrar detalles

DR

Escriba un comentario...

### Administración de las dimensiones:

Etapa 3: Administración de las dimensiones

en la lista IN PROGRESS

MIEMBROS

JG

+

ETIQUETAS

IN PROGRESS

+

VENCIMIENTO

☐

26 de nov. a las 23:59

▼

Descripción

Editar

Describir la manera en que se administraron las dimensiones: Alcance, Tiempo y Presupuesto para cumplir con los requerimientos (Diagrama de Gantt)

Actividad

Mostrar detalles

DR

Escriba un comentario...

### Diagrama de Flujo de datos:

Etapa 3: Diagrama Flujo de Datos

en la lista IN PROGRESS

MIEMBROS

DR

+

ETIQUETAS

DONE

+

VENCIMIENTO

☐

26 de nov. a las 23:59

▼

Descripción

Editar

Describir el flujo de los datos en la solución del reto, con énfasis en la manera en que los temas vistos en clase (cada módulo) se usan para permitir que los datos sean utilizados en la implementación de IoT.

Actividad

Mostrar detalles

DR

Escriba un comentario...

### Construcción de los sensores:

Miembros

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreyna...)

✓

AM

Alan Estiel Aguirre Mohar (alanestie...)

AR

Alejandro Hernandez Ramos (alejan...)

JG

Jonatan Hernández García (jonatan...)

MH

Manuel Hernandez (manuelhernan...)

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreynareyes)

JG

Jonatan Hernández García (jona...)

✓

AM

Alan Estiel Aguirre Mohar (alanestie...)

AR

Alejandro Hernandez Ramos (alejan...)

MH

Manuel Hernandez (manuelhernan...)

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreyna...)

✓

AM

Alan Estiel Aguirre Mohar (alanestie...)

AR

Alejandro Hernandez Ramos (alejan...)

JG

Jonatan Hernández García (jonatan...)

MH

Manuel Hernandez (manuelhernan...)

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreyna... ✓

AM

Alan Estiel Aguirre Mohar (alan... ✓

JG

Jonatan Hernández García (jona... ✓

AR

Alejandro Hernandez Ramos (alejan...

MH

Manuel Hernandez (manuelhernan...

LM

Luis Montesinos (luismontesinos5)



Rolando Vallejo Clemente (rfvallejo)



Sergio Ruiz Loza (sergioruizloza)

### Implementación de los programas:

Miembros

X

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreyna... ✓

AM

Alan Estiel Aguirre Mohar (alan... ✓

MH

Manuel Hernandez (manuelher... ✓

AR

Alejandro Hernandez Ramos (alej... ✓

JG

Jonatan Hernández García (jonatan... ✓

LM

Luis Montesinos (luismontesinos5) ✓



Rolando Vallejo Clemente (rvallejo) ✓



Sergio Ruiz Loza (sergioruizloza) ✓

### Conclusiones por temas:

Miembros

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoireynareyes)

JG

Jonatan Hernández García (jona... ✓)

AM

Alan Estiel Aguirre Mohar (alanestiel...)

AR

Alejandro Hernandez Ramos (alejant...)

MH


Manuel Hernandez (manuelherman...)

LM

Luis Montesinos (luismontesinos5)



Rolando Vallejo Clemente (rfvallejo)



Sergio Ruiz Loza (sergioruizloza)

La segunda forma de división dividió el desarrollo de acuerdo a los módulos. Esta se dividió de la siguiente manera:

## Módulo 1:

Etapa 3: Módulo 1: Sistema digitales

en la lista [IN PROGRESS](#)

MIEMBROS

DR

+

ETIQUETAS

IN PROGRESS

+

VENCIMIENTO

☐

26 de nov. a las 23:59

▼

Descripción

Editar

4.1.1 Validar el correcto funcionamiento de Humedad Relativa y Temperatura.

4.1.2 Contar ocurrencias de PM2.5 y PM10.

4.1.3 Máquina de estados de riesgos.

4.1.4 Evidencia de correcta conexión con la nube.

Actividad

Mostrar detalles

DR

Escriba un comentario...

## Módulo 2:

Etapa 3: Módulo 2: Bases de datos

en la lista [IN PROGRESS](#)

MIEMBROS

MH

+

ETIQUETAS

IN PROGRESS

+

VENCIMIENTO

☐

26 de nov. a las 23:59

▼

Descripción

Editar

4.2.1 DB en MySQL con datos.

4.2.2 DB completa y normalizada hasta 3NF (imagen del modelo relacional desde PHPMyAdmin / tab Designer).

4.2.3 Diagrama ER (imagen).

4.2.4 Modelo Relacional (imagen).

Actividad

Mostrar detalles

DR

Escriba un comentario...

## Módulo 3:

Etapa 3: Módulo 3: Recursos de un Sistema computacional

en la lista [IN PROGRESS](#)

MIEMBROS

AR

+

ETIQUETAS

IN PROGRESS

+

VENCIMIENTO

☐

26 de nov. a las 23:59

▼

Descripción

Editar

4.3.1 Aplicación Productor-Consumidor (Base de Datos y Sensores Simulados):  
Funcionamiento de la aplicación de acuerdo a los requerimientos.  
Respuesta en equipo: ¿Qué temas del Módulo 3 se utilizan en su aplicación y de qué manera?  
De manera individual, responder: Menciona un elemento técnico relevante que justifica la importancia de aprender cada tema del Módulo (lista en Sesión 1), en relación con su aplicación en el Reto.

Actividad

Mostrar detalles

DR

Escriba un comentario...

## Módulo 4:

Miembros

×

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreyna...)

AM

Alan Estiel Aguirre Mohar (alanestie...)

AR

Alejandro Hernandez Ramos (alejan...)

JG

Jonatan Hernández García (jonatan...)

MH

Manuel Hernandez (manuelhernan...)

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rfvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

×

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreynareyes)

MH

Manuel Hernandez (manuelher...)

AM

Alan Estiel Aguirre Mohar (alanestie...)

AR

Alejandro Hernandez Ramos (alejan...)

JG

Jonatan Hernández García (jonatan...)

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rfvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

×

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreynareyes)

AR

Alejandro Hernandez Ramos (al...)

AM

Alan Estiel Aguirre Mohar (alanestie...)

JG

Jonatan Hernández García (jonatan...)

MH

Manuel Hernandez (manuelhernan...)

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rfvallejo)

Sergio Ruiz Loza (sergioruizloza)

Etapa 3: Modulo 4: Diseño Interactivo

en la lista [IN PROGRESS](#)

MIEMBROS

AM MH +

ETIQUETAS

IN PROGRESS +

VENCIMIENTO

☐ 26 de nov. a las 23:59 ▾

Descripción

Editar

4.4.1 Actualización final de la página web del equipo en línea. Incluir:

- Vista final del dashboard con vista de datos desde la DB.
- Prueba heurística. Evaluar su dashboard utilizando las 10 heurísticas de Nielsen. Incluir en la página web una página con los resultados de esta evaluación.

Actividad

Mostrar detalles

DR

Escriba un comentario...

### Módulo 5:

Etapa 3: Modulo 5: Administración de proyectos

en la lista [IN PROGRESS](#)

MIEMBROS

DR +

ETIQUETAS

IN PROGRESS +

VENCIMIENTO

☐ 26 de nov. a las 23:59 ▾

Descripción

Editar

4.5.1 Seguimiento puntual de las actividades registradas en Trello de la 3ta etapa y la entrega final con las fechas y responsables de cada equipo.

Actividad

Mostrar detalles

DR

Escriba un comentario...

### Módulo 6:

Etapa 3: Modulo 6: IoT

en la lista [IN PROGRESS](#)

MIEMBROS

AM JG +

ETIQUETAS

IN PROGRESS +

VENCIMIENTO

☐ 26 de nov. a las 23:59 ▾

Descripción

Editar

4.6.1 Documentación de la topología y el proceso.  
4.6.2 Debe incluirse un diagrama con las direcciones y los protocolos involucrados.  
4.6.3 así como la explicación de la interacción entre los diferentes dispositivos.  
4.6.4 Se deben incluir 4 sensores uno de CO, otro CO2 y otros 2 a decidir por el equipo justificando su elección y explicando cómo estos sensores podrían ayudar en la interpretación de los datos

Actividad

Mostrar detalles

DR

Escriba un comentario...

Miembros

×

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreynareyes)

AM

Alan Estiel Aguirre Mohar (alan... ✓)

MH

Manuel Hernandez (manuelher... ✓)

AR

Alejandro Hernandez Ramos (alejan...

JG

Jonatan Hernández García (jonatan...

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rfvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

×

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreyna... ✓)

AM

Alan Estiel Aguirre Mohar (alanestie...

AR

Alejandro Hernandez Ramos (alejan...

JG

Jonatan Hernández García (jonatan...

MH

Manuel Hernandez (manuelhernan...

LM

Luis Montesinos (luismontesinos5)

Rolando Vallejo Clemente (rfvallejo)

Sergio Ruiz Loza (sergioruizloza)

Miembros

×

Buscar miembros

MIEMBROS DEL TABLERO

DR

Diego Reyna Reyes (diegoreynareyes)

AM

Alan Estiel Aguirre Mohar (alan... ✓)

JG

Jonatan Hernández García (jona... ✓)

AR

Alejandro Hernandez Ramos (alejan...

MH

Manuel Hernandez (manuelhernan...

LM

Luis Montesinos (luismontesinos5)

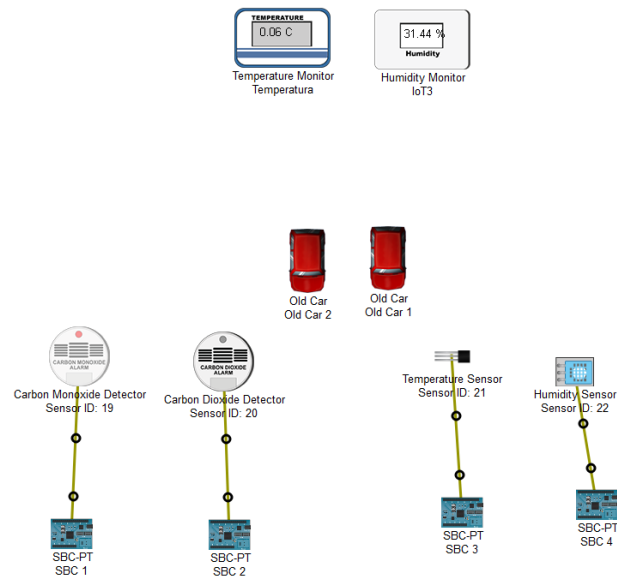
Rolando Vallejo Clemente (rfvallejo)

Sergio Ruiz Loza (sergioruizloza)



## 4.6. Internet de las Cosas

### 4.6.1. Documentación de la topología y el proceso.



Se utilizaron 4 dispositivos SBC para leer los datos que enviaba cada sensor (CO, CO2, temperatura y humedad), cada uno con su respectivo ID. Estos dispositivos SBC se programaron para que sean capaces de enviar los datos de lectura a la base de datos en Helio Host por protocolo HTTP.

Los sensores por su parte recopilan los datos de un entorno simulado gracias a las funciones de Packet Tracer. Los coches son los únicos actores de aumento de CO2 y CO en el entorno. Los monitores solo están con fines ilustrativos.

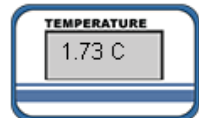
### 4.6.2. Direcciones y protocolos utilizados

Como en toda estructura de IOT las conexiones que existen entre los diferentes dispositivos se rigen bajo ciertos protocolos, los cuales podríamos definir como reglas bajo las cuales los dispositivos se comunican e identifican con los demás dispositivos, esto con la finalidad de garantizar que la información es recibida de forma correcta o si ha habido algún problema en la transferencia de datos. En esta aplicación se emplea el protocolo HTTP el cual es usado comúnmente por navegadores para solicitar y recibir archivos HTML. En este caso es lo que ocurre entre el servidor y el navegador que ocupamos para la visualización de datos. Cabe mencionar que este protocolo está basado en TCP, el cual, junto a la IP, garantiza que se transmitan datos de manera adecuada a través de Internet e implementa la comunicación entre cliente y servidor.

Ahora, en cuanto a las direcciones que emplean los dispositivos es empleado el direccionamiento de Internet IPv4, las cuales se caracterizan por tener una longitud de cuatro bytes y la mayoría de las redes de internet están hechas utilizando esta versión de IP.

### 4.6.3. Así como la explicación de la interacción entre los diferentes dispositivos.

Los monitores de temperatura y humedad no tienen otro fin más que el de ayudarnos a visualizar los datos que deberían estar mandando los sensores a los SBC.



Temperature Monitor  
Temperatura



Humidity Monitor  
IoT3

Las variables que nos ayuda a simular el CO y el CO2 son los coches, los cuales desprenden estas emisiones.



Old Car  
Old Car 2



Old Car  
Old Car 1

Nuestros 2 sensores de CO y CO2 tienen un ID y se encuentran enviando los datos que recopilan a las SBC. Las SBC 1 y 2 se encargan de recibir y mandar estas lecturas a nuestra base de datos en Helio Host bajo protocolo HTTP.

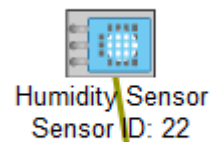
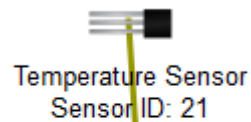


SBC-PT  
SBC 1



SBC-PT  
SBC 2

De igual forma actúan los 2 sensores de temperatura y humedad, pero con los sensores SBC 3 y 4. Esto lo hacen debido a que los sensores no están capacitados para mandar los datos al servidor.



SBC-PT  
SBC 3



SBC-PT  
SBC 4



**4.6.4. Se deben incluir 4 sensores uno de CO, otro CO<sub>2</sub> y otros 2 a decidir por el equipo justificando su elección y explicando cómo estos sensores podrían ayudar en la interpretación de los datos.**

Los sensores que se decidieron implementar fueron de CO, CO<sub>2</sub>, temperatura y humedad ya que unos de los principales contaminantes que afectan la salud de las personas y que favorecen al efecto invernadero en mala medida son el Dióxido de Carbono y Monóxido de Carbono, estos indicadores podríamos decir que afectan directamente la calidad del aire, mientras que otros indicadores lo hacen de manera indirecta como la temperatura y la humedad ya que la diferencia entre la temperatura del ambiente y el gas emitido con contaminantes puede provocar que los contaminantes suban y se dispersen pero que si chocan con una capa de aire más caliente se queden debajo de ella, favoreciendo así la concentración de contaminantes. De igual forma un estudio realizado por el departamento de ingeniería de la Universidad de Zulia encontró que la humedad relativa afecta la concentración de PM<sub>10</sub> en el ambiente, ya que entre más humedad relativa haya, mayor concentración de PM<sub>10</sub> existe.

Debido a la relación que existe entre los elementos que mediremos y la calidad del aire es que yace la importancia del análisis de los datos obtenidos. Todo con la finalidad de proveer la información necesaria para determinar si la calidad del aire es óptima para los ciudadanos o se deben tomar medidas que ayuden a mitigar los contaminantes presentes en el aire.

## **5. Conclusiones**

### **5.1. Conclusiones por módulo**

#### **5.1.1. Sistemas digitales.**

Los temas de sistemas digitales fueron de los más complejos y retadores en el módulo. Se empezó de forma tranquila con compuertas lógicas y álgebra booleana, lo cual sentó las bases para los circuitos con lógica combinacional y posteriormente lógica secuencial. Gracias a lo anterior pudimos trabajar en MATLAB con SIMULINK y así empezar a tomar el enfoque del reto en el módulo para la lectura de datos reales previamente tomados en el sistema. Por último, los flip flops, registros y contadores nos está brindando más herramientas para la máquina de estados lo cual presenta un reto ya por sí solo.

#### **5.1.2. Software**

En software se vieron los temas más satisfactorios del módulo, ya que estos nos brindaron herramientas, información y recursos para la creación del servidor y bases de datos para la resolución del reto. Estos temas nunca los habíamos visto y nos tranquilizó tener las herramientas casi de inmediato para poder empezar a visualizar la dirección del reto. Por otra parte, el tema de entidad relación nos facilitó la construcción, acomodo y control de nuestros datos, mientras que el tema de normalización fue una excelente herramienta complementaria para ahondar aún más en el manejo de los datos. Por último, SQL nos permitió manejar y obtener de forma fácil datos de nuestra BD o incluso implementar una.

#### **5.1.3. Administración de recursos del sistema**

Diseñar o seguir una arquitectura es de suma importancia para poder seguir un proceso controlado de intercambio de datos, en nuestro caso Von Neuman nos permitió trabajar con los datos de los sensores y el almacenamiento de estos. Gracias a el tema de conceptos de un proceso permitió que supiéramos nuestras limitantes, con los recursos de un proceso, la administración, para evitar malas subidas de datos o bugs, y finalmente la sincronización de datos para que estos mismos se modifiquen de tal forma los receptores de los datos siempre los puedan leer con el formato adecuado. Por último. la programación para el recibo y envío de datos nos permitió enviar datos en cualquier momento, ya que recibirlos no afecta su funcionamiento.

#### **5.1.4. Introducción al diseño Interactivo**

Diseño interactivo fue el módulo que más nos acercó al usuario y al entorno de trabajo con el cliente, se vieron temas con poco nivel de complejidad pero que son recursos esenciales para la práctica, ya que, al desarrollar, el usuario es el que más importa a la hora de resultados.

#### **5.1.5. Procesos y administración de proyectos**

Desarrollar un proyecto de alta complejidad y trabajo es una tarea sumamente difícil para una sola persona, por ello es necesario para colaborar con más personas y aprender a trabajar en equipo. La importancia de las herramientas brindadas por este módulo radica en el manejo del tiempo, presupuesto y recursos que se tienen para desarrollar trabajos de alta escala.

#### **5.1.6. Internet of things**

La tecnología avanza a pasos gigantescos y con ella las ciudades. Cada vez son más frecuentes los dispositivos inteligentes capaces de conectarse con otros y es importante tener los conocimientos para poder manipular estas redes para así seguir desarrollando o simplemente seguir administrando las mismas. Packet Tracer de Cisco es una herramienta increíble para aprender y practicar con este tipo de dispositivos como nosotros hicimos con nuestros sensores y dispositivos SBC.

### **5.2. Conclusiones individuales**

#### **• Alan Estiel Aguirre Mohar**

Realizar este proyecto requirió de una buena comunicación dentro del equipo de trabajo, ya sea para pedir ayuda, informar sobre los avances que se llevaban y los tiempos en los que se realizaban las tareas. Además del desarrollo de estas competencias y habilidades de trabajo en equipo fue necesario poner en práctica los diferentes conocimientos técnicos adquiridos en los diferentes módulos, los cuales contribuyen a nuestro crecimiento profesional en ámbitos de preparación teórica y de ganancia de experiencia para la creación de un producto que sea de utilidad para el usuario final, además de estar desarrollado bajo diferentes estándares que garantizan una experiencia óptima mediante una interfaz intuitiva que sea capaz de mostrar el resultado del análisis de los datos recibidos por sensores y almacenados en bases de datos.

#### **• Alejandro Hernández Ramos**

Con este reto logramos entender e implementar el internet de las cosas mediante una simulación de sensores utilizando Packet Tracer. Para lograr todo ello, fue necesario entender varios temas tales como la creación de bases de datos relacionales y la normalización de estas, para así poder optimizarlas y eliminar la redundancia de información. También fue necesario entender la administración de recursos de un sistema computacional, con lo cual aprendimos cómo es que llega la información desde los sensores hasta poder ser mostrada en un sitio web, todo esto mediante procesos. Para lograr crear una buena experiencia para el usuario final, fue necesario aprender diseño interactivo, con lo cual comprendí que los diseños de las interfaces de usuario tienen que ser sencillas, eficientes y seguras.

- **Diego Reyna Reyes**

Siendo esta la primera vez que he implementado algo usando la nube, debo admitir que hubo bastante confusión al principio del proyecto. Sin embargo, con el tiempo se lograron solventar estas dudas. Debo mencionar que mi principal problema con el reto fue la falta de una implementación física. Entiendo las dificultades que la situación actual nos presenta, sin embargo, creo que se pudo haber realizado la implementación tomando en cuenta la gran cantidad de facilidades para encontrar los componentes del reto. Debo admitir que esta falta de implementación electrónica redujo mi interés por el reto, debido a mi carrera IRS. Igual quiero recalcar que incluir cuestiones de administración de proyectos es un gran acierto, ya que permitió una mejor organización dentro del equipo.

- **Jonatan Hernández García**

En un inicio me sentí abrumado por todos los conocimientos que no tenía y que iba a necesitar para trabajar el reto, sin embargo, los temas se vieron en cada módulo con su respectivo paso. Algunos fueron super retadores como sistemas digitales, software o internet of things (principalmente por todo el conocimiento nuevo) sin embargo, también hubo algunos otros como recursos, diseño interactivo y procesos y administración en que fueron breves, pero de mucha utilidad. Para esta última etapa me siento confiado sobre los conocimientos adquiridos para poder desarrollarlos aún más en un futuro cercano.

- **Manuel Hernández Bravo**

En este proyecto pude apreciar todos los aspectos involucrados a la hora de realizar un proyecto de programación, desde la generación y recolección de datos hasta el diseño de una página web para que un usuario pueda consultar la información recolectada. Creo que este proyecto me sirvió para conocer las bases del desarrollo de software y todo lo que conlleva para poder aplicarlo en proyectos más complejos y estructurados.

## 6. Bibliografía

1. *Calidad del aire*. (s. f.). OPS/OMS | Organización Panamericana de la Salud. Recuperado 5 de noviembre de 2020, de <https://www.paho.org/es/temas/calidad-aire>
2. *Normas Oficiales Mexicanas (NOM) de Calidad del Aire Ambiente*. (s. f.). gob.mx. Recuperado 5 de noviembre de 2020, de <https://www.gob.mx/cofepris/acciones-y-programas/4-normas-oficiales-mexicanas-nom-de-calidad-del-aire-ambiente>
3. World Health Organization: WHO. (2018, 2 mayo). *Calidad del aire y salud*. Organización Mundial de la Salud. [https://www.who.int/es/news-room/fact-sheets/detail/ambient-\(outdoor\)-air-quality-and-health](https://www.who.int/es/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health)
4. Rojano, Roberto, Pérez, Jhonny, & Freyle, Edesnel. (2012). Efecto de la humedad relativa en la determinación de PM10 utilizando un DataRam 4, en una zona costera de Colombia. *Revista Técnica de la Facultad de Ingeniería Universidad del Zulia*, 35(2), 204-212. Recuperado de [http://ve.scielo.org/scielo.php?script=sci\\_arttext&pid=S0254-07702012000200010&lng=es&tlng=es](http://ve.scielo.org/scielo.php?script=sci_arttext&pid=S0254-07702012000200010&lng=es&tlng=es).
5. Fernández, L. (2020, March 21). *Protocolos de redes: la guía completa con todos los protocolos básicos*. RedesZone. <https://www.redeszone.net/tutoriales/internet/protocolos-basicos-redes/>

## 7. Anexos

### 7.1. Módulo 1

- 7.1.1. Simulink, primera y segunda solución:  
[https://drive.google.com/drive/folders/1r-2AkBu8-TFgrZnLRptcW\\_41NVgdYisH?usp=sharing](https://drive.google.com/drive/folders/1r-2AkBu8-TFgrZnLRptcW_41NVgdYisH?usp=sharing)
- 7.1.2. Archivos PHP para subir datos:  
[https://drive.google.com/file/d/1bIVP6fEmUau1N266Kls\\_rJF0KXcsdf57/view?usp=sharing](https://drive.google.com/file/d/1bIVP6fEmUau1N266Kls_rJF0KXcsdf57/view?usp=sharing)

### 7.2. Módulo 2

- 7.2.1. Archivo SQL de la base de datos:  
[https://drive.google.com/file/d/1dHe7TLp3MmRB4\\_jogOSB9nbnyv6Dnugb/vi ew?usp=sharing](https://drive.google.com/file/d/1dHe7TLp3MmRB4_jogOSB9nbnyv6Dnugb/vi ew?usp=sharing)
- 7.2.2. Archivos de Matlab:  
[https://drive.google.com/file/d/1YEL4bYKnRshE4CGJmXpFM6vNKriY7M\\_u/v i ew?usp=sharing](https://drive.google.com/file/d/1YEL4bYKnRshE4CGJmXpFM6vNKriY7M_u/v i ew?usp=sharing)

### 7.3. Módulo 3

- 7.3.1. Repl.it de la Tarea: <https://repl.it/@diegs117/Tarea-Mod-3-Final#main.py>

### 7.4. Módulo 4

- 7.4.1. Archivos php del Dashboard:  
<https://drive.google.com/file/d/1E42hxCQJvZ7ahqAZz1501ieXrZfwOt7h/view ?usp=sharing>

### 7.5. Módulo 5

- 7.5.1. Diagrama Flujo de datos: <https://drive.google.com/file/d/1BMI3omVi-QXmNltzIXo5GwT3s67qF2xP/view?usp=sharing>

### 7.6. Módulo 6

- 7.6.1. Archivo PKT: <https://drive.google.com/file/d/1vJJGFOI8aGKE-LQ3zsLVuYZGzkzOHWkv/view?usp=sharing>
- 7.6.2. Archivo PHP para subir datos:  
<https://drive.google.com/file/d/1uXNdSWTTCNxWFXL4vRije8M0IIHzSwcM/vi ew?usp=sharing>

### 7.7. Carpeta con los archivos anteriores

- 7.7.1. Carpeta Drive:  
<https://drive.google.com/drive/folders/1LhxacoP5mJ3PP4VGZsMAI lweA1jrRi c3?usp=sharing>