



UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

VISUALIZACIÓN DEL ESTADO DE SENSORES UTILIZANDO EL
MICROCONTROLADOR ARDUINO PARA LA CONSOLIDACION DE
DATOS.

AYALA MENDIETA JOFFRE MANUEL
INGENIERO DE SISTEMAS

MACHALA
2018



UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

VISUALIZACIÓN DEL ESTADO DE SENSORES UTILIZANDO EL
MICROCONTROLADOR ARDUINO PARA LA CONSOLIDACION DE
DATOS.

AYALA MENDIETA JOFFRE MANUEL
INGENIERO DE SISTEMAS

MACHALA
2018



UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

EXAMEN COMPLEXIVO

VISUALIZACIÓN DEL ESTADO DE SENSORES UTILIZANDO EL
MICROCONTROLADOR ARDUINO PARA LA CONSOLIDACION DE DATOS.

AYALA MENDIETA JOFFRE MANUEL
INGENIERO DE SISTEMAS

CÁRDENAS VILLAVICENCIO OSCAR EFRÉN

MACHALA, 05 DE JULIO DE 2018

MACHALA
05 de julio de 2018

Nota de aceptación:

Quienes suscriben, en nuestra condición de evaluadores del trabajo de titulación denominado Visualización del estado de sensores utilizando el microcontrolador arduino para la consolidación de datos., hacemos constar que luego de haber revisado el manuscrito del precitado trabajo, consideramos que reúne las condiciones académicas para continuar con la fase de evaluación correspondiente.



CARDENAS VILLAVICENCIO OSCAR EFRÉN
0703935312
TUTOR - ESPECIALISTA 1



NOVILLO VICUÑA JOHNNY PAUL
0702947409
ESPECIALISTA 2



RIVAS ASANZA WILMER BRAULIO
0702580192
ESPECIALISTA 3

Fecha de impresión: martes 10 de julio de 2018 - 07:58

Urkund Analysis Result

Analysed Document: AyalaMendietaJoffreManuel20181.pdf (D40261866)
Submitted: 6/19/2018 3:05:00 PM
Submitted By: oecardenas@utmachala.edu.ec
Significance: 0 %

Sources included in the report:

Instances where selected sources appear:

0

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, AYALA MENDIETA JOFFRE MANUEL, en calidad de autor del siguiente trabajo escrito titulado Visualización del estado de sensores utilizando el microcontrolador arduino para la consolidacion de datos., otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.


El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 05 de julio de 2018



AYALA MENDIETA JOFFRE MANUEL
0704877638

DEDICATORIA

Dedico mi trabajo a Dios por permitirme mantenerme con salud para cumplir con mis metas, a mi madre por ser el pilar fundamental para mi desarrollo durante el transcurso de mi vida y brindarme sus sabios consejos que ayudaron a la culminación de mi formación académica, a mis hijos por mi inspiración de superación, y a los docentes de la UTMACH por su orientación durante el proceso de mi formación académica.

Sr. Joffre Manuel Ayala Mendieta

AGRADECIMIENTO

Agradezco a Dios por haberme brindado salud y sabiduría a lo largo de mis estudios académicos, a mi madre Nancy Mendieta por su demostración de superación y perseverancia y enseñarme a luchar por cumplir mis objetivos con constancia sin rendirme ante ningún obstáculo, a todos los que conforman la Carrera de Ingeniería de Sistemas de la Universidad Técnica de Machala por brindarme sus sabios consejos en el desarrollo de mi formación, en especial al Ing. Jhonny Novillo y al Ing. Oscar Cárdenas por su paciencia y apoyo para ayudarnos como guía durante todo el proceso de culminación de mi carrera.

Sr. Joffre Manuel Ayala Mendieta

RESUMEN

VISUALIZACIÓN DEL ESTADO DE SENSORES UTILIZANDO EL MICRONTROLADOR ARDUINO PARA LA CONSOLIDACIÓN DE DATOS.

Ayala Mendieta Joffre Manuel, 0704877638

La implementación de un servidor web en la tarjeta Arduino, permite observar de manera sencilla y rápida los estados actuales de varios sensores que se encuentren conectados a la misma, almacenando la información obtenida por los dispositivos en la base de datos SQLite. El firmware creado para la tarjeta microcontroladora Arduino Mega levanta un servidor web popularmente conocido como servduino utilizando un Ethernet Shield, logrando una efectiva comunicación con la red y aprovechando los beneficios que brinda como es la gran cantidad de puertos disponibles y su estabilidad en producción. Los sensores utilizados en el proyecto para la lectura de los datos son los siguientes: de Temperatura y Humedad, proximidad ultrasónico, campos magnéticos, fuego y nivel de agua, los registros generados se envían a través del protocolo MQTT al bróker Mosquitto y este a su vez al middleware programado en Python, para la visualización, monitorización y consolidación de los datos.

PALABRAS CLAVES: Arduino, servidor, consolidación, sensores, middleware.

ABSTRACT

VISUALIZATION OF THE SENSORS STATE USING THE MICROPROCESSOR DEVELOPMENT CARD ARDUINO FOR DATA CONSOLIDATION.

Ayala Mendieta Joffre Manuel, 0704877638

The implementation of a web server on an Arduino board allows to monitor easily and quickly the current states of several sensors that are connected to it, storing the information obtained by the devices in the SQLite database. The firmware created for the Arduino Mega microcontroller board raises a web server popularly known as servduino using an Ethernet Shield, achieving effective communication with the network and taking advantage of the large number of ports available and its stability in production. The sensors used in the project to generate data are the following: Temperature and Humidity, ultrasonic proximity, magnetic fields, fire and water level, the generated records are sent through the MQTT protocol to the Mosquitto broker and this to the middleware programmed in Python, for the visualization, monitoring and consolidation of the data.

KEYWORDS: arduino, server, consolidate, sensors, middleware.

ÍNDICE DE CONTENIDO

| | |
|--|-----------|
| DEDICATORIA | 3 |
| AGRADECIMIENTO | 4 |
| RESUMEN | 5 |
| ABSTRACT | 6 |
| ÍNDICE DE FIGURAS | 9 |
| ÍNDICE DE TABLAS | 10 |
| ÍNDICE DE ANEXOS | 11 |
| 1. INTRODUCCIÓN | 12 |
| 1.1. MARCO CONTEXTUAL | 13 |
| 1.2. PROBLEMA | 13 |
| 1.3. OBJETIVO GENERAL | 13 |
| 2. DESARROLLO | 14 |
| 2.1. MARCO TEÓRICO | 14 |
| 2.1.1. Arduino Mega | 14 |
| 2.1.2. Ethernet Shields | 14 |
| 2.1.3. Servidor web en arduino (servduino) | 15 |
| 2.1.4. Sensores | 15 |
| 2.1.5. Python | 17 |
| 2.1.6. Paho | 17 |
| 2.1.7. Base de datos embebida SQLite | 17 |
| 2.1.8. Protocolo de comunicación MQTT | 18 |
| 2.2. SOLUCIÓN DEL PROBLEMA | 19 |
| 2.3. RESULTADOS | 19 |
| 3. CONCLUSIONES | 21 |
| 4. REFERENCIAS BIBLIOGRÁFICAS | 21 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1: Mediciones de latencia de envío de mensajes. | 17 |
| Figura 2: Sitio web. | 18 |
| Figura 3: Middleware. | 19 |
| Figura 4: Información consolidada en la base de datos SQLite. | 19 |

ÍNDICE DE TABLAS

Tabla 1: Funcionamiento de Sensores.

15

ÍNDICE DE ANEXOS

| | |
|---|----|
| Anexo A: Diagrama de flujo de Arduino Mega. | 22 |
| Anexo B: Diagrama de flujo de Middleware. | 23 |
| Anexo C: Diagrama de Red. | 24 |
| Anexo D: Diagrama de conexión. | 25 |
| Anexo E: Diagrama Esquemático. | 26 |

1. INTRODUCCIÓN

En la actualidad la automatización de procesos son indispensables para el desarrollo de instituciones o desenvolvimiento de las personas en especiales adultos mayores o discapacitados ya que permite automatizar los procesos y obtener mejores resultados en las actividades realizadas. [1].

El auge de las tecnologías tecnifican una gran parte de los procesos de producción y ayudan a los usuarios a realizar menos trabajo manual, obteniendo resultados más gratificantes. La mayoría de automatización de procesos son realizados en computadores ya sea de mesa o portátiles que tengan un microprocesador y memorias que almacenen los datos en el mismo ensamblado. [2]

Existen diferentes tecnologías que ayudan al aprendizaje y enseñanza en las instituciones académicas como es el caso de las tarjetas: Raspberry, Arduino que permiten realizar prácticas electrónicas a los estudiantes con la ayuda de los dispositivos inalámbricos de buena calidad que aportan al correcto funcionamiento de sus tareas escolares. [3] [4]

Arduino es una tarjeta microprocesadora de fácil uso para su programación de proyectos como monitoreo de sensores o dispositivos, es de bajos costos con entorno amigable para estudiantes. La utilización de la tarjeta Arduino graba instrucciones programadas, permite la conexión de dispositivos de entrada y salida con conexión a un computador. [5]

Es importante poder visualizar los datos obtenidos de las instrucciones ejecutadas en la tarjeta arduino, para ello se necesita la elaboración de una aplicación donde se detallen la información consolidada.

El proyecto tiene como objetivo visualizar los estados de sensores en la tarjeta microprocesadora Ardino para la consolidación de los datos, mediante el desarrollo de un servidor web que permita el monitoreo de los estados producidos por los sensores: de nivel de agua, temperatura y humedad, proximidad, fuego y campos magnéticos.

.

.

1.1. MARCO CONTEXTUAL

En la actualidad, el internet de las cosas ha ganado una importancia muy notoria dentro de la informática y electrónica debido a que permite la automatización y monitoreo de procesos. Gracias al rápido avance tecnológico en esta área se realizan tareas complejas como es el caso del registro de estados de sensores a través de un computador para lograr control y consolidación de datos sin necesidad de intervención humana. El proyecto realizado es un sistema web implementado en la placa Arduino Mega el cual permite la obtención de valores que presentan los sensores de nivel de agua – FUNDUINO water sensor, temperatura y humedad – KY 015, de proximidad ultrasónico – HC SR04, de campos magnéticos – KY 025, y de fuego – KY 026, para su monitorización y almacenamiento en la base de datos SQLite con la ayuda del protocolo MQTT y un middleware escrito en Python.

1.2. PROBLEMA

El monitoreo de los datos producidos por los sensores es una tecnología de punta necesaria para poder controlar fácilmente la información que se presentan desde una aplicación web a través de un dispositivo electrónico. Por tal motivo se pretende visualizar los estados de los sensores nivel de agua, fuego, temperatura y humedad, campos magnéticos y proximidad, utilizando la tarjeta microprocesadora Arduino para la consolidación de los datos.

1.3. OBJETIVO GENERAL

Visualizar los estados de sensores utilizando la tarjeta microprocesadora Arduino para la consolidación de datos.

2. DESARROLLO

2.1. MARCO TEÓRICO

2.1.1. Arduino Mega

Las placas Arduino fueron originalmente creadas como placas de prototipado sencillo de aprendizaje para estudiantes, sin embargo, han evolucionado hasta alcanzar un nivel muy alto de calidad y con estándares que permiten a estas ser usadas en proyectos altamente complejos. [6]

Arduino mega es una placa de hardware libre desarrollada por la compañía Arduino basada en el procesador ATmega2560 programable a través de un puerto USB montado sobre la placa. Cuenta con un total de 54 pines digitales y 16 analógicos, lo que lo hace ideal para realizar proyectos de gran escala. Es una placa robusta y compatible con la mayoría de las placas de Arduino. La programación de la placa se realiza en lenguaje C y la compañía creadora de esta distribuye de manera gratuita un IDE para la compilación de los programas.

El IDE de Arduino posee también conexión con un repositorio de librerías para su fácil acceso, además de un asistente de instalación de librerías para librerías fuera de repositorio. Posee además tarjetas de expansión o shields que se utilizan para agregar nuevas funcionalidades a las tarjetas de Arduino tales como comunicación Ethernet, Wireless, Bluetooth, entre otras, y cuentan con sus respectivos controladores en el repositorio. [7]

2.1.2. Ethernet Shields

Ethernet es uno de los medios de comunicación más utilizados a nivel mundial debido a su robustez y facilidad de manejo e instalación. Esto implica que las redes montadas sobre ethernet pueden proporcionar un buen rendimiento con la posibilidad de escalar en tamaño. [8]

El Ethernet shield es una tarjeta de expansión que permite a las placas Arduino conectarse a una red mediante un cable ethernet, se adapta a cualquier placa de Arduino que tenga

una configuración de pines similar a la de la tarjeta Arduino UNO. Para la programación de la comunicación se requiere el uso de una librería llamada Ethernet, misma que se encuentra en el repositorio del IDE Arduino.

2.1.3. Servidor web en arduino (servduino)

Los servidores web son el conjunto de software que utiliza el protocolo HTTP que utilizan el modelo cliente-servidor para realizar conexiones con clientes a través de internet y proveerles toda la información requerida para mostrar una página web. Los servidores web son un componente vital para la difusión de información en la actualidad llegando a tener que manejar millones de conexiones por día.


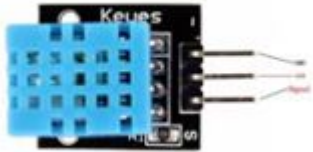



Uno de los servidores web menos conocidos es el que se puede crear con las tarjetas Arduino en conjunto con un EthernetShield. Arduino es un componente muy versátil que facilita la conectividad a la red y que permite levantar protocolos de alto nivel, tal como el http [9], es decir, levanta sitios web directamente desde la placa microprocesadora, permitiendo de esta manera mitigar la carga de servidores centrales o simplemente mostrando páginas definidas dentro de la tarjeta. Para realizar este proceso se utiliza la librería ethernet con la que el ethernet shield podrá responder y recibir peticiones HTTP.

2.1.4. Sensores

Los sensores son dispositivos electrónicos que se encargan de monitorear el ambiente en búsqueda de cambios en el mismo y lo reporta mediante un valor que puede ser análogo o digital. Son un componente básico en todos los sistemas electrónicos que se encuentran implementados en la actualidad. [10]

En la tabla 1 se puede observar el funcionamiento de algunos sensores utilizados en el proyecto:

Tabla 1: Funcionamiento de Sensores

| Sensores | Funcionamiento | Imagen |
|--|---|--|
| Sensor de nivel del agua - FUNDUINO Water Sensor | Mide el nivel de condensación de su entorno inmediato y devuelve un valor análogo |  |
| Sensor de temperatura y humedad – KY 015 | Sensor que determina la temperatura y humedad, devuelve un valor análogo |  |
| Sensor de proximidad – HC SR04 | El sensor de proximidad permite determina la distancia de un objeto frente al sensor |  |
| Sensor de campos magnéticos – KY 025 | Sensor que permite detector si existen campos magnéticos en la proximidad, devuelve valores digitales y análogos según la necesidad |  |
| Sensor de fuego – KY 026 | Sensor que permite detector si existe fuego en la proximidad, devuelve valores digitales y análogos según la necesidad |  |

Fuente: El autor

2.1.5. Python

Python es un lenguaje de programación interpretado y dinámico que hace énfasis en un código fácilmente legible y sencillo de entender, posee librerías muy poderosas para funciones específicas, haciéndolo especialmente atractivo para la comunidad científica. Python se han vuelto una elección popular para el desarrollo de aplicaciones web, prototipos rápidos, scripts, entre muchas otras funciones que requieran adaptarse a cambios rápidamente. [11]

Python fue diseñado como un lenguaje de vinculación de sistemas, un bridge o middleware de datos generados en sistemas de alto nivel a lenguajes de bajo nivel, interacción entre lenguaje interpretado y código compilado. [12]

2.1.6. Paho

Paho es un Proyecto de Eclipse Foundation financiado activamente por IBM que busca brindar a los desarrolladores implementaciones de clientes MQTT y MQTT-SN de código abierto y para protocolos nuevos que puedan aparecer en el aspecto IoT.

Paho permite superar muchas de las limitaciones que se tenían en la implementación de sistemas que conecten dispositivos mediante protocolo MQTT facilitando la creación y crecimiento de soluciones que funcionen de manera individual o middlewares en múltiples lenguajes en los que se puede ejecutar. Este proyecto cuenta además con apoyo de la comunidad e intenta avivar el ideal de mantener abierto y provocar el crecimiento del mercado de proyectos IoT.

2.1.7. Base de datos embebida SQLite

Sqlite es un Sistema de base de datos embebido que implementa ACID (atomicidad, consistencia, aislamiento y durabilidad) y la mayor parte del estándar SQL. Permite almacenar una base de datos relacional en sistemas con bajo consumo de recursos y es ampliamente utilizada por sistemas web para almacenamiento local, aplicaciones de dispositivos móviles, y en general por dispositivos con recursos limitados. [13]

Posee el beneficio de ser independiente de la plataforma en la que se ejecute y tiene las operaciones que se realizan en la base de datos son dadas directamente por la aplicación

que la usa y el lenguaje de programación que esta use. Esto le da una base sólida de trabajo multiplataforma. Además de poder residir o ser transferido de manera sencilla en dispositivos de almacenamiento portables como memorias flash. [14]

Sin embargo se debe tomar en cuenta que escribir datos a lo largo del tiempo consume más tiempo, el desenvolvimiento de las aplicaciones y porta tanto la experiencia del usuario se verá afectada, especialmente en dispositivos con baja capacidad. [15]

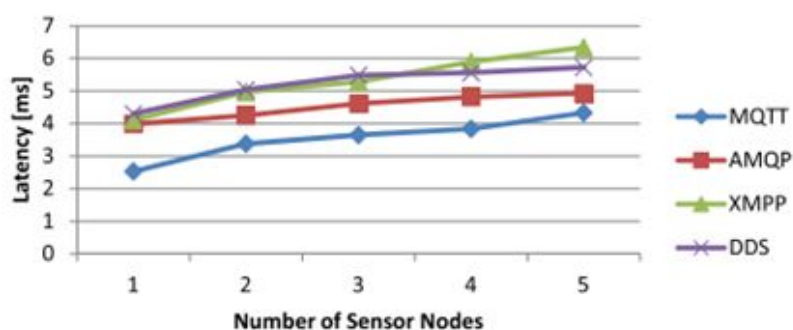
2.1.8. Protocolo de comunicación MQTT

Message Queuing Telemetry Transport, más conocido por su abreviatura MQTT, se basa en la arquitectura publicador/subscriptor con un servidor central llamado bróker que trabaja sobre TCP/IP. Fue creado en el año 1999 por la empresa IBM para lograr una comunicación que requiera un mínimo de ancho de banda, codificación sencilla y bajo consumo de energía. [16] La comunicación se realiza mediante la definición de tópicos a los que llegan los mensajes que son enviados por los publicadores y de estos reciben la información los subscriptores. [17]

MQTT se caracteriza por el uso de tópicos y está principalmente orientado a dispositivos. Es un sistema que permite transmisión de mensajería de manera más rápida que otros protocolos, véase figura 1., lo que lo ha hecho uno de los protocolos más utilizados dentro de proyectos que involucran IoT. [18].

Figura 1: Mediciones de latencia de envío de mensajes

| Protocol | Number of Sensor Nodes | 1 Sensor Node | 2 Sensor Nodes | 3 Sensor Nodes | 4 Sensor Nodes | 5 Sensor Nodes |
|-------------|------------------------------|---------------|----------------|----------------|----------------|----------------|
| | Message Payload Size [chars] | 1475-1477 | 2896-2901 | 4316-4321 | 5734-5746 | 7157-7169 |
| MQTT | <i>Ttransfer</i> [ms] | 2.53 | 3.38 | 3.65 | 3.84 | 4.33 |
| AMQP | <i>Ttransfer</i> [ms] | 3.99 | 4.25 | 4.61 | 4.82 | 4.93 |
| XMPP | <i>Ttransfer</i> [ms] | 4.11 | 4.97 | 5.28 | 5.89 | 6.34 |
| DDS | <i>Ttransfer</i> [ms] | 4.3 | 5.04 | 5.48 | 5.56 | 5.72 |



Fuente: [16]

2.2. SOLUCIÓN DEL PROBLEMA

En una tarjeta microcontroladora Arduino Mega en conjunto con una tarjeta de expansión de ethernet (Ethernet Shield) se implementó un sitio web escrito en C (véase anexo 1) para la visualización de los estados de los sensores y envío de datos para la consolidación de los datos en un middleware escrito en Python (véase anexo 2) y alojado en un computador o servidor.

Para la solución se implementó una red local (base anexo 3) en el que el MCU (Unidad microcontroladora) representa a la Placa Arduino Mega. Los componentes interactúan entre sí como se muestra en el diagrama esquemático (véase anexo 4) y se encuentran conectados a la placa como se muestra en el diagrama de conexión (véase anexo 5).

2.3. RESULTADOS

Como se puede observar en la figura 2, se implementó un sitio web en cual permite visualizar y monitorear los estados de los sensores.

Figura 2: Sitio web



Fuente: El autor

En la figura 3 se observa el middleware realizado en python con la librería Paho:

Figura 3: Middleware

```
C:\Users\USUARIO\Desktop\arduinov4>python sv.py
Creando instancia del cliente
Conectar al broker
Subscribiendose al topico sensores
Estado recibido # T: 24.00
T: 24.00
Estado recibido # D: 122
D: 122
Estado recibido # A: 0
A: 0
Estado recibido # F: Si
F: Si
Estado recibido # C: No
C: No
```

Fuente: El autor

La figura 4 se detalla los datos almacenados en la base de datos realizada en SQLite:

Figura 4: Información consolidad en la base de datos SQLite

| | sensor | tipo | dato | fec |
|----|--------------|-----------------------------|-------|----------------------------|
| 1 | KY015 | Sensor de fuego | 23.00 | 2018-06-10 22:42:51.258704 |
| 2 | HC-SR04 | Sensor de proximidad | 48 | 2018-06-10 22:42:51.378385 |
| 3 | Water Sensor | Sensor de agua | 1 | 2018-06-10 22:42:51.500062 |
| 4 | KY026 | Sensor de fuego | No | 2018-06-10 22:42:51.665617 |
| 5 | KY025 | Sensor de campos magneticos | No | 2018-06-10 22:42:51.789286 |
| 6 | KY015 | Sensor de fuego | 23.00 | 2018-06-10 22:43:01.102383 |
| 7 | HC-SR04 | Sensor de proximidad | 49 | 2018-06-10 22:43:01.429508 |
| 8 | Water Sensor | Sensor de agua | 0 | 2018-06-10 22:43:01.594071 |
| 9 | KY026 | Sensor de fuego | No | 2018-06-10 22:43:01.789546 |
| 10 | KY025 | Sensor de campos magneticos | No | 2018-06-10 22:43:01.946125 |
| 11 | KY015 | Sensor de fuego | 23.00 | 2018-06-10 22:43:11.205369 |
| 12 | HC-SR04 | Sensor de proximidad | 48 | 2018-06-10 22:43:11.335020 |
| 13 | Water Sensor | Sensor de agua | 0 | 2018-06-10 22:43:11.433756 |
| 14 | KY026 | Sensor de fuego | No | 2018-06-10 22:43:11.534487 |
| 15 | KY025 | Sensor de campos magneticos | No | 2018-06-10 22:43:11.634221 |
| 16 | KY015 | Sensor de fuego | 23.00 | 2018-06-10 22:43:21.032092 |
| 17 | HC-SR04 | Sensor de proximidad | 49 | 2018-06-10 22:43:21.201637 |
| 18 | Water Sensor | Sensor de agua | 0 | 2018-06-10 22:43:21.403100 |
| 19 | KY026 | Sensor de fuego | No | 2018-06-10 22:43:21.535745 |
| 20 | KY025 | Sensor de campos magneticos | No | 2018-06-10 22:43:21.703295 |
| 21 | KY015 | Sensor de fuego | 23.00 | 2018-06-10 22:43:31.185970 |
| 22 | HC-SR04 | Sensor de proximidad | 48 | 2018-06-10 22:43:31.291658 |
| 23 | Water Sensor | Sensor de agua | 1 | 2018-06-10 22:43:31.392392 |
| 24 | KY026 | Sensor de fuego | No | 2018-06-10 22:43:31.503094 |
| 25 | KY025 | Sensor de campos magneticos | No | 2018-06-10 22:43:31.604042 |
| 26 | KY015 | Sensor de fuego | 23.00 | 2018-06-10 22:43:41.314102 |
| 27 | HC-SR04 | Sensor de proximidad | 48 | 2018-06-10 22:43:41.537778 |
| 28 | Water Sensor | Sensor de agua | 0 | 2018-06-10 22:43:41.636511 |
| 29 | KY026 | Sensor de fuego | No | 2018-06-10 22:43:41.738244 |

Fuente: El autor

3. CONCLUSIONES

Mediante la realización del proyecto se llegó a la conclusión de que es técnicamente factible la implementación de un servidor web en la placa Arduino mega utilizando el ethernet shield para la visualización de los estados de sensores. Utilizando el protocolo de comunicación MQTT se puede realizar una rápida difusión de información que mediante un middleware escrito en Python con el uso de la librería paho pueden ser leídos, interpretados y almacenados en la base de datos embebida SQLite.

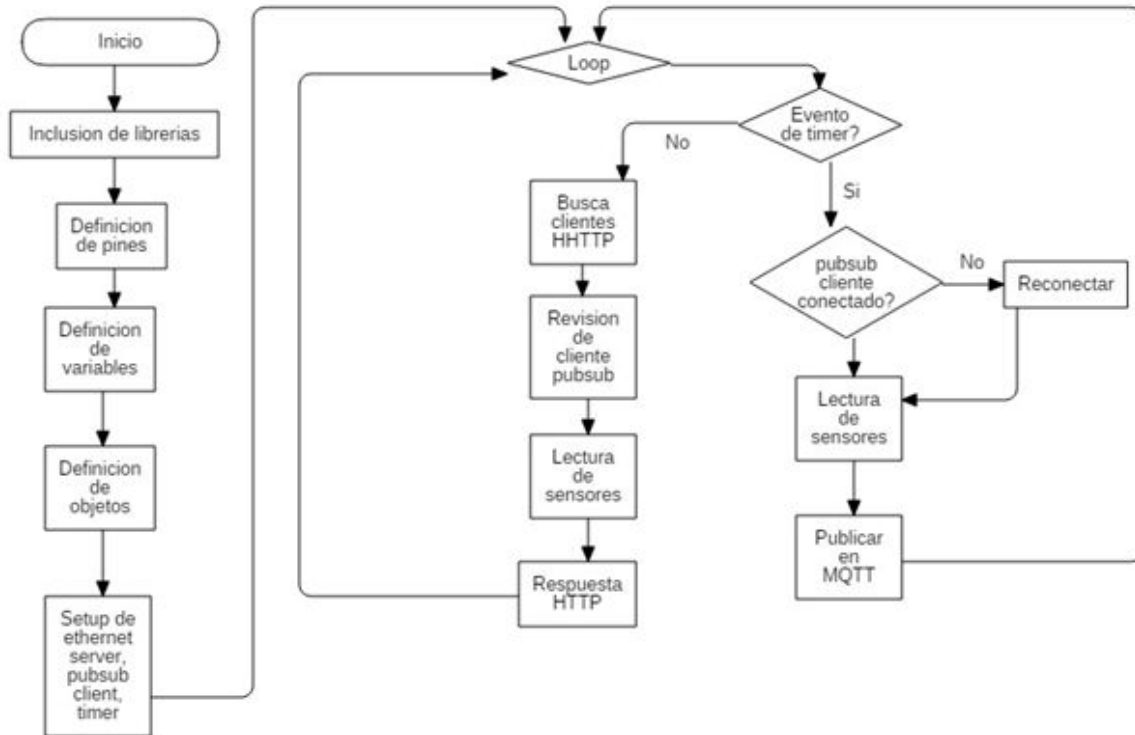
4. REFERENCIAS BIBLIOGRÁFICAS

- [1] A. B. González Rogado, A. M. Vivar Quintana y L. Lavandero Mayo, «Evaluation of the Use of Technology to Improve Safety in the Teaching Laboratory,» *IEEE*, vol. 12, nº 16756767, pp. 17 - 23, 2017.
- [2] T. Adegbija, A. Rogacs, C. Patel y A. Gordon-Ross, «Microprocessor Optimizations for the Internet of Things: A Survey,» *IEEE*, vol. 37, nº 17466887, pp. 7 - 20, 2017.
- [3] J. P. Novillo Vicuña, F. F. Redrován Castillo, F. Espinoza y J. R. Molina Ríos, «Raspberry Analysis in the Teaching of Computer Sciences,» *Research India Publications*, nº 12, pp. 1182-1189, 2017.
- [4] O. Cárdenas Villavicencio, J. Molina Ríos, M. Zea Ordoñez, J. Armijos Carrión y R. Elizalde López, «IMPACTO TECNOLÓGICO DE LOS DISPOSITIVOS INALÁMBRICOS,» *CIEG, REVISTA ARBITRADA DEL CENTRO DE INVESTIGACIÓN Y ESTUDIOS GERENCIALES*, pp. 109-118, 2017.
- [5] M. S. Perez y E. Carrera, «Time synchronization in Arduino-based wireless sensor networks,» *IEEE*, vol. 13, nº 14981660, pp. 455 - 461, 2015.
- [6] C. Severance, «Massimo Banzi: Building Arduino,» *IEEE*, vol. 47, nº 1, pp. 11 - 12, 2014.
- [7] J. Martínez-Santos, S. Contreras-Ortiz y O. Acevedo-Patiño, «Influence of Arduino on the Development of Advanced Microcontrollers Courses,» *IEEE*, vol. 12, nº 14, pp. 208 - 217, 2017.
- [8] Y. Xiong, G. He, X. Li, C. Zhao y H. Xiong, «Improving the serialisation of Trajectory approach for switched Ethernet network,» *IEEE*, vol. 53, nº 13, pp. 845-847, 2017.
- [9] A. Vega, F. Santamaría y E. Rivas, «Internet de los objetos empleando arduino para la gestión eléctrica domiciliaria,» *Revista Escuela de Administración de Negocios*, nº 0120-8160, 2014.
- [10] S. Luiz Stevan, G. N. Dalla Stella y M. M. Dias Santos, «Monitoring of Sleep Apnea Through Thoracic Expansion with Capacitive and Inductive Sensors,» *IEEE*, vol. 14, nº 1, pp. 102 - 108, 2016.
- [11] «A Comprehensive Evaluation of Common Python Implementations,» *IEEE*, vol. 32, nº 15258515, pp. 76 - 84, 2014.
- [12] S. Guelton, «Pythran: Crossing the Python Frontier,» *IEEE*, vol. 20, nº 17630429, pp. 83-89, 2018.
- [13] M. Owens, *The Definitive Guide*, New York: Apress, 2006.

- [14] J. Wang, K.-Y. Lam, Y.-H. Chang, J.-W. Hsieh y P.-C. Huang, «Block-Based Multi-Version B+-Tree for Flash-Based Embedded Database Systems,» *IEEE*, vol. 64, n° 14982411, pp. 925 - 940, 2015.
- [15] S. Ryu, K. Lee y H. Han, «In-memory write-ahead logging for mobile smart devices with NVRAM,» *IEEE*, vol. 61, n° 15000750, pp. 39 - 46, 2015 .
- [16] Z. Babovic, J. Protic y V. Milutinovic, «Web Performance Evaluation for Internet of Things Applications,» *IEEE*, vol. 4, n° 16433307, pp. 6974 - 6992, 2016.
- [17] D. Yacchirema y C. Palau, «Smart IoT Gateway For Heterogeneous Devices Interoperability,» *IEEE*, vol. 14, n° 16561991, pp. 3900 - 3906, 2016.
- [18] Y.-t. Lee, W.-h. Hsiao, C.-m. Huang y S.-c. Chou, «An integrated cloud-based smart home management system with community hierarchy,» *IEEE*, vol. 62, n° 15917898, pp. 1 - 9, 2016.

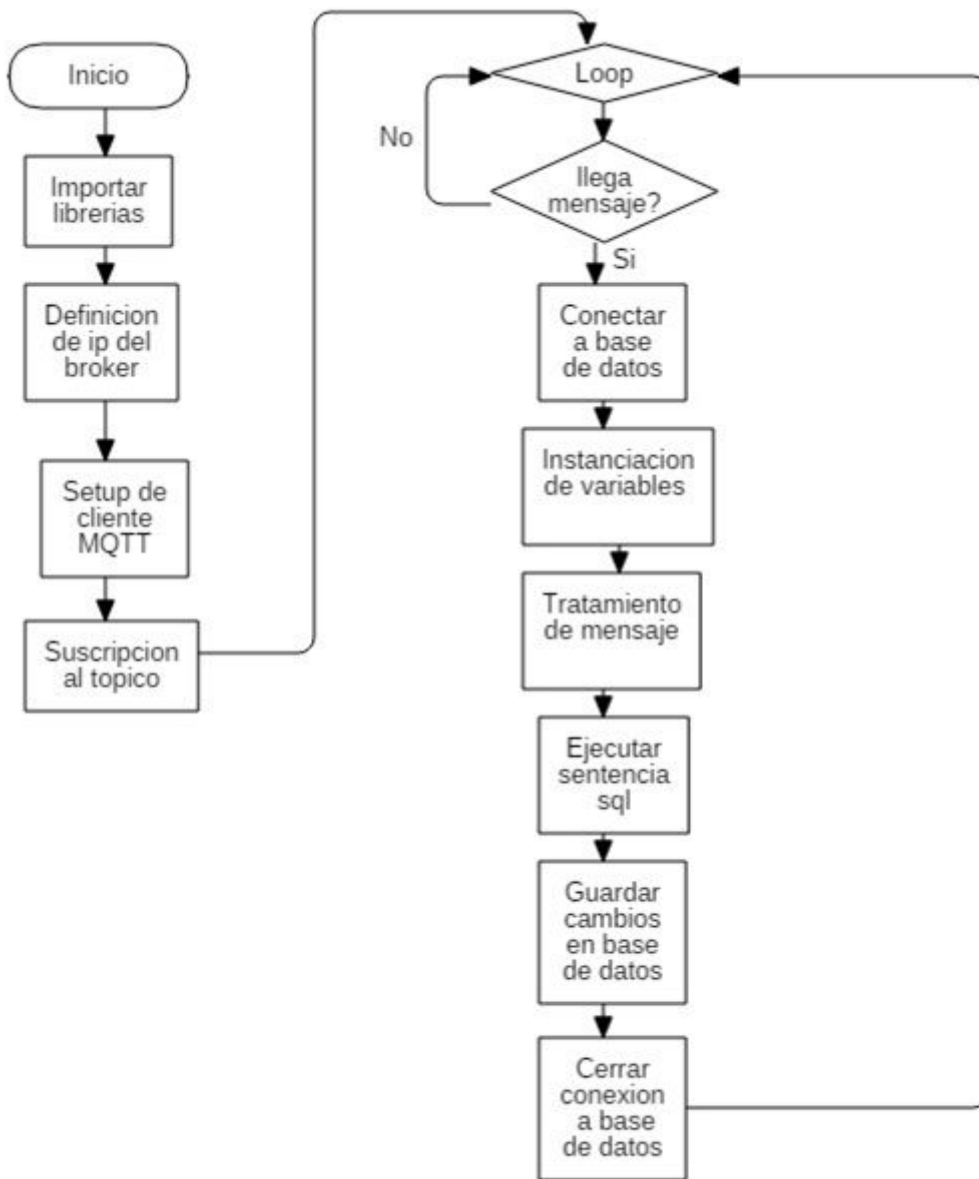
ANEXOS

Anexo A: Diagrama de flujo de Arduino Mega



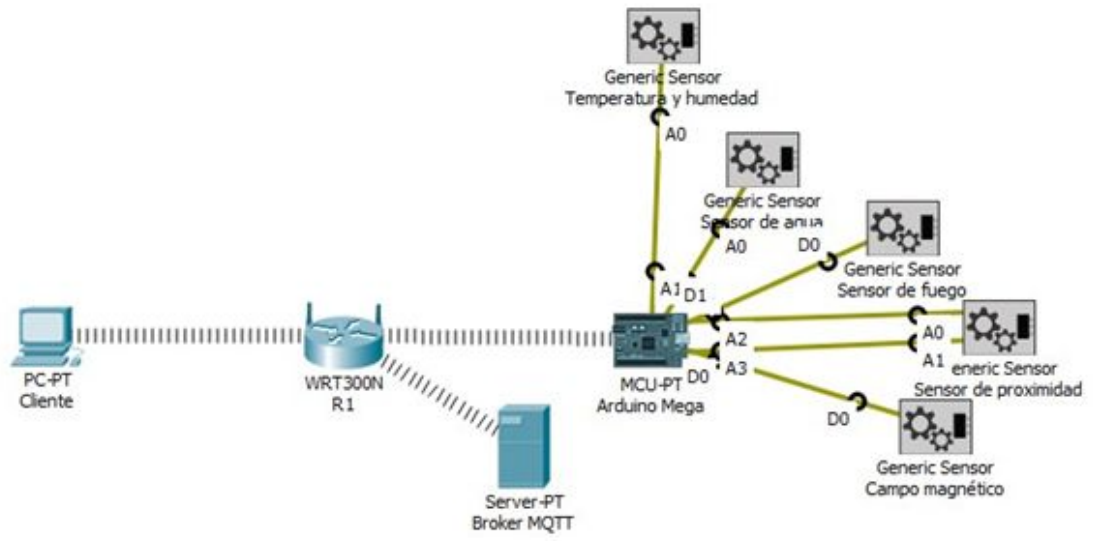
Fuente: El autor

Anexo B: Diagrama de flujo de Middleware



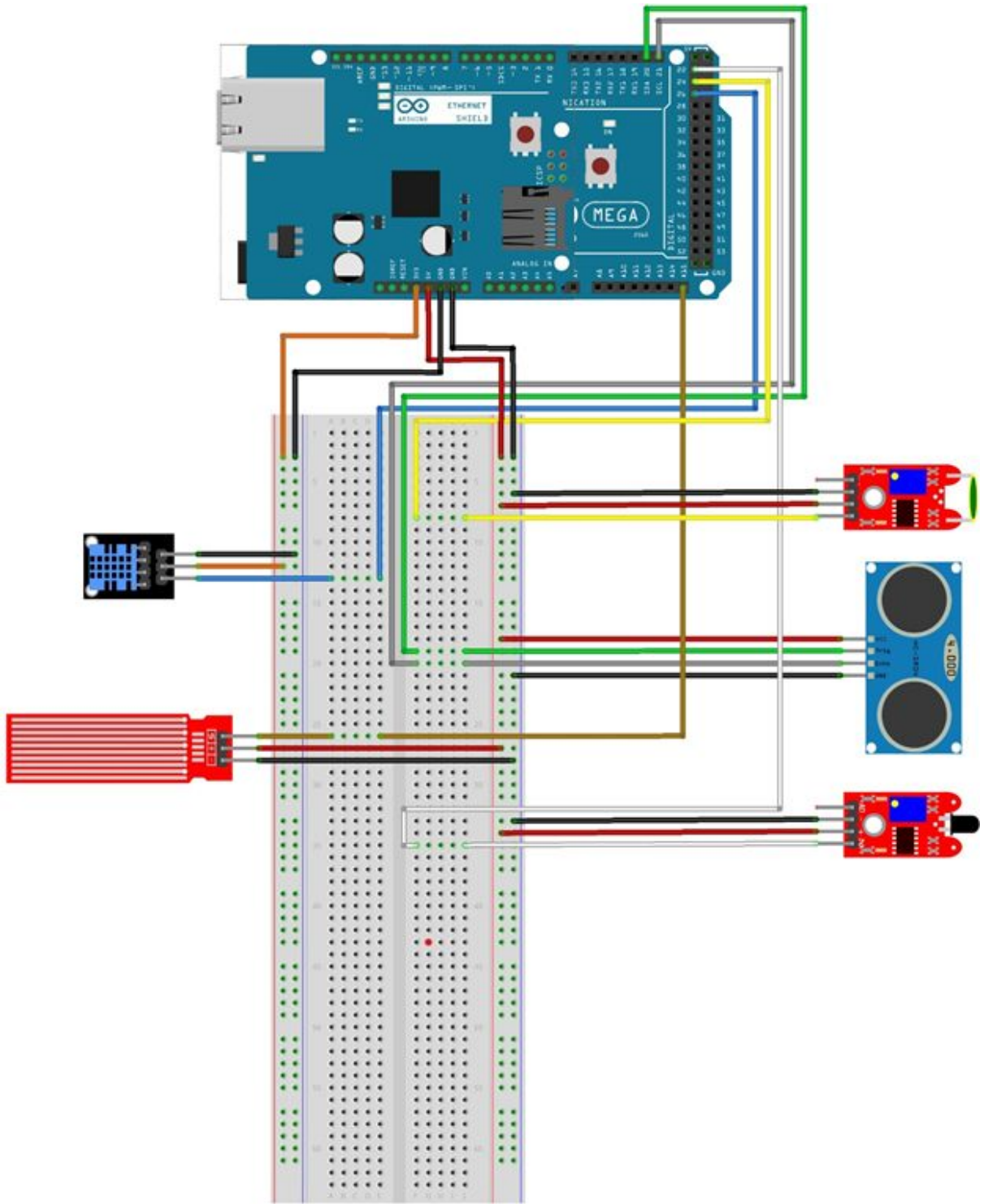
Fuente: El autor

Anexo C: Diagrama de Red



Fuente: El autor

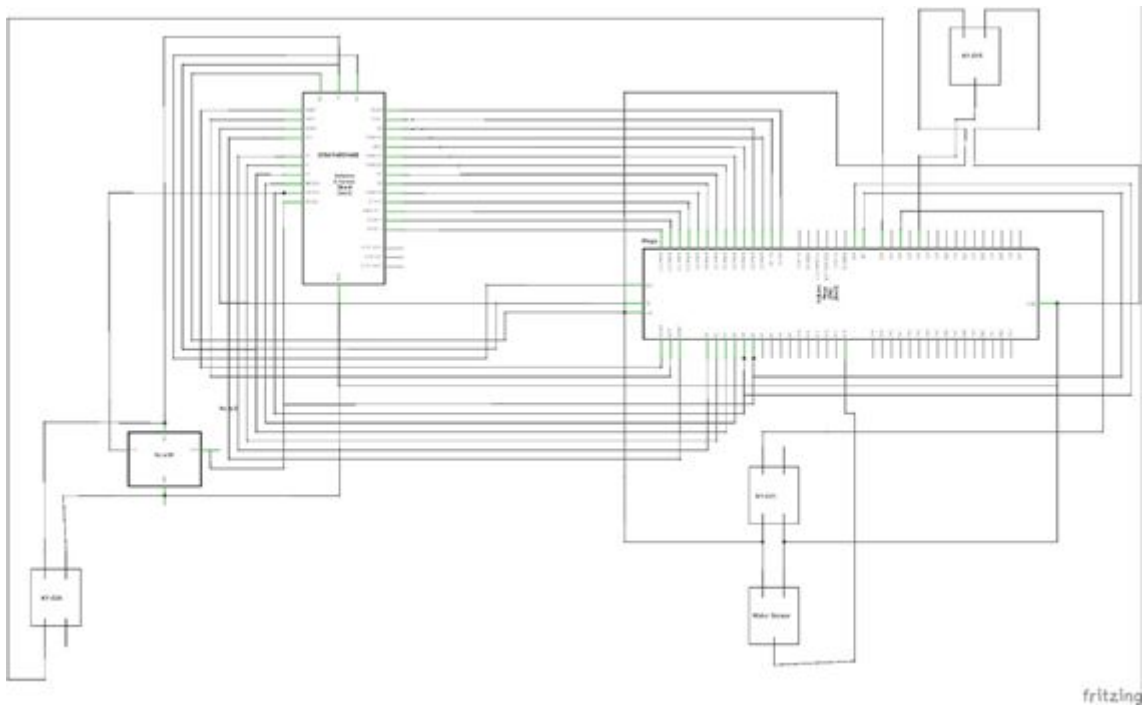
Anexo D: Diagrama de conexión



fritzing

Fuente: El autor

Anexo E: Diagrama Esquemático



Fuente: El autor