

Desarrollo de sitio web para IoT con Arduino

IoT website with Arduino

Gloria Mónica **Martínez Aguilar**¹, Abel **Sosa Escobedo**², Vanessa Maribel **Morales Ibarra**³

Universidad Tecnológica de Torreón, MÉXICO

¹<https://orcid.org/0000-0003-3834-4880> | gmartinez@utt.edu.mx

²<https://orcid.org/0000-0002-4236-326X> | asosa@utt.edu.mx

³<https://orcid.org/0000-0003-3764-8858> | vmorales@utt.edu.mx

Recibido 08-01-2022, aceptado 30-01-2022

Resumen

Internet de las cosas (IoT) es un arquetipo tecnológico que puede describirse como el conjunto de elementos que tienen pequeños sistemas integrados (sensores, microprocesadores, módulos de interconexión, actuadores, entre otros), que se comunican entre sí, para compartir información útil y mejorar los procesos. Por lo que el IoT industrial (IIoT) representa una oportunidad atractiva para aplicaciones de medición de nueva generación, que se basan cada vez más en eficientes y fiables sistemas de comunicación para permitir la amplia disponibilidad de datos permitiendo un análisis en tiempo real. Una de las cosas a tomar en cuenta con estos sistemas es la interoperabilidad de los componentes que lo integran, se han desarrollado una gran diversidad de soluciones para esto. Este tipo de sistemas son una herramienta ampliamente utilizada para la Industria 4.0 y es un tema en auge en campos industriales en los últimos años. Por lo que es de vital importancia la descripción de metodologías para la implementación de estos sistemas con tecnología asequible y amigable la cual brinde interoperabilidad, repetibilidad, monitoreo y control en tiempo real. En este trabajo se presenta una metodología para la implementación de un sitio web con IoT utilizando la tarjeta Arduino, el cual se puede orientar a diferentes áreas de operabilidad.

Palabras clave: IoT, Apache, Arduino, sitio web.

Abstract

The Internet of Things (IoT) is a technological archetype that can be described as the set of elements that have small integrated systems (sensors, microprocessors, interconnection modules, actuators, among others), which communicate with each other, to share useful information and improve processes. So Industrial IoT (IIoT) represents an attractive opportunity for next-generation measurement applications, which increasingly rely on efficient and reliable communication systems to enable the wide availability of data allowing real-time analysis. One of the things to take into account with these systems is the interoperability of the components that make it up. A wide variety of solutions have been developed for this. This type of system is a widely used tool for Industry 4.0 and is a booming topic in both industrial and non-industrial fields in recent years. Therefore, the description of methodologies for the implementation of these systems with affordable and friendly technology is vitally important, which provides interoperability, repeatability, monitoring and control in real time. In this work, a methodology for the implementation of a website with IoT using the Arduino board is presented, which can be oriented to different areas of operability.

Index terms: IoT, Apache, Arduino, website.

I. INTRODUCCIÓN

Internet de las cosas (IoT) es un paradigma tecnológico que está ganando terreno, tanto en el campo de las telecomunicaciones como en el campo computacional [1]. Se describe como un término que agrupa diferentes elementos (maquinaria, vehículos, mercancías, electrodomésticos, ropa y “cosas” en general), que poseen pequeños sistemas embebidos conformados por sensores, microprocesadores, interfaces de conexión cableada/inalámbrica y actuadores, los cuales se comunican, entre sí y con diversas aplicaciones a través de internet. Además, tienen la posibilidad de crear servicios inteligentes y autónomos. Esto permite el auge de nuevas tendencias, basadas en el almacenamiento de datos a gran escala, el análisis de cantidades robustas de información y la incorporación de sistemas ciberfísicos en comunicaciones máquina-a-máquina (M2M) [2].

IoT nace del concepto de la industria 4.0 o IIoT, implicando una evolución de las máquinas hacia un contexto de interconexión mutua para formar una comunidad colaborativa, enfocada en mejorar la productividad, reducir costos y ofrecer servicios más eficientes [3], [4]. De acuerdo a la pirámide, la automatización [5] cierra la brecha entre los niveles donde se encuentran los dispositivos de campo con los niveles donde es vital la recolección de la información para tomar decisiones inteligentes [6], [7].

Por lo que es necesario el empleo de herramientas avanzadas de control y monitoreo, que ayuden al tratamiento sistemático de diversos datos para convertirlos en información útil. De esta forma, se pueden explicar incertidumbres y optimizar procesos industriales. Ante esta situación, para el tratamiento de los datos obtenidos, es necesario la implementación de sistemas embebidos de bajo costo, por lo que varios fabricantes están invirtiendo grandes recursos investigativos en su diseño y producción, basándose en hardware y software, tanto licenciado como *open source* [8].

Entre las propuestas más relevantes, se pueden encontrar Arduino y Raspberry Pi, los cuales son dispositivos que cuentan con interfaces programables y altamente compatibles con diferentes tipos de sensores (temperatura, detección de poluciones, ruido, intensidad lumínica, entre otras) [9]. Sin embargo, aunque el uso de sistemas embebidos de bajo costo ha facilitado el despliegue de tendencias IoT, la amplia diversidad de proyectos existentes ha provocado que la mayoría de las soluciones diseñadas trabajen de manera aislada. Esto dificulta su integración en una plataforma centralizada, conllevando a la existencia de inconsistencias o falta de continuidad en los registros enviados por cada sensor.

Es por esto que en el presente trabajo se presenta una metodología para la implementación de un sitio web con IoT utilizando la tarjeta Arduino, el cual se puede orientar a diferentes áreas de operabilidad. El artículo se divide de la siguiente manera: la sección II *Metodología* en la cual se abordan *grosso modo* las características de las tecnologías utilizadas en esta propuesta, así como el desarrollo del sistema, la sección III *Análisis de resultados* donde se muestra el funcionamiento del sistema de acuerdo a lo diseñado, la sección IV *Conclusiones* donde se remarcan los puntos clave de lo obtenido en la implementación del sistema y por último las referencias.

II. METODOLOGÍA

A. Diseño del sistema

El diseño del sistema propuesto (véase Fig. 1) se basa en la implementación de un sistema web que puede ser accesado desde cualquier punto que cuente con internet, este establece una conexión a una base de datos que es alimentada por los datos monitoreados mediante un Arduino Uno acoplada con un Shield Ethernet.



Fig. 1. Diagrama del diseño del sistema.

En el desarrollo del sistema se utilizaron tecnologías digitales open source front end y back end tanto para destacar la asequibilidad, así como fácil implementación del mismo.

Una de las partes primordiales para la conectividad del sistema fue el uso de XAMPP, la cual es una distribución de Apache que incluye diferentes softwares libres. Combina un servidor web Apache, lenguaje web dinámico y una base de datos que sirve para el almacenamiento de información para servicios web. En las versiones actuales de XAMPP la base de datos se ha sustituido por MariaDB. Su ventaja de uso, es evitar la descarga e instalación de cada componente por separado y para crear o editar sus ficheros de configuración manualmente, XAMPP sólo requiere una pequeña fracción del tiempo necesario para descargar y ejecutar. Puede encontrarse en versión completa o en una versión más ligera, y permite configurar los componentes necesarios del servidor web mediante una misma y sencilla interfaz web. XAMPP puede ser ejecutado en diferentes sistemas operativos, es software de distribución libre, segura, y no requiere pago de licencias.

Las páginas del sistema se programaron en Hypertext Preprocessor (PHP), que es un lenguaje de código abierto muy popular, especialmente adecuado para el desarrollo web y que puede ser incrustado en Hypertext Markup Language (HTML), que es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. PHP se utiliza fundamentalmente para realizar páginas web dinámicas y este lenguaje es procesado en servidores. Lo mejor de esto es que al ser instalado en el servidor, el usuario no puede modificarlo en su ordenador y gracias a esto se pueden hacer aplicaciones más seguras. Desde este punto se definió la seguridad del sistema.

En el front-end se usaron hojas de estilo en cascada (CSS), las cuales son un lenguaje de estilos utilizado para describir la presentación de documentos HTML que permiten describir cómo serán renderizados los elementos estructurados de una página web en la pantalla, dándoles la apariencia que se desee, como: colores, espacios entre elementos, tipos de letra, separando de esta forma, la estructura de la presentación, entre otros. También se utilizó JavaScript el cual es un lenguaje de programación o de secuencias de comandos que permite implementar funciones complejas en páginas web. Por ejemplo, cada vez que una página web hace algo más que mostrar información estática como: muestras oportunas actualizaciones de contenido, mapas interactivos, animación de gráficos 2D/3D, desplazamiento de máquinas reproductoras de vídeo, etc., se está haciendo uso de este lenguaje.

Se usaron ambas tecnologías con la finalidad de desplegar los datos adquiridos por el sistema de una manera amena para el usuario, y se implementaron desde Bootstrap, que es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios web y aplicaciones web.

Dentro del back-end, para la parte de la base datos se utilizó MySQL Workbench que ofrece un modelo relacional, el cual almacena los datos en tablas, realiza consultas, búsquedas, filtros, entre otras cosas. Se decidió utilizar esta tecnología porque es una herramienta que permite modelar diagramas de entidad-relación para

bases de datos MySQL. Puede utilizarse para diseñar el esquema de una base de datos nueva, documentar una ya existente o realizar una migración compleja, dado que es un software de código abierto y multiplataforma de aplicaciones de escritorio para la administración de servidores de bases de datos MySQL puede construir y manipular los datos de estas.

Parte de la visualización de datos en tiempo real fue implementada con CanvasJS, que es una biblioteca para entorno WEB con tipos diferentes de gráficos y representaciones en dispositivos, incluidos iPhone, iPad, Android, Mac y PC. Esto permite crear cuadros de mando completos que funcionan en todos los dispositivos sin comprometer la capacidad de mantenimiento o la funcionalidad de su aplicación web. Los gráficos incluyen varios temas atractivos y son diez veces más rápidos que las bibliotecas de gráficos convencionales basadas en Flash/SVG, lo que da como resultado paneles de control livianos, visuales y receptivos.

La adquisición de datos se realizó mediante un Arduino Uno y una Ethernet Shield (véase Fig. 2). El Arduino es una tarjeta electrónica que tiene todos los elementos necesarios para conectar periféricos a las entradas y salidas de un microcontrolador. Es decir, es una placa impresa con los componentes necesarios para que funcione el microcontrolador y su comunicación con un ordenador a través de la comunicación serial. Mientras que el Ethernet Shield es una tarjeta electrónica que permite a un Arduino tener una conectividad al protocolo TCP/IP. Este tipo de protocolo se creó para comunicar a los dispositivos digitales mediante una conexión en red. Ambas placas son asequibles, así como de fácil programación y una de las principales ventajas es el modularidad que tienen, lo cual permite que el sistema sea de arquitectura abierta, permitiendo así la reconfiguración de los datos a adquirir sin importar la naturaleza de estos.

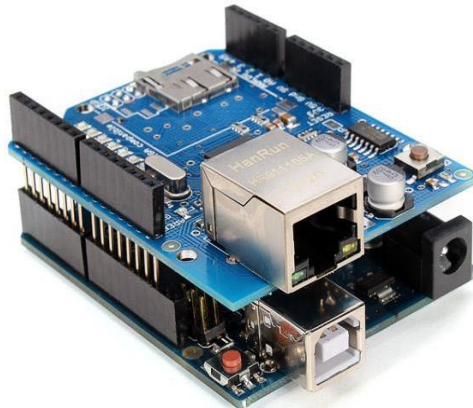


Fig. 2. Acoplamiento de Arduino Uno y Ethernet Shield.

B. Desarrollo del sistema

El primer paso del desarrollo del sistema es crear una base de datos en MySQL usando el entorno PHPMyAdmin en la cual se almacenará la información de lo recibido por Arduino vía Ethernet.

La base de datos se creó con tres tablas (véase Fig. 3). La primera es 'usuario' la cual tiene tres campos:

- iduser de tipo entero (int),
- usuario de tipo de texto (varchar),
- contraseña de tipo texto (varchar).

El objetivo de la tabla es almacenar a usuarios que tengan permiso de acceder al sistema. La segunda es la tabla 'registro', la cual tiene los campos:

- nombre, apellido, correo electrónico, usuario y contraseña de tipo texto (varchar),
- y una referencia o llave foránea (int), que conecta con la otra tabla que referencia a que usuario se está ingresando.

Con esta tabla se establece un control de acceso de los usuarios al sistema. Por último, la tabla datos, consiste en tres campos:

- entradas de tipo decimal (float),
- salidas de tipo decimal (float) y
- fecha de tipo fecha-hora (datetime).

Esta tabla es el punto central del sistema, dado que es la encargada de almacenar la información que manda la Arduino y de ahí partir con la funcionalidad del sistema, como lo es graficar y generar archivos CSV a partir de esta tabla de información.

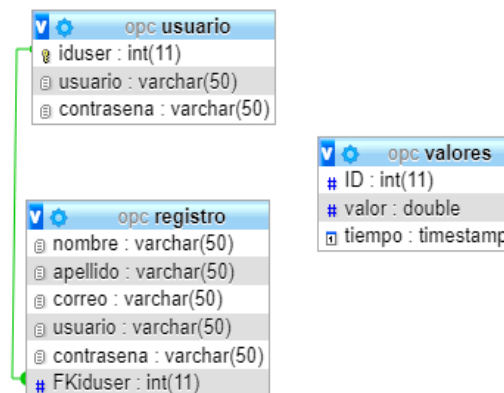


Fig. 3. Diagrama modelo entidad-relación de la base de datos.

Después de la creación de la base de datos, se configura el servidor apache desde el panel de control de XAMPP (véase Fig. 4), donde es necesario seleccionar al botón de config de la opción del servidor Apache, el cual despliega un menú contextual y se selecciona Apache (httpd.conf), lo cual abre un bloc de notas con la configuración del servidor, se busca localhost y el puerto de salida, y ahí se cambia la IP asignada como servidor (véase Fig. 5, lo que está en color azul es la dirección IP del servidor).

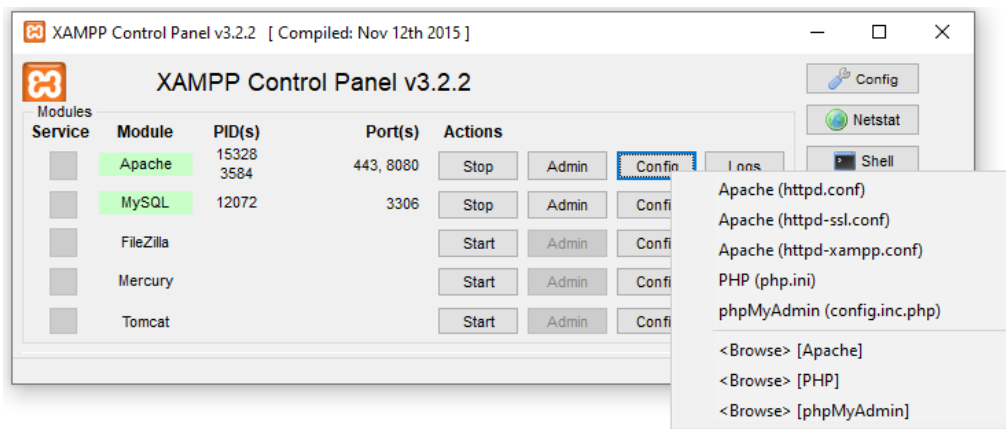


Fig. 4. Panel de control de XAMPP.

```
# If your host doesn't have a registered DNS
name, enter its IP address here.
#
#ServerName localhost:8080
ServerName 192.168.40.100:8080
```

Fig. 5. Archivo de configuración Apache.

6

Una vez configurada la base de datos y el servidor Apache, se programa el Arduino, para la conexión de este con el servidor Apache y la escritura de registros en la base de datos. En la Fig. 6 se muestra un ejemplo del envío de un dato analógico de prueba que simulara un sensor de temperatura. Cabe destacar que los datos enviados pueden ser digitales o analógicos y pueden ser todos los que sean necesarios de acuerdo a las necesidades de la implementación de este sistema.

```
ethernetbd
#include <Ethernet.h>
#include <SPI.h>

// Configuración del Ethernet Shield
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFF, 0xEE}; // Dirección MAC
byte ip[] = { 192,168,40,102 }; // Dirección IP del Arduino
byte server[] = { 192,168,40,100 }; // Dirección IP del servidor
EthernetClient client;
float temperatura;
int analog_pin = 0;
|
void setup(void) {
  Ethernet.begin(mac, ip); // Inicializamos el Ethernet Shield
  delay(1000); // Esperamos 1 segundo de cortesía
}

void loop(void) {
  temperatura = analogRead(analog_pin);
  temperatura = 5.0*temperatura*100.0/1024.0;
  //Display in Serial Monitor
  Serial.print(temperatura); //Return temperature to Monitor
  Serial.println(" oC");
  // Proceso de envío de muestras al servidor
  Serial.println("Connecting...");
  if (client.connect(server, 8080)>0) { // Conexión con el servidor
    client.print("GET /opc/iot.php?valor="); // Enviamos los datos por GET
    client.print(temperatura);
    client.println(" HTTP/1.0");
    client.println("User-Agent: Arduino 1.0");
    client.println();
    Serial.println("Conectado");
  } else {
    Serial.println("Fallo en la conexión");
  }
  if (!client.connected()) {
    Serial.println("Disconnected!");
  }
  client.stop();
  client.flush();
  delay(60000); // Espero un minuto antes de tomar otra muestra
}
```

Fig. 6. Programación de Arduino para envío de dato analógico de temperatura.

El diagrama de conexión para la prueba del sistema se muestra en la Fig. 7.

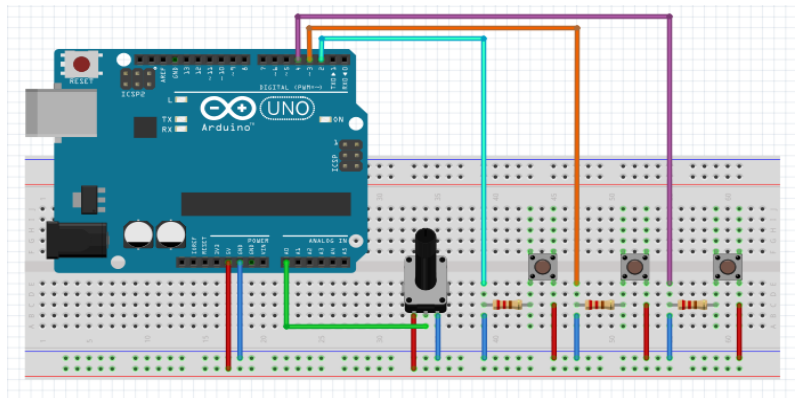


Fig. 7. Conexión de Arduino para envío de datos.

En este diagrama se adquiere una señal analógica por la entrada A0 del Arduino la cual varía con la perilla del potenciómetro el cual simulara un sensor de temperatura, también se adquieren 3 señales digitales en los pines D3, D4 y D5. Como ya se mencionó previamente este método de adquisición permite la modularidad lo cual implica que dependiendo de las necesidades de aplicación se pueden configurar y conectar una gran diversidad de sensores y actuadores.

El sistema hasta este punto está listo para envío de datos al servidor, solo falta la creación del sitio web que permitirá el monitoreo de los datos guardados en la base de datos. Para esto lo primero es crear una carpeta (en este caso se llamó proyecto) en la siguiente ruta: C:\xampp\htdocs\, la cual se encarga de arrancar el servidor Apache con las páginas WEB en conjunto con la parte lógica y el PHP. En esta carpeta se crean dos archivos, el primero llamado config.php el cual contiene la información para establecer una conexión con la base de datos, el servidor, el usuario y contraseña del servidor Apache. El segundo es iot.php, su funcionalidad es subir los datos recibidos de los sensores a la base de datos. Los datos se cargan a la base de datos a través del método GET.

Con estos archivos se asegura el arranque del servidor, del sistema web y la conectividad de la base de datos desde el Arduino. Después de esto, se realiza la programación de las páginas del sitio web, donde se utilizaron librerías de Bootstrap el cual mejora la apariencia de estas, además están en PHP para que corran desde el servidor. El sitio consta de 6 páginas:

- La primera página web es la de inicio de sesión y tiene como objetivo que el usuario pueda ingresar al sistema tecleando credenciales de acceso como: un usuario y una contraseña.
- La segunda página web consiste en el registro del usuario, si este aún no cuenta con credenciales de acceso, podrá registrarse para obtenerlas.
- La tercera página web consta de un menú de opciones, que despliega dos acciones: en primera instancia un monitoreo de tiempo real de los datos obtenidos y de manera simultánea guardados en la base de datos desde el Arduino Uno y la Ethernet Shield, en esta parte se da la opción de guardado de las sesiones en un archivo .CSV lo cual da pie a la segunda opción que consiste en la carga de estos archivos y la visualización de estos para la creación de reportes.

8

- La cuarta página permite la visualización de gráficos en tiempo real, se implementó la librería Canvas JS, la cual permite generar las ondas de las señales (analógicas) adquiridas de manera coordinada directamente en el espacio establecido para su representación. Así mismo cuenta con opción de hacer zoom en una parte específica de la gráfica y mover de un lado a otro o bien regresar al estado inicial. También se muestran las señales (digitales) adquiridas a modo de pequeños leds. Desde aquí se pueden guardar los datos en un archivo como previamente se describió para su futura visualización según se requiera.
- La quinta página web consiste en la simulación de los archivos previamente guardados que contienen la información adquirida en tiempo real. El objetivo de esta página es visualizar los datos mediante una gráfica de manera similar a la de tiempo real.
- Y por último la sexta página consiste en visualizar dos gráficas, la primera es de la simulación del archivo previamente guardado (CSV) que contiene la información adquirida en tiempo real, y en la segunda un zoom de la primera.

En la Fig. 8 se presenta el código QR para acceder a la programación del sitio web propuesto para los lectores que quieran replicarlo.



Fig. 8. Código QR para acceder a la programación del sitio web propuesto.

Una vez terminadas las páginas del sitio web se procede a realizar pruebas de conexión y visualización del sitio.

III. ANÁLISIS DE RESULTADOS

Para el testeo del sitio web se hicieron una serie de pruebas de conectividad y funcionalidad. Primero se realizó una prueba de conexión del Arduino con la base de datos, donde se observó el correcto guardado de información de los datos adquiridos (véase Fig. 9).

EntradaAnalogica	EntradaDigital	fecha
200	0	2021/06/23 15:40:00
155	1	2021/06/22 14:40:00
222	0	2021/06/20 14:45:00
156	1	2021/06/15 12:40:00
187	0	2021/06/05 14:45:00

Fig. 9. Tabla con información de monitoreo.

Después se realizó el análisis de funcionalidad de cada una de las páginas del sitio web. La primera página, la de inicio de sesión (véase Fig. 10) se probó dando de alta usuarios.

9



Fig. 10. Página de inicio de sesión.

Donde el registro de los usuarios solicita los siguientes datos: nombre completo, apellido, correo electrónico, contraseña y confirmar contraseña (véase Fig. 11).



Fig. 11. Página creación de cuenta nueva.

Una vez llenados estos datos se registran los usuarios, y se guardan en la base de datos donde se verificó el correcto funcionamiento de esta página (véase Fig. 12).

iduser	usuario	contrasena
1	asosa	aEsa1@
2	maguilar	mMarA9@
3	mmorales	orAles1@

Fig. 12. Usuarios registrados en base de datos.

Con los usuarios registrados se procedió a acceder al sistema donde se presenta una página con un menú de opciones (véase Fig. 13), donde primero se eligió la de monitoreo en tiempo real.



Fig. 13. Página con menú de opciones del sistema.

En la página que se abre para el monitoreo de datos en tiempo real (véase Fig. 14), se visualiza en una gráfica los datos analógicos adquiridos desde Arduino y al lado izquierdo de esta los datos digitales en forma de leds. También se tiene un botón del lado superior izquierdo con el cual se inicializa el grabado de estos datos en un archivo CSV, en la parte inferior de la gráfica se da la opción para el guardado de este archivo.

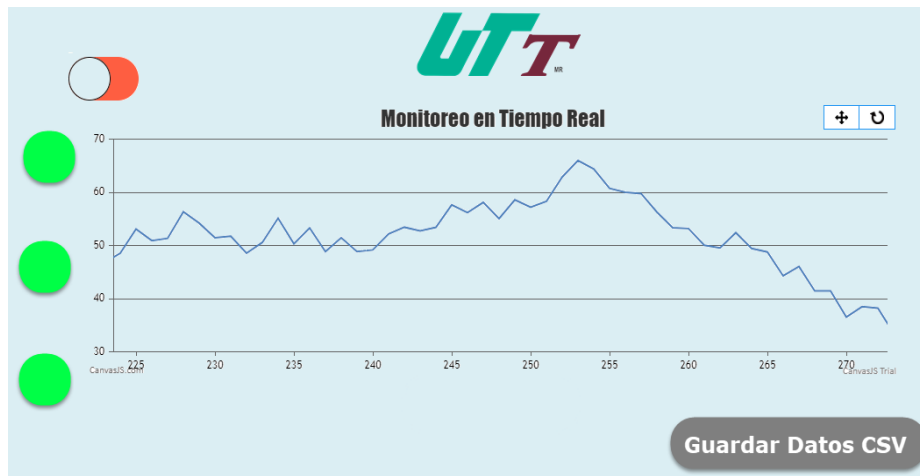


Fig. 14. Gráfica generada en tiempo real.

En la gráfica de esta página se pueden ejecutar acercamientos posicionando el puntero del ratón en la parte que se desea visualizar, en seguida, en la parte superior derecha de la gráfica aparece dos iconos, uno en forma de cruz, el cual el usuario puede mover la gráfica hacia la derecha o izquierda según sea la necesidad, el segundo icono que se encuentra en la parte superior derecha es regresar la gráfica en un estado normal.

Por último, se comprobó la opción de simulación donde la página solicita el archivo previamente guardado con formato CSV para cargar los datos a simular (véase Fig. 15).

11

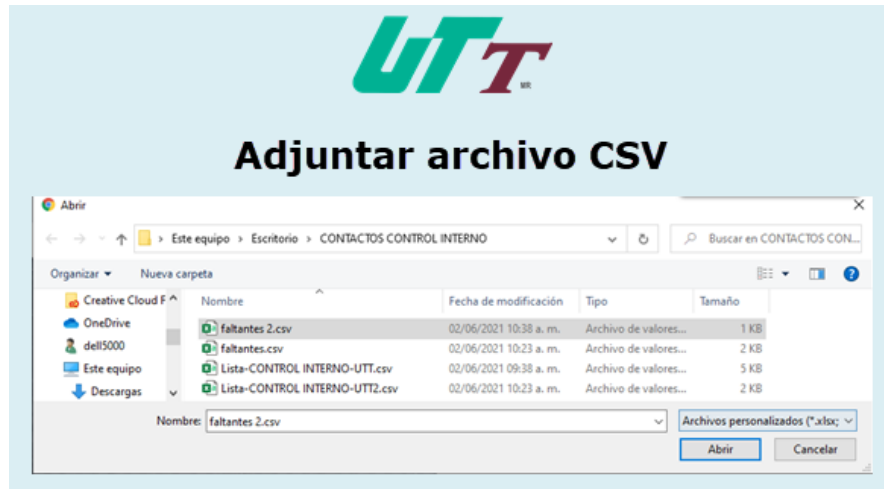


Fig. 15. Página donde se adjunta archivo CSV.

Una vez cargado el archivo CSV seleccionado se abre la página con las gráficas de la simulación de los datos contenidos en este (véase Fig. 16). Cabe destacar que este tipo de archivo es un tipo de formato que contiene información separada por comas y se puede ver en Excel o bloc de notas.

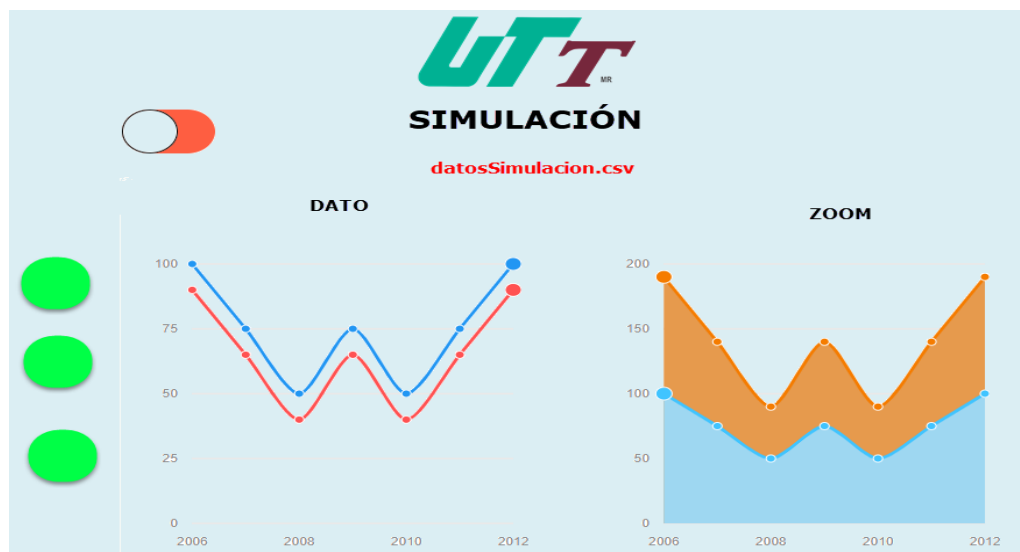


Fig. 16. Página de simulación de datos.

La página de simulación de datos está conformada por dos gráficas, la primera con los datos tomados del archivo seleccionado y la segunda de realizar un zoom a la primera.

De manera satisfactoria se probó el correcto funcionamiento del sitio desarrollado, sin presentarse ningún problema de conectividad o funcionalidad en las páginas que lo conforman.

IV. CONCLUSIONES

12

La IoT promete abrir la puerta a un mundo revolucionario, un mundo “inteligente” totalmente interconectado en el cual las relaciones entre los objetos y su entorno y las personas se entrelazan aún más. Cada día existen más soluciones para la implementación en diferentes entornos de este arquetipo.

El sistema desarrollado cumple con el objetivo de implementación para el monitoreo de manera remota de datos, resaltando la modularidad del sistema que ofrece una arquitectura abierta. Además de su interoperabilidad por las tecnologías implementadas y su método de adquisición, lo cual le da versatilidad para poder usarse en varios dispositivos, así como en diversas áreas de aplicación. Destacando que el usuario final podrá acceder a través de cualquier navegador instalado en cualquier sistema operativo (Windows, Linux, Unix, Macintosh, etcétera).

Los tiempos de respuesta entre el mando vía Web y la ejecución en el dispositivo electrónico dependen de la velocidad de Internet utilizada, pero no se requieren grandes velocidades para conseguir un manejo óptimo aproximado al tiempo real.

La descripción del desarrollo del sistema deja abierta la posibilidad de personificación de este de acuerdo a las necesidades de aplicación, de una manera sencilla tanto en software como en hardware, permitiendo de esta manera la interconexión característica de un sistema IoT.

REFERENCIAS

- [1] M. Mardiana, H. Wan, “Current research on Internet of Things (IoT) security: A survey,” *Computer Networks*, vol. 148, 2019, pp. 283-294, <https://doi.org/10.1016/j.comnet.2018.11.025>
- [2] A. J. Jara, D. Genoud, Y. Bocchi, *Big Data for Cyber Physical Systems: An Analysis of Challenges, Solutions and Opportunities*, 2014. <https://doi.org/376-380>. 10.1109/IMIS.2014.139
- [3] M. Okuda, T. Mizuya, and T. Nagao, “Development of iot testbed using opc ua and database on cloud,” *2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, 2017.
- [4] P. L. Gallegos-Segovia, J. F. Bravo-Torres, J. J. Argudo-Parra, E. J. Sacoto-Cabrera, V. M. Larios-Rosillo, “Internet of Things as an attack vector to critical infrastructures of cities,” *2017 International Caribbean Conference on Devices, Circuits and Systems (ICCDACS)*, Cozumel, 2017, pp. 117-120.
- [5] A. Contan, C. Dehelean, L. Miclea, “Test automation pyramid from theory to practice,” *2018 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, 2018, pp. 1-5, doi: <https://doi.org/10.1109/AQTR.2018.8402699>
- [6] J. J. Lee, H. A. Kao, S. Yang, “Service innovation and smart analytics for industry 4.0 and big data environment,” *Procedia Cirp*, vol. 16, pp. 3-8, 2014.
- [7] S. Bhardwaj, P. Larbig, R. Khondoker, K. Bayarou, “Survey of domain specific languages to build packet parsers for industrial protocols,” *2017 20th International Conference of Computer and Information Technology (ICCIT)*, 2017.
- [8] N. Vikram, K. Harish, M. Nihaal, R. Umesh, A. Shetty, A. Kumar, “A low cost home automation system using wi-fi based wireless sensor network incorporating internet of things (iot),” *2017 IEEE 7th International Advance Computing Conference (IACC)*, 2017.
- [9] N. Agrawal, S. Singhal, “Smart drip irrigation system using raspberry pi and arduino,” in *Computing, Communication & Automation (ICCCA), 2015 International Conference on. IEEE*, 2015, pp. 928-932.