



**EDUCACIÓN**

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO

# Tecnológico Nacional de México

Centro Nacional de Investigación  
y Desarrollo Tecnológico

## Tesis de Maestría

Desarrollo de un sistema para la conexión automática de  
sensores de Internet de las Cosas a la plataforma FIWARE

presentada por

**Ing. Juan José Flores Sedano**

como requisito para la obtención del grado de  
**Maestría en Ciencias de la Computación**

Director de tesis

**Dr. Hugo Estrada Esquivel**

Codirector de tesis

**Dr. Alicia Martínez Rebollar**

Cuernavaca, Morelos, México. Julio de 2022.

Cuernavaca, Mor., **22/04/2022**

OFICIO No. DCC/056/2022  
Asunto: Aceptación de documento de tesis  
CENIDET-AC-004-M34-OFCIO

DR. CARLOS MANUEL ASTORGA ZARAGOZA  
SUBDIRECTOR ACADÉMICO  
PRESENTE

Por este conducto, los integrantes de Comité Tutorial del C. JUAN JOSÉ FLORES SEDANO, con número de control M20CE084, de la Maestría en Ciencias de la Computación, le informamos que hemos revisado el trabajo de tesis de grado titulado "DESARROLLO DE UN SISTEMA PARA LA CONEXIÓN AUTOMÁTICA DE SENSORES DE INTERNET DE LAS COSAS A LA PLATAFORMA FIWARE", y hemos encontrado que se han atendido todas las observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

  
DR. HUGO ESTRADA ESQUIVEL  
Director de tesis

  
DRA. ALICIA MARTÍNEZ REBOLLAR  
Codirectora de Tesis

DR. JAVIER ORTIZ HERNÁNDEZ  
Revisor 1

  
DR. NIMRÓD GONZÁLEZ FRANCO  
Revisor 2

\_\_\_\_\_  
Revisor 3

Ccp. Dpto. Servicios Académicos  
Expediente / Estudiante  
JCGS/iam



Walter Herrerías Patrón 634, Car. Reforma, C. P. 62400 Cuernavaca, México  
Tel. 01 (777) 9627770 ext. 8200, e-mail: dca@cenidet.tecnico.mx / dca@cei.unam.mx





Cuernavaca, Mor.,  
No. De Oficio:  
Asunto:

**30/junio/2022**  
**SAC/108/2022**  
**Autorización de**  
**Impresión de tesis**

**JUAN JOSÉ FLORES SEDANO**  
**CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS**  
**DE LA COMPUTACIÓN**  
**P R E S E N T E**

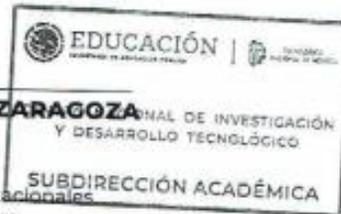
Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado "DESARROLLO DE UN SISTEMA PARA LA CONEXIÓN AUTOMÁTICA DE SENSORES DE INTERNET DE LAS COSAS A LA PLATAFORMA FIWARE", ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

**ATENTAMENTE**  
Excelencia en Educación Tecnológica®  
"Educación Tecnológica al Servicio de México"

**DR. CARLOS MANUEL ASTORGA ZARAGOZA**  
**SUBDIRECTOR ACADÉMICO**

C. c. p. Departamento de Ciencias Computacionales  
Departamento de Servicios Escolares



CMAZ/CHG



## Dedicatoria

Este trabajo lo dedico a mi familia, amigos y profesores que durante toda mi carrera me han apoyado de manera incondicional, jamás dejaron que me diera por vencido y sin ellos este, que hasta ahora es el mayor logro de mi vida no habría sido posible.

En especial va dedicado a la mi madre que durante toda mi vida me apoyó y sin ella no estaría aquí, a mis hermanas que en momentos de estrés y desesperación me brindaron todo su apoyo, a mi hermano que siempre esta cuando más lo necesito, también quiero dedicar este logro a mi segunda madre mi abuela quien ha estado en los momentos más difíciles de mi vida y a pesar de todo sigue conmigo, por último quiero hacer una dedicatoria a mi padre que desde el cielo está mirando mi más grande logro, tengo la esperanza de que el este feliz y orgulloso de mi.



## Agradecimientos

Quiero agradecer a Dios por darme la oportunidad de vivir esta maravillosa experiencia, agradecer a los profesores que siempre estuvieron para darme consejos y ánimos durante toda la carrera, en especial al Dr. Hugo Estrada Esquivel y a la Dra. Alicia Martínez Rebollar quien durante los últimos 2 años de mi carrera me han apoyado y brindado la oportunidad de trabajar con ellos y experimentar el mundo de la investigación, si no fuera por ellos no me hubiera animado a estudiar un posgrado. También quiero agradecer a mis amigos Manuel Erazo, Daza, y Juan Antonio, quienes hicieron de la carrera la más divertida e interesante, con quienes pasé los mejores momentos de mi vida, agradezco al CONACyT por la beca otorgada y al TecNM por facilitar sus instalaciones. Por último, pero no menos importante quiero agradecer a mi familia quienes en todo momento estuvieron conmigo apoyándome y dándome ánimos, mil, mil gracias a todos ustedes por estar conmigo.

## Resumen

Hoy en día existen varias plataformas que permiten el desarrollo de aplicaciones inteligentes con enfoque de Internet de las Cosas. Estas plataformas procesan la información de contexto que producen las aplicaciones inteligentes a través de diferentes dispositivos y sensores conectados a Internet. FIWARE es una plataforma de código abierto que impulsa la creación de estándares para el desarrollo de aplicaciones y servicios inteligentes de diferentes dominios, en la nube. Sin embargo, uno de los principales problemas en el desarrollo de aplicaciones inteligentes con estas plataformas ha sido la complejidad asociada a la conexión entre hardware y software. Este documento presenta un nuevo enfoque para llevar a cabo la creación automática de componentes de software que gestionan la comunicación entre los sensores y la Plataforma FIWARE. El enfoque propuesto se basa en los mecanismos de agente IoT propuestos por la plataforma de FIWARE como intermediario entre los dispositivos y una plataforma IoT. El documento presenta un caso de estudio de la utilización del sistema de software que implementa el enfoque propuesto para producir automáticamente el código de los módulos que administran la conexión entre un dispositivo IoT y la plataforma FIWARE.

## Abstract

Today there are several platforms that allow the development of smart applications with the Internet of Things approach. These platforms process the context information produced by smart applications through different devices and sensors connected to the Internet. FIWARE is an open-source platform that impulses the standards creation for the development of smart applications and services of different domains, in the cloud. However, one of the main issues in the development of smart applications with these platforms has been the complexity associated with the connection between hardware and software. This document presents a new approach for carrying on the automatic creation of software components that manage the communication between sensors and the FIWARE Platform. The proposed approach is based on the IoT agent mechanisms proposed by FIWARE Platform as an intermediary between devices and an IoT Platform. The paper also presents some examples of the software system that implements the proposed approach to automatically produce the code of the modules to manage the connection between sensor and FIWARE IoT Platform.

<b>CAPÍTULO 1 INTRODUCCIÓN</b>	<b>1</b>
<b>1.1 INTRODUCCIÓN</b>	<b>2</b>
<b>1.2 PLANTEAMIENTO DEL PROBLEMA</b>	<b>2</b>
<b>1.3 OBJETIVO</b>	<b>5</b>
1.3.1 OBJETIVOS ESPECÍFICOS	5
1.3.2 ALCANCES	6
1.3.3 LIMITACIONES	6
<b>1.4 ESTRUCTURA DEL DOCUMENTO</b>	<b>7</b>
<b>CAPÍTULO 2</b>	<b>8</b>
<b>MARCO TEÓRICO</b>	<b>8</b>
<b>2.1 INTERNET DE LAS COSAS EN LA NUBE</b>	<b>9</b>
<b>2.2 PLATAFORMAS DEL INTERNET DE LAS COSAS</b>	<b>9</b>
<b>2.3 PLATAFORMA FIWARE</b>	<b>9</b>
<b>2.4 CONECTIVIDAD CON LA PLATAFORMA FIWARE</b>	<b>9</b>
<b>2.5 ORION CONTEXT BROKER COMO MANEJADOR DE DATOS DE CONTEXTO DE FIWARE</b>	<b>10</b>
<b>2.6 MODELO NGSII PARA REPRESENTAR ENTIDADES EN LA PLATAFORMA FIWARE</b>	<b>10</b>
<b>2.7 COMPONENTE GRAFANA PARA GRAFICAR DATOS</b>	<b>10</b>
<b>2.8 AGENTE DEL INTERNET DE LAS COSAS EN FIWARE</b>	<b>11</b>
<b>2.9 PROTOCOLOS DE COMUNICACIÓN DE DISPOSITIVOS IOT</b>	<b>11</b>
<b>CAPÍTULO 3 ESTADO DEL ARTE</b>	<b>13</b>
<b>3.1 TRABAJOS RELACIONADOS CON LA SEGURIDAD DE LOS DATOS EN DISPOSITIVOS IOT</b>	<b>14</b>
3.1.1 MEJORA DE LA SEGURIDAD EN APLICACIONES DEL IOT BASADAS EN LA PLATAFORMA FIWARE	14
3.1.2 SEGURIDAD EN FLUJO PARA DISPOSITIVOS IOT UTILIZANDO UNA PUERTA DE ENLACE SDN	15
<b>3.2 TRABAJOS RELACIONADOS AL USO DE AGENTES PARA EL IOT</b>	<b>16</b>
3.2.1 UN MODELO DE CONFIANZA DE AGENTES AUTÓNOMO PARA SISTEMAS DE IOT	16
3.2.2 ARQUITECTURA DE APROVISIONAMIENTO DE SERVICIOS RESILIENTE Y AUTÓMATA BASADO EN MÚLTIPLES AGENTES PARA EL INTERNET DE LAS COSAS	17
<b>3.3 TRABAJOS RELACIONADOS A LA CONECTIVIDAD ENTRE DISPOSITIVOS Y LA NUBE</b>	<b>18</b>
3.3.1 HABILITACIÓN DE LA CONECTIVIDAD DEL IOT PARA SENSORES MODBUS TCP	18
3.3.2 PLATAFORMA SEGURA DE AGENTES IOT CON MECANISMO DE CÁLCULO ESTADÍSTICOS DISTRIBUIDO M-CLOUD	19
<b>CAPÍTULO 4 SISTEMA PARA LA CONEXIÓN AUTOMÁTICA DE DISPOSITIVOS IOT A FIWARE</b>	<b>21</b>
<b>4.1 ARQUITECTURA DEL SISTEMA</b>	<b>22</b>
4.1.1 CAPA DE APLICACIÓN	23
4.1.2 CAPA DE SERVICIOS	23
4.1.3 ECOSISTEMA FIWARE	23

<b>4.2</b>	<b>IMPLEMENTACIÓN DE LA ARQUITECTURA PROPUESTA</b>	<b>24</b>
4.2.1	MÓDULO GESTOR	25
4.2.2	MÓDULO GENERADOR DE CONEXIÓN DE FIWARE	26
4.2.3	MÓDULO RECEPTOR	31
<b>4.3</b>	<b>SISTEMA WEB PARA LA CONEXIÓN ENTRE DISPOSITIVOS IOT A LA PLATAFORMA FIWARE</b>	<b>34</b>

---

**CAPÍTULO 5 PRUEBAS Y RESULTADOS** **41**

<b>5.1</b>	<b>OBJETIVO DE LA EXPERIMENTACIÓN</b>	<b>42</b>
<b>5.2</b>	<b>PARTICIPANTES</b>	<b>43</b>
<b>5.3</b>	<b>RECURSOS UTILIZADOS</b>	<b>45</b>
<b>5.4</b>	<b>PROCEDIMIENTO DE LA EXPERIMENTACIÓN</b>	<b>47</b>
<b>5.5</b>	<b>RESULTADOS DE LA EVALUACIÓN PRELIMINAR DE LA FACILIDAD DE USO DE CADA UNA DE LAS FUNCIONALIDADES DEL SISTEMA</b>	<b>50</b>
<b>5.6</b>	<b>RESULTADOS DE LA EVALUACIÓN DEL TIEMPO UTILIZADO PARA CADA UNA DE LAS TAREAS</b>	<b>51</b>
<b>5.7</b>	<b>RESULTADOS DEL GRUPO C PARTICIPANTES AVANZADOS</b>	<b>52</b>
<b>5.8</b>	<b>RESULTADOS DEL GRUPO B PARTICIPANTES CON EXPERIENCIA MEDIA</b>	<b>54</b>
<b>5.9</b>	<b>RESULTADOS DEL GRUPO A PARTICIPANTES SIN EXPERIENCIA</b>	<b>56</b>
<b>5.10</b>	<b>RESULTADOS DE LA EVALUACIÓN DE LA INTERFAZ GRÁFICA</b>	<b>58</b>

---

**CAPÍTULO 6 CONCLUSIONES Y TRABAJO A FUTURO** **61**

<b>6.1</b>	<b>CONCLUSIONES</b>	<b>62</b>
<b>6.2</b>	<b>TRABAJO A FUTURO</b>	<b>62</b>
	<b>REFERENCIA</b>	<b>64</b>
	<b>ANEXO A</b>	<b>66</b>
	<b>ANEXO B</b>	<b>67</b>
	<b>ANEXO C</b>	<b>68</b>

Figura 1-1.	Arquitectura de los agentes del Internet de las Cosas y el Orion Context Broker[5].....	4
Figura 2-1.	Interacción entre el Orion Context Broker, agentes IoT y protocolos de comunicación [12] .....	11
Figura 3-1.	Arquitectura de seguridad en aplicación IoT propuesta en [16] .....	15
Figura 3-2.	Arquitectura de seguridad para dispositivos utilizando SDN propuesta [17].....	16
Figura 3-3.	Arquitectura de un modelo de confianza en sistemas IoT propuesta en [18] .....	17
Figura 3-4.	Arquitectura de aprovisionamiento de servicios basada en múltiples agentes propuesta en [19] .....	18
Figura 3-5.	Arquitectura de conectividad para sensores Modbus TCP propuesta en [20] .....	19
Figura 3-6.	Arquitectura de una plataforma segura de agentes IoT propuesta en [21] .....	20
Figura 4-1.	Arquitectura del sistema propuesto. ....	23
Figura 4-2.	Arquitectura específica del sistema propuesto .....	25
Figura 4-3	Modelo relacional de la base de datos del sistema.....	26
Figura 4-4	Modelo de datos de una entidad.....	27
Figura 4-5.	Entidad NGSI de un dispositivo IoT .....	28
Figura 4-6	Interacción entre agente IoT y el Orion Context Broker .....	29
Figura 4-7	Código para levantar servicios de FIWARE.....	30
Figura 4-8	Confirmación de un agente IoT para protocolo Ultralight .....	31
Figura 4-9	Diagrama transformación a formato NGSI.....	32
Figura 4-10	Suscripción de la entidad NGSI .....	33
Figura 4-11	Diagrama de interacción de la suscripción NGSI .....	33



Figura 4-12 Diagrama de interacción con la nube de FIWARE .....	34
Figura 4-13 Dispositivo Arduino D1 con sensor Dht11 .....	35
Figura 4-14 Formulario de registro de usuario.....	35
Figura 4-15 formulario del registro del dispositivo.....	36
Figura 4--16 Código Arduino.....	36
Figura 4-17 Tabla de datos recibidos por el sistema .....	37
Figura 4-18 Suscripción de la entidad NGSi del dispositivo .....	38
Figura 4-19 Tabla en CrateDB .....	38
Figura 4-20 Código de conexión a FIWARE.....	39
Figura 4-21 Módulo de Grafana.....	40
Figura 5-1 Gráfica del tiempo esperado de cada tarea .....	51
Figura 5-2 Gráfica del tiempo promedio de cada tarea .....	52
Figura 5-3 Tiempo promedio del grupo C .....	52
Figura 5-4 Gráfica del tiempo del participante 2 .....	53
Figura 5-5 Gráfica del tiempo del participante 3 .....	53
Figura 5-6 Grafica del tiempo del participante 4 .....	54
Figura 5-7 Gráfica del tiempo promedio del grupo b .....	54
Figura 5-8 Gráfica del tiempo del participante 1 .....	55
Figura 5-9 Gráfica del tiempo del participante 6 .....	55
Figura 5-10 Gráfica del tiempo del participante 8.....	56
Figura 5-11 Gráfica del tiempo promedio del grupo a .....	57
Figura 5-12 Gráfica del tiempo del participante 5.....	57
Figura 5-13 Gráfica del tiempo del participante 7.....	58
Figura 5-14 Resultados del cuestionario de usabilidad del grupo c .....	59
Figura 5-15 Resultados del cuestionario de usabilidad del grupo b.....	59
Figura 5-16 Resultados del cuestionario de usabilidad del grupo a .....	60
Figura 5-17 Resultados del cuestionario de usabilidad .....	60
Tabla 5-1 Participantes .....	43
Tabla 5-2 Grupo a de participantes.....	44
Tabla 5-3 Grupo b de participantes .....	44
Tabla 5-4 Grupo c de los participantes .....	44
Tabla 5-5 Grupos de participantes.....	44
Tabla 5-6 Participantes grupo c y sensores.....	53
Tabla 5-7 Participantes grupo b y sensores.....	55
Tabla 5-8 Participantes gripo a y sensores.....	57

# Capítulo 1

# Introducción

---

En este capítulo se presenta la problemática que se intenta resolver, el objetivo general y objetivos específicos de la tesis, así como los alcances y límites del sistema de software propuesto en este proyecto de investigación. Esta sección incluye la forma en la cual está estructurado el resto del documento de tesis.

## 1.1 Introducción

El Internet de las Cosas (IoT por sus siglas en inglés), es una red de dispositivos que están embebidas con software, sensores, y conectividad en la red que recolectan e intercambian datos. El Internet de las Cosas permite que los objetos cotidianos sean “inteligentes” al permitir la transmisión de datos y automatiza tareas, sin requerir ninguna intervención manual. IoT ofrece diferentes oportunidades y mejoras en la vida de las personas, siendo las ciudades inteligentes un concepto que permite unir múltiples tecnologías de este paradigma. Las ciudades inteligentes proporcionan diversos servicios que permiten sensorizar y automatizar diferentes aspectos de la ciudad como es el caso del uso de energía eléctrica, el manejo de basura, la administración del agua de la ciudad, la contaminación en las diferentes zonas de la ciudad, el ruido en las calles, etc.

La proliferación de servicios inteligentes basados en IoT se debe en gran medida al abaratamiento en el precio de los sensores y de los servicios de comunicación. Este abaratamiento de los dispositivos electrónicos ha popularizado la creación de aplicaciones que utilicen datos de sensores para conocer el contexto en el cual opera un objeto y el desarrollo de aplicaciones que capturen y analicen estos datos de contexto para la toma de decisiones.

El desarrollo de aplicaciones IoT implica una gran responsabilidad en el desarrollo de software, debido a que los objetos necesitan estar en constante comunicación con las plataformas IoT, lo que implica que los objetos permanezcan conectados todo el tiempo a internet. Esta situación provoca que la información producida por los sensores quede expuesta a amenazas informáticas, las cuales deben ser atendidas de manera inmediata para así garantizar la integridad de los sistemas. Como consecuencia de esta situación, se crearon los agentes inteligentes como un intermediario entre la información proveniente de los sensores y las plataformas software y para gestionar además los aspectos de seguridad producida.

## 1.2 Planteamiento del problema

En la actualidad, la complejidad de las soluciones en Internet de las Cosas ha requerido de plataformas que ayuden a administrar esta complejidad en dispositivos físicos y sistemas software. Las aplicaciones enfocadas al IoT implementan las tecnologías del Internet del Futuro [1] las cuales utilizan plataformas tales como: Google Cloud Platform [2], Amazon Web Services IoT [3] o FIWARE [4], estas plataformas también gestionan la parte de la seguridad e integridad de los datos que reciben de los dispositivos mediante sus habilitadores genéricos. La Plataforma FIWARE provee de componentes para el envío de datos de los sensores a la nube de datos FIWARE para que puedan estar disponibles en forma abierta. Un componente

clave de FIWARE es el Orion Context Broker el cual permite la publicación de información de contexto por entidades. El Orion Context Broker es un servidor que implementa una interfaz de programación de aplicaciones, la cual está basada en un modelo de información NGSI, por la cual se pueden realizar varias operaciones como:

- Registrar aplicaciones de contexto
- Actualizar información de contexto.
- Ser notificado cuando surjan cambios en la información del contexto.
- Consultar información de contexto.

El componente Orion Context Broker gestiona la información producida por los dispositivos IoT, por lo que las consultas para conocer el estado actual de los valores de un dispositivo deben realizarse a este componente de la plataforma FIWARE. El componente Orion Context Broker permite la publicación de los datos de contexto más recientes provenientes de los dispositivos. Con la finalidad de registrar el histórico de los datos se incorpora una base de datos en MongoDB que permite acceder a los datos producidos en el pasado por los sensores del dispositivo IoT.

En el caso de la Plataforma FIWARE se cuenta con agentes de IoT que permiten gestionar grupos de dispositivos IoT e integrar los datos que recopilan en un flujo combinado y enviarlos de vuelta a las aplicaciones IoT. La figura 1-1 muestra la arquitectura entre los agentes IoT y el Orion Context Broker [5].

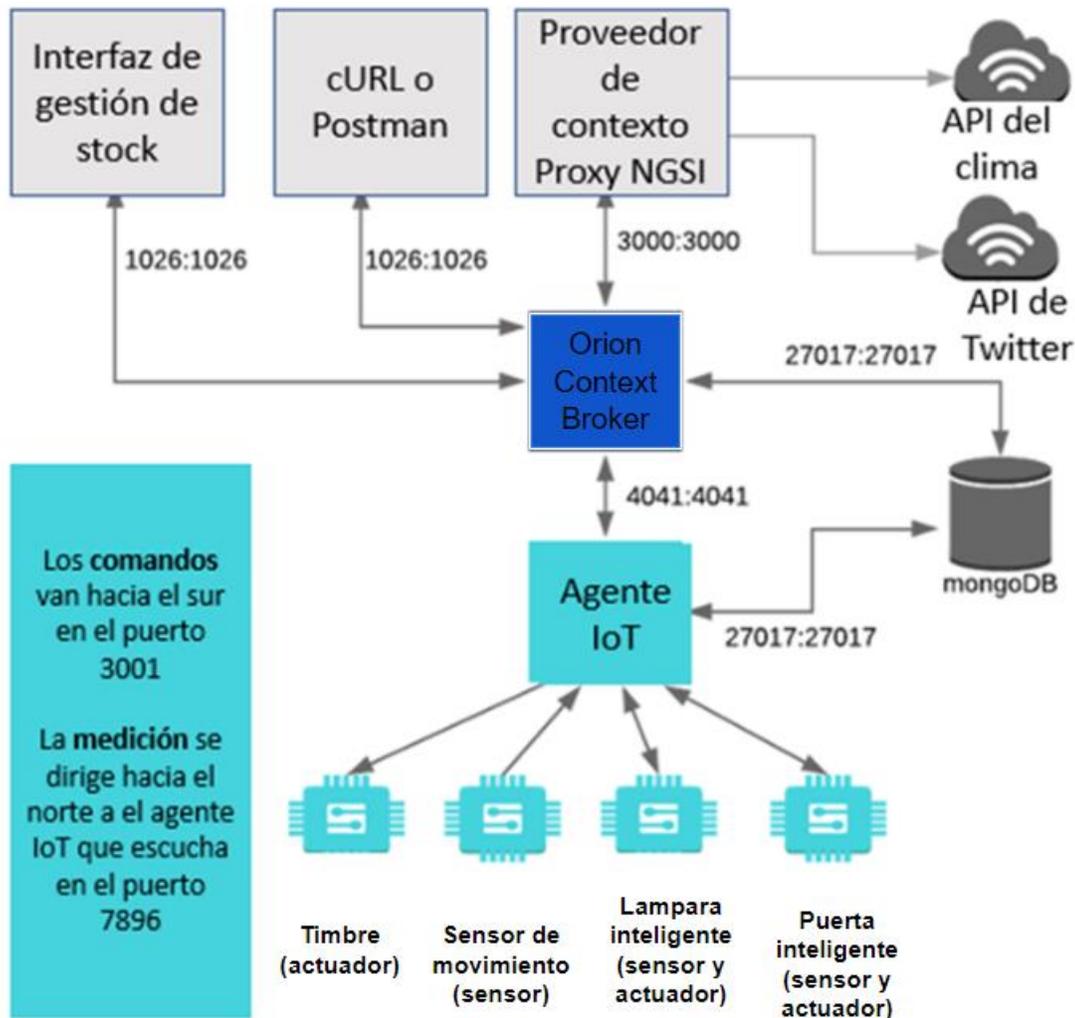


Figura 1-1. Arquitectura de los agentes del Internet de las Cosas y el Orion Context Broker[5]

La conexión entre los dispositivos físicos y la plataforma FIWARE debe realizarse utilizando los agentes IoT. Esta conexión que actualmente se realiza de forma manual implica la configuración de los dispositivos físicos para poder enlazarlos con el agente IoT adecuado. Este agente IoT requiere ser configurado para cada protocolo de comunicación, lo que implica conocer a fondo el funcionamiento del protocolo y el agente en cuestión. En caso de que se use un protocolo de comunicación que aún no cuente con un estándar para el agente IoT, se requiere crear un agente IoT específico, lo que implica codificar los comandos y atributos necesarios para que el agente IoT funcione de manera correcta. Con el agente IoT adecuado se puede realizar la conexión con el Orion Context Broker. Esta conexión implica el conocimiento del dispositivo físico, el conocimiento de los agentes IoT y del Orion Context Broker.

La dificultad de realizar esta conexión es la que ha originado que en la actualidad la conexión se realice en forma directa entre los dispositivos y el Orion Context Broker, lo cual rompe con el esquema ideal y permite el acceso no autorizado a los datos leídos de los sensores. Por lo cual la seguridad es una parte fundamental del desarrollo de aplicaciones

IoT, así como la conectividad entre los dispositivos físicos y la plataforma FIWARE, haciendo uso de los protocolos y los agentes del Internet de las Cosas asegurando el correcto funcionamiento de la recepción y envío de datos.

En el trabajo “Una revisión concisa sobre Internet de las Cosas (IoT): problemas, desafíos y oportunidades” [6] se describen los problemas, retos y oportunidades del Internet de las Cosas, siendo la seguridad una de estas. Los autores del trabajo explican que, actualmente, la integración de la aplicación en la infraestructura de red se centra solo en lograr la funcionalidad en lugar de considerar de manera integral los requisitos de seguridad cuando la aplicación está siendo diseñada. Esto deja la puerta abierta a ataques e intentos de pirateo por ejemplo robo de datos, ataque de denegación de servicio, secuestro de datos para hogares o coches inteligentes, etc. Los expertos en ciberseguridad han advertido que IoT es una de las tecnologías más vulnerables y que esperan ataques más dirigidos a las infraestructuras existentes y emergentes.

Actualmente, ante la proliferación de aplicaciones de IoT, se requiere de nuevos enfoques que nos permitan reducir la complejidad de la creación de aplicaciones IoT que utilicen Agentes IoT como intermediarios entre los dispositivos IoT y plataformas IoT.

El problema de realizar una conexión manual entre los dispositivos y la plataforma FIWARE es la dificultad y el tiempo necesario para conocer el dispositivo, sus protocolos de comunicación, los agentes que se enlazan con los protocolos y el Orion Context Broker, además de entender el modelo de datos de FIWARE para hacer uso de las entidades y suscripciones NGSI.

Con esto en mente el objetivo de reducir la complejidad del proceso de mapeo entre dispositivos y la plataforma FIWARE se requiere que las actividades de configuración de agentes puedan realizarse de forma automática y que pueda automatizarse también el proceso de comunicación con una plataforma IoT.

## 1.3 Objetivo general

Desarrollar un sistema de software para la conexión automática de sensores IoT a la plataforma FIWARE. El sistema a desarrollar es capaz de gestionar los dispositivos, seleccionar y modificar los agentes IoT de acuerdo con el protocolo del dispositivo, ya sea COAP o MQTT. Cada aplicación de IoT considera dispositivos diferentes, por lo que se propuso generar de manera automática el módulo específico para cada aplicación de IoT que necesite enviar datos de sensores utilizando la plataforma FIWARE.

### 1.3.1 Objetivos específicos

- Desarrollar un módulo para la gestión de usuarios y dispositivos, con la finalidad de administrar la información de los usuarios y dispositivos, para su uso en el módulo de conexión.
- Desarrollar un módulo que genere el código de conexión a FIWARE con la configuración necesaria para levantar los servicios de FIWARE y la configuración del agente IoT.

- Desarrollar un módulo que genere el código Arduino, que se compila en el dispositivo IoT para el envío de datos al sistema web.
- Desarrollar un módulo que permita recibir y almacenar los datos enviados por el dispositivo.
- Desarrollar un submódulo que genere las entidades NGSI para el envío de datos.

### 1.3.2 Alcances

- El sistema cuenta con un módulo que permite registrar dispositivos IoT considerando los diversos sensores del dispositivo y el tipo de lecturas de cada uno de estos sensores. Este módulo permite definir el protocolo de comunicación del dispositivo que está siendo registrado.
- El sistema define el agente IoT utilizando los protocolos de comunicación de cada sensor registrado en el sistema.
- El sistema modifica los agentes IoT seleccionando cada sensor registrado asignando la información del sensor y configurando la dirección del Orion Context Broker.
- El sistema genera el código para la conexión de sensores a la plataforma FIWARE, incorporando las ligas para la conexión a la plataforma FIWARE e incorporando el código de la interfaz del módulo. Este código es compilado por el usuario para generar el módulo que gestiona la comunicación entre el sensor y FIWARE.

### 1.3.3 Limitaciones

- El límite de sensores que puede tener el dispositivo IoT depende del tipo de placa base en el cual se coloquen los sensores (Raspberry, Arduino, Cloudino, etc.).
- La solución propuesta en esta tesis solo considera los protocolos de comunicación COAP, MQTT.

## 1.4 Estructura del documento

La tesis está organizada con los siguientes capítulos.

- Capítulo 2 Marco teórico

Este capítulo provee de los conceptos principales que componen esta tesis, como el Internet de las Cosas, los dispositivos del Internet de las Cosas, protocolos, agentes y las plataformas del Internet de las Cosas, con el propósito de definir los conceptos utilizados durante la tesis.

- Capítulo 3 Estado del arte

En este capítulo se provee de una revisión al estado del arte sobre algunos de los temas más relevantes de esta tesis. La conexión de agentes del Internet de las Cosas, la seguridad e integridad de los datos de los dispositivos y la conectividad entre dispositivo y la nube.

- Capítulo 4 Sistema para la conexión entre dispositivos IoT a FIWARE.

En este capítulo muestra los módulos que conforman al sistema desarrollado. El módulo gestor, módulo de conexión y módulo receptor de información. Mostrando la funcionalidad del sistema para realizar la conexión y envío de datos a la plataforma FIWARE.

- Capítulo 5 Pruebas y resultados.

En este capítulo se detalla la importancia de la experimentación para evaluar el funcionamiento del sistema desarrollado, así como del procedimiento para realizar las pruebas. La selección de los participantes, la ejecución de las tareas, formularios y la presentación de pruebas y resultados.

- Capítulo 6 Conclusiones y trabajos a futuros.

En este capítulo se habla sobre las conclusiones del documento, las cuales incluyen el análisis de resultado y cómo se obtuvieron, así como los trabajos a futuro.

# Capítulo 2

## Marco teórico

---

## 2.1 Internet de las Cosas en la nube

El constante crecimiento en el número de “cosas ” conectadas a una red, ha generado una gran cantidad de soluciones que fueron necesarias, siendo el desarrollo de aplicaciones enfocadas al Internet de las Cosas un parte crucial, para la recopilación y análisis de los datos de los dispositivos, es así como el Internet de las Cosas en la nube nace, dado que admite dispositivos y aplicaciones de IoT, la cual incluye los servicios y estándares necesarios para conectar, administrar y proteger diferentes dispositivos y aplicaciones de IoT. Estos dispositivos pueden ser sensores, actuadores, dispositivos móviles, etc. Todos los datos recopilados por estos dispositivos son almacenados en la nube, para así tener un fácil acceso a ellos, además de proporcionar una mayor seguridad en caso de un percance [7].

## 2.2 Plataformas del Internet de las Cosas

Con la llegada del Internet de las Cosas fue necesario desarrollar soluciones utilizadas para administrar todas las interacciones entre el hardware y las capas de aplicación, soluciones que fueran capaz de administrar la enorme cantidad de datos que los dispositivos inteligentes recolectaban, naciendo así las plataformas del Internet de las Cosas, hoy en día hay una gran cantidad de plataformas IoT, entre las cuales en se encuentran las plataformas de Amazon, Google, Azure de Microsoft, IBM y FIWARE [8].

## 2.3 Plataforma FIWARE

Es una de las plataformas del Internet de las Cosas emergentes para el desarrollo y despliegue de aplicaciones de Internet del Futuro. FIWARE intenta proveer de una arquitectura totalmente abierta la cual permita a los desarrolladores, proveedores de servicios, empresas y otras organizaciones desarrollar productos que satisfagan sus necesidades. FIWARE comprende un conjunto de tecnologías llamados habilitadores genéricos los cuales proveen interfaces abiertas para las API y soportar interoperabilidad con otros habilitadores genéricos; estos habilitadores como el Orion Context Broker, QuantumLeap surgen como una propuesta para dar respuesta a la necesidad de enfoques a las ciudades inteligentes [8]

## 2.4 Conectividad con la plataforma FIWARE

Uno de los retos actuales en el Internet de las Cosas es la conexión actual entre los dispositivos IoT y la plataforma FIWARE, esta conexión actualmente se realiza de forma manual lo cual implica la configuración de los dispositivos físicos para poder enlazarlos con el IoT Agente adecuado. Este agente IoT requiere ser configurado para cada protocolo de munición, lo que implica conocer a fondo el funcionamiento del protocolo y el agente en cuestión. En caso de que se use un protocolo de comunicación que aún no cuente con un estándar para el agente IoT, se requiere crear un agente IoT específico, lo que implica codificar los comandos y atributos necesarios para que el agente IoT funcione de manera correcta. Con el agente IoT adecuado se puede realizar la conexión con el Orion Context Broker. Lo cual requiere conocer la conexión y configuración de los dispositivos físicos, conocer la

configuración de los agentes IoT y del Orion Context Broker. La dificultad de realizar esta conexión es la que ha originado que en la actualidad la conexión se realice de forma directa entre los dispositivos y el Orion Context Broker, lo cual rompe con el esquema ideal y permite el acceso no autorizado a los datos leídos de los sensores [9].

## 2.5 Orion Context Broker como manejador de datos de contexto de FIWARE

El Orion Context Broker permite la publicación de información de contexto por entidades, es un servidor que implementa una interfaz de programación de aplicaciones, la cual está basada en un modelo de información NGSI, por lo cual se pueden realizar varias operaciones como [10]:

- Registrar aplicaciones de contexto.
- Actualizar información de contexto.
- Ser notificado cuando surjan cambios en la información de contexto.
- Consultar información de contexto.

## 2.6 Modelo NGSI para representar entidades en la plataforma FIWARE

Dentro de la plataforma FIWARE, una entidad representa el estado de un objeto físico o conceptual que existe en el mundo real. Las entidades al representar objetos contienen un identificador único, así como atributos asociados al objeto que describen sus características, además las características del objeto pueden tener sus propias características, teniendo así una relación directa formada entre el objeto (entidad) y sus características (atributos).

La información de contexto en FIWARE está representada a través de estructuras de datos genéricos referidos como elementos de contexto el cual se refiere a la información que es producida y que es relevante para su uso, además un elemento de contexto proporciona información relevante a una entidad en particular. Un elemento de contexto contiene un id y un tipo que identifica exclusivamente a una entidad. Dicho lo anterior, pueden existir metadatos, sin embargo, la existencia de metadatos vinculados a un atributo de elemento de contexto es opcional. Toda la comunicación al Orion Context Broker se realiza a través de la API RESTful NGSI v2, la cual define un modelo de datos para la información de contexto, basada en un modelo de información simple usando la noción de entidades de contexto, proporciona una interfaz de datos de contexto para el intercambio de información por medio de operaciones de consulta, suscripción y actualización [10].

## 2.7 Componente Grafana para graficar datos

Grafana es una aplicación web de visualización interactiva y análisis de código abierto multiplataforma. Proporciona tablas, gráficos y alertas para la web cuando se conecta a fuentes de datos compatibles. Es ampliable mediante un sistema enchufable. Los usuarios finales pueden crear paneles de monitoreo complejos utilizando generadores de consultas interactivos [11].

## 2.8 Agente del Internet de las Cosas en FIWARE

Conectar “objetos” o “cosas” implica la necesidad de superar un conjunto de problemas que surgen en las diferentes capas del modelo de comunicación. Debido a la falta de estándares aceptados a nivel mundial, se requiere de un entorno heterogéneo para los dispositivos que utilizan diferentes protocolos. Por esta razón es que los agentes IoT fueron creados junto con la plataforma FIWARE, para así permitir a un grupo de dispositivos enviar sus datos al Context Broker usando sus propios protocolos nativos, son intermediarios entre los dispositivos y plataformas IoT, razón por la cual se debe determinar primero el protocolo con el cual, el dispositivo se comunica y así seleccionar el agente IoT adecuado en la figura 2-1 se observa la relación entre los protocolos de comunicación, agentes IoT y el Orion Context Broker [12].

Los siguientes son ejemplos de agentes IoT que ya existen para los protocolos de comunicación.

- IoTAgent-JSON.
- IoTAgent-LWM2M.
- IoTAgent-Ultralight.
- IoTAgent loRanWAN.
- IoTAgent OPC-UA.
- IoTAgent sigfox.

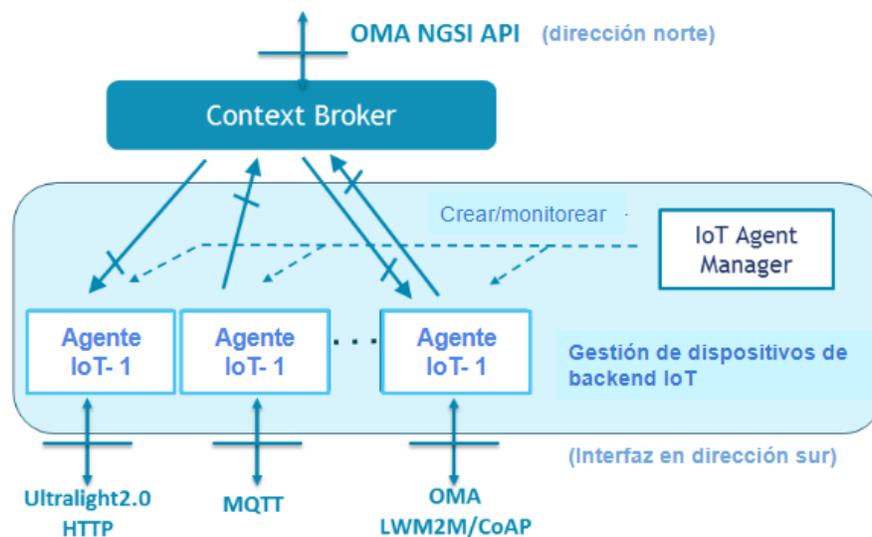


Figura 2-1. Interacción entre el Orion Context Broker, agentes IoT y protocolos de comunicación [12]

## 2.9 Protocolos de comunicación de dispositivos IoT

Un aspecto crucial en el Internet de las Cosas es la estandarización y conectividad para así poder seguir evolucionando y ampliando soluciones entre los distintos dispositivos, los protocolos de comunicación que existen hoy en día proporcionan la conectividad necesaria

para enviar los datos que los dispositivos manejan. En este trabajo de tesis los protocolos de comunicaciones y los agentes IoT son de suma importancia debido a que el módulo de comunicación a desarrollar utilizará los protocolos y agentes como base para el código de conexión con FIWARE.

FIWARE considera los siguientes protocolos de comunicación con los dispositivos:

- MQTT (MQ Telemetry Transport) es un protocolo que actúa sobre TCP (protocolo de control de transmisión), el cual destaca por ser ligero y sencillo de implementar. Usualmente útil para los dispositivos de baja frecuencia [13].
- COAP (Constrained Application Protocol) es un protocolo de transferencia web especializado para usarse con nodos y redes restringidas. Está diseñado para permitir que los dispositivos simples se unan a través de redes de bajo ancho de banda y baja disponibilidad [13].
- HTTP (Hypertext Transfer Protocol) es un protocolo utilizado para transferir datos, igual que el MQTT actúa sobre TCP (Transmission Control Protocol) en sus versiones 1 y 2, mientras que la versión 3 admite el uso de UDP (User Datagram Protocol), una gran ventaja de utilizar HTTP es su capacidad para enviar y recibir grandes cantidades de datos de manera eficiente [13]
- LoRanWAN es un protocolo de control de acceso a medios para redes de área amplia, diseñado para permitir que los dispositivos de baja potencia se comuniquen con aplicaciones a través de conexiones inalámbricas de largo alcance [13].
- Sigfox enfocado en la autonomía, simplicidad y eficiencia de costo, es un protocolo de solución económica y confiable de bajo consumo utilizado para conectar sensores y dispositivos [14].
- Lightweight M2M (Machine to Machine) es un protocolo de gestión de sensores diseñado para redes de sensores y las demandas de un entorno máquina a máquina [15]
- NGSI (Next Generation Service Interfaz) es un protocolo desarrollado por OMA para manejar el contexto de información, el cual provee operaciones como: gestionar la información de contexto sobre las entidades. Acceder a la información de contexto disponible sobre entidades de contexto [16].

# Capítulo 3

## Estado del arte

---

En este capítulo se provee de una revisión al estado del arte sobre algunos de los temas más relevantes de esta tesis: la seguridad e integridad de los datos de los dispositivos, Agentes para el Internet de las Cosas y la conectividad entre dispositivos y la nube. A continuación, se presenta cada uno de los trabajos recopilados de la literatura. Los trabajos relacionados con este proyecto de investigación han sido clasificados en tres categorías, de acuerdo con los objetivos que se persiguen en esta tesis, las cuales son:

#### Seguridad e integridad de datos en dispositivos IoT

En esta categoría se analizan trabajos que presentan desarrollos para mejorar la seguridad e integridad de los datos recopilados y enviados de los dispositivos IoT.

#### Uso de agentes para el IoT

En esta categoría se analizan trabajos que presentan desarrollos para el uso y configuración de agentes utilizados en el Internet de las Casas.

#### Conectividad entre dispositivos y la nube

En esta categoría se analizan trabajos que presentan desarrollos para la mejora en la conectividad con los dispositivos que trabajan en conjunto con la nube del Internet de las Cosas.

#### Clasificación de los trabajos del estado del arte en navegación en interiores con nuevas tecnologías

Las categorías utilizadas para clasificar los tipos de navegación analizados son:

### 3.1 Trabajos relacionados con la seguridad de los datos en dispositivos IoT

La seguridad en IoT es el acto de proteger los dispositivos de Internet y las redes a las que están conectados frente a amenazas e infracciones al proteger, identificar y monitorear los riesgos, al tiempo que ayuda a corregir las vulnerabilidades de una variedad de dispositivos que pueden presentar riesgos de seguridad para su negocio.

#### 3.1.1 Mejora de la seguridad en aplicaciones del IoT basadas en la plataforma FIWARE [16]

En este trabajo el autor propone un mecanismo de seguridad con los protocolos de comunicación (MQTT, LWM2M/COAP y HTTP) Este trabajo el objetivo propuesto por el autor es el de incorporar servicios de seguridad de un extremo a otro utilizando cifrado y control de acceso en todas las solicitudes NGSI (API RESTful), también la implementación del protocolo DTLS 1.2 en NodeJs para admitir el protocolo LWM2M / CoAP y su adición al agente de IoT. Además, este trabajo también permite al agente de IoT de la plataforma FIWARE admitir la comunicación TLS con el protocolo MQTT. Para lograr estos objetivos el autor configuró dos objetivos de prueba. El primero implica la creación de un escenario sin requisitos de seguridad y el segundo considera el uso de cifrado para mensajes intercambiados entre los dispositivos,

complementado con autenticación y autorización. Para los protocolos MQTT y LWM2M/CoAP, el autor condujo inspecciones de paquetes para validar la secuencia de mensajes necesaria, además de la medición de latencia en cada escenario, en este contexto la latencia es el tiempo de comunicación entre un dispositivo y el Orion Context Broker. El protocolo MQTT obtuvo un tiempo en promedio de 0.08815 segundos con los servicios de seguridad y 0.06119 segundos sin servicios de seguridad, por otro lado, el protocolo LWM2M/CoAP obteniendo un tiempo de 0.02845 segundos sin los servicios de seguridad y 0.03444 segundos con los servicios de seguridad. Los resultados del protocolo LWM2M/CoAP son más bajos que los del protocolo MQTT. La figura 3-1 muestra un diagrama de la solución propuesta por los autores de este trabajo de investigación.

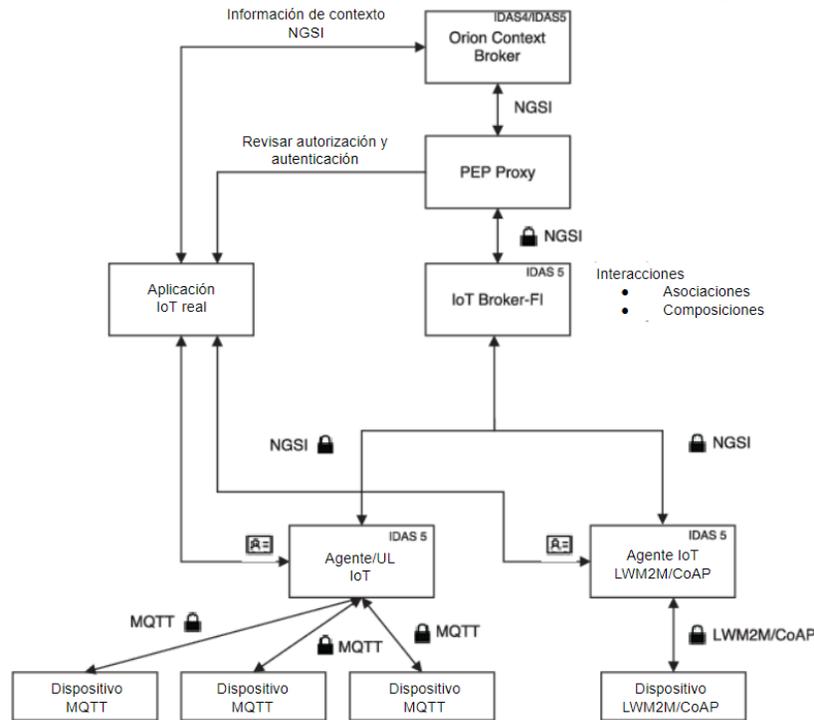


Figura 3-1. Arquitectura de seguridad en aplicación IoT propuesta en [16]

### 3.1.2 Seguridad en flujo para dispositivos IoT utilizando una puerta de enlace SDN [17]

En el trabajo de investigación el autor propone el uso de una red definida por software (Software-Defined Networking, SDN por sus siglas en inglés) como un medio para monitorear el tráfico originado por los dispositivos del Internet de las Cosas. Con esto se puede detectar el comportamiento malicioso y realizar una respuesta apropiada. La experimentación que realizó el autor consistió en desarrollar una topología simple, la cual consiste en un dispositivo IoT, un destino sincronizado y un nodo de ataque. La experimentación se centró en el uso de ataques basados en protocolo de control de transmisión y protocolos de control de mensajes de Internet, en el escenario 1 se hizo uso de un ataque TCP del atacante al dispositivo. El tráfico genuino se envió mediante TCP a aproximadamente 1,5 Mbps. La herramienta Low Orbit Ion Cannon se utilizó para enviar 2 Mbits por segundo de tráfico, saturando así el enlace. A los 5 segundos, se inició el ataque y aproximadamente a los 10 segundos, el ataque se mitiga bloqueando este flujo. En el escenario 2 se realiza un ataque mediante TCP que transmite a aproximadamente 1,5 Mbps y a los 5 segundos se inicia un ataque basado en

ICMP en el dispositivo de IoT, y aproximadamente a los 9 segundos el ataque se bloquea con éxito. Este trabajo confirma la capacidad del mecanismo SDN para bloquear y mitigar los ataques de este tipo. La figura 3-2 muestra un diagrama de la solución propuesta por los autores de este trabajo de investigación.

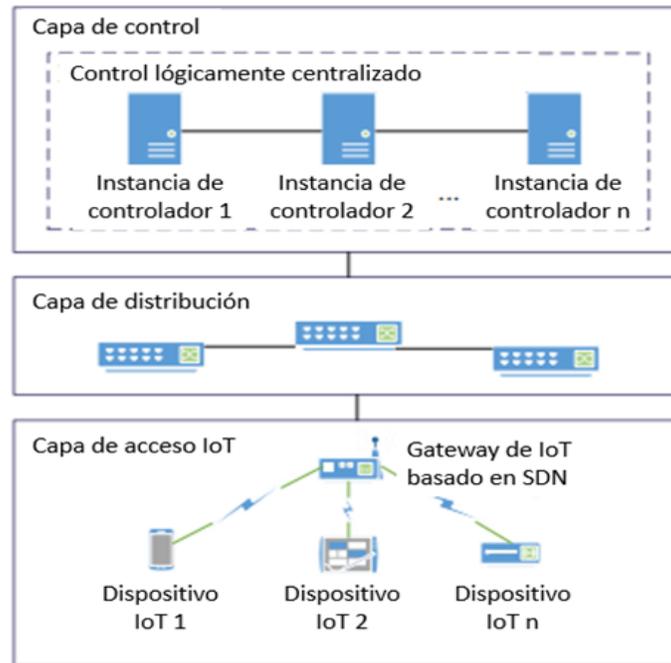


Figura 3-2. Arquitectura de seguridad para dispositivos utilizando SDN propuesta [17]

## 3.2 Trabajos relacionados al uso de agentes para el IoT

Los agentes del Internet de las Cosas son de suma importancia en el envío de datos, mejorando la seguridad y asegurando el formato de los datos enviados a través de los protocolos. Los agentes de IoT suelen ser componentes de una plataforma de IoT más amplia y, por lo general, los administran dispositivos de puerta de enlace de IoT ubicados en el perímetro de la red.

### 3.2.1 Un modelo de confianza de agentes autónomo para sistemas de IoT [18]

En este trabajo el autor propone un modelo de confianza de agente autónomo para disminuir los problemas de seguridad, aumentar la confiabilidad y credibilidad en entornos dinámicos de IoT, el autor propone una arquitectura llamada TAEC (Trustworthy Agent Execution Chip), utilizando una plataforma de software y hardware de alta seguridad y rentable para el funcionamiento seguro del Agente.

El modelo propuesto por el autor funciona en una topología centralizada o distribuida de IoT. Cada nodo del sensor puede registrar su nombre de nodo, dirección de red, requisito de seguridad y certificado digital. Este último es emitido por el fabricante TAEC (TAECM). TAECM y verificado por la copia de la clave pública de TAEC, el cual es responsable de utilizar el

coprocesador de cifrado específico para generar la clave de par para garantizar que la clave privada no pueda ser espiada por nadie. El autor selecciona el chip de silicón con una arquitectura y funciones completas del sistema para ser el hardware de TAEC, además se seleccionó el sistema operativo Linux integrado para ser el sistema operativo que ejecute la plataforma, y al final se usó el J2ME para ser la herramienta de desarrollo. La figura 3-3 muestra un diagrama de la solución propuesta por los autores de este trabajo de investigación.

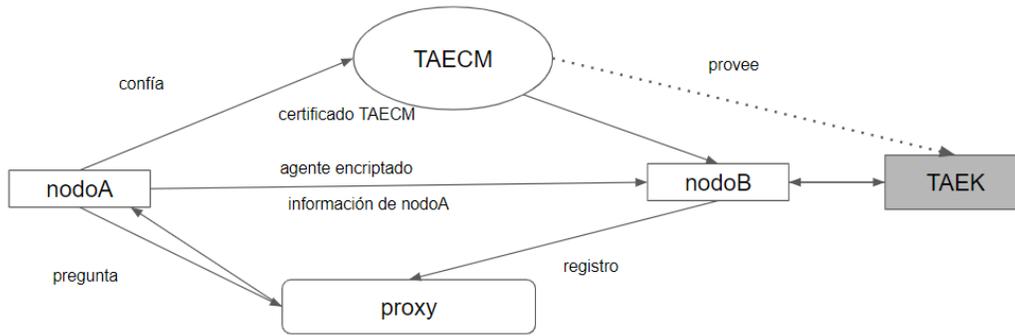


Figura 3-3. Arquitectura de un modelo de confianza en sistemas IoT propuesta en [18]

### 3.2.2 Arquitectura de aprovisionamiento de servicios resiliente y autómatas basado en múltiples agentes para el Internet de las Cosas [19]

En este trabajo de investigación el autor propone una arquitectura de IoT basada en agentes y su middleware para realizar autónomos y resilientes prestaciones de servicios en sistemas IoT, la arquitectura de aplicación y del Internet de las Cosas basado en agentes (Agent-based Internet of Things por sus siglas en inglés AIoT) de dispositivos IoT, así como el esquema autónomo de organización y reorganización para que de forma autónoma se pueda componer y operar la aplicación usando el agente IoT de los dispositivos. El autor implementa una aplicación de logística autónoma utilizando pequeños robots con sensores, mediante las pruebas se confirma que, en cuanto a diseño e implementación, la arquitectura del Internet de las Cosas basado en agentes mitiga la carga administrativa en la organización y reorganización de IoT. Los resultados de la comparación a nivel de arquitectura entre la aplicación logística basada en arquitectura Internet de las Cosas basado en agentes, y la aplicación logística basada en arquitectura convencional revela la flexibilidad arquitectónica y la resistencia de la aplicación Internet de las Cosas basado en agentes. La figura 3-4 muestra un diagrama de la solución propuesta por los autores de este trabajo de investigación.

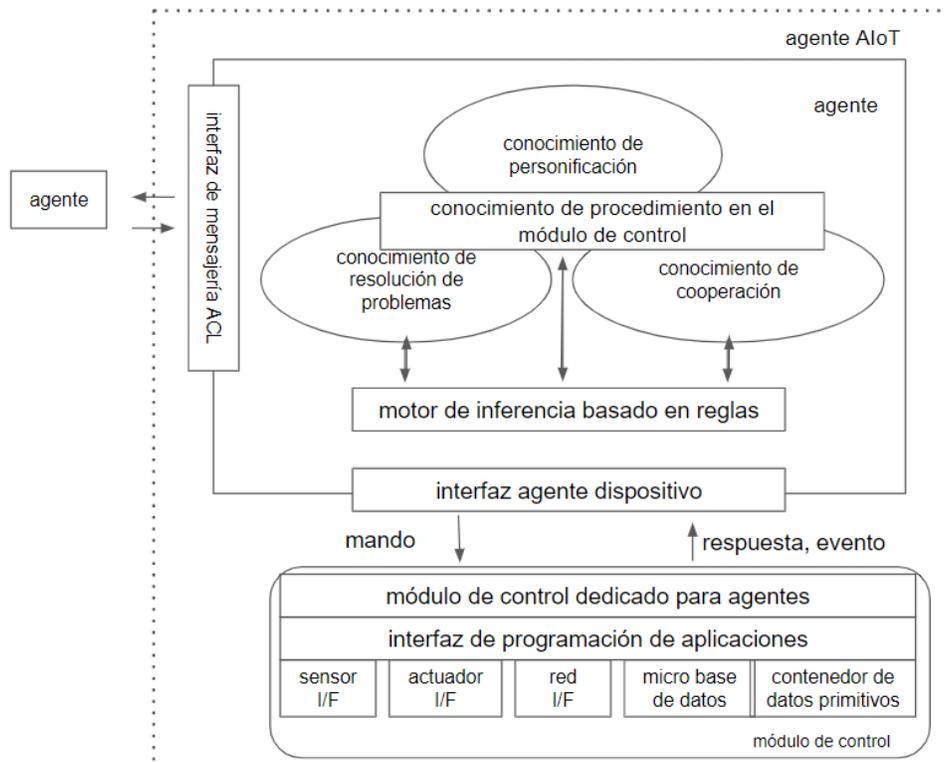


Figura 3-4. Arquitectura de aprovisionamiento de servicios basada en múltiples agentes propuesta en [19]

### 3.3 Trabajos relacionados a la conectividad entre dispositivos y la nube

La nube almacena los datos de IoT y se puede acceder a ellos en cualquier momento y en cualquier lugar, por lo que es ideal para múltiples ubicaciones u operaciones dispersas. Combinadas, las soluciones de IoT y la nube apoyan la captura de datos con control en tiempo real y monitoreo de inteligencia. Razón por la cual es importante contar con una conectividad estable y segura.

#### 3.3.1 Habilitación de la conectividad del IoT para sensores Modbus TCP [20]

En este trabajo el autor presenta una arquitectura que permite correr PCL de generaciones viejas, con el fin de adquirir más valor de los datos de los sensores a través de tecnologías IoT, tomando el enfoque para capturar datos de los sensores con el protocolo Modbus, para así transformar los datos a un estándar de comunicación IoT, además de incluir un método que almacene, procese y visualice los datos del sensor conectado. La arquitectura propuesta fue aprobada en un área con tres sensores de temperatura y tres sensores de CO2 conectados a un controlador lógico programable (Programmable Logic Controller PLC por sus siglas en inglés), en la figura 3.5 se muestra un diagrama de la solución propuesta en este trabajo.

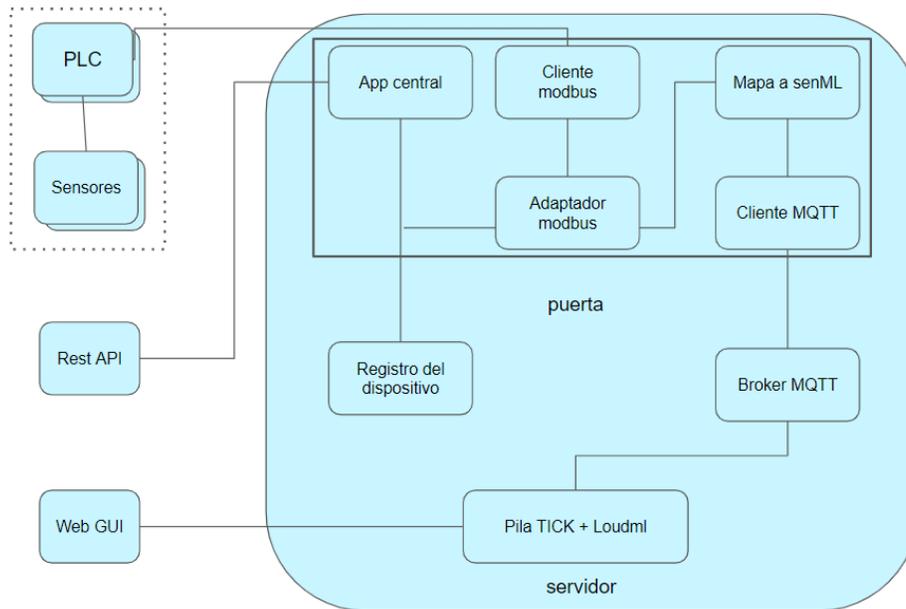


Figura 3-5. Arquitectura de conectividad para sensores Modbus TCP propuesta en [20]

### 3.3.2 Plataforma segura de agentes IoT con mecanismo de cálculo estadísticos distribuido m-Cloud [21]

En este trabajo el autor realiza un análisis de las brechas fundamentales que existen entre los dispositivos físicos y el software integrado de IoT, el autor llega a la conclusión de que a dificultad crítica en la seguridad de IoT está en la implementación de las uncciones en los dispositivos físicos, además de proponer una arquitectura de plataforma de agente IoT donde se albergarán clones virtuales de los dispositivos físicos, al separar los dispositivos de las funciones que realizan, esto permite ejecutar las funciones de forma aislada. El autor se enfoca en el servicio que obtiene índices estadísticos en lugar de tener valores de sensores individuales, La arquitectura restringe servicios de aplicaciones por naturaleza, por ejemplo, servicios personalizados basados en información privada. Como resultados, la plataforma de agentes IoT proporciona una forma más segura de gestionar los valores de los sensores en la nube sin ninguna modificación de los programas de agentes/ dispositivo IoT. La Figura 3.6 muestra un diagrama de la solución propuesta por los autores de este trabajo de investigación.

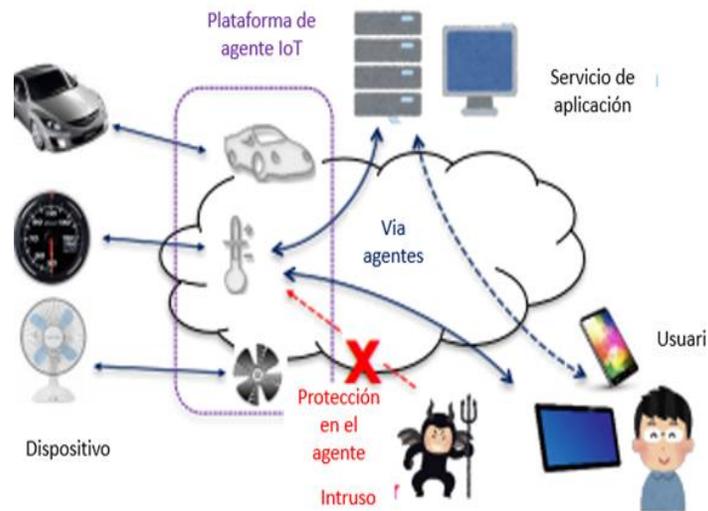


Figura 3-6. Arquitectura de una plataforma segura de agentes IoT propuesta en [21]

Durante la investigación de los trabajos relacionados y estado del arte se realizó la búsqueda de mejoras en la conexión entre los dispositivos y las distintas plataformas del Internet de las Cosas, aprendiendo que la conexión de los dispositivos utilizando los protocolos de comunicación y los agentes IoT es de crucial importancia no solo por la conectividad también por cuestiones de seguridad reforzando el acceso a los datos enviados a través de los protocolos, en concreto en los trabajos relacionados con la conectividad de los dispositivos físicos y la nube al utilizar clones de los dispositivos físicos y trabajar con los clones y no directamente con los dispositivos permitió reforzar la seguridad de los datos, con esto en mente se propuso realizar un almacenamiento temporal de los datos antes de enviarlos a la plataforma FIWARE. Además, en los trabajos relacionados con la seguridad en los dispositivos IoT, nos permitió entender de mejor manera del uso de los protocolos de comunicación con la finalidad de reforzar la seguridad de envío de datos en los protocolos.

# Capítulo 4

## Sistema para la conexión automática de dispositivos IoT a FIWARE

---

Este capítulo muestra la arquitectura del sistema, explicando cada una de las capas que la componen además de los módulos que conforman al sistema desarrollado: el módulo gestor de dispositivos y usuarios encargado de administrar y almacenar la información de los usuarios y dispositivos para su posterior uso en los otros dos módulos, el módulo de conexión entre el dispositivo, el sistema web generando los códigos de conexión para el dispositivo y la nube de FIWARE , además el módulo receptor de información de datos de los dispositivos al sistema web para su posterior envío a FIWARE utilizando el modelo de datos NGSI. Se muestra también la funcionalidad del sistema que realiza la construcción de los códigos de conexión y recepción y envío de datos a la plataforma FIWARE.

## 4.1 Arquitectura del sistema

La arquitectura propuesta utiliza los componentes de la plataforma FIWARE para el desarrollo de la aplicación con el objetivo de generar un sistema de código totalmente abierto, el cual se adapte a los estándares que provee esta plataforma IoT y que además utilice el concepto de Agentes IoT que fue desarrollado por FIWARE para mejorar la seguridad entre la conexión de dispositivos y la nube. Los componentes FIWARE utilizados en esta tesis son: el Orion Context Broker, CrateDB y Grafana. La interacción entre FIWARE y el sistema web creado se hace mediante una API RESTful. Las APIs RESTful son servicios con los cuales podemos intercambiar información mediante protocolos y estándares establecidos como es, por ejemplo, HTTP. En la figura 4.1 se muestra la arquitectura propuesta la cual se compone de las siguientes capas: Capa de aplicación y Ecosistema FIWARE.

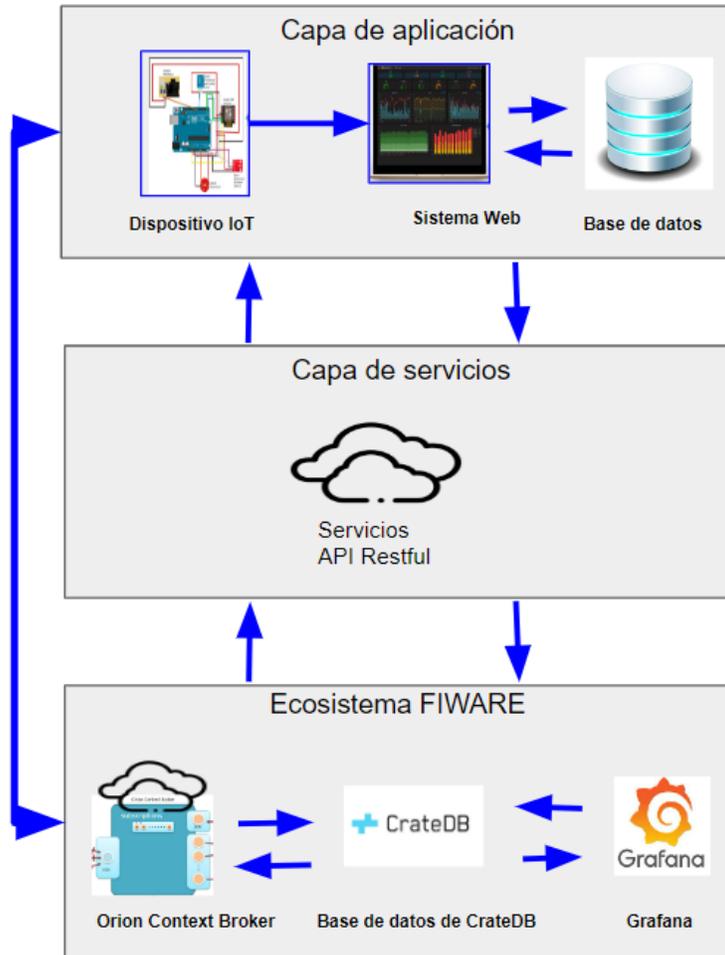


Figura 4-1. Arquitectura del sistema propuesto.

#### 4.1.1 Capa de aplicación

La capa de aplicación contiene el sistema web es el encargado de recibir, almacenar, procesar y enviar los datos de los dispositivos. Para realizar estas tareas, el sistema cuenta con tres módulos, los cuales son módulo gestor, módulo generador de código para la conexión con FIWARE y el módulo receptor, con los módulos, el sistema se conecta a una base de datos diseñada para almacenar la información de los usuarios y los dispositivos con la finalidad de que el sistema pueda gestionar esta información y utilizarla para generar los códigos de conexión, la recepción y el envío de datos del dispositivo a FIWARE, el envío de datos es a través de los servicios RESTful.

#### 4.1.2 Capa de servicios

En la capa de servicios se utilizan los servicios RESTful los cuales permiten la interacción entre la capa de aplicación y el ecosistema FIWARE, mediante solicitudes para el envío de datos.

#### 4.1.3 Ecosistema FIWARE

El ecosistema de FIWARE se encarga de recibir los datos enviados a través del sistema web, con la finalidad de manejar los datos recibidos por el Orion Context Broker. Se realiza una conexión con el módulo de CrateDB para crear una tabla donde se almacenan todos los

datos históricos enviados a FIWARE, con la finalidad de conectar al módulo de Grafana para visualizar los datos de manera gráfica.

## 4.2 Implementación de la arquitectura propuesta

La implementación de arquitectura propuesta se realizó mediante el desarrollo de un sistema web la cual implementa las funcionalidades de gestionar la información de los usuarios y dispositivos, así como manejar la autenticación de los usuarios, generar los códigos necesarios para la recepción y el envío de datos a FIWARE, así como la recepción y gestión de los datos enviados por el dispositivo IoT, para lograr esto se desarrollaron tres módulos específicos para cada una de las tareas los cuales son:

- **El módulo gestor-** Este módulo tiene como objetivo administrar las características de los sensores, la información de los usuarios, los aspectos de autenticación y los detalles de la Nube FIWARE. Este módulo está compuesto por tres submódulos: Gestión de usuarios, Gestión de dispositivos y Gestión de autenticación.
- **El módulo generador de código para la conexión a FIWARE-** Este módulo tiene como objetivo generar el código necesario para definir las entidades FIWARE correspondientes a los dispositivos que proporcionarán datos al sistema IoT, la creación de entidades NGSI y la generación de código para la conexión con FIWARE. Este módulo está compuesto por tres submódulos: generador de códigos para crear entidades NGSI, generador de códigos para configurar un agente IoT y generador de códigos para la conexión con FIWARE.
- **El módulo receptor de información-** Este módulo tiene como objetivo producir el código necesario para que el sistema IoT pueda obtener datos de los sensores previamente registrados. Además, este módulo tiene como objetivo generar el código que permita la transformación de los datos de los sensores al formato NGSI (NGSI REST API) para ser enviados al Orion Context Broker a través de un agente IoT específico. Este módulo está compuesto por 3 submódulos: Generador de código para recibir datos de sensores, Generador de código para traducir datos a formato NGSI y Generador de código para enviar datos para FIWARE usando un agente IoT.

Estos tres módulos componen el sistema y permiten la gestión, conexión, recepción y envío de datos a FIWARE, en la figura 4.2 se muestra la arquitectura específica del sistema web donde se puede apreciar las entradas del sistema (la información del usuario, dispositivo y la nube de FIWARE), estas entradas son procesadas en los tres módulos del sistema para obtener como salida los códigos de conexión y las entidades NGSI.

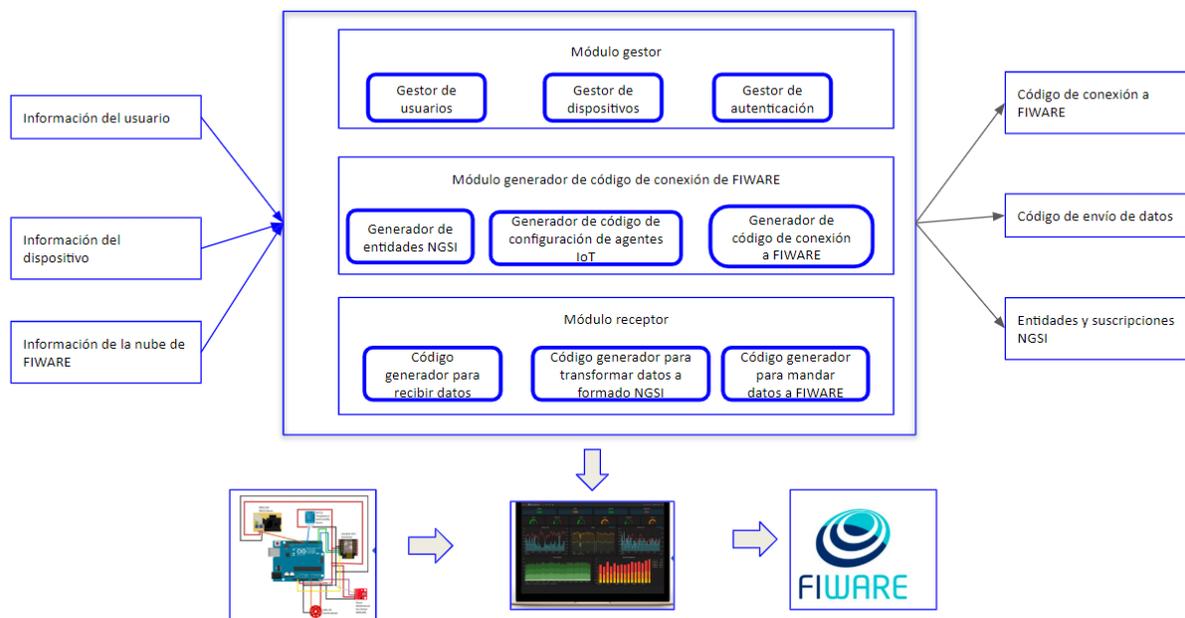


Figura 4-2. Arquitectura específica del sistema propuesto

### 4.2.1 Módulo gestor

El módulo gestor permite al sistema el administrar la información de los usuarios (nombre, correo y contraseña), dispositivos (nombre, lecturas, unidades, protocolos) y los detalles de la nube de FIWARE (dirección del Orion Context Broker y agente IoT), además de encargarse de la autenticación de los usuarios que se registran en el sistema, este módulo se conecta con la base de datos para almacenar toda la información que recibe en particular la información de los dispositivos debido a que, esta información es la que se utiliza para crear los códigos de conexión y recepción de datos del dispositivo.

La figura 4.3 se muestra el diagrama relacional utilizado para crear la base de datos con la cual se conecta el sistema desarrollado, el diagrama está compuesto por la entidad de usuario (atributos: id, nombre, correo y contraseña) la cual tiene una relación de 1 a muchos con la entidad de dispositivo (atributos: id, nombre), donde la entidad de dispositivo a su vez tiene una relación de 1 a muchos con la tabla de sensor (atributos: id, lectura, unidad), una relación de 1 a 1 con la entidad códigoFIWARE (atributos: id, OCB, host, versión) y una relación de 1 a 1 con la entidad protocolo (atributos: id, nombre, host, versión, mensaje), la entidad agenteloT (atributos: id, host, puerto, url) tiene una relación de 1 a 1 con la tabla protocolo.

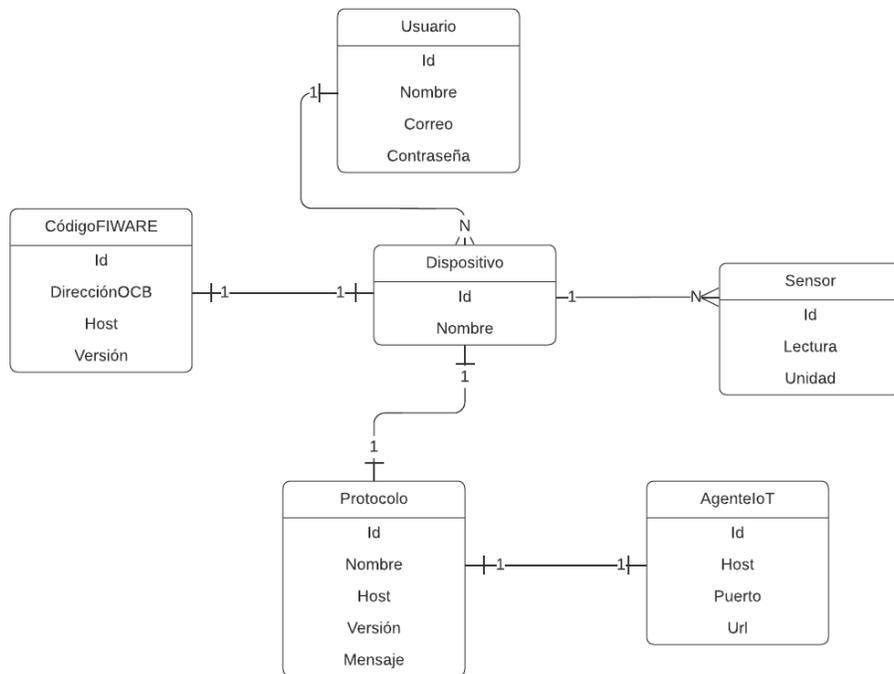


Figura 4-3 Modelo relacional de la base de datos del sistema

El módulo Gestor está compuesto por tres submódulos: Gestión de usuarios, Gestión de dispositivos y Gestión de autenticación.

El submódulo de Gestión de Usuarios ofrece la funcionalidad de registrar usuarios en el sistema mediante un formulario donde a los usuarios se les pide ingresar nombre, correo y contraseña. El sistema verifica los campos con la finalidad de que no se ingresen caracteres inválidos en los campos.

El sistema solo puede ser utilizado por usuarios registrados que proporcionaron su nombre de usuario, correo electrónico y contraseña. La validación del usuario se realiza mediante el submódulo de gestión de autenticación. Solo los usuarios autorizados pueden registrar sensores en el sistema. Es importante señalar que se registran los sensores que proporcionarán datos al sistema IoT final que se desarrollará automáticamente.

Los usuarios registrados deben definir el nombre de los sensores y el tipo de lecturas de cada sensor mediante el submódulo de gestión de dispositivos. Estas características de los sensores son relevantes porque se utilizan para crear los códigos utilizados para la conexión, envío y recepción de datos, el funcionamiento y relevancia de estos códigos se explican en el módulo generador de código de conexión a FIWARE y el módulo receptor.

## 4.2.2 Módulo generador de conexión de FIWARE

Este módulo tiene como objetivo generar el código necesario para definir las entidades FIWARE correspondientes a los dispositivos que proporcionarán datos al sistema IoT, la creación de entidades NGSI y la generación de código para la conexión con FIWARE. Este módulo está compuesto por tres submódulos: generador de códigos para crear entidades

NGSI, generador de códigos para configurar un agente IoT y generador de códigos para la conexión con FIWARE.

El enfoque FIWARE se basa en el concepto de entidades que relacionan y entidad con atributos y metadatos (figura 4.4). Dado que toda la comunicación entre los distintos componentes de la arquitectura del OCB se realiza a través de la API RESTful NGSI v2, la cual está inspirada y basada en la especificación OMA NGSI. Por lo tanto, la información de contexto en FIWARE está representada a través de estructuras de datos genéricas referidos como elementos de contexto (un elemento de contexto se refiere a la información que es producida, recopilada u observada y que puede ser relevante para su procesamiento, análisis y extracción de nuevo conocimiento). La información de contexto en OMA NGSI se representa a través de estructuras de datos llamadas elementos de contexto, los elementos de contexto están compuestos por un identificador, un tipo asociado a su entidad, además de tener atributos los cuales contiene un nombre, tipo y valor del atributo, estos atributos a su vez tienen meta datos los cuales tienen un nombre, tipo y valor, los metadatos son opcionales al momento de construir los elementos de contexto.

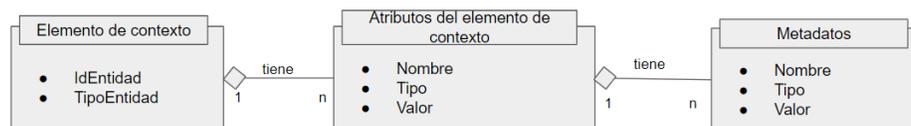


Figura 4-4 Modelo de datos de una entidad

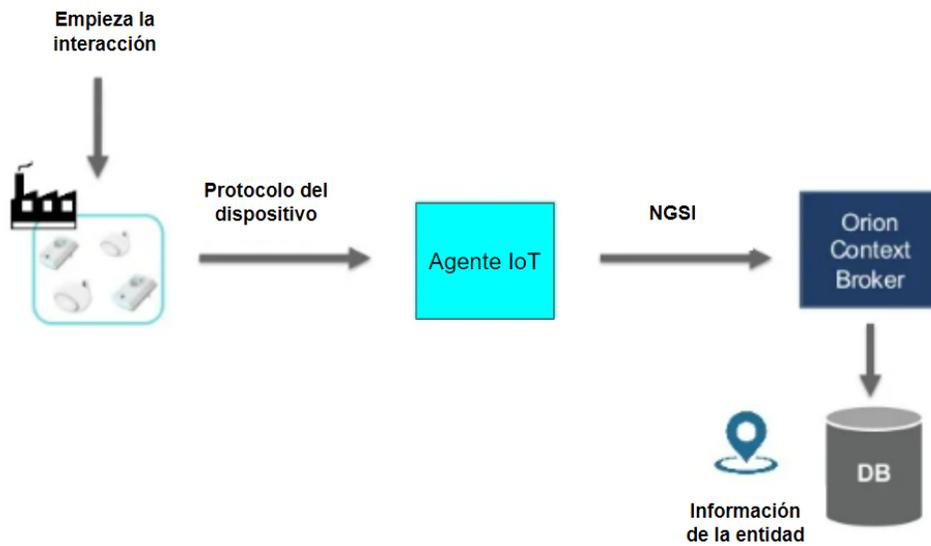
Los valores que se capturan de los sensores deben representarse utilizando el concepto de Entidades. Los usuarios deben utilizar el generador de códigos para crear las entidades NGSI para definir el mapa entre los diversos sensores definidos en el sistema y las diversas entidades predefinidas por FIWARE, siendo las entidades NGSI, formatos de datos tipo JSON que representan un objeto u cosa del mundo real con atributos o características tangibles en la figura 4.5 se muestra un ejemplo de una entidad NGSI con los elementos de contexto y atributos que lo componen, mostrando el “id” de la entidad la cual es “device1” el tipo de entidad la cual es “device1”, teniendo además dos atributos uno con el nombre “humedad”, el tipo “number” y el valor asociado a este atributo el cual es “1”, el segundo con el nombre “temperatura”, el tipo “number” y el valor asociado a ese atributo es “327.83”.

```
1 // 20211129204633
2 // http://192.168.0.12:1026/v2/entities
3
4 [
5   {
6     "id": "device1",
7     "type": "device1",
8     "humedad": {
9       "type": "Number",
10      "value": 1,
11      "metadata": {
12
13      }
14    },
15    "temperatura": {
16      "type": "Number",
17      "value": 327.83,
18      "metadata": {
19
20      }
21    }
22  }
23 ]
```

Figura 4-5. Entidad NGSI de un dispositivo IoT

Estas entidades ayudan al envío de información a FIWARE más en concreto al componente del Orion Context Broker, el cual utiliza la base de datos en mongoDB para almacenar el estatus actual de las entidades, no almacena información histórica de sus cambios.

Con las entidades creadas los usuarios deben seleccionar un agente de IoT específico utilizando el generador de código del submódulo para configurar un agente de IoT de acuerdo con el protocolo definido en el sensor. Un agente IoT es un componente el cual permite a un grupo de dispositivos mandar y gestionar sus datos por el Orion Context Broker además de manejar aspectos de la seguridad de la plataforma FIWARE (autenticación y autorización del canal), debido a que la seguridad es un aspecto importante de FIWARE, el correcto uso de los agentes es crucial para el funcionamiento del sistema web. La forma en la que el sistema realiza esta tarea es que utiliza plantillas de código JSON ya establecidas para cada agente IoT, en las plantillas se encuentra la configuración del agente IoT que utiliza el protocolo de comunicación establecido al momento de dar de alta el dispositivo en la figura 4.5 se muestra un ejemplo de interacción entre el protocolo de comunicación, el agente IoT y el Orion Context Broker de FIWARE. En la figura 4.6 se observa que la interacción empieza con los dispositivos enviando su información a través de los protocolos de comunicación que utilizan, esta información es recibida por el agente IoT el cual la manda al Orion Context Broker mediante el formato NGSI donde el Orion Context Broker almacena la información de contexto en una base de datos en mongoDB. Entonces el submódulo generador de código para la conexión con FIWARE produce el código para la conexión del agente IoT seleccionado con la nube FIWARE específica que el usuario definió en el módulo de gestión. Este código permite que el sistema automáticamente establezca un enlace estable con la nube FIWARE para enviar los datos de los sensores representados en las entidades FIWARE. Esta conexión entonces establece la relación entre los dispositivos, protocolos de comunicación, agentes IoT y la nube de FIWARE más en concreto el módulo del Orion Context Broker.



*Figura 4-6 Interacción entre agente IoT y el Orion Context Broker*

En la figura 4.7 se muestra un ejemplo del código generado para levantar los servicios de FIWARE, en este código se observa que está compuesto por la versión de FIWARE, el servicio del Orion Context Broker, las dependencias como mongodb, los puertos que utiliza además de incluir la configuración del agente IoT que se desea utilizar. La información se obtiene de los datos que el usuario da de alta al registrar el dispositivo, con el código generado el usuario puede compilar el código en el servidor donde desea alojar los servicios de FIWARE.

```

version: "3.8"
services:
  # Orion is the context broker
  orion:
    labels:
      org.fiware: 'tutorial'
    image: fiware/orion:${ORION_VERSION}
    hostname: orion
    container_name: fiware-orion
    depends_on:
      - mongo-db
    networks:
      - default
    expose:
      - "${ORION_PORT}"
    ports:
      - "${ORION_PORT}:${ORION_PORT}" # localhost:1026
    command: -dbhost mongo-db -logLevel DEBUG
    healthcheck:
      test: curl --fail -s http://orion:${ORION_PORT}/version || exit 1
      interval: 5s

  # IoT-Agent is configured for the UltraLight Protocol
  iot-agent:

```

Figura 4-7 Código para levantar servicios de FIWARE

En la Figura 4.8 se muestra el código de configuración del agente IoT ultralight, donde se puede observar que se debe especificar la imagen del agente con la versión a utilizar, el nombre del host, las dependencias que en este caso en particular es mongo debe, la red, así como los puertos por los cuales se comunica el agente y por último el ambiente el cual es de suma importancia debido a que se especifica la información no solo del agente sino que también el del servidor donde está alojado el Orion Context Broker, como los puertos por donde él se envía la información, la base de datos donde se almacena la información del dispositivo, la versión NGSI que se utiliza, la url URL pasada a Context Broker cuando se registran comandos, utilizada como ubicación de URL de reenvío cuando Context Broker emite un comando a un dispositivo, así como el broker del mqtt con su puerto.

```

iot-agent:
  image: fiware/iotagent-ul:latest
  hostname: iot-agent
  container_name: fiware-iot-agent
  depends_on:
    - mongo-db
  networks:
    - default
  expose:
    - "4041"
  ports:
    - "4041:4041"
  environment:
    - IOTA_CB_HOST=orion
    - IOTA_CB_PORT=1026
    - IOTA_NORTH_PORT=4041
    - IOTA_REGISTRY_TYPE=mongodb
    - IOTA_LOG_LEVEL=DEBUG
    - IOTA_TIMESTAMP=true
    - IOTA_CB_NGSI_VERSION=v2
    - IOTA_AUTOCAST=true
    - IOTA_MONGO_HOST=mongo-db
    - IOTA_MONGO_PORT=27017
    - IOTA_MONGO_DB=iotagentul
    - IOTA_PROVIDER_URL=http://iot-agent:4041
    - IOTA_MQTT_HOST=mosquitto
    - IOTA_MQTT_PORT=1883

```

Figura 4-8 Confirmación de un agente IoT para protocolo Ultralight

### 4.2.3 Módulo receptor

Este módulo tiene como objetivo producir el código necesario para que el sistema IoT pueda obtener datos de los sensores previamente registrados. Además, este módulo tiene como objetivo generar el código que permita la transformación de los datos de los sensores al formato NGSI (NGSI REST API) para ser enviados al Orion Context Broker a través de un agente IoT específico. Este módulo está compuesto por 3 submódulos: Generador de código para recibir datos de sensores, Generador de código para traducir datos a formato NGSI y Generador de código para enviar datos para FIWARE usando un agente IoT.

El submódulo Generador de códigos para recibir datos de sensores genera el código para acceder a los diversos sensores definidos en el módulo de gestión. En este caso no es

necesaria la intervención del usuario porque el sistema recupera la información de los sensores. Este código está escrito en c/c++ utilizado por el IDE de arduino, compuesto por:

- La dirección del sistema la cual indica a donde se envían los datos.
- Las librerías del dispositivo utilizadas para que el código compilado pueda leer los datos de los sensores del dispositivo.
- Las variables de lecturas del dispositivo, variables que se agregan a la dirección del sistema con los valores de las lecturas del dispositivo.
- El protocolo por el cual se envían los datos, protocolo de comunicación que permite el envío de datos al sistema web.

Este código, que se compila posteriormente, permite que el sistema IoT tome datos de cada sensor y pasarlos al módulo encargado de transformarlos en el formato NGSI para su posterior envío al Orion Context Broker

El submódulo generador de código para traducir datos a formato NGSI genera el código para mapear los datos de los sensores con las entidades definidas en el módulo anterior. En este caso, no es necesaria la intervención del usuario. Este código permite que el sistema IoT mapee las lecturas de sensores específicos en valores de los atributos de la entidad. En la figura 4.9 se muestra una representación gráfica de cómo el sistema recibe los datos del sensor utiliza el modelo de datos de FIWARE, así como el agente IoT para generar los datos de contexto que serán enviados al Orion Context Broker.

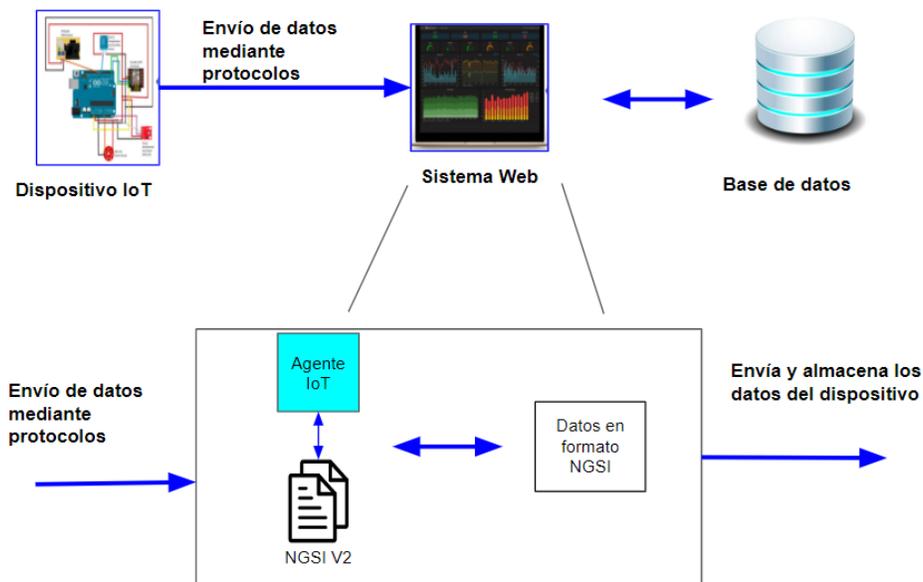


Figura 4-9 Diagrama transformación a formato NGSI

El submódulo generador de código para enviar datos a FIWARE utiliza el código generado en módulo generador de código para la conexión a FIWARE, donde se crea el código de conexión y las entidades NGSI, para crear las suscripciones de las entidades, siendo una suscripción un componente el cual permite notificar cuando surge un cambio en la información del contexto por lo cual no es necesario estar realizando solicitudes continuas al Orion Context Broker. Las suscripciones (figura 4.11) están compuestas por la información de las entidades y la información del dispositivo, donde se observa que cuenta con un

identificador único, una descripción, el estatus, los atributos que son las lecturas del dispositivo, así como los atributos de la notificación que son los que disparan las notificaciones del envío de datos y la url del servidor donde está alojado el Orion Context Broker.

```

1 // 20211129204804
2 // http://192.168.0.12:1026/v2/subscriptions
3
4 [
5   {
6     "id": "619bf7e659ee18128df27ad0",
7     "description": "Suscripcion device1",
8     "status": "active",
9     "subject": {
10      "entities": [
11        {
12          "idPattern": ".*",
13          "type": "device1"
14        }
15      ],
16      "condition": {
17        "attrs": [
18          "temperatura",
19          "humedad"
20        ]
21      }
22    },
23    "notification": {
24      "timesSent": 21,
25      "lastNotification": "2021-11-30T02:40:32.00Z",
26      "attrs": [
27        "id",
28        "temperatura",
29        "humedad"
30      ],
31      "attrsFormat": "normalized",
32      "http": {
33        "url": "http://192.168.0.12:8668/v2/notify"
34      },
35      "lastSuccess": "2021-11-30T02:40:32.00Z"
36    }
37  },
38 ]

```

Figura 4-10 Suscripción de la entidad NGSI

En la figura 4.12 está una representación gráfica de cómo interactúan las suscripciones, con los datos en formato NGSI para su posterior envío a FIWARE.

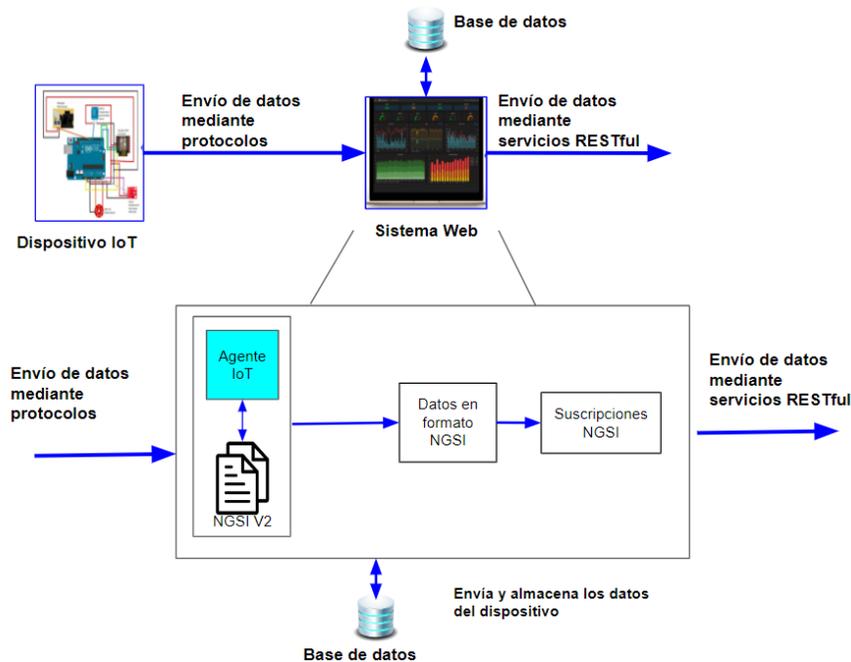


Figura 4-11 Diagrama de interacción de la suscripción NGSI

Además de crear una base de datos en CrateDB para almacenar los datos históricos del dispositivo, esta base de datos está compuesta por el identificador único, nombre, lecturas, hora y fecha en la que se recibieron los datos, para su posterior uso.

Una vez ejecutados los tres módulos del sistema, se ha generado todo el código y se debe compilar para implementar las funcionalidades con la finalidad de: obtener datos de los sensores, transformar estos datos en entidades FIWARE, establecer la comunicación con un agente IoT que será el intermediario entre los sensores y FIWARE, y finalmente, enviar los datos a la nube FIWARE para su posterior uso como la visualización de los datos en el módulo de Grafana la cual permite la visualización y el formato de datos métricos además de permitir la creación de cuadros de mando y gráficos a partir de múltiples fuentes, incluidas bases de datos de series de tiempo, Grafana se conecta con la base de datos en CrateDB para usar la información del dispositivo y los datos almacenados, en la figura 4.13 se muestra la interacción que tiene estos submódulos con el ecosistema de FIWARE.

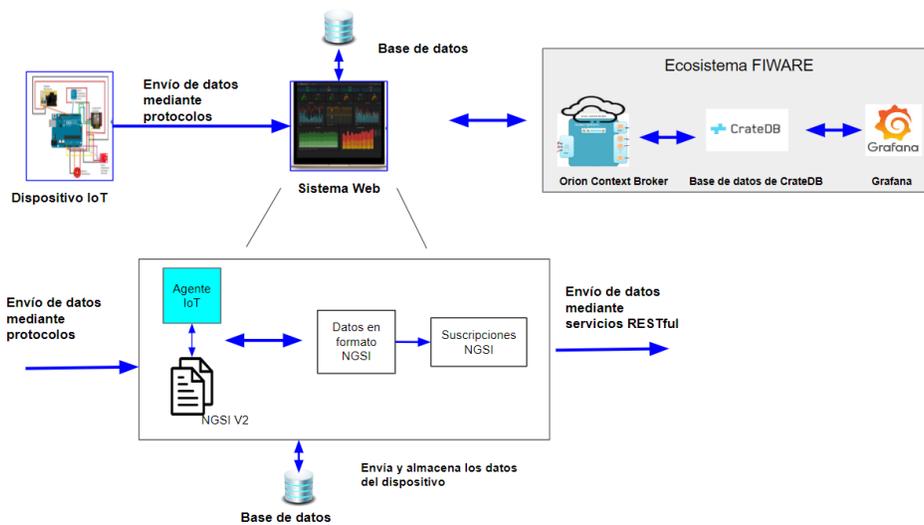


Figura 4-12 Diagrama de interacción con la nube de FIWARE

### 4.3 Sistema web para la conexión entre dispositivos IoT a la plataforma FIWARE

El caso de estudio se realizó utilizando una placa Wemos D1 wifi uno esp8266 y un sensor de temperatura y humedad dht11, en la figura 4.12 se muestra la placa con el sensor conectado.

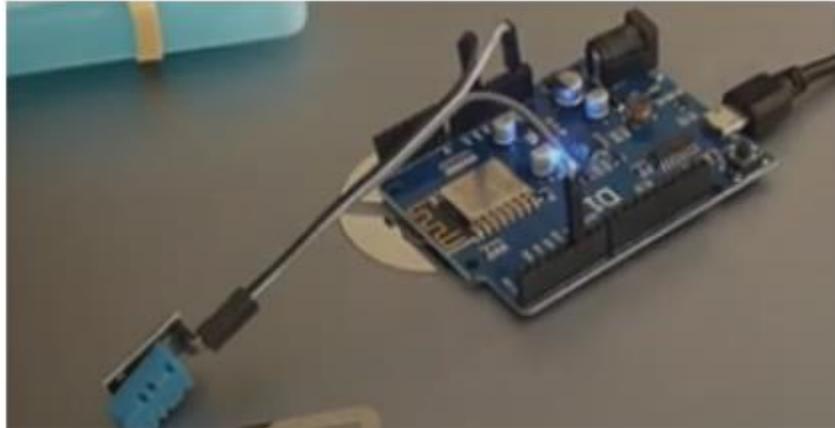


Figura 4-13 Dispositivo Arduino D1 con sensor Dht11

Con el dispositivo conectado y configurado el usuario accede al sistema web donde tiene que registrarse como usuario nuevo, para ello en la pantalla de registro de usuario (ver figura 4.13) llena el formulario ingresando su nombre de usuario, correo electrónico y contraseña.

Figura 4-14 Formulario de registro de usuario

Una vez que el usuario ha iniciado sesión se procede a registrar el dispositivo con el cual se desea realizar la conexión y envío de datos a FIWARE, en la pantalla de registro de dispositivo (ver figura 4.15) el usuario llena el formulario con el nombre del dispositivo, unidades, valor máximo y mínimo de la lectura del dispositivo, la dirección del servidor del Orion Context Broker, el protocolo de comunicación y el agente del Internet de las Cosas que utiliza el dispositivo y las lecturas del dispositivo.

Dashboard User lists Device lists Add device Add user Profile Logout

### Add New Device

Name  
device1

Units  
C

Max value  
100

Min value  
0

OCB Address  
192.168.0.11

Choose a protocol:  
MQTT

Choose an IoT Agent:  
IoTAgent-UL

readings  
 Temperatura  humedad

[Register](#)

Figura 4-15 formulario del registro del dispositivo

Una vez realizado el registro de dispositivo, el usuario procede a cargar el código Arduino para el envío de datos, en la pantalla del código Arduino (Figura 4.16) se carga el archivo Arduino para que el sistema inserte el código con los datos necesarios para realizar el envío de datos, el código está compuesto por el id y las variables de la lectura del dispositivo, la dirección del sistema web, así como las librerías necesarias el funcionamiento del código el cual el usuario descarga y compila para realizar el envío de datos al sistema web.

Dashboard User lists Device lists Add device Add user Profile Logout

Seleccionar archivo Ningún archivo seleccionado [Read Contents](#)

Contents of wifi.ino File

temp/wifi.ino

### Arduino CODE

```

1 // DHT Pro Shield - Simple
2 *
3 * Example testing sketch for various DHT humidity/temperature sensors
4 * Written by ladyada, public domain
5 *
6 * Depends on Adafruit's DHT Arduino library
7 * https://github.com/adafruit/DHT-sensor-library
8 *
9 #include "ESP8266WiFi.h"
10 #include "WiFiClient.h"
11 #include "ESP8266WebServer.h"
12 #include "ESP8266HTTPClient.h"
13 const char* ssid = "IZZI-F506";
14 const char* password = "888u_1234_?";
15 const char* host = "http://192.168.0.16/";
16 const char* server = "localhost";
17 int sensorPin = D5; // select the input pin for the potentiometer
18 double temperatura;
19 double humedad;
20 String id="id=82";
21 double Thermistor(int RawADC) {
22   double Temp;
23   Temp = log(10899.0*((1024.0/RawADC-1)));
24   Temp = 1 / (0.00112946 + (0.00023425 + (0.0000000876741 * Temp * Temp))) * Temp ;
25   Temp = Temp - 273.15; // Convert Kelvin to Celsius
26   //Temp = (Temp * 9.0)/ 5.0 + 32.0; // Convert Celsius to Fahrenheit
27   return Temp;
28 }
29 void setup() {

```

[download](#)

Figura 4--16 Código Arduino

Una vez compilado el código Arduino, el sistema web recibe los datos y los almacena en una tabla que crea con la información del dispositivo, en la figura 4.17 se muestra los datos que recibe el sistema web donde se tienen las lecturas, la fecha y hora en la que se recibieron los datos, así como un identificador único para cada dato.

temperatura	humedad	link	cur_time	id
327.83	250.01	82	2021-11-15 14:17:09	27
317.83	247.5	82	2021-11-15 14:19:05	28
307.83	231.4	82	2021-11-15 14:19:48	29
327.83	237.6	82	2021-11-15 14:19:58	30
327.83	1.00	82	2021-11-15 14:20:08	31
327.83	1.00	82	2021-11-15 14:20:18	32
327.83	1.00	82	2021-11-15 14:20:28	33
327.83	1.00	82	2021-11-15 14:20:38	34
327.83	1.00	82	2021-11-15 14:20:48	35
327.83	1.00	82	2021-11-15 14:21:06	36
327.83	1.00	82	2021-11-15 14:21:16	37
327.83	1.00	82	2021-11-15 14:21:26	38

*Figura 4-17 Tabla de datos recibidos por el sistema*

Con la entidad creada el sistema entonces procede a crear la suscripción del dispositivo para guardar los datos históricos en la figura 4.19 se muestra la suscripción del dispositivo, donde se observa que cuenta con un identificador único, una descripción, el estatus, los atributos que son las lecturas del dispositivo, así como los atributos de la notificación que son los que disparan las notificaciones del envío de datos y la url del servidor donde está alojado el Orion Context Broker.

```

1 // 20211129204804
2 // http://192.168.0.12:1026/v2/subscriptions
3
4 [
5   {
6     "id": "619bf7e659ee18128df27ad0",
7     "description": "Suscripcion device1",
8     "status": "active",
9     "subject": {
10      "entities": [
11        {
12          "idPattern": ".*",
13          "type": "device1"
14        }
15      ],
16      "condition": {
17        "attrs": [
18          "temperatura",
19          "humedad"
20        ]
21      }
22    },
23    "notification": {
24      "timesSent": 21,
25      "lastNotification": "2021-11-30T02:40:32.00Z",
26      "attrs": [
27        "id",
28        "temperatura",
29        "humedad"
30      ],
31      "attrsFormat": "normalized",
32      "http": {
33        "url": "http://192.168.0.12:8668/v2/notify"
34      },
35      "lastSuccess": "2021-11-30T02:40:32.00Z"
36    }
37  },
38 ]

```

Figura 4-18 Suscripción de la entidad NGSI del dispositivo

Con las entidades y suscripciones creadas el sistema crea una tabla en CrateDB para guardar los valores históricos del dispositivo en la figura 4.20 se muestra la tabla creada.

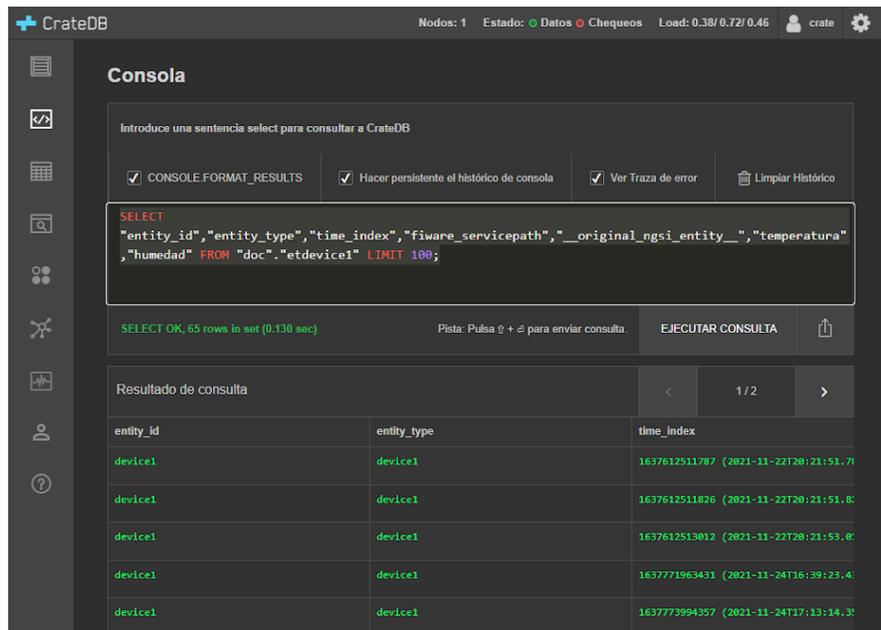


Figura 4-19 Tabla en CrateDB

Para finalmente realizar la conexión con el Orion Context Broker, en la pantalla del código de FIWARE (ver figura 4.21) se muestran los datos del dispositivo, así como el código necesario para levantar los servicios de FIWARE con los puertos que utilizan y el agente IoT con su

configuración. Este código se compila en el servidor donde se desea alojar los servicios de FIWARE.

FIWARE Code Back

Device name

readings

readings

Orion Context Broker address

IoT Agent

**FIWARE CONNECTION CODE** yaml

```
1 version: 3
2 services:
3   orion:
4     image: 'fiware/orion:${ORION_VERSION:-2.0.0}'
5     port: '1026:1026'
6     commands: 'logLevel DEBUG -noCache -dbhost mongo'
7     depends_on: '- mongo'
8     healthcheck:
9       test: ['CMD', 'curl', '-f', 'http://0.0.0.0:1026/version']
10      interval: 1m
11      timeout: 10s
12      retries: 3
13   iot-agent:
14     image: 'fiware/iotagent-ul:latest'
15     hostname: iot-agent
16     container_name: fiware-iot-agent
17     depends_on: mongo-db
18     networks: '- default'
19     expose:
20       - '4041'
21       - '7896'
22     ports:
23       - '4041:4041'
24       - '7896:7896'
25     environment:
26       - IOTA_CB_HOST=orion
27       - IOTA_CB_HOST_PORT=1026
```

download

submit Delete

Figura 4-20 Código de conexión a FIWARE

En la pantalla de Grafana (ver figura 4.22) se muestra el módulo de Grafana de FIWARE donde se puede observar las lecturas del dispositivo alojadas en FIWARE en una gráfica.

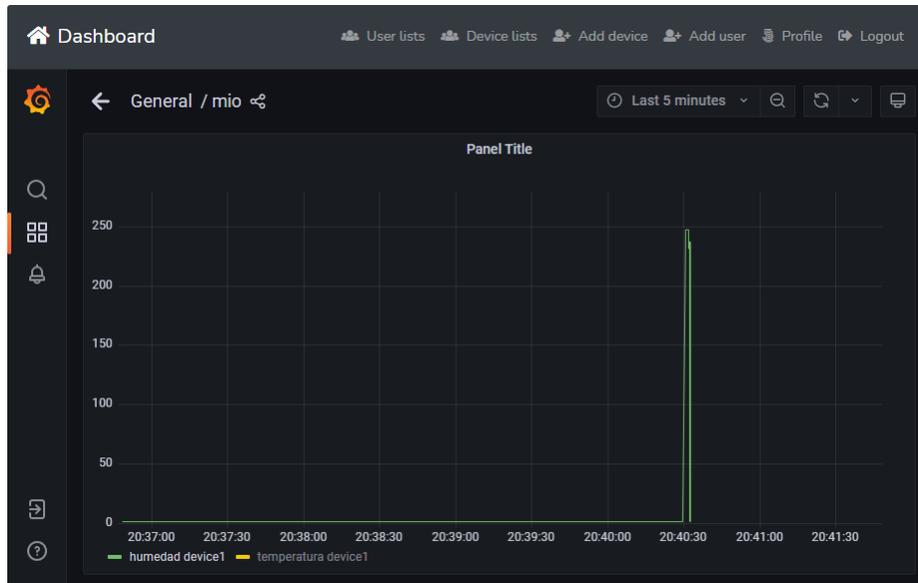


Figura 4-21 Módulo de Grafana

La implementación del sistema web desarrollado permitió a los dispositivos IoT conectarse a la plataforma FIWARE, enviar los datos generados por los sensores y mostrarlos mediante el componente de Grafana, así cumpliendo con los objetivos especificados en esta tesis. El caso de estudio permitió validar cada componente del sistema desarrollado con la finalidad de confirmar que el enfoque propuesto cumpliera con los requisitos especificados durante su diseño.

# Capítulo 5

## Pruebas y resultados

---

En este capítulo se describe el plan de pruebas establecido para llevar a cabo la evaluación de las aplicaciones que se generaron y el algoritmo usado.

Durante el diseño de un experimento es crucial tener presente ciertos conceptos como: el por qué y para qué se está desarrollando la experimentación, quiénes son los sujetos que participarán en las pruebas, cuáles y cuántos dispositivos se utilizarán durante las pruebas y las medidas de análisis estadístico. Este documento presenta el diseño experimental de la tesis de maestría "Sistema para la conexión automática de sensores IoT a la Plataforma FIWARE". El documento explica los conceptos principales utilizados durante las pruebas al sistema desarrollado.

## 5.1 Objetivo de la experimentación

La experimentación tiene como objetivo realizar pruebas sobre el sistema web desarrollado en esta tesis de tal forma que se pueda verificar su funcionalidad de acuerdo a los objetivos planteados. El sistema cuenta con las siguientes funcionalidades:

Inicio de sesión- Función principal es el acceso controlado al sistema web.

Registro de usuario- Función principal es agregar nuevos usuarios para que puedan hacer uso del sistema.

Registro del dispositivo- Función principal es agregar nuevos dispositivos para generar el código de conexión y envío de datos.

Subir código Arduino al dispositivo IoT- Función principal es de generar el código de envío de datos al sistema web.

Subir código de conexión con FIWARE en el dispositivo Arduino- El participante debe utilizar el sistema Web para generar el código YAML que contiene la configuración del servidor de FIWARE.

Recepción de los datos del dispositivo - Función principal mostrar los datos enviados del dispositivo al sistema web.

Mostrar datos con Grafana- Función principal mostrar los datos almacenados por el Orion Context Broker utilizando Grafana.

Además de la evaluación de la funcionalidad del sistema, también se realizó una evaluación preliminar de la usabilidad y del tiempo requerido para completar cada una de las funciones que tiene el sistema web.

La experimentación se realizó en tres fases:

Fase 1. En esta fase se entregó a los participantes un documento de consentimiento informado donde se explica el objetivo de la experimentación, los procedimientos que se siguen durante las pruebas y se establece que su información personal no será difundida. Los participantes leyeron el documento de consentimiento y lo firmaron estableciendo su conformidad con el plan de pruebas.

- Fase 2. En esta fase se realiza la evaluación de la funcionalidad del sistema web desarrollado en esta tesis. La evaluación requiere del desarrollo de 7 tareas. Cada una de las tareas debe ser evaluada con un cuestionario que permita conocer la

funcionalidad del sistema y los tiempos utilizados para que el participante desarrolle cada una de las tareas.

- Fase 3. En esta etapa se realiza la evaluación preliminar de la interfaz gráfica del sistema web. La evaluación consiste en un cuestionario de tipo likert de 12 preguntas relacionadas a la interacción de los participantes con las interfaces del sistema Web con la finalidad de evaluar la usabilidad del sistema web.

A continuación, se presentan cada uno de los elementos relevantes en la experimentación: participantes, recursos utilizados, procedimientos de la experimentación y resultados.

## 5.2 Participantes

La selección de los participantes es una de las tareas esenciales durante un experimento, debido a que los participantes pueden contar con características específicas, que pueden ser determinantes para ser incluidos o excluidos para el experimento.

Para la experimentación del sistema Web presentado en la tesis se reclutaron 8 sujetos masculinos (con edades que oscilan entre 23 a 26 años) con diferentes conocimientos en cuanto al desarrollo en entornos IoT y en el armado y uso de dispositivos IoT. El criterio de inclusión fue (a) saber codificar en lenguaje C o C++, (b) saber utilizar una computadora, (c) que como mínimo estuvieran cursando la licenciatura o ingeniería en computación. Con relación a la escolaridad, 4 participantes se encuentran cursando un posgrado de maestría y 4 de los participantes se encuentran cursando la licenciatura en computación (Tabla 1). Los participantes fueron citados a un aula donde se encontraban la computadora con el Sistema Web desarrollado en la tesis y los dispositivos Arduino utilizados para el envío de datos. En este entorno se realizaron las 3 fases de la experimentación de forma continua, finalizando con la evaluación preliminar de la interfaz gráfica.

*Tabla 5-1 Participantes*

No. del participante	Edad	Maestría/ Otro
1	23	Maestría
2	23	Maestría
3	26	Maestría
4	23	Maestría
5	24	Licenciatura
6	22	Licenciatura
7	21	Licenciatura
8	24	Licenciatura

Los participantes fueron divididos en tres grupos de acuerdo a su experiencia con dispositivos, el manejo de agentes IoT y el envío de datos a la nube, sea de FIWARE o de otro proveedor de servicios en la nube. La división se realizó mediante un breve cuestionario donde se les preguntó a los participantes sobre su experiencia utilizando dispositivos IoT, el manejo y

configuración de agentes del Internet de las Cosas, la programación con Arduino y la configuración de la nube de FIWARE, obteniendo así tres grupos

- Grupo a) sin experiencia en IoT (tabla 5-2). Los participantes de este grupo no han trabajado con sensores, configuración de agentes IoT o envío de datos a la nube, este grupo consta de 2 participantes.

*Tabla 5-2 Grupo a de participantes*

Participante	Edad	Maestría/otro
<b>5</b>	24	Licenciatura
<b>7</b>	21	Licenciatura

- Grupo b) Experiencia media en IoT (tabla 5-3). Los participantes de este grupo han trabajado con dispositivos o con la plataforma FIWARE, pero no han configurado agentes IoT, este grupo consta de 3 participantes.

*Tabla 5-3 Grupo b de participantes*

Participante	Edad	Maestría/otro
<b>1</b>	23	Maestría
<b>6</b>	22	Licenciatura
<b>8</b>	24	Licenciatura

- Grupo c) Experiencia avanzada en IoT (tabla 5-4). Los participantes de este grupo han trabajado con sensores y con el envío de datos a la nube de la Plataforma FIWARE. Este grupo consta de 3 participantes.

*Tabla 5-4 Grupo c de los participantes*

Participante	Edad	Maestría/otro
<b>2</b>	23	Maestría
<b>3</b>	26	Maestría
<b>4</b>	23	Maestría

La tabla 5-5 presenta la organización de los grupos de participantes de acuerdo a su experiencia en el uso de IoT o envío de datos a la nube.

*Tabla 5-5 Grupos de participantes*

Participantes	Edad	Maestría/otro
<b>5 y 7</b>	21-24	Sin experiencia
<b>1,6 y 8</b>	22-24	Experiencia media
<b>1,3 y 4</b>	23-26	Experiencia avanzada

### 5.3 Recursos utilizados

En esta sección se presentan los recursos utilizados para la experimentación: como es el caso de los dispositivos electrónicos, el documento de consentimiento informado y los cuestionarios para obtener la retroalimentación de los participantes.

a) Dispositivos electrónicos:

Con el objetivo de reducir la problemática en la disponibilidad de dispositivos para la prueba se pusieron a disposición de los participantes un conjunto de dispositivos que se utilizaron en las sesiones de prueba. Los dispositivos que se pusieron a disposición para los participantes durante las sesiones de prueba son los siguientes:

3 placas Arduino genéricas Wemos D1.

1 sensor de agua -SL067

1 sensor de gas propano – MQ-5

1 sensor de pulso - MLM604362783

1 sensor de temperatura y humedad - Dht11A

1 acelerómetro -MPU6050

1 laptop Lenovo con Windows 10, IDE de Arduino en su última versión. En este sistema se instaló el Sistema Web desarrollado en esta tesis.

b) Consentimiento informado

El documento de consentimiento informado tiene el propósito de informar a los participantes sobre los objetivos de la fase de pruebas, el procedimiento para la ejecución de la experimentación, y, sobre todo, hacer explícita la aprobación de los sujetos para participar en el plan de pruebas y para dejar constancia de que los datos personales de los participantes serán anónimo. El documento de consentimiento se encuentra descrito en el Anexo A.

c) Cuestionarios de evaluación

El cuestionario de evaluación se realizó después de completar cada una de las tareas donde el participante expresaba la complejidad de realizar la tarea.

Fecha	
Nombre del participante	
Nombre de la herramienta	
Número de la tarea	

Instrucciones: marque con una X la opción que mejor se adecue a su respuesta en cada una de las preguntas.

1. ¿Qué tan fácil le resultó esta tarea?

Muy fácil                      Fácil                      Neutral                      Difícil                      Muy difícil

--	--	--	--	--

2. ¿Cómo considera las actividades que realizó para completar la tarea?

Muy fácil                      Fácil                      Neutral                      Difícil                      Muy difícil

--	--	--	--	--

3. ¿Para completar la tarea usted requirió de un aprendizaje previo o explicación detallada en el manejo de la herramienta?

SI

NO

--	--

4. ¿Al realizar la tarea, de qué manera identificar cada elemento de la pantalla y su funcionalidad?

Muy rápido                      Rápido                      Regular                      Lento                      Muy Lento

--	--	--	--	--

5. ¿Durante el tiempo en el que llevó a cabo la tarea, considera que el modo de respuesta de la herramienta fue práctico y entendible?

Muy fácil                      Fácil                      Neutral                      Difícil                      Muy difícil

--	--	--	--	--

6. ¿Durante el tiempo en el que llevó a cabo la tarea, requirió del manual de usuario?

Mucho                      Regular                      Poco                      Muy Poco                      Nada

--	--	--	--	--

El cuestionario de usabilidad se realizó después de completar todas las tareas, en este cuestionario los participantes contestaron cada una de las preguntas de acuerdo a su criterio al momento de utilizar el sistema web.

**Fecha**

**Nombre del participante**

<b>Nombre de la herramienta</b>	
<b>Número de la tarea</b>	

Instrucciones: responda a cada una de las afirmaciones del listado marcando con una X la casilla que se adapte a su respuesta, considerando que 1 está totalmente en desacuerdo y 5 totalmente de acuerdo.

	1	2	3	4	5
<b>El diseño y los colores son consistentes en la herramienta.</b>					
<b>Los íconos y elementos gráficos concuerdan con las funcionalidades de cada vista.</b>					
<b>No existen indicaciones o pantallas difíciles de entender dentro de la herramienta.</b>					
<b>El menú de navegación es de utilidad para ubicar cuáles son las secciones de la herramienta.</b>					
<b>El diseño y formato de presentación del menú es familiar y consistente en toda la herramienta.</b>					
<b>Las opciones de navegación entre las vistas de la herramienta son sencillas y visibles.</b>					
<b>La herramienta tiene una navegación accesible y es de fácil entendimiento.</b>					
<b>La forma de navegar entre las pantallas de la herramienta es familiar a la navegación de otras aplicaciones.</b>					
<b>Es posible cambiar entre funciones de la herramienta de forma accesible.</b>					
<b>Las funcionalidades que ofrece la herramienta son de fácil aprendizaje y utilización.</b>					
<b>Los tiempos de respuesta para cada tarea o función de la herramienta son los adecuados.</b>					
<b>No existe confusión en la visualización de los contenidos que muestra cada vista en pantalla.</b>					

El plan de pruebas considera que los participantes del experimento puedan realizar una serie de tareas con el sistema Web con el objetivo de enviar datos generados por sensores a la nube de FIWARE, pero utilizando el concepto de Agentes IoT como intermediarios. Los cuestionarios desarrollados para este plan de pruebas tienen el objetivo realizar una evaluación de la facilidad de uso de las funcionalidades del sistema, el tiempo que se utilizó para desempeñar cada una de las tareas y la usabilidad de cada uno de los módulos.

## 5.4 Procedimiento de la experimentación

La idea principal de la realización de las tareas de la experimentación es realizar la conexión entre dispositivos IoT y la plataforma FIWARE utilizando el sistema Web desarrollado en la tesis.

El sistema por su parte realiza, de forma automática, las tareas de transformación de los datos leídos de los sensores a entidades FIWARE en formato NGSI y acordes a un modelo de datos específico. A su vez el sistema realiza, también de forma automática, el envío de datos a la plataforma FIWARE.

## **Fase 1 Inicio de la experimentación**

En esta fase se entregó a los participantes un documento de consentimiento informado donde se explica el objetivo de la experimentación, los procedimientos que se siguen durante las pruebas y se establece que su información personal no será difundida. Los participantes leyeron el documento de consentimiento y lo firmaron estableciendo su conformidad con el plan de pruebas. Con el objetivo de clasificar a los participantes, se realizó un cuestionario donde se les solicitó que describiera sus conocimientos (en una escala de 0 al 10) con respecto a los siguientes aspectos: armado y programación de Arduino, programación con agentes IoT y desarrollo de aplicaciones inteligentes utilizando la plataforma FIWARE.

## **Fase 2 Ejecución de tareas utilizando el sistema Web**

En esta fase se realiza la evaluación de la funcionalidad del sistema web desarrollado en esta tesis. La evaluación requiere del desarrollo de 7 tareas. El objetivo de las tareas es permitir que el participante pueda conectar un dispositivo IoT al sistema Web, y posteriormente realizar las tareas de configuración que le permitan leer los datos de los sensores del dispositivo y realizar el envío de los datos estandarizados a la nube de FIWARE. Cada una de las tareas debe ser evaluada con un cuestionario que permita conocer la usabilidad del sistema y los tiempos utilizados para que el participante desarrolle cada una de las tareas.

A continuación, se describe cada una de las tareas que fueron realizadas por los participantes a través del sistema Web. Es importante comentar que las 7 tareas mostradas a continuación permiten cumplir los objetivos de la tesis que son:

- Desarrollar un módulo para la gestión de usuarios y dispositivos, con la finalidad de administrar la información de los usuarios y dispositivos, para su uso en el módulo de conexión.
- Desarrollar un módulo que genere el código de conexión a FIWARE con la configuración necesaria para levantar los servicios de FIWARE y la configuración del agente IoT.
- Desarrollar un módulo que genere el código Arduino, que se compila en el dispositivo IoT para el envío de datos al sistema web.
- Desarrollar un módulo que permita recibir y almacenar los datos enviados por el dispositivo.
- Desarrollar un submódulo que genere las entidades NGSI para el envío de datos.

- Tarea 1: Registro de usuario  
En esta tarea el participante debe realizar su registro de usuario en el sistema Web completando el formulario con sus datos. El participante debe proporcionar un

nombre de usuario, un correo electrónico y una contraseña que le permitirá iniciar sesión posteriormente en el sistema.

- **Tarea 2: Inicio de sesión en el sistema**  
En esta tarea el participante debe iniciar su sesión en el sistema Web con la información de correo electrónico y contraseña que proporcionó en la tarea 1.
- **Tarea 3: Registro de dispositivo en el sistema**  
En esta tarea el participante debe dar de alta un dispositivo IoT con información que le solicita un formulario desplegado en el sistema Web. La información que se provee para el dispositivo es nombre, dirección de Orion Context Broker de FIWARE, el protocolo de comunicación que utiliza el dispositivo, el tipo de Agente IoT y el tipo de lecturas, como es el caso de temperatura, humedad, luminosidad, etc. Esto se realiza con el objetivo de generar el código fuente necesario para realizar la conexión entre los dispositivos físicos y el sistema Web.
- **Tarea 4: Subir código Arduino al dispositivo IoT**  
En esta tarea el participante debe utilizar el sistema Web propuesto para subir el código generado por el sistema al dispositivo Arduino que se pretende conectar a la nube de FIWARE. Esto con el objetivo de que el dispositivo Arduino cuente con el código necesario para enviar los datos al sistema Web. La tarea consiste en seleccionar un dispositivo en el listado provisto por el sistema y descargar el código Arduino en ese dispositivo para permitir la comunicación en el sistema Web.
- **Tarea 5: Subir el código de conexión con FIWARE en el dispositivo Arduino**  
En esta tarea, el participante debe utilizar el sistema Web para generar el código fuente que permita la conexión entre el dispositivo y la plataforma FIWARE para enviar los datos de los sensores representados en entidades FIWARE. El participante debe utilizar el sistema Web para generar el código YAML que contiene la configuración del servidor de FIWARE. Este código debe ser descargado y compilado en el dispositivo para permitir la conexión y el envío de datos del sistema web a la plataforma FIWARE. Es importante comentar que este proceso requiere contar con servidor FIWARE previamente configurado. En el caso de nuestra experimentación se cuenta ya un servidor previamente configurado, pero en caso de no contar con un servidor se debe seguir los pasos descritos en: <https://tesis-jjfs.readthedocs.io/en/latest/servidor.html>
- **Tarea 6: Recepción de los datos de sensores**  
En esta tarea el participante debe ser capaz de visualizar los datos que han sido enviados del dispositivo IoT al sistema web, visualizar los datos de las lecturas de los

sensores acoplados al dispositivo IoT. Cada lectura debe contar con un identificador único además de la fecha y hora en la que los datos fueron recibidos por el sistema Web. Este proceso genera, de forma automática una base de datos en el manejador de datos históricos CreateDB.

- Tarea 7: Visualización de datos con Grafana:  
En esta tarea el participante debe ser capaz de visualizar los datos del dispositivo de la nube de FIWARE utilizando el módulo de Grafana. Cuando el servidor de FIWARE recibe los datos de los sensores que se encuentran ligados al sistema web, también se almacenan de forma automática en una base de datos histórica, denominada CreateDB, lo cual permite su despliegue de gráficas con esos datos utilizando el módulo Grafana.

### **Fase 3 Evaluación de la interfaz gráfica**

Una vez que se realizaron las 7 tareas utilizando el sistema Web, se realizó la evaluación preliminar de la interfaz gráfica del sistema web con la finalidad de evaluar la usabilidad del sistema. La evaluación consiste en un cuestionario de tipo likert de 12 preguntas ver Anexo C.

Una vez finalizadas las tareas y los cuestionarios se da por terminada la evaluación, la cual tuvo un promedio de 15 minutos.

## **5.5 Resultados de la evaluación preliminar de la facilidad de uso de cada una de las funcionalidades del sistema**

Con el objetivo de tener una primera aproximación para medir la facilidad de uso de cada funcionalidad del sistema se aplicó un cuestionario de funcionalidad (Anexo B) que tenía como objetivo evaluar la complejidad al momento de realizar las tareas de la experimentación.

El resultado de esta evaluación permitió determinar que todos los usuarios pudieron completar todas las funcionalidades esperadas. El nivel de dificultad de las tareas promedio se muestra a continuación:

Tarea 1 Registro de usuario. Nivel de dificultad (Fácil 8 participantes, Medio 0 participantes y Difícil 0 participantes).

Tarea 2 Inicio de sesión. Nivel de dificultad (Fácil 8 participantes, Medio 0 participantes y Difícil 0 participantes).

Tarea 3 Registro del dispositivo. Nivel de dificultad (Fácil 5 participantes, Medio 2 participantes y Difícil 1 participante).

Tarea 4 Subir el código Arduino al dispositivo IoT. Nivel de dificultad (Fácil 6 participantes, Medio 2 participantes y Difícil 0 participantes).

Tarea 5 Subir código de conexión con FIWARE. Nivel de dificultad (Fácil 8 participantes, Medio 0 participantes y Difícil 0 participantes).

Tarea 6 Recepción de los datos de los sensores. Nivel de dificultad (Fácil 7 participantes, Medio 1 participante y Difícil 0 participantes).

Tarea 7 Visualización de datos con Grafana. Nivel de dificultad (Fácil 8 participantes, Medio 0 participantes y Difícil 0 participantes).

Los resultados de esta evaluación permiten establecer que el sistema no solo realiza las tareas de manera exitosa, sino que también establece que las funciones son sencillas de seguir y realizar incluso para los usuarios menos experimentados en el área del Internet de las Cosas.

## 5.6 Resultados de la evaluación del tiempo utilizado para cada una de las tareas

La presente sección se compone de los resultados obtenidos a partir de la experimentación realizada: Los resultados que se tomaron en cuenta fueron el tiempo que les tomó a los participantes realizar cada tarea y las respuestas del cuestionario de usabilidad del sistema. A continuación, se muestran dos gráficas donde en la figura 5-1 se muestra el tiempo que se esperaba que les tomará a los participantes la ejecución de cada tarea es importante mencionar que el tiempo se basa en el tiempo promedio que se esperaba que los usuarios completaras las tareas tomando como base el tiempo que como autores del sistema utilizamos para las tareas además de agregar tiempo extra a cada tarea considerando la falta de experiencia de los usuarios en el uso del sistema, por ejemplo para el inicio de sesión se estimó un tiempo corto de (1.30 minutos) y un tiempo más grande para el registro de dispositivos ( 3.30 minutos) y en la figura 5-2 se muestra el tiempo promedio que les tomó a los participantes.

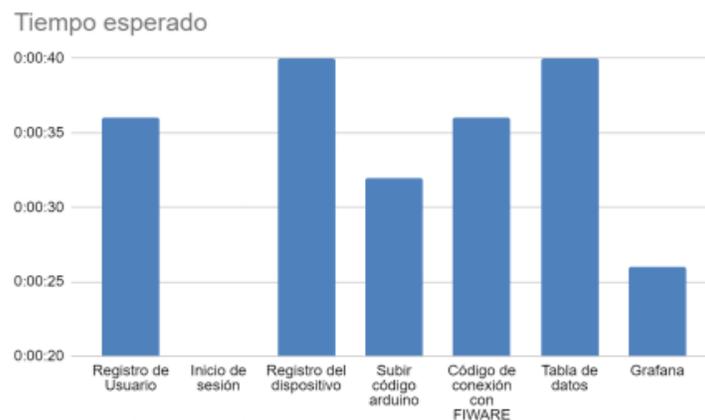


Figura 5-1 Gráfica del tiempo esperado de cada tarea

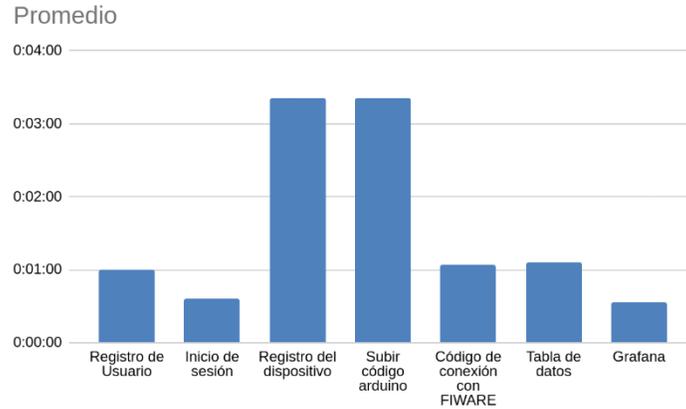


Figura 5-2 Gráfica del tiempo promedio de cada tarea

El análisis del tiempo utilizado permitió detectar que los tiempos utilizados en la práctica por los participantes del experimento fueron menores a los esperados.

Además del tiempo promedio general de todos los participantes, también se muestra el tiempo promedio que les tomó a cada uno de los grupos mencionados en la sección de participantes de este documento. A continuación, se muestran los resultados para cada uno de los grupos de participantes.

## 5.7 Resultados del grupo c participantes avanzados

La figura 5-3 muestra el tiempo promedio que le tomó al grupo C (Participante avanzado) para realizar las tareas donde se puede observar que las tareas que tomaron más tiempo fueron la tarea 3 “registro del dispositivo” y la tarea 4 “subir código Arduino”. Es importante mencionar que los participantes del grupo avanzado utilizaron más sensores que los otros dos grupos. La tabla 5-6 muestra los sensores utilizados por cada participante del grupo de experiencia avanzada.



Figura 5-3 Tiempo promedio del grupo C

Tabla 5-6 Participantes grupo c y sensores

Participantes	Sensores utilizados
2	1 acelerómetro- MPU6050 1 sensor de agua- SL067
3	1 sensor de pulso MLM6040362783
4	1 sensor de temperatura y humedad Dht11A

En las Figuras 5-4, 5-5 y 5-6 se muestran los tiempos que les tomo realizar a cada uno de los participantes del grupo C.

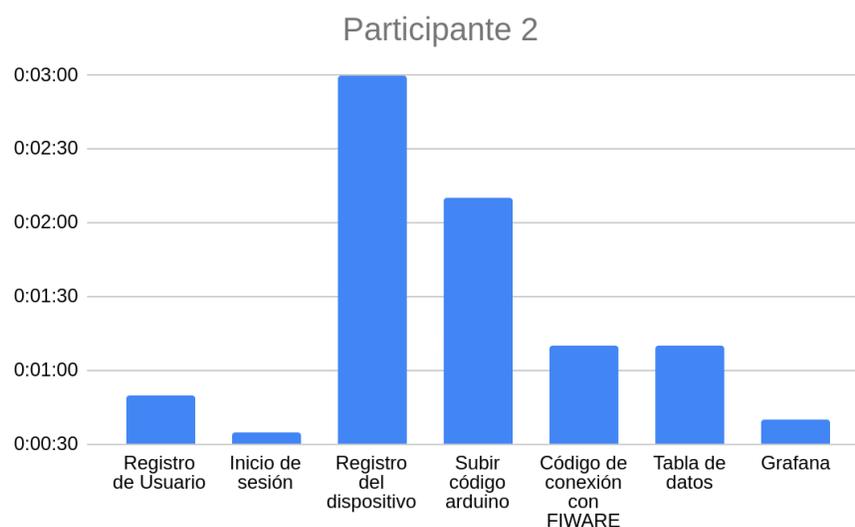


Figura 5-4 Gráfica del tiempo del participante 2

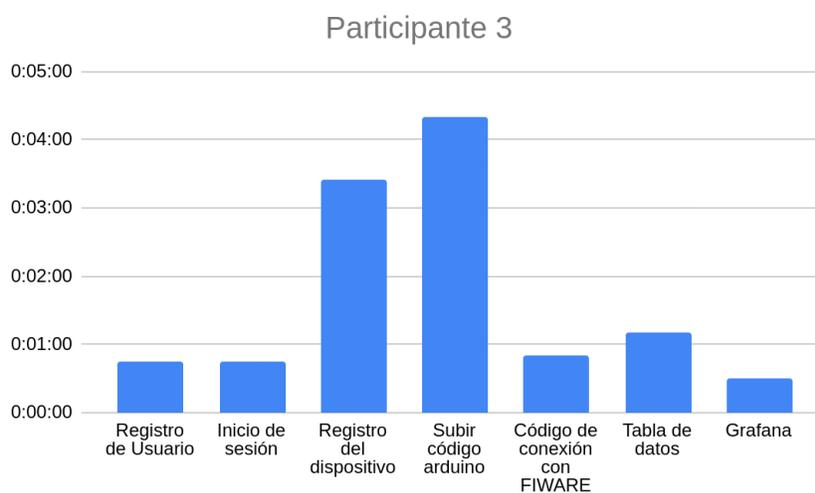


Figura 5-5 Gráfica del tiempo del participante 3

En la figura 5-6 se observa que el participante 4 obtuvo mejores tiempos en las tareas 3 y 4 que los participantes 3 y 2.

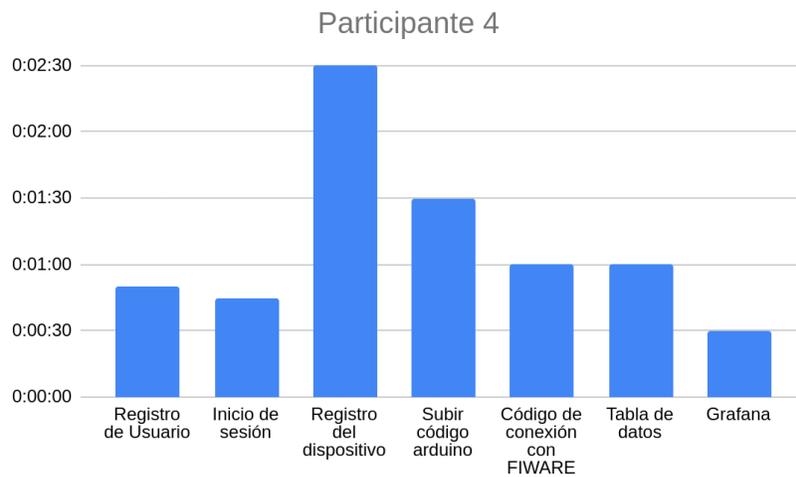


Figura 5-6 Grafica del tiempo del participante 4

## 5.8 Resultados del grupo b participantes con experiencia media

La figura 5-7 muestra el tiempo promedio que le tomó al Grupo B (experiencia media) para realizar las tareas y así como los resultados del grupo C, se puede observar que las tareas que tomaron más tiempo fueron la 3 y 4, además que con respecto a las otras tareas también tuvieron un tiempo más elevado, en la tabla 5-7 se observan los sensores que utilizaron los participantes del grupo B observando que utilizaron los participantes 1, 6 utilizaron menos sensores que los participantes 2 y 3 del grupo C.

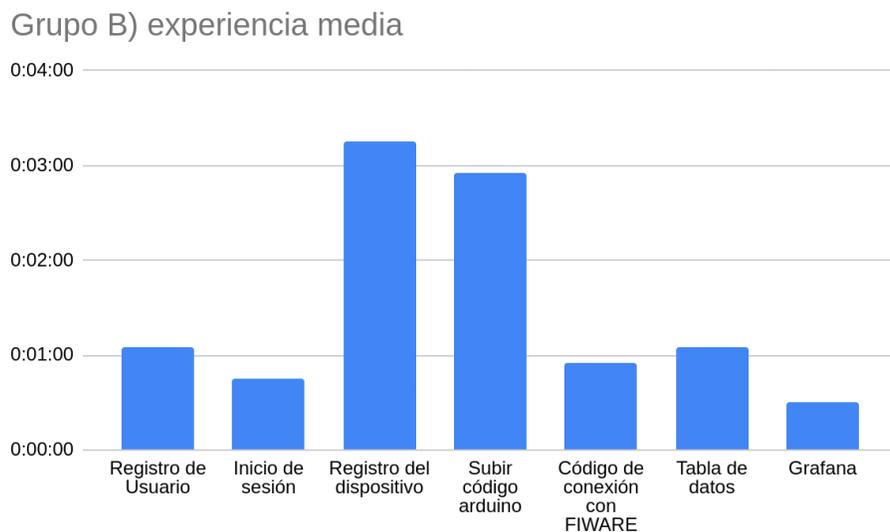


Figura 5-7 Gráfica del tiempo promedio del grupo b

Tabla 5-7 Participantes grupo b y sensores

Participantes	Sensores utilizados
1	1 sensor de agua- SL067
6	1 sensor de temperatura y humedad- Dht11A
8	1 sensor de gas propano- MQ-5 1 sensor de pulso MLM60436283

A continuación, se presentan los tiempos para cada uno de los participantes del grupo b.

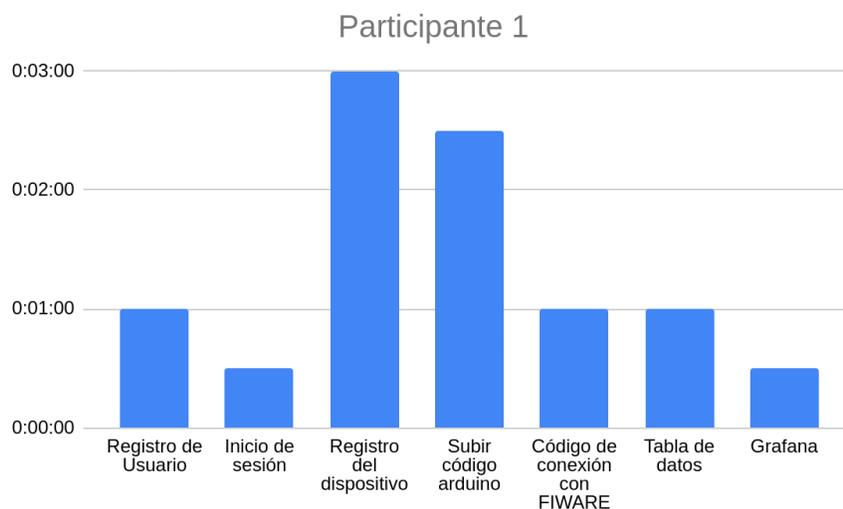


Figura 5-8 Gráfica del tiempo del participante 1

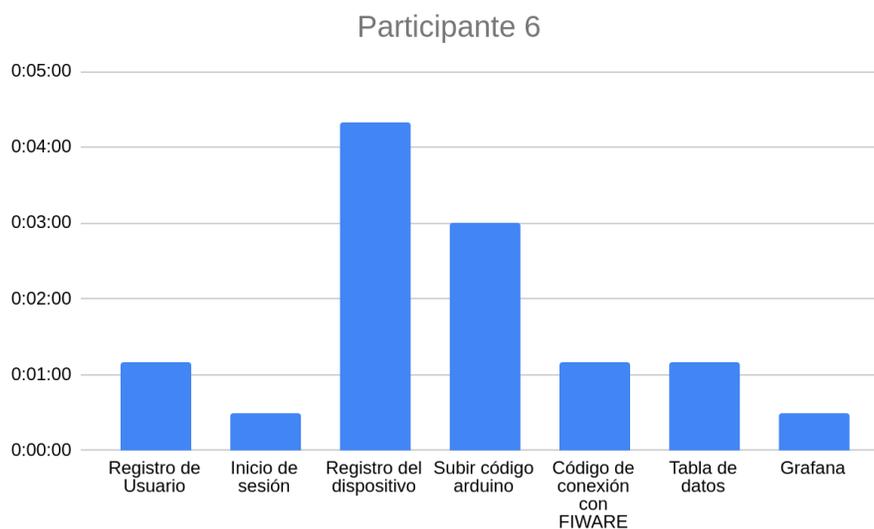


Figura 5-9 Gráfica del tiempo del participante 6

Los tiempos de los participantes 6 y 8 son muy parecidos, consideramos que esto se debe a que ambos tuvieron respuestas muy similares en el cuestionario sobre su conocimiento con respecto a los aspectos: armado y programación de Arduino, programación con agentes IoT

y desarrollo de aplicaciones inteligentes utilizando la plataforma FIWARE. Esto puede ser un indicativo de un nivel muy semejante de conocimientos de ambos participantes.

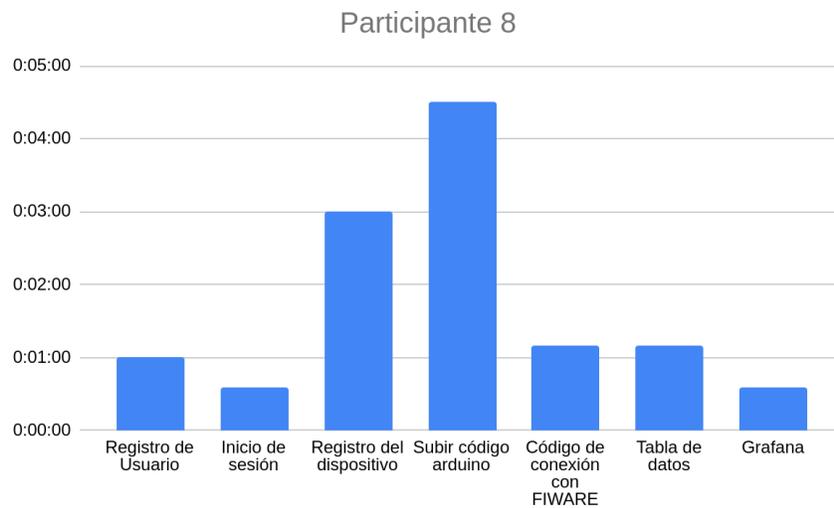


Figura 5-10 Gráfica del tiempo del participante 8

## 5.9 Resultados del grupo a participantes sin experiencia

La figura 5-11 muestra el tiempo promedio que le tomó al grupo A (Participantes sin experiencia) para realizar las tareas y así como los resultados del grupo C y B se puede observar que las tareas que tomaron más tiempo fueron la 3 y 4, teniendo tiempos un poco más elevados que los otros dos grupos. Los participantes de este grupo solo utilizaron 1 sensor

cada uno para realizar la experimentación debido a su poca experiencia con los dispositivos y el envío de datos, en la tabla 5-8 se muestra a los participantes y los sensores que eligieron.

Tabla 5-8 Participantes grupo a y sensores

Participantes	Sensores utilizados
5	1 sensor de agua- SL067
7	1 sensor de pulso- MLM604362783

Grupo A) sin experiencia

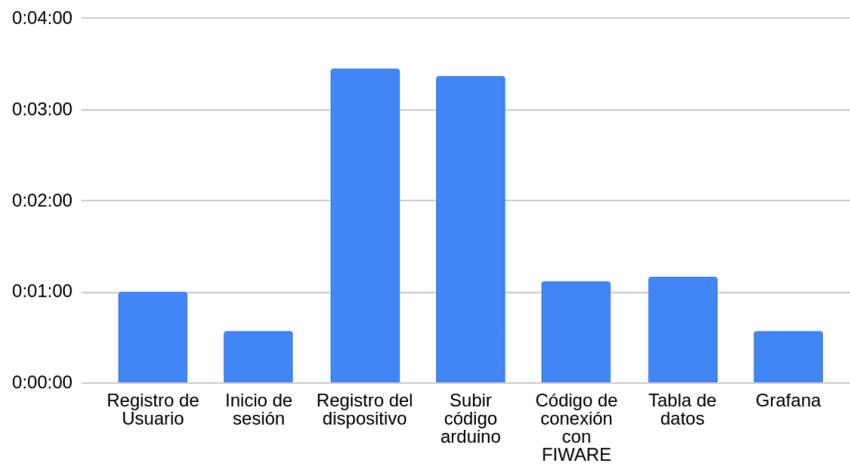


Figura 5-11 Gráfica del tiempo promedio del grupo a

A continuación, se presentan los tiempos para cada uno de los participantes del grupo A, donde en las figuras 5-12 y 5-13 se observa que los tiempos no eran muy diferentes a los del grupo B, debido a que, si bien este grupo de participantes no tenían mucha experiencia con el Internet de las Cosas, fueron los que utilizaron sólo un sensor para las pruebas, lo cual ocasionó que los tiempos de este grupo fueran cortos.

Participante 5

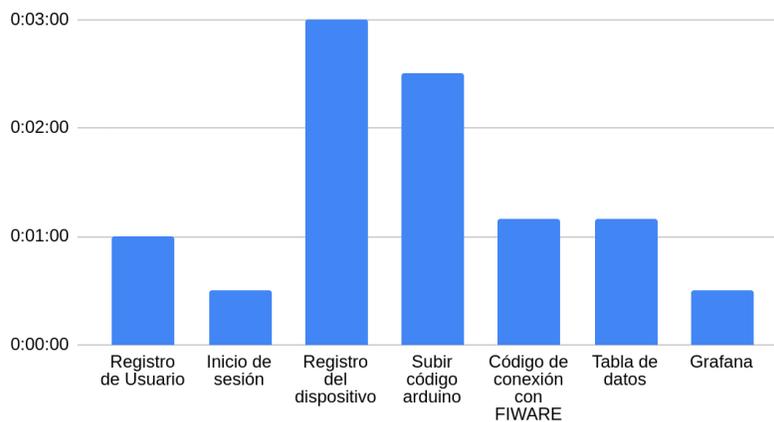


Figura 5-12 Gráfica del tiempo del participante 5

De igual manera que los tiempos de los participantes 5 y 7 sean muy parecidos se debe a que en el cuestionario sobre su conocimiento con respecto a los aspectos: armado y programación de Arduino, programación con agentes IoT y desarrollo de aplicaciones inteligentes utilizando la plataforma FIWARE, tuvieron respuestas muy similares.

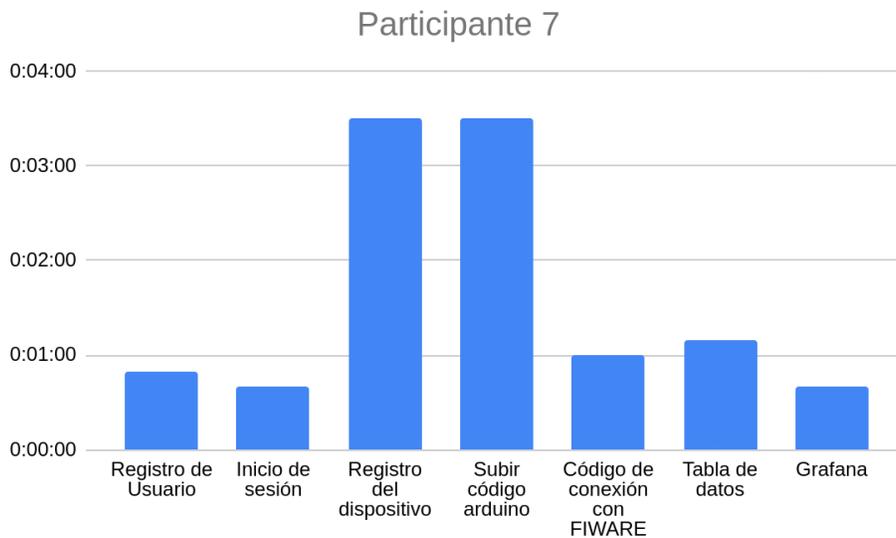


Figura 5-13 Gráfica del tiempo del participante 7

## 5.10 Resultados de la evaluación de la interfaz gráfica

Como último punto la gráfica 12 muestra los resultados del cuestionario de usabilidad (ver anexo c) que cada participante contestó al finalizar todas las tareas del experimento. Las preguntas de este cuestionario son las siguientes:

1. El diseño y los colores son consistentes en la herramienta.
2. Los íconos y elementos gráficos concuerdan con las funcionalidades de cada vista.
3. No existen indicaciones o pantallas difíciles de entender dentro de la herramienta.
4. El menú de navegación es de utilidad para ubicar cuáles son las secciones de la herramienta.
5. El diseño y formato de presentación del menú es familiar y consistente en toda la herramienta.
6. Las opciones de navegación entre las vistas de la herramienta son sencillas y visibles.
7. La herramienta tiene una navegación accesible y es de fácil entendimiento.
8. La forma de navegar entre las pantallas de la herramienta es familiar a la navegación de otras aplicaciones.
9. Es posible cambiar entre funciones de la herramienta de forma accesible.
10. Las funcionalidades que ofrece la herramienta son de fácil aprendizaje y utilización.
11. Los tiempos de respuesta para cada tarea o función de la herramienta son los adecuados.
12. No existe confusión en la visualización de los contenidos que muestra cada vista en pantalla.

A continuación, se presentan los resultados obtenidos para cada uno de los grupos de participantes.

La figura 5-14 muestra los resultados del grupo c.

### Grupo C) experiencia avanzada

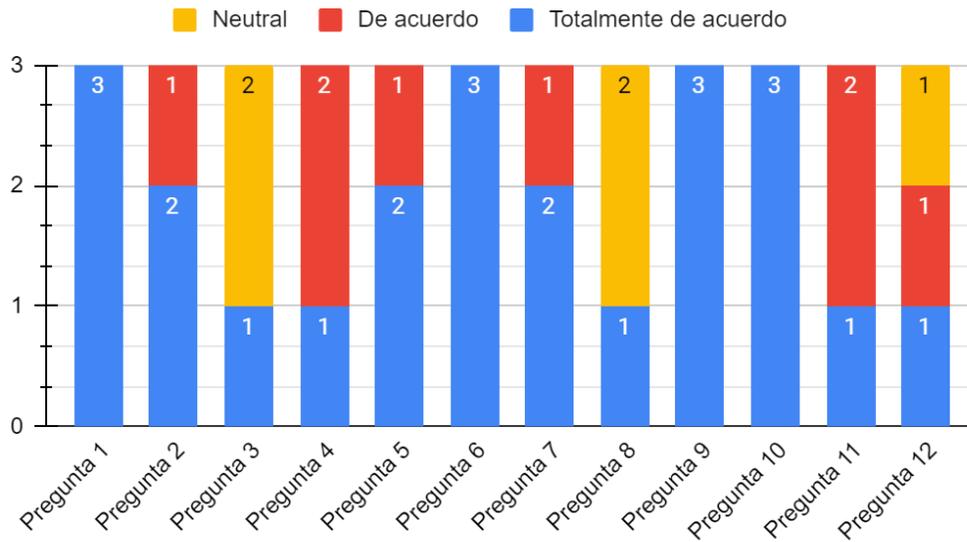


Figura 5-14 Resultados del cuestionario de usabilidad del grupo c

La figura 5-15 muestra los resultados del grupo b.

### Grupo B) experiencia media

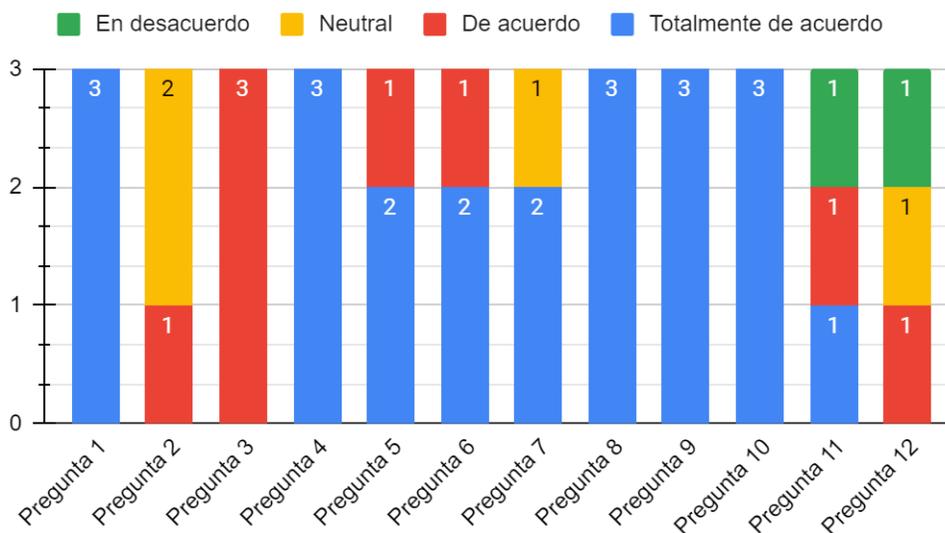


Figura 5-15 Resultados del cuestionario de usabilidad del grupo b

La figura 5-16 muestra los resultados del grupo a.

## Grupo A) Sin experiencia

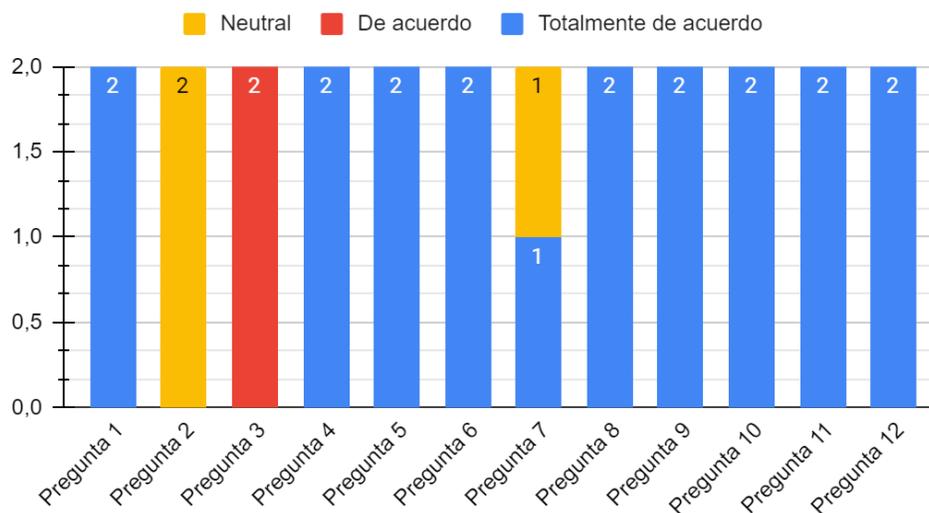


Figura 5-16 Resultados del cuestionario de usabilidad del grupo a

La figura 5-17 muestra un resumen de la cantidad de participantes que estaban en total acuerdo, de acuerdo, neutral, en desacuerdo o totalmente en desacuerdo en cada una de las preguntas del cuestionario.

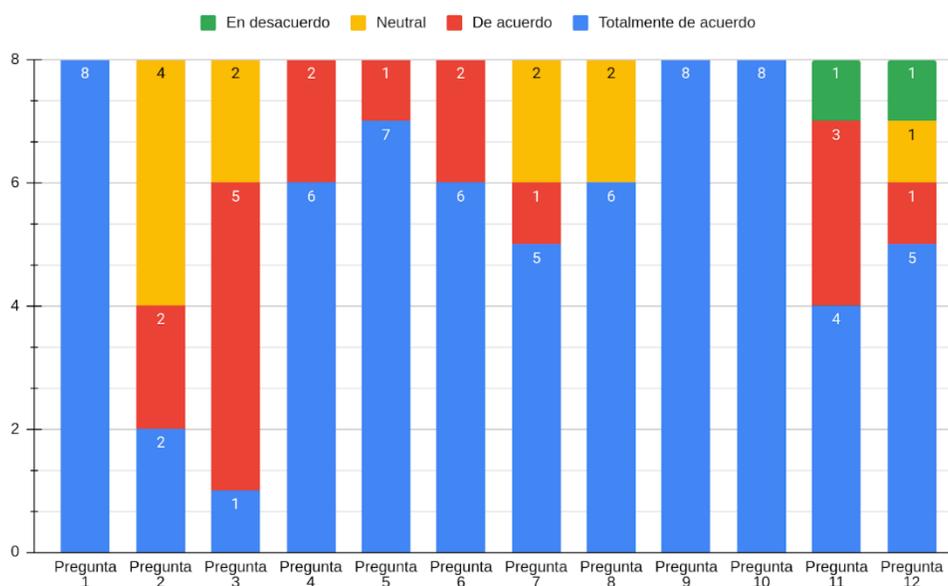


Figura 5-17 Resultados del cuestionario de usabilidad

# Capítulo 6

## Conclusiones y trabajo a futuro

---

En esta sección se presentan las conclusiones que se han generado a partir de este proyecto de investigación. Así también, se describen los trabajos futuros que se pueden derivar a partir de esta tesis.

## 6.1 Conclusiones

En la presente tesis se muestra que el sistema desarrollado permite de forma automática la conexión a la plataforma FIWARE usando los agentes del Internet de las Cosas.

En la presente tesis se muestra el sistema desarrollado permite enviar datos de forma automática a la plataforma FIWARE usando los protocolos CoAP y MQTT en conjunto con los agentes del Internet de las Cosas.

En la presente tesis se muestra que el sistema desarrollado permite g

El sistema fue sometido a una serie de pruebas, teniendo como resultado con una efectividad del 100% en todas sus funcionalidades.

El sistema desarrollado permite al usuario ahorrarse una cantidad de tiempo significativa al realizar la conexión y envió de datos a la plataforma FIWARE.

Este tipo de sistemas impulsa la transformación de ciudades tradicionales a ciudades inteligentes, las cuales tienen como obtuvo mejorar la calidad de vida de sus habitantes.

El sistema presentado en esta tesis al ser un intermediario entre los dispositivos y la plataforma FIWARE, permite manejar de forma segura los datos recopilados por los sensores de los dispositivos, para posteriormente enviarlos de forma segura y automática al Orion Context Broker.

## 6.2 Trabajo a futuro

Los trabajos futuros que se proponen para ampliar y/o mejorar este proyecto de investigación se listan a continuación:

- Desarrollar la opción para utilizar una raspberry pie además de los Arduinos.
- Implementar un módulo que permita compilar directamente en el sistema los códigos de conexión y envío de datos.
- Trabajar con más protocolos de comunicación y agentes IoT.
- Implementar los códigos de conexión y envío de datos directamente en dispositivo IoT si hacer uso del sistema web.



## Referencia

1. Aldein Mohammed, Z., Ali Ahmed, E.: Internet of Things Applications, Challenges and Related Future Technologies. World Scientific News Journal 67(2), pp. 126–148 (2017).
2. Google Cloud. (s.f). “Cloud Computing, servicios de alojamiento y APIs de Google Cloud | Google Cloud.” [Online].
3. Amazon Web Services. (s.f) “Plataforma AWS IoT – Amazon Web Services,” 2017. [Online]. <https://aws.amazon.com/es/IoT-platform/>.
4. IBM Cloud. (s.f). “About Watson IoT Platform”. [https://console.bluemix.net/docs/services/IoT/IoTplatform\\_overview.html#about\\_IoTplatform](https://console.bluemix.net/docs/services/IoT/IoTplatform_overview.html#about_IoTplatform).
5. Fiware-orion. (s.f) «Welcome to Orion Context Broker». [Online]. <https://fiwareorion.readthedocs.io/en/master/>
6. Ray, P. P. (2016). A survey of IoT cloud platforms. Future Computing and Informatics Journal, 1(1-2), 35-46. <https://www.sciencedirect.com/science/article/pii/S2314728816300149>.
7. Bliznakoff del Valle, D. J. (2014). IoT: Tecnologías, usos, tendencias y desarrollo futuro. <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/40044/6/dbliznakoffTFM0115memoria.pdf>.
8. FIWARE-ABOUT US, “ABOUT US» FIWARE,” 2016. [Online]. <https://www.fiware.org/about-us/>.
9. APRENDE FIWARE EN ESPAÑO, “Plataforma FIWARE,” 2016. [Online]. [https://fiware-training.readthedocs.io/es\\_MX/latest/ecosistemaFIWARE/plataformaFIWARE/](https://fiware-training.readthedocs.io/es_MX/latest/ecosistemaFIWARE/plataformaFIWARE/).
10. APRENDE FIWARE EN ESPAÑO, “Orion Context Broker” 2016. [Online]. [https://fiware-training.readthedocs.io/es\\_MX/latest/ecosistemaFIWARE/ocb/](https://fiware-training.readthedocs.io/es_MX/latest/ecosistemaFIWARE/ocb/).
11. FIWARE, “Tutorial time series data” 2018. [Online]. <https://github.com/FIWARE/tutorials.Time-Series-Data>.
12. FIWARE, “Tutorial time series data” 2018. [Online]. <https://github.com/FIWARE/tutorials.IoT-Agent>.
13. Alliance, O. M. (2018). Lightweight Machine to Machine Technical Specification: Core. Aug, 6, 21-28. Recuperado de [http://www.openmobilealliance.org/release/LightweightM2M/V1\\_1-20180710-A/OMA-TS-LightweightM2M\\_Core-V1\\_1-20180710-A.pdf](http://www.openmobilealliance.org/release/LightweightM2M/V1_1-20180710-A/OMA-TS-LightweightM2M_Core-V1_1-20180710-A.pdf).
14. CÁTEDRA TELEFÓNICA (UNIVERSIDAD DE EXTREMADURA). (2017, 24 octubre). FIWARE, el estándar que necesita el Internet de las Cosas. Transformación Digital del Sector Agroganadero. Recuperado 27 de junio de 2022, de <http://catedratelefonica.unex.es/fiware-el-estandar-que-necesita-el-internet-de-las-cosas/>.
15. Alliance, O. M. (2012). NGSI Context Management. OMA, OMA-TSNGSI Context Management-V1 CÁTEDRA TELEFÓNICA (UNIVERSIDAD DE EXTREMADURA). (2017, 24 octubre). *FIWARE, el estándar que necesita el Internet de las Cosas*. Transformación Digital del Sector Agroganadero. Recuperado 27 de junio de 2022, de <http://catedratelefonica.unex.es/fiware-el-estandar-que-necesita-el-internet-de-las-cosas/>.
16. Oliveira, C. T., Moreira, R., de Oliveira Silva, F., Miani, R. S., & Rosa, P. F. (2018, May). Improving security on IoT applications based on the FIWARE platform. In 2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA) (pp. 686-693). IEEE. Recuperado de <https://ieeexplore.ieee.org/document/8432306>.
17. Bull, P., Austin, R., Popov, E., Sharma, M., & Watson, R. (2016, August). Flow based security for IoT devices using an SDN gateway. In 2016 IEEE 4th international conference on future internet of things and cloud (FiCloud) (pp. 157-163). IEEE. Recuperado de <https://ieeexplore.ieee.org/document/7575858>.
18. Xu, X., Bessis, N., & Cao, J. (2013). An autonomic agent trust model for IoT systems. Procedia Computer Science, 21, 107-113. Recuperado de <https://www.sciencedirect.com/science/article/pii/S1877050913008090>.
19. Kato, T., Takahashi, H., & Kinoshita, T. (2017). Multiagent based autonomic and resilient service provisioning architecture for the Internet of Things. IJCSNS, 17(6), 36 [https://www.researchgate.net/publication/318338583\\_Multiagent-based\\_Autonomic\\_and\\_Resilient\\_Service\\_Provisioning\\_Architecture\\_for\\_the\\_Internet\\_of\\_Things](https://www.researchgate.net/publication/318338583_Multiagent-based_Autonomic_and_Resilient_Service_Provisioning_Architecture_for_the_Internet_of_Things).
20. John, T., & Vorbröcker, M. (2020, September). Enabling IoT connectivity for ModbusTCP sensors. In 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA) (Vol. 1, pp. 1339-1342). IEEE. Recuperado de <https://ieeexplore.ieee.org/document/9211999>.
21. Shimojo, S., & Nakagawa, I. (2018, July). Secure IoT Agent Platform with m-Cloud Distributed Statistical Computation Mechanism. In 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC) (Vol. 2, pp. 528-533). IEEE. Recuperado de <https://ieeexplore.ieee.org/document/8377917>.



## Anexo A

### CONSENTIMIENTO INFORMADO

Consentimiento informado entregado a los participantes de la experimentación, donde se explica de qué trata el experimento, así como de las recompensas por participar y el cómo se utilizarán los datos generados, así como su anonimato.

### CONSENTIMIENTO INFORMADO

Tiempo de lectura: 3 minutos aprox.

#### Introducción

En el Centro Nacional de Investigación y desarrollo Tecnológico, estamos llevando a cabo herramientas computacionales que exploren nuevas tecnologías como el Internet de las Cosas, y FIWARE. En este estudio forma parte de la tesis de maestría denominada: Sistema para la conexión automática de sensores IoT a la plataforma FIWARE, la cual tiene como objetivo crear una herramienta que permite a dispositivos conectarse y enviar datos a la nube de FIWARE de manera más sencilla que el esquema actual de la plataforma FIWARE.

#### Procedimiento

Requerimos su colaboración para participar en un conjunto de evaluaciones de funcionalidad y usabilidad del sistema web. Todo el procedimiento tiene una duración aproximada de 20 minutos. Y está organizada en dos etapas:

- Fase 1. A los participantes se les entregó un documento de consentimiento informado donde se explica la experimentación, los procedimientos y que su información personal no será difundida el cual leyeron y firmaron
- Fase 2. Evaluación de funcionalidad del sistema web, esta etapa tiene como objetivo evaluar las funcionalidades de la herramienta. La evaluación requiere del desarrollo de 7 tareas, cada una de ellas debe ser evaluada con un breve cuestionario, que permita conocer la usabilidad del sistema.
- Fase 3. Evaluación de la interfaz gráfica del sistema web, esta etapa tiene como objetivo evaluar la interfaz gráfica de la herramienta. La evaluación consiste en un cuestionario de tipo Likert de 13 preguntas

#### Riesgos y beneficios

No existen riesgos asociados a los participantes de este estudio. Su participación en este estudio será de gran ayuda en el desarrollo de esta tesis de maestría. Además, no se cuenta con beneficios adicionales para alentarlos a participar.

#### Derechos

Su participación es voluntaria. Usted tiene derecho a abstenerse de participar o retirarse de las pruebas en cualquier momento, sin que exista ningún problema.

#### Alcance del anonimato y la confidencialidad

Los resultados obtenidos de este estudio se mantendrán estrictamente confidenciales. La información recopilada se utilizará únicamente con fines de la investigación, por lo que se mantendrá la total confidencialidad de los datos personales de los participantes.

#### Compensación

Su participación es voluntaria y no remunerada.

Las pruebas consisten en la conexión de los dispositivos a la plataforma FIWARE, las pruebas están compuestas por dos partes, la primera tiene como objetivo probar las funcionalidades de la herramienta y está compuesta por 7 tareas y tendrán una duración de 15 minutos.

La segunda es el cuestionario de usabilidad, que tiene como objetivo evaluar el qué tan fácil es el utilizar la herramienta y está compuesto por 13 preguntas y tendrá una duración de 5 minutos

#### Responsabilidades del participante y permiso

Yo voluntariamente estoy de acuerdo en participar en este estudio, y no conozco ninguna razón por la que no pueda participar. He leído y comprendo el consentimiento informado y las condiciones de este estudio. He tenido todas mis preguntas respondidas. Por la presente reconozco lo anterior y doy mi consentimiento voluntario para participar en este estudio. Si participo, puedo retirarme en cualquier momento sin penalización. Estoy de acuerdo en cumplir con las reglas de este estudio.

\_\_\_\_\_  
\_Nombre

\_\_\_\_\_  
Firma

\_\_\_\_\_  
Fecha

Número de documento de identificación oficial: \_\_\_\_\_

## Anexo B

El cuestionario de evaluación se realizó después de completar cada una de las tareas donde el participante expresaba la complejidad de realizar la tarea.

Fecha	
Nombre del participante	
Nombre de la herramienta	
Número de la tarea	

Instrucciones: marque con una X la opción que mejor se adecue a su respuesta en cada una de las preguntas.

1. ¿Qué tan fácil le resultó esta tarea?

Muy fácil

Fácil

Neutral

Difícil

Muy difícil

--	--	--	--	--

2. ¿Cómo considera las actividades que realizó para completar la tarea?

Muy fácil                  Fácil                  Neutral                  Difícil                  Muy difícil

--	--	--	--	--

3. ¿Para completar la tarea usted requirió de un aprendizaje previo o explicación detallada en el manejo de la herramienta?

SI

NO

--	--

4. ¿Al realizar la tarea, de qué manera identificar cada elemento de la pantalla y su funcionalidad?

Muy rápido                  Rápido                  Regular                  Lento                  Muy Lento

--	--	--	--	--

5. ¿Durante el tiempo en el que llevó a cabo la tarea, considera que el modo de respuesta de la herramienta fue práctico y entendible?

Muy fácil                  Fácil                  Neutral                  Difícil                  Muy difícil

--	--	--	--	--

6. ¿Durante el tiempo en el que llevó a cabo la tarea, requirió del manual de usuario?

Mucho                  Regular                  Poco                  Muy Poco                  Nada

--	--	--	--	--

## Anexo C

El cuestionario de usabilidad se realizó después de completar todas las tareas, en este cuestionario los participantes contestaron cada una de las preguntas de acuerdo a su criterio al momento de utilizar el sistema web.

Fecha

<b>Nombre del participante</b>	
<b>Nombre de la herramienta</b>	
<b>Número de la tarea</b>	

Instrucciones: responda a cada una de las afirmaciones del listado marcando con una X la casilla que se adapte a su respuesta, considerando que 1 está totalmente en desacuerdo y 5 totalmente de acuerdo.

	1	2	3	4	5
<b>El diseño y los colores son consistentes en la herramienta.</b>					
<b>Los íconos y elementos gráficos concuerdan con las funcionalidades de cada vista.</b>					
<b>No existen indicaciones o pantallas difíciles de entender dentro de la herramienta.</b>					
<b>El menú de navegación es de utilidad para ubicar cuáles son las secciones de la herramienta.</b>					
<b>El diseño y formato de presentación del menú es familiar y consistente en toda la herramienta.</b>					
<b>Las opciones de navegación entre las vistas de la herramienta son sencillas y visibles.</b>					
<b>La herramienta tiene una navegación accesible y es de fácil entendimiento.</b>					
<b>La forma de navegar entre las pantallas de la herramienta es familiar a la navegación de otras aplicaciones.</b>					
<b>Es posible cambiar entre funciones de la herramienta de forma accesible.</b>					
<b>Las funcionalidades que ofrece la herramienta son de fácil aprendizaje y utilización.</b>					
<b>Los tiempos de respuesta para cada tarea o función de la herramienta son los adecuados.</b>					
<b>No existe confusión en la visualización de los contenidos que muestra cada vista en pantalla.</b>					