



ESPE

**UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA**

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

**PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL**

AUTOR: ALVARO SOTALIN, GABRIELA ALEXANDRA

**TEMA: DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO
CONTROLADO Y MONITORIZADO DE FORMA REMOTA UTILIZANDO EL
ESTÁNDAR DE TECNOLOGÍA INALÁMBRICA ZIGBEE Y LA PLATAFORMA IDIGI
DEVICE CLOUD.**

DIRECTOR: ING. GORDILLO, RODOLFO

CODIRECTOR: ING. ORTIZ, HUGO

SANGOLQUÍ, AGOSTO 2014

Certificado de tutoría

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

CERTIFICADO

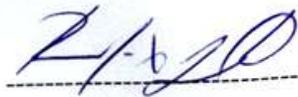
Ing. Rodolfo Gordillo

Ing. Hugo Ortiz

CERTIFICAN

Que el trabajo titulado "DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO CONTROLADO Y MONITORIZADO DE FORMA REMOTA UTILIZANDO EL ESTÁNDAR DE TECNOLOGÍA INALÁMBRICA ZIGBEE Y LA PLATAFORMA DE IDIGI DEVICE CLOUD", realizado por Gabriela Alexandra Alvaro Sotalin, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la institución, en el Reglamento de Estudiantes de la Universidad de las Fuerzas Armadas-ESPE.

Sangolquí, Agosto de 2014



Ing. Rodolfo Gordillo
DIRECTOR



Ing. Hugo Ortiz
CODIRECTOR

Declaración de responsabilidad

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

DECLARACIÓN DE RESPONSABILIDAD

GABRIELA ALEXANDRA ALVARO SOTALIN

DECLARO QUE:

El proyecto de grado denominado "DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO CONTROLADO Y MONITORIZADO DE FORMA REMOTA UTILIZANDO EL ESTÁNDAR DE TECNOLOGÍA INALÁMBRICA ZIGBEE Y LA PLATAFORMA DE IDIGI DEVICE CLOUD", ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme a las fuentes que se incorporan en la biografía.

Consecuentemente este trabajo es de mi autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Agosto de 2014



Gabriela Alexandra Alvaro Sotalin

Autorización de publicación

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

AUTORIZACIÓN

GABRIELA ALEXANDRA ALVARO SOTALIN

Autorizo a la Universidad de las Fuerzas Armadas-ESPE la publicación, en la biblioteca virtual de la Institución, del trabajo "DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO CONTROLADO Y MONITORIZADO DE FORMA REMOTA UTILIZANDO EL ESTÁNDAR DE TECNOLOGÍA INALÁMBRICA ZIGBEE Y LA PLATAFORMA DE IDIGI DEVICE CLOUD", cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.

Sangolqui, Agosto de 2014



Gabriela Alexandra Alvaro Sotalin

DEDICATORIA

A Dios por derramar sus bendiciones sobre mí y llenarme de fuerzas para seguir adelante y vencer todos los obstáculos que se presentaron durante el camino.

A mis padres por brindarme todo el amor, la comprensión y el apoyo incondicional a lo largo de mis estudios universitarios.

A mis hermanos que siempre me alentaron y me empujaron adelante para no dejarme caer en los momentos difíciles.

Gabriela Alvaro S.

AGRADECIMIENTO

A Dios por ser mi guía y mi apoyo en los momentos más difíciles, tu luz siempre me ilumina y me da sabiduría para tomar decisiones importantes y permitirme día a día luchar por conseguir mis objetivos.

A mis queridos padres por su amor incondicional, su continuo apoyo, gracias por brindarme sus sabios consejos y por ser un excelente ejemplo de vida a seguir.

A mis hermanos por permanecer a mi lado en todo momento, por su cariño, apoyo y comprensión.

A mi director y codirector del proyecto por la orientación y ayuda que me brindaron para la realización de este proyecto.

Y por último a todos mis familiares y amigos que de una forma u otra me brindaron su apoyo incondicional y me motivaron a luchar con más energía.

Gabriela Alvaro S.

TABLA DE CONTENIDO

CAPÍTULO 1	1
INTRODUCCIÓN	1
1.1 ANTECEDENTES	1
1.2 JUSTIFICACIÓN E IMPORTANCIA	3
1.3 ALCANCE DEL PROYECTO.....	4
1.4 OBJETIVOS	5
1.4.1 General	5
1.4.2 Específicos	6
CAPÍTULO 2	7
MARCO TEÓRICO	7
2.1 DOMÓTICA	7
2.1.1 Prestaciones de un sistema domótico	8
2.1.2 Componentes básicos de un sistema domótico	9
2.1.3 Tipos de arquitecturas de un sistema domótico	11
2.1.4 Medios de transmisión de los sistemas domóticos	13
2.2 COMUNICACIONES INALÁMBRICAS	13
2.2.1 Clasificación de las redes inalámbricas.....	14
2.2.2 Red inalámbrica de sensores	16
2.3 PROTOCOLO ZIGBEE Y ESTÁNDAR IEEE 802.15.4	16
2.3.1 Introducción.....	16
2.3.2 El estándar IEEE 802.15.4	16
2.3.2.1 Capa física.....	18
2.3.2.2 Control de acceso.....	18
2.3.3 ZigBee.....	19
2.3.3.1 Características del protocolo ZigBee para domótica	20

2.3.3.2	Tipos de dispositivos.....	21
2.3.3.3	Funcionalidad.....	23
2.3.3.4	Topología.....	24
2.3.3.5	Aplicaciones.....	25
2.4	CLOUD COMPUTING.....	25
2.4.1	Modelos de servicio.....	26
2.5	DESCRIPCIÓN DE LA PLATAFORMA DIGI.....	28
2.5.1	Digi International® Inc.....	28
2.5.2	Nube de dispositivos Etherios.....	29
2.5.2.1	Comunicación con la nube.....	29
2.6	DESCRIPCIÓN DE LOS DISPOSITIVOS XBEE UTILIZADOS.....	30
2.6.1	ConnectPort X4 Gateway.....	31
2.6.2	Xbee Wall Router.....	32
2.6.3	Adaptador Xbee DIO.....	34
2.7	SENSORES Y ACTUADORES (Pallás).....	36
2.7.1	Sensores.....	36
2.7.1.1	Sensor PIR DC-SS502V110.....	37
2.7.1.2	Sensor magnético.....	39
2.7.2	Actuadores.....	40
2.7.2.1	Relé.....	40
CAPÍTULO 3.....		42
DISEÑO E IMPLEMENTACIÓN DEL SISTEMA.....		42
3.1	REQUERIMIENTOS DEL SISTEMA.....	42
3.2	DISEÑO DEL SISTEMA.....	43
3.2.1	Funcionalidad del sistema.....	43
3.2.2	Arquitectura del sistema.....	45
3.2.3	Diseño de la red.....	46
3.2.4	Diseño de los circuitos de acondicionamiento.....	48
3.3	CONFIGURACIÓN DE LOS DISPOSITIVOS DE LA RED.....	49

3.4	DEASARROLLO DEL SOFTWARE.....	52
3.4.1	Software de gestión remota	52
3.4.2	Configuración de los controladores de los dispositivos.....	53
3.4.3	Ejecución de la aplicación DIA	60
3.4.4	Conexión de los dispositivos Xbee al servidor	61
3.4.5	Almacenamiento de datos.....	62
3.5	DESARROLLO DE LA INTERFAZ DE USUARIO	64
3.5.1	Descripción de la plataforma Google App Engine	64
3.5.2	Creación de cuenta en Google App Engine.....	65
3.5.3	Diseño de la aplicación web.....	67
3.5.4	Programación de la aplicación web	72
3.5.4.1	Lenguajes de programación utilizados	72
3.5.4.2	Archivos de la aplicación	73
3.6	IMPLEMENTACIÓN DEL PROTOTIPO	92
	CAPÍTULO 4.....	95
	PRUEBAS Y RESULTADOS	95
4.1	MEDICIÓN DE COBERTURA Y SENSIBILIDAD DE LOS NODOS	95
4.2	TIEMPO DE RESPUESTA DE LA RED	99
4.3	CONSUMO DE ENERGÍA	100
4.4	PRUEBA GENERAL DEL PROTOTIPO	102
	CAPÍTULO 5.....	106
	CONCLUSIONES Y RECOMENDACIONES.....	106
5.1	CONCLUSIONES.....	106
5.2	RECOMENDACIONES.....	108
	GLOSARIO.....	110
	BIBLIOGRAFÍA.....	112

ÍNDICE DE FIGURAS

Figura 1. Servicios de un sistema doméstico.....	9
Figura 2. Tipos de sensores.	10
Figura 3. Arquitectura centralizada.....	12
Figura 4. Arquitectura distribuida.....	12
Figura 5. Tipos de redes inalámbricas.	15
Figura 6. Capas del estándar IEEE 802.15.4.....	17
Figura 7. Capas añadidas por ZigBee.....	20
Figura 8. Tipos de dispositivos ZigBee.....	22
Figura 9. Tipos de topologías.	24
Figura 10. Modelos de servicio.....	26
Figura 11. ConnectPort X4 Gateway.....	32
Figura 12. Xbee Wall Router.....	33
Figura 13. Adaptador Xbee DIO.....	34
Figura 14. Sensor PIR DC-SS502V110.....	37
Figura 15. Diagrama del rango de detección.....	39
Figura 16. Sensor magnético.....	40
Figura 17. Relé SRD-05VDC-SL-C.....	41
Figura 18. Diagrama de bloques del prototipo.....	43
Figura 19. Arquitectura del sistema.....	46
Figura 20. Red tipo estrella.....	47
Figura 21. Circuito de acondicionamiento del actuador.....	49
Figura 22. Configuración del dispositivo coordinador.....	50
Figura 23. Configuración del dispositivo final.....	51
Figura 24. Nodos asociados a la red.....	51
Figura 25. Creación proyecto DIA.....	52
Figura 26. Configuración proyecto DIA.....	53
Figura 27. Controlador Xbee Wall Router.....	54
Figura 28. Controlador del adaptador Xbee DIO.....	55
Figura 29. Controlador del canal transform0.cid.....	57
Figura 30. Controlador del canal transform1.cin.....	58
Figura 31. Controlador del canal transform2.ddi.....	59
Figura 32. Presentación web de la aplicación DIA.....	60
Figura 33. Interfaz Nube de dispositivos.....	61
Figura 34. Dispositivos conectados a la red Xbee.....	62
Figura 35. Solicitud HTTP POST.....	63

Figura 36. Fichero almacén de datos	63
Figura 37. Interfaz de la plataforma Google App Engine	65
Figura 38. Consola de Administración de las aplicaciones GAE	66
Figura 39. Página para la creación de aplicaciones en GAE	67
Figura 40. Interfaz ingreso de contraseña	68
Figura 41. Interfaz página principal	69
Figura 42. Interfaz pestaña Monitoreo	70
Figura 43. Interfaz pestaña Control	71
Figura 44. Interfaz página de Error	72
Figura 45. Archivos de la aplicación	74
Figura 46. Archivo app.yaml	75
Figura 47. Archivo cron.yaml	76
Figura 48. Diagrama de flujo de la clase Manual	80
Figura 49. Diagrama de flujo de la clase Automa	81
Figura 50. Diagrama de flujo de la clase Automatico	82
Figura 51. Diagrama de flujo de la clase Modos	83
Figura 52. Diagrama de flujo de la clase Alarma	84
Figura 53. Diagrama de flujo de la clase LoggingHandler	86
Figura 54. Diagrama de flujo de la clase MonitoreoPage función post	87
Figura 55. Diagrama de flujo de la clase ControlPage	88
Figura 56. Placa del control de iluminación	92
Figura 57. Placa del detector de intrusión	93
Figura 58. Módulo del prototipo	94
Figura 59. Módulo del prototipo terminado	94
Figura 60. Medición del parámetro RSSI en la ventana Radio Range Test	96
Figura 61. Medición del parámetro LQI	98
Figura 62. Resultados del parámetro LQI	98
Figura 63. Prueba de control de Iluminación por modos de configuración	103
Figura 64. Prueba del control Automático de iluminación	103
Figura 65. Visualización de valores de temperatura y luminosidad	104
Figura 66. Activación del detector de intrusión e ingreso de email	105
Figura 67. Activación de los sensores y ventana de alarma	105

ÍNDICE DE TABLAS

Tabla 1. Configuración de pines del adaptador Xbee DIO	35
Tabla 2. Especificaciones y características de las entradas y salidas	35
Tabla 3. Configuración de pines	38
Tabla 4. Canales utilizados en la variable cid	56
Tabla 5. Canales utilizados en la variable ddi	59
Tabla 6. Resultados del parámetro RSSI	97
Tabla 7. Resultados del tiempo de respuesta en red local	99
Tabla 8. Resultados del tiempo de respuesta en red externa	100
Tabla 9. Resultados del consumo de corriente en los nodos	101

RESUMEN

El presente proyecto tiene por finalidad realizar el diseño y la implementación del prototipo de un sistema domótico, controlado y monitorizado de forma remota utilizando el estándar de tecnología inalámbrica ZigBee y la plataforma de nube pública. Los servicios domóticos elegidos para este prototipo son: confort y seguridad. En lo referente a aplicaciones de confort se tiene monitorización de temperatura y control de iluminación. El control de iluminación se lleva a cabo mediante dos configuraciones: modalidad día y modalidad noche. Además se añade un control manual y automático de la luz. En aplicación de seguridad se tiene el detector de intrusión, si se activa, el sistema emitirá una alarma y a la vez se enviará un correo electrónico notificando la activación de la misma. El prototipo está formado por una red de sensores inalámbricos con cuatro nodos Xbee, se utiliza el kit iDigi® ZigBee Gateway de la marca Digi, que cuenta con dos dispositivos finales, un coordinador y un router. Para recopilar los datos y enviarlos a la nube de dispositivos Etherios se utiliza el software DIA de Digi ESP para Python. También se desarrolla una aplicación web cliente en Google App Engine que maneja de forma remota, a través de Internet, los dispositivos conectados a la red, permitiendo en tiempo real, visualizar los datos de los sensores y controlar los dispositivos finales.

PALABRAS CLAVE:

Domótica, WSN, ZigBee, Digi, Cloud Computing, Google App Engine

ABSTRACT

This project has by purpose to design and implement the prototype of a home automation system, controlled and monitored remotely using the standard of ZigBee wireless technology and the public cloud platform-as-a-service (PaaS). Home automation services chosen for this prototype are: comfort and security. In regard of comfort applications have temperature monitoring and lighting control. The lighting control is carried out through two configurations: day mode and night mode. Also added a manual and automatic lighting control. In security applications has the intrusion detector, if enabled, the system will issue an alarm and at the same time will send a notification email. The prototype consists of a wireless sensor network with four Xbee nodes, used iDigi® ZigBee Gateway Development kit of Digi, that has two end devices, a coordinator and router. To collect data and send them to the device cloud by Etherios uses Digi ESP for Python software. Also it has developed a client web application in Goggle App Engine to manage remotely, through Internet, devices connected to the network, allowing in real time display data from the sensors and control the end devices.

KEY WORDS:

Home automation, WSN, ZigBee, Digi, Cloud Computing, Google App engine

CAPÍTULO 1

INTRODUCCIÓN

1.1 ANTECEDENTES

El nacimiento de la Domótica es un proceso evolutivo que comenzó con las redes de control de los edificios inteligentes y se ha ido adaptando a las necesidades propias de la vivienda.

En la década de 1980 aparecen los subsistemas de automatización de intrusión, seguridad e iluminación, evidenciando un grado de integración entre sus componentes. Es así que en 1984 se lanza el Proyecto "Smart House", cuyo principio es la utilización de un cable unificado que sustituye a los distintos sistemas que pueden existir en una vivienda actual: electricidad, antenas, periféricos de audio-video, teléfono, alarmas, etc., dando de esta manera el primer paso hacia un hogar digital y la domótica (Vasco, Sintés, & Lagos, 2003).

En los últimos años el mundo de las Comunicaciones Inalámbricas experimentó la aparición de un gran número de estándares, durante el 2005 se

han consolidado las aplicaciones de estos nuevos estándares, mientras que algunos de ellos todavía siguen avanzando en su proceso de desarrollo como Wi-Fi, Bluetooth, y otras venideras WiMax, USB inalámbrico, etc. Pero el problema es que estas tecnologías no satisfacen los requerimientos de la Domótica.

ZigBee es una alianza, de más de 100 empresas, con el objetivo de auspiciar el desarrollo e implantación de una tecnología inalámbrica de bajo coste (Domodesk). ZigBee utiliza una norma del IEEE, lo cual hace factible que los dispositivos se comuniquen e interoperen, se ha convertido en una potente herramienta para la domótica ya que define características para su implementación donde uno de los principales aspectos tomados en cuenta es su bajo consumo de energía, su sistema de comunicaciones vía radio y su integración que permite fabricar nodos con muy poca electrónica.

A través del proyecto de investigación “Diseño, implementación y validación de soluciones inalámbricas para Automatización Industrial” realizado en la Universidad de las Fuerzas Armadas-ESPE, se adquirieron equipos para realizar aplicaciones de telemetría, monitoreo o control utilizando el estándar IEEE 802.15.4, ZigBee y Red inalámbrica de Sensores (WSN). Entre estos está el kit de desarrollo iDigi® ZigBee Gateway. Este kit permite realizar la configuración de una red ZigBee y proporcionar conectividad sin fisuras a la nube de dispositivos Etherios, para la integración de servicios web a las aplicaciones empresariales estándar a través de Internet.

En el artículo “Interoperabilidad en Sistemas Domóticos Mediante Pasarela Infrarrojos-ZigBee” se desarrolla un prototipo para mejorar la integración de los sistemas domóticos con los dispositivos que son controlados mediante infrarrojos, creando una pasarela inalámbrica que permite a una red domótica el envío de tramas infrarrojos (Asencio, y otros, 2011).

También se menciona el artículo “ZigBee y sus aplicaciones”. En el artículo se expone el funcionamiento, principales características de la tecnología, los tipos de dispositivos que utiliza, además de las principales aplicaciones, en donde se recalca que el mayor campo de aplicación de la tecnología ZigBee es en casas domóticas debido a la baja transmisión de datos que esta demanda y al bajo consumo de sus dispositivos (Salgado, 2011).

1.2 JUSTIFICACIÓN E IMPORTANCIA

La instalación de un sistema domótico proporciona un sinnúmero de beneficios y ventajas que una vivienda tradicional no puede ofrecer, fundamentalmente en lo que respecta a seguridad, confort, comodidad, ahorro energético e información, elementos que se combinan para ofrecer aumento en la calidad de vida de los usuarios.

En la actualidad el número de viviendas domotizadas es relativamente bajo respecto al número total de viviendas, no obstante, el interés en su adopción está creciendo progresivamente.

A esto se suma el gran desarrollo de dispositivos y tecnologías inalámbricas para aplicaciones de automatización industrial que facilitan en algunos aspectos la implementación de dispositivos con mandos remotos, ya sea para monitoreo o control.

La utilización del kit de desarrollo iDigi® ZigBee Gateway permite configurar una red ZigBee con dispositivos Xbee que permiten el ahorro de consumo energético y ofrecen un rango de cobertura de 10 a 75 metros.

Además el uso de la nube de dispositivos Etherios, ofrece la infraestructura necesaria para acceder, controlar, configurar y actualizar los dispositivos de la red de forma segura vía Internet, simplificando la gestión de los dispositivos de red y el desarrollo de aplicaciones.

Al disponer de estos recursos el presente proyecto tiene por finalidad realizar el diseño y la implementación de un sistema domótico con dispositivos capaces de tomar medidas de sensores y llevar a cabo la activación de actuadores de forma inalámbrica, además de evaluar el desempeño de la plataforma de nube pública Etherios.

El trabajo servirá de base para el desarrollo de futuras aplicaciones similares, además permitirá desarrollar una aplicación de servicios web proporcionada por la nube de dispositivos Etherios.

1.3 ALCANCE DEL PROYECTO

En el presente proyecto se utilizará el kit de desarrollo iDigi® ZigBee Gateway para realizar el diseño y la implementación del prototipo de un sistema domótico, el sistema estará formado por una red de sensores inalámbricos empleando nodos Xbee, las lecturas tomadas por los sensores de los dispositivos Xbee serán reportadas al gateway el mismo que pasará la información a la nube de dispositivos Etherios.

Los servicios domóticos elegidos para este prototipo son: confort y seguridad.

En lo referente a aplicaciones de confort se tiene monitorización de temperatura y control de iluminación. El control de iluminación se lleva a cabo mediante dos configuraciones: modalidad día y modalidad noche. En la modalidad día, el encendido y apagado de la luz depende de las señales generadas por los sensores de presencia y luminosidad, de modo que si se detecta presencia pero existe un nivel de luminosidad adecuado, la luz permanecerá apagada. En cambio, en la modalidad noche el encendido y apagado de la luz depende directamente de la señal generada por el sensor de presencia. Además se añade un control manual y automático de la luz, en el control automático se debe ingresar la hora de encendido de la luz.

En aplicaciones de seguridad se tiene el detector de intrusión, una vez activado recibe las señales del sensor magnético y de presencia, si se activan las dos señales, el sistema emitirá una alarma y a la vez se enviará un correo electrónico notificando la activación de la misma.

Se desarrolla un software de gestión remota en Digi ESP™ utilizando el lenguaje de programación Python, ya que es un lenguaje de código abierto y cuenta con una amplia biblioteca de secuencias de comandos para personalizar o automatizar tareas gestionadas por los gateways.

1.4 OBJETIVOS

1.4.1 General

Utilizar el estándar de tecnología inalámbrica ZigBee y la plataforma de Digi para diseñar e implementar un sistema domótico controlado y monitorizado de forma remota.

1.4.2 Específicos

- Analizar las principales características y funcionamiento de los dispositivos Xbee en aplicaciones domóticas.
- Implementar una red inalámbrica de sensores y actuadores.
- Desarrollar un software de gestión remota en el entorno de Python Digi ESP™.
- Utilizar la plataforma de nube pública para realizar una aplicación de servicios web.
- Evaluar el sistema domótico mediante pruebas que confirmen el correcto funcionamiento del prototipo.

CAPÍTULO 2

MARCO TEÓRICO

2.1 DOMÓTICA

La Domótica se refiere a la integración de la tecnología en el hogar para mejorar la calidad de vida de sus habitantes, automatizando procesos domésticos que pueden ser desde un simple temporizador para encender y apagar una luz o aparato a una hora determinada, hasta los más complejos sistemas capaces de interactuar con cualquier elemento eléctrico del hogar (Domínguez & Sáez, 2006).

Tiene por finalidad mejorar la seguridad, el confort, la flexibilidad, el ahorro energético, que pueden estar conectados por medio de redes interiores y exteriores de comunicación, para facilitar el control general de los usuarios de los (Domodesk)sistemas y ofrecer nuevos servicios.

2.1.1 Prestaciones de un sistema domótico

Un sistema domótico ofrece al usuario una serie de servicios relacionados con aspectos muy diferentes de la actividad que se desarrolla en el hogar, todas dirigidas a mejorar la calidad de vida de los usuarios como se muestra en la Figura 1; estas prestaciones son:

- **Seguridad:** en este apartado se hace referencia a los componentes del sistema domótico encargados de garantizar los posibles fallos en cualquiera de las instalaciones de la vivienda (fugas de gas, inundaciones), además de asegurar la integridad física de los habitantes. Algunas de las aplicaciones que se implementan son: simulación de presencia, detectores de intrusión, movimiento; detección de inundaciones, fugas de gas e incendios.
- **Confort:** la calidad de vida en el hogar puede ser incrementada mediante una gama muy amplia de servicios. Por ejemplo, control de la climatización, iluminación, monitorización de la temperatura, automatización de determinados electrodomésticos.
- **Ahorro energético:** la optimización del gasto energético es otro de los pilares básicos de la domótica. La gestión inteligente de las instalaciones de climatización, de los electrodomésticos o de los sistemas de iluminación mediante diversos dispositivos permite un mejor aprovechamiento de recursos naturales y mejora claramente los costes energéticos.

- **Comunicaciones:** mediante diferentes sistemas, la domótica permite a los usuarios enterarse de las incidencias y del estado de funcionamiento de los equipos empleando la gestión remota del hogar.

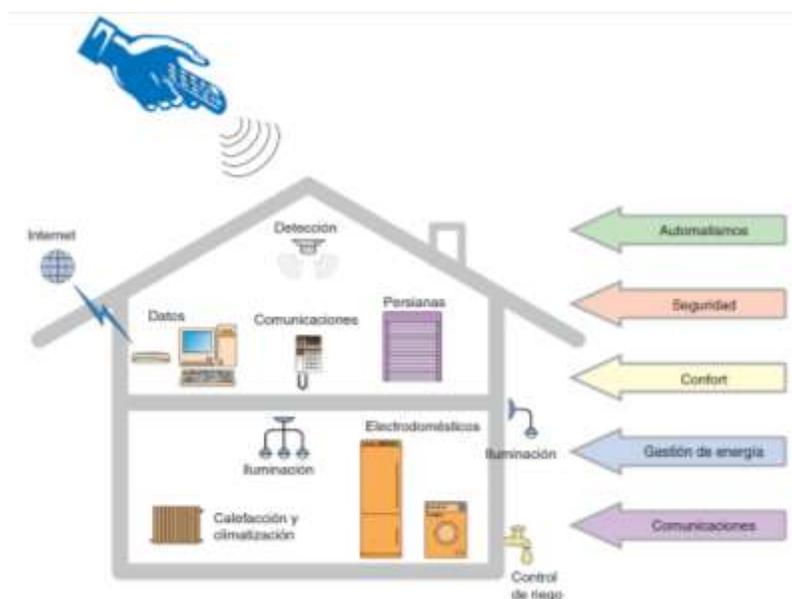


Figura 1. Servicios de un sistema domótico

Fuente (Domodesk)

2.1.2 Componentes básicos de un sistema domótico

Un sistema domótico está compuesto por una serie de elementos, que van desde una central de gestión hasta un mando automático a distancia. Dentro de esta variedad de elementos se tiene:

- **Sensores:** estos dispositivos son los encargados de recoger la información de los distintos parámetros que controlan (el nivel de presión de una tubería, la temperatura ambiente, el suministro de gas natural) y de transmitir esta información para su procesamiento. Por lo general, los sensores no se conectan a la red eléctrica, sino que incorporan baterías de larga duración. De esta manera se consigue una gran flexibilidad en su instalación (Domínguez & Sáez, 2006). Existen varios tipos de sensores como se muestra en la Figura 2.



Figura 2. Tipos de sensores.

Fuente (Domínguez & Sáez, 2006)

- **Actuadores:** son elementos capaces de recibir una orden del sistema de control y realizar una acción que modifique el estado de ciertos equipos e instalaciones como: encendido o apagado, subida o bajada, apertura o cierre. Entre los más utilizados tenemos contactores, electroválvulas de corte de suministro para luz o gas, relés, motores, etc.

- **Controladores:** son los elementos encargados de gestionar la información (señales) que les llega a través de los sensores y/o actuadores, enviando acciones para que actúen dentro del sistema.
- **Pasarela:** es el dispositivo que interconecta los distintos elementos destinados a la automatización de la vivienda, haciendo de interfaz común de todos ellos hacia las redes externas.
- **Bus:** es el medio de comunicación que transporta la información entre los distintos dispositivos por un cableado propio, por redes de otros sistemas (red eléctrica, red telefónica, red de datos) o de forma inalámbrica.

2.1.3 Tipos de arquitecturas de un sistema domótico

La arquitectura de un sistema domótico depende de la distribución y ubicación de los nodos de la red, que son los elementos de control que se encuentran dentro del hogar. Existen dos arquitecturas básicas:

- **Arquitectura centralizada:** es aquella en la que el controlador es el eje central del sistema, recibe la información de los sensores, la procesa y genera las órdenes para los actuadores, según la información o configuración que reciba por parte del usuario como se muestra en la Figura 3 (Junestrand, Passaret, & Vázquez, 2005).

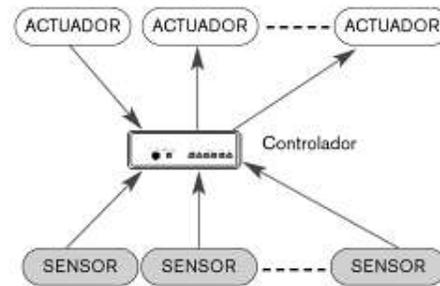


Figura 3. Arquitectura centralizada

Fuente (Junestrand, Passaret, & Vázquez, 2005)

- **Arquitectura distribuida:** este tipo de arquitectura no tiene un controlador centralizado, sino que cada sensor y actuador es capaz de analizar la información. Estos dispositivos están conectados a través de un bus como se muestra en la Figura 4 (Junestrand, Passaret, & Vázquez, 2005).

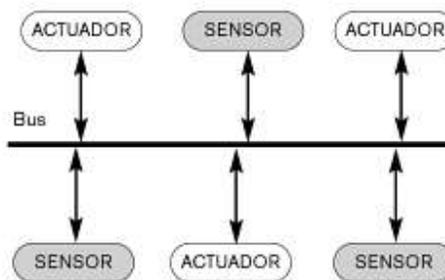


Figura 4. Arquitectura distribuida

Fuente (Junestrand, Passaret, & Vázquez, 2005)

Algunos sistemas usan un enfoque mixto, son sistemas con arquitectura descentralizada que disponen de varios dispositivos capaces de adquirir y procesar la información de los sensores y transmitirlos al resto de dispositivos distribuidos en el domicilio.

2.1.4 Medios de transmisión de los sistemas domóticos

Los sistemas domóticos utilizan distintos medios de transmisión de las señales de control de los diferentes dispositivos, entre los estos tenemos:

- **Cableados:** todos los dispositivos del sistema están cableados a la central o entre ellos. El cableado puede ser coaxial, par trenzado, por fibra óptica o mediante la línea eléctrica de la vivienda llamado “corrientes portadoras”.
- **Inalámbricos:** en este caso se usan sensores inalámbricos alimentados por pilas o baterías y transmiten vía radio la información de los eventos a la central. Entre estos tenemos Wifi, Bluetooth, radiofrecuencia, infrarrojos, ZigBee.

2.2 COMUNICACIONES INALÁMBRICAS

La comunicación inalámbrica posibilita el intercambio de información entre usuarios sin mediación de cables, utilizando la modulación de ondas electromagnéticas a través del espacio.

Entre algunas ventajas que proporcionan los sistemas inalámbricos, se puede mencionar las siguientes:

- La capacidad de cruzar zonas de agua, en donde una instalación con cable requeriría un trato especial para prevenir las filtraciones sobre los conductores de cobre.
- Suelen ser más baratos y permiten una gran movilidad dentro del alcance de la red.

2.2.1 Clasificación de las redes inalámbricas

Según su cobertura se pueden clasificar en diferentes tipos como se muestra en la Figura 5:

- **Red inalámbrica de Área Personal (WPAN):** es una red de computadoras para la comunicación entre distintos dispositivos cercanos al punto de acceso. Este tipo de red tiene una cobertura de pocos metros. Existen tecnologías basadas en Home RF, Bluetooth, ZigBee, RFID.
- **Red inalámbrica de Área Local (WLAN):** este tipo de redes tiene una cobertura de un área equivalente a la red local de una empresa, con un

alcance aproximado de 100 metros. En las redes de área local podemos encontrar tecnologías basadas en HiperLAN, Wifi (IEEE 802.11).

- **Red inalámbrica de Área Metropolitana (WMAN):** este tipo de red pretende cubrir el área de una ciudad. Para las WMAN se encuentran tecnologías basadas en WiMAX, un estándar de comunicación basado en la norma IEEE 802.16. WiMAX puede alcanzar una velocidad aproximada de 70 Mbps en un radio de varios kilómetros.
- **Red inalámbrica de Área Extensa (WWAN):** tiene el alcance más amplio de todas las redes inalámbricas, se basan en la tecnología celular y han aparecido como evolución de comunicaciones de voz. Las principales tecnologías son GSM (Global System for Mobile Communication), GPRS (General Packet Radio Service) y UMTS (Universal Mobile Telecommunication System).

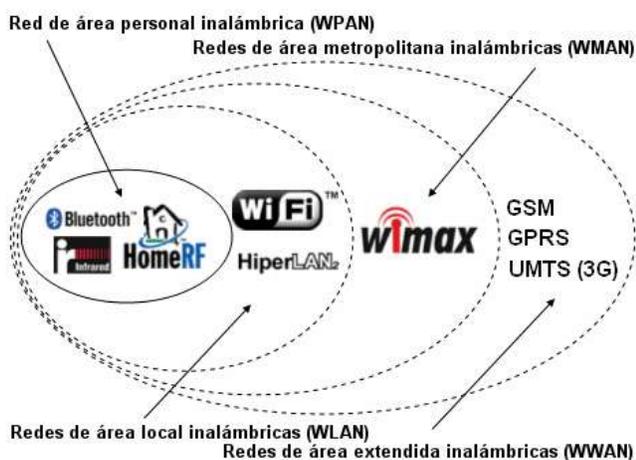


Figura 5. Tipos de redes inalámbricas.

Fuente (Redes inalámbricas)

2.2.2 Red inalámbrica de sensores

Una red inalámbrica de sensores, o Wireless Sensor Network (WSN), es una red de un gran número de pequeños dispositivos capaces de medir diferentes variables de ambiente en el que se encuentran, y de procesar y comunicar la información de manera inalámbrica.

2.3 PROTOCOLO ZIGBEE Y ESTÁNDAR IEEE 802.15.4

2.3.1 Introducción

ZigBee es el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radios digitales de bajo consumo que se basa en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (Wireless Personal Area Network, WPAN).

Su objetivo son las aplicaciones que requieren comunicaciones seguras con baja tasa de envío de datos y maximización de la vida útil de sus baterías.

2.3.2 El estándar IEEE 802.15.4

Es el estándar que define el nivel físico (PHY) y el control de acceso al medio (MAC) de redes inalámbricas de área personal con tasas bajas de transmisión de datos como se muestra en la Figura 6. La capa física define la

frecuencia, potencia, modulación y otras condiciones del enlace inalámbrico. El control de acceso define el formato de la manipulación de datos.

El estándar IEEE 802.15.4 soporta múltiples topologías para su conexión en red, entre ellas la topología tipo Estrella y la topología Punto a Punto. La topología a escoger es una elección de diseño y va a estar dado por la aplicación a la que se desee orientar; algunas aplicaciones como periféricos e interfaces de PC, requieren de conexiones de baja potencia de tipo estrella, mientras que otros como los perímetros de seguridad requieren de una mayor área de cobertura por lo que es necesario implementar una red Punto a Punto (Higuera, Kartsakli, & Valenzuela)

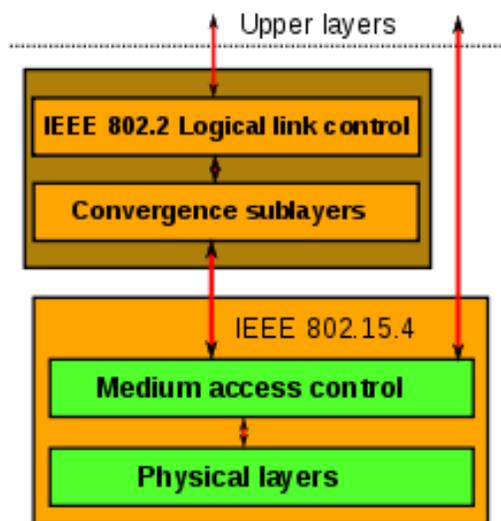


Figura 6. Capas del estándar IEEE 802.15.4

Fuente (Higuera, Kartsakli, & Valenzuela)

2.3.2.1 Capa física

La capa física es la capa de red más básica, proporcionando únicamente los medios para transmitir bit a bit sobre un enlace de datos físico conectado a nodos de red. Las cadenas de bits pueden ser agrupadas en palabras codificadas o símbolos, y convertidas a señales físicas, que son transmitidas sobre un medio de transmisión físico. La capa física proporciona un interfaz eléctrico, mecánico y procedimental para el medio de transmisión. En este nivel se especifican las características de los conectores eléctricos, sobre qué frecuencias retransmitir, qué esquema de modulación usar y parámetros de bajo nivel similares (Capa física).

La característica de la capa física es el rango de transmisión. La capa física de 2.4 GHz permite un rango de transmisión de 250 kb/s, mientras que la capa física de los 868/915 MHz ofrece rangos de transmisión de 20 kb/s y 40 kb/s respectivamente.

La principal diferencia entre ambas capas físicas radica en la banda de frecuencias. La capa física de los 2.4 GHz, especifica operación en la banda industrial, médica y científica (ISM), que está disponible a nivel mundial, mientras que la capa física de los 868/915 MHz especifica operaciones en la banda de 865 MHz en Europa y 915 MHz en la banda ISM en Estados Unidos.

2.3.2.2 Control de acceso

El control de acceso al medio transmite tramas MAC usando para ello el canal físico. Además del servicio de datos, ofrece un interfaz de control y regula

el acceso al canal físico y al balizado de la red. También controla la validación de las tramas y las asociaciones entre nodos, y garantiza slots de tiempo. Por último, ofrece puntos de enganche para servicios seguros.

El formato general de las tramas MAC se diseñó para ser muy flexible y que se ajustara a las necesidades de las diferentes aplicaciones con diversas topologías de red al mismo tiempo que se mantenía un protocolo simple.

El estándar no define niveles superiores ni subcapas de interoperabilidad. Existen extensiones, como la especificación de ZigBee, que complementan al estándar en la propuesta de soluciones completas.

2.3.3 ZigBee

El término ZigBee describe un protocolo inalámbrico normalizado para la conexión de una Red de Área Personal (WPAN: Wireless personal area network). ZigBee es una alianza, de más de 100 empresas, la mayoría de ellas fabricantes de semiconductores, con el objetivo de auspiciar el desarrollo e implantación de una tecnología inalámbrica de bajo coste (Domodesk).

El estándar ZigBee enfoca a un segmento del mercado no atendido por los estándares existentes, con baja transmisión de datos, bajo ciclo de servicio de conectividad. La razón de promover un nuevo protocolo como un estándar, es para permitir la interoperabilidad entre dispositivos fabricados por compañías diferentes.

El estándar ZigBee amplía el estándar IEEE 802.15.4 aportando una capa de red (NWK) que gestiona las tareas de enrutado y de mantenimiento de los nodos de la red; y un entorno de aplicación que proporciona una subcapa de aplicación (APS) que establece una interfaz para la capa de red, y los objetos de los dispositivos tanto de ZigBee como del diseñador. Así pues, los estándares IEEE 802.15.4 y ZigBee se complementan proporcionando una pila completa de protocolos como se muestra en la Figura 7, estos permiten la comunicación entre multitud de dispositivos de una forma eficiente y sencilla (Ocampo & Serna, 2010).

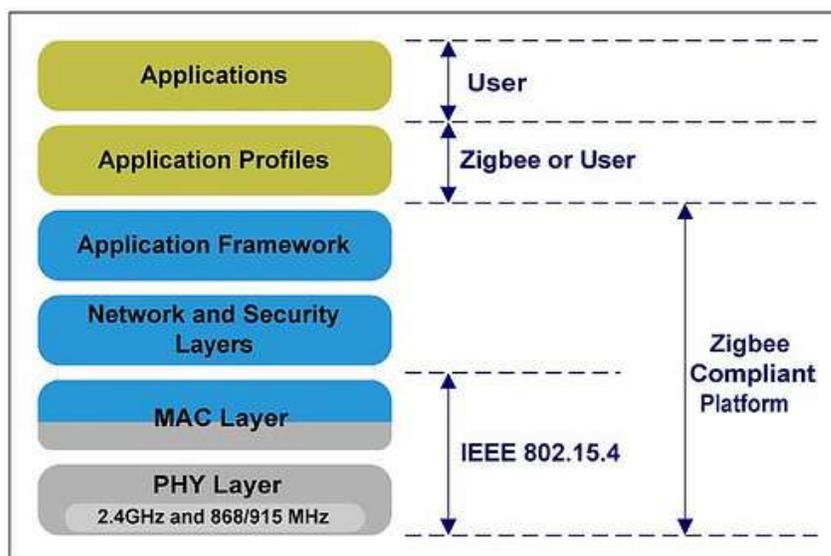


Figura 7. Capas añadidas por ZigBee.

Fuente (Ocampo & Serna, 2010)

2.3.3.1 Características del protocolo ZigBee para domótica

ZigBee es un sistema ideal para redes domóticas, específicamente diseñado para reemplazar la proliferación de sensores/actuadores individuales.

ZigBee fue creado para cubrir la necesidad del mercado de un sistema a bajo coste, un estándar para redes Wireless de pequeños paquetes de información, bajo consumo, seguro y fiable. A continuación se tiene algunas de sus características:

- Los rangos de alcance son de 10 m a 75 m.
- Puede usar las bandas libres ISM de 2,4 GHz (Mundial), 868 MHz (Europa) y 915 MHz (EEUU).
- La fabricación de un transmisor ZigBee consta de menos circuitos analógicos de los que se necesitan habitualmente.
- La escalabilidad de red es un mejor soporte para las redes más grandes, ofrece más opciones de gestión, flexibilidad y desempeño.
- La gestión automatizada de direcciones de dispositivos permite optimizar la gestión de red agregada y herramientas de configuración en grandes redes.

2.3.3.2 Tipos de dispositivos

ZigBee tiene tres tipos de dispositivos como se muestra en la Figura 8, estos son:

- **Coordinador ZigBee (ZC):** es el tipo de dispositivo más completo. Debe existir uno por red, entre sus tareas están las de formar y gestionar la red, requiere de memoria y capacidad de computación.
- **Router ZigBee (ZR):** además de ofrecer un nivel de aplicación para la ejecución de código de usuario, puede actuar como router

interconectando dispositivos separados en la topología de la red. Son que tienen la capacidad de enviar, recibir y enrutar información. Pueden actuar como mensajeros entre dispositivos que están muy alejados para establecer una comunicación directamente, los dispositivos ZigBee no están pensados para comunicaciones de largas distancias, sino para trabajar en redes de sensores y cubrir esas largas distancias pasando la información entre los distintos nodos.

- **Dispositivo final (ZED):** estos serían los dispositivos de bajo consumo. El coordinador y los routers deben estar siempre encendidos ya que pueden actuar como mensajeros entre otros dispositivos, por lo que lo normal puede ser conectarlos a la red eléctrica. Los end devices pueden enviar y recibir información pero no pueden actuar como mensajeros entre otros dos dispositivos de la red, lo normal es que el end device esté en un modo de bajo consumo y se despierte cuando quiere enviar o recibir información, alargando la duración de sus baterías durante mucho tiempo. Un ZED tiene requerimientos mínimos de memoria y es por tanto significativamente más barato.

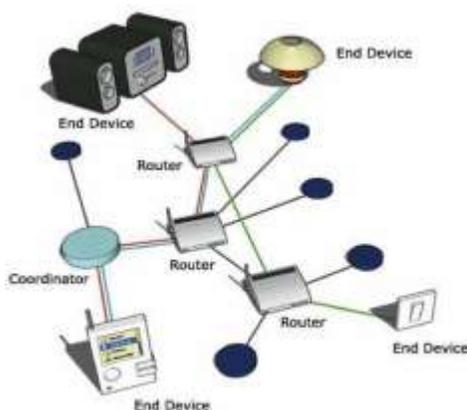


Figura 8. Tipos de dispositivos ZigBee.

Fuente (Esquivel, y otros, 2011)

2.3.3.3 Funcionalidad

Basándose en su funcionalidad, puede plantearse una segunda clasificación:

- **Dispositivo de funcionalidad completa (FFD):** También conocidos como nodo activo. Es capaz de recibir mensajes en formato 802.15.4. Gracias a la memoria adicional y a la capacidad de computar, puede funcionar como Coordinador o Router ZigBee, o puede ser usado en dispositivos de red que actúen de interfaces con los usuarios.

- **Dispositivo de funcionalidad reducida (RFD):** También conocido como nodo pasivo. Tiene capacidad y funcionalidad limitadas con el objetivo de conseguir un bajo coste y una gran simplicidad. Básicamente, son los sensores/actuadores de la red. Un nodo ZigBee (tanto activo como pasivo) reduce su consumo gracias a que puede permanecer dormido la mayor parte del tiempo (incluso muchos días seguidos). Cuando se requiere su uso, el nodo ZigBee es capaz de despertar en un tiempo ínfimo, para volverse a dormir cuando deje de ser requerido. Un nodo cualquiera despierta en aproximadamente 15 ms. Además de este tiempo, se muestran otras medidas de tiempo de funciones comunes:
 - Nueva enumeración de los nodos esclavo (por parte del coordinador): aproximadamente 30 ms.

 - Acceso al canal entre un nodo activo y uno pasivo: aproximadamente 15 ms.

2.3.3.4 Topología

ZigBee usa las topologías de IEEE 802.15.4 para transferencia de datos y agrega las topologías de árbol y de malla. Debido al poco alcance de cada nodo, frecuentemente un paquete debe ser retransmitido varias veces por intermedio de ruteadores. En la Figura 9 se muestra los tipos de topologías.

- **Estrella:** el coordinador de la red se sitúa en el centro de la red y el resto de dispositivos pertenecientes a la red, sean FFD o RFD se comunican directamente con él.
- **Árbol:** el coordinador de la red se comunica con los dispositivos FFDs, los cuales a su vez comunican de forma directa con los nodos de su clúster. Un ejemplo sería un edificio en el que se establece un clúster por planta, los ZEDs que estén instalados se comunicaran únicamente con el FFD de dicha planta, que hará de router, poniendo en contacto así a dichos dispositivos con el coordinador, que será la raíz del árbol.
- **Malla:** al menos uno de los nodos tendrá más de dos conexiones, con lo cual existen muchas rutas para hacer llegar mensajes al coordinador de red en caso de que una ruta falle.

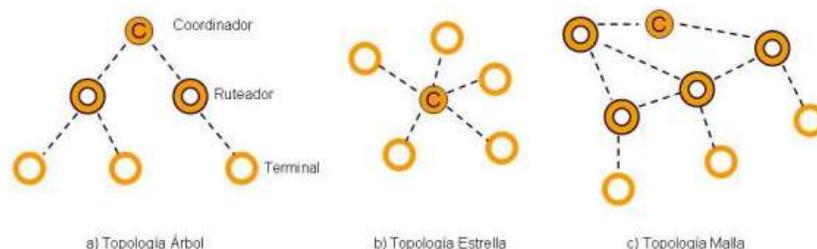


Figura 9. Tipos de topologías.

Fuente (Longares)

2.3.3.5 Aplicaciones

Básicamente ZigBee tiene su principal aplicación cuando sus nodos se unen para trabajar juntos y formar lo que se denomina “Red de Sensores”. El estándar ZigBee se ha hecho a medida para la monitorización y para aplicaciones de control. Por lo tanto, los principales campos de aplicación son:

- Red de sensores.
- Automatización de edificios y hogares.
- Control industrial.
- Agricultura y control ambiental.
- Cuidados médicos.

2.4 CLOUD COMPUTING

Cloud computing es un modelo de computación basado en Internet, permite el acceso de red bajo demanda a un conjunto compartido de recursos informáticos configurables (redes, servidores, almacenamiento de datos, aplicaciones y servicios) que pueden ser rápidamente almacenados y liberados con un esfuerzo mínimo de gestión o interacción del proveedor de servicios (Mell & Grance, 2011).

Las nubes pueden ser privadas o públicas. La nube pública es la que vende servicios en Internet a cualquier usuario. Las nubes privadas son una red o centro de datos que pertenece a una organización y que ofrece servicios de hosting a un número limitado de personas. Sea privada o pública, el objetivo de

la computación en la nube es ofrecer acceso a recursos de computación y servicios de TI de forma sencilla y escalable.

Está compuesto por tres modelos de servicio como se muestra en la Figura 10: Infraestructura como servicio (IaaS), Plataforma como servicio (PaaS) y Software como servicio (SaaS).



Figura 10. Modelos de servicio.

Fuente (Top cloud computing providers, 2010)

2.4.1 Modelos de servicio

- **Infraestructura como servicio (IaaS):** es una manera de entregar la infraestructura de la nube (servidores, almacenamiento, sistema operativo) como un servicio compartido, para reducir el costo de inversión, mantenimiento operativo y de hardware. La infraestructura

debe ser confiable y flexible para una fácil implementación y operación de aplicaciones (Kepes, 2011).

- **Plataforma como servicio (PaaS):** es la plataforma que permite la creación de aplicaciones web de forma rápida y sencilla, PaaS ofrece el despliegue de aplicaciones, reduce el costo y la complejidad de la compra y mantenimiento de hardware y software y aprovisionamiento de la capacidad de alojamiento.

PaaS es análogo a SaaS, salvo que, en lugar de ser software entregado a través de Internet, es una plataforma para la creación de software, entregado a través de Internet.

Ejemplos: Google App Engine, Heroku, Force.com

- **Software como servicio (SaaS):** es un modelo de distribución de software en el que el software y los datos asociados están alojados en la nube. Las aplicaciones son accesibles desde varios dispositivos de cliente a través de una interfaz ligera, como un navegador web o una interfaz de programa. El consumidor no administra ni controla la infraestructura de nube. Ejemplos: Google Docs, Salesforce, Dropbox, Gmail.

2.5 DESCRIPCIÓN DE LA PLATAFORMA DIGI.

2.5.1 Digi International® Inc.

Digi International se formó en 1985. La sede central se encuentra en Minnesota, y tiene oficinas de ventas regionales en América del Norte, Europa y Asia.

Digi International es experto en comunicación M2M, combinando productos y servicios como soluciones de extremo a extremo para impulsar la eficiencia de los negocios. Ofrece una amplia gama de productos inalámbricos, una plataforma Cloud computing para los dispositivos y el desarrollo de servicios.

Todo el conjunto de soluciones que ofrecen están diseñados para permitir que cualquier dispositivo se comunique con cualquier aplicación, en cualquier lugar del mundo.

Digi ofrece la Nube de dispositivos Etherios para gestionar los dispositivos, simplificando la integración, instalación y administración del sistema.

Productos

Los productos que ofrece son módulos Xbee, microprocesadores, productos de comunicación por satélite, kits de desarrollo y software, Gateway, adaptadores de comunicación inalámbrica (ZigBee, Wi-Fi, RF), servidores de serie, servidores de consola inteligente, productos conectados USB, productos de visualización a distancia, cámaras y sensores (Digi, About us).

2.5.2 Nube de dispositivos Etherios

Es una plataforma de nube pública como servicio (PaaS) que proporciona la integración de aplicaciones con las redes de los dispositivos en tiempo real. Proporciona mensajería segura, almacenamiento de datos y gestión de dispositivos.

Permite una comunicación bidireccional entre cualquier dispositivo y la nube para acceder, controlar, configurar y actualizar los dispositivos ilimitadamente a través de Internet utilizando una plataforma segura y escalable.

Ofrece a los clientes la gestión centralizada de los gateways y dispositivos finales conectados facilitando la integración a través de APIs. El usuario puede conectar un dispositivo físico a la nube y usar una aplicación web para el acceso remoto.

Creando una cuenta en la nube de dispositivos Etherios se tiene acceso para diseñar y probar un prototipo, y evaluar libremente toda la funcionalidad y características de los dispositivos de la nube.

2.5.2.1 Comunicación con la nube

Para comunicarse con un dispositivo mediante programación, se utiliza Servicios web.

Las peticiones de Servicios web se envían desde una aplicación remota a la nube de dispositivos, que a su vez se comunica directamente con el dispositivo. Esto permite una comunicación bidireccional máquina a máquina.

La nube proporciona una API HTTP estándar que permite acceder a los datos de distintas maneras. Los usuarios pueden escribir solicitudes HTTP en cualquier tipo de lenguaje de programación para obtener los datos de dispositivos y utilizarlos o mostrarlos de la manera que deseen.

Ejemplos de este tipo de clientes son las páginas Web y los programas escritos en un lenguaje como Python o Java. Los clientes envían peticiones al servidor mediante solicitudes HTTP estándar. Las solicitudes HTTP que soporta la nube de dispositivos son GET, PUT, POST y DELETE. El servidor admite la autenticación HTTP básica y sólo los usuarios válidos pueden acceder a la base de datos.

Una de las ventajas de utilizar servicios web es la independencia de la plataforma y del lenguaje, ya que utilizan lenguaje XML estándar.

2.6 DESCRIPCIÓN DE LOS DISPOSITIVOS XBEE UTILIZADOS.

Los dispositivos ZigBee utilizados para la implementación del sistema son los siguientes:

2.6.1 ConnectPort X4 Gateway

El ConnectPort X4 Gateway ofrece una variedad de opciones de interfaz LAN/WAN para redes de extremo a extremo de dispositivos remotos. Es la puerta de enlace entre distintas tecnologías como Ethernet y Xbee.

El gateway recoge los datos de los nodos finales, los procesa, y envía una solicitud principal. Cuenta con un entorno de desarrollo y utiliza Digi ESP™ IDE y lenguajes de programación DIA/Python.

El ConnectPort X4 Gateway es compatible con adaptadores Xbee, módulos Xbee y accesorios Xbee, proporcionando a los usuarios una flexibilidad sin precedentes y la comunicación de extremo a extremo entre los dispositivos (Digi, ConnectPort® X4 Family). En la Figura 11 se muestra el ConnectPort X4 Gateway.

Características

- Tiene interfaces Ethernet, serie, USB
- Funciona como coordinador en la red ZigBee.
- Protocolos de red: UDP/TCP, DHCP
- Tiene una interfaz web HTTP/HTTPS



Figura 11. ConnectPort X4 Gateway

Fuente (Digi, ConnectPort® X4 Family)

2.6.2 Xbee Wall Router

El Xbee Wall Router de Digi es un dispositivo que se utiliza para crear o ampliar una red ZigBee permitiendo que otros dispositivos ZigBee se comuniquen sin problemas entre sí.

Además, los sensores ambientales incluidos en el router permiten a los usuarios del ConnectPort X4 Gateway monitorear la temperatura y las condiciones de luz en toda la zona de despliegue. El router Xbee es totalmente interoperable con otros dispositivos Xbee ZB, incluyendo módulos Xbee, adaptadores, sensores y puertas de enlace.

El adaptador Xbee entrega conectividad inalámbrica a los dispositivos electrónicos en las redes de cable y los sensores Xbee proporcionan información del ambiente en tiempo real. En la Figura 12 se muestra el adaptador Xbee.

Características

- Se conecta a una toma de corriente estándar para una fácil implementación.
- Extiende intensidad de la señal y el rango de una red de malla Xbee ZB.
- Crea rutas de red adicionales para la creación de redes de malla más fiable.
- Sensores de luz y temperatura integrados.



Figura 12. Xbee Wall Router

Fuente (Digi, XBee Wall Router)

2.6.3 Adaptador Xbee DIO

El adaptador digital Xbee proporciona conectividad inalámbrica de corto alcance para cualquier dispositivo digital. Es un producto diseñado para tomar una entrada digital o mediante programación validar una salida digital y comunicar los cambios de E/S a través de la red Xbee. En la Figura 12 se muestra el adaptador Xbee DIO.



Figura 13. Adaptador Xbee DIO

Fuente (Inc. D. I., 2014)

Características

- Terminales de E/S digitales de colector abierto.
- Batería opcional para lugares aislados.
- Proporciona conectividad inalámbrica instantánea a los dispositivos digitales existentes.

En la Tabla 1 y Tabla 2 se muestran consideraciones importantes que se deben tomar en cuenta para trabajar con el dispositivo.

Tabla 1

Configuración de pines del Adaptador Xbee DIO

PIN	FUNCIÓN
1	Estos pines son configurados como entradas o salidas digitales.
2	
3	
4	
5	GND
6	+12VDC 50mA max

Fuente (Inc. D. I., 2014)

Tabla 2

Especificaciones y características de las entradas y salidas

MODO	ESPECIFICACIÓN	VALOR
Entrada digital	Entrada máx.	+30VDC
	Entrada mín.	-.5VDC
	Impedancia de entrada	~1.5 MΩ
Salida digital	Corriente máx.	1.8ADC
	Voltaje máx.	+30VDC
	Resistencias pullups	10KΩ a 3VDC

Fuente (Inc. D. I., 2014)

2.7 SENSORES Y ACTUADORES (Pallás)

Los sensores y actuadores son los elementos que conectan un sistema con su entorno físico.

2.7.1 Sensores

Son dispositivos que convierten magnitudes físicas o químicas (luz, presión, temperatura, movimiento, etc) en variables eléctricas para poder medirlas y controlarlas.

Según la forma de la señal de salida pueden ser analógicos o digitales. Los sensores digitales son aquellos que frente a un estímulo pueden cambiar de estado ya sea de cero a uno o de uno a cero, en este caso no existen estados intermedios y los valores de tensión que se obtienen son únicamente dos, 5VDC y 0VDC o valores muy próximos. La salida de los sensores analógicos es un nivel de tensión o corriente que varía de forma continua con la variable a medir.

Sensor de presencia

Es un tipo de sensor que activa o desactiva automáticamente el mecanismo eléctrico al que está conectado, cuando detecta o no, la presencia de personas en movimiento u objetos.

Tipos de sensores

- Sensor infrarrojo pasivo (PIR)
- Sensor ultrasónico

2.7.1.1 Sensor PIR DC-SS502V110

El sensor PIR es un dispositivo electrónico que mide la luz infrarroja (IR) radiada de los objetos situados en su campo de visión. Funciona detectando el calor (rayos infrarrojos) del cuerpo humano. Cuando una persona entra dentro del área de detección del sensor, la cantidad de radiación incidente de infrarrojos en el sensor varía en función de la diferencia de temperatura entre la temperatura ambiente y la de la persona. Como respuesta al cambio el sensor cambia el nivel lógico de un pin. En la Figura 14 se muestra el sensor utilizado en el prototipo.



Figura 14. Sensor PIR DC-SS502V110

Fuente (Inc. S. E., 2009)

Características

- Amplio rango de detección.
- Voltaje de alimentación: 5-24VDC o 3.3VDC
- Sensibilidad ajustable

En la Tabla 3 se muestra en detalle la configuración de pines del sensor de presencia.

Tabla 3

Configuración de pines

NOMBRE	NÚMERO PIN	FUNCIÓN
+3.3V	1	Voltaje de alimentación: 3.3VDC
+Vcc	2	Voltaje de alimentación: 5-24VDC
OUTPUT	3	Pin salida
GND	4	GND
Relay2	5	Pin 2 del relé
Relay1	6	Pin 1 del relé

Fuente (Inc. S. E., 2009)

Rango de detección

El rango de detección diseñado para el módulo es de 3 a 4 metros, y su ángulo de detección es de 100° como se muestra en la Figura 15.

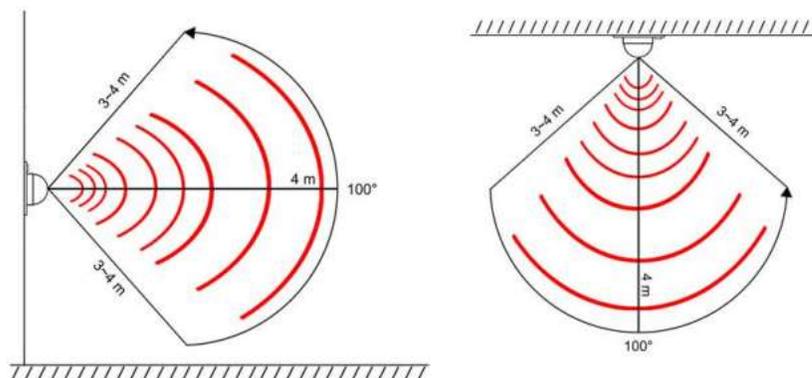


Figura 15. Diagrama del rango de detección

Fuente (Inc. S. E., 2009)

2.7.1.2 Sensor magnético

Detecta el campo magnético que provoca un imán. Eléctricamente, es un interruptor momentáneo y se presenta con contacto normalmente abierto (NA) o normalmente cerrado (NC).

Consiste en dos piezas apareadas, el imán y el contacto de ampolla. Cuando existe la presencia del campo magnético del imán, los contactos de la ampolla cambian de posición abriendo o cerrando el circuito.

Este tipo de sensor se utiliza para proteger puertas, ventanas y en general todo tipo de apertura. En la Figura 16 se muestra el sensor magnético utilizado en el prototipo.



Figura 16. Sensor magnético

Fuente (Aliexpress)

2.7.2 Actuadores

Son dispositivos que transforman una señal eléctrica de bajo nivel en otra de mayor potencia o magnitud física para actuar sobre un elemento final.

Los actuadores se suelen diferenciar por el tipo de acción que realizan, por ejemplo electroválvulas de agua, caldera, lavadora, etc, aunque también existen actuadores genéricos, como un contactor o un relé, que simplemente adaptan la potencia de la salida para actuar sobre cualquier elemento.

2.7.2.1 Relé

Es un interruptor mecánico accionado eléctricamente, actúa de modo automático como interruptor, accionando o desconectando un circuito. Permite conmutar circuitos de potencia más elevada mediante una señal de baja potencia.

Relé SRD-05VDC-SL-C

Puede controlar dispositivos de alta potencia (hasta 10A). Tiene dos partes diferenciadas, una parte que irá conectada a VAC y otra que va conecta a VDC.

En la parte de AC hay 3 contactos, uno normalmente abierto, uno normalmente cerrado y otro común, esto sirve para que, cuando el relé esté inactivo, el circuito quede abierto o cerrado. En la Figura 17 se muestra el relé utilizado en el prototipo.

Características

- Tamaño: 15.5mm x 15.0mm 19.00mmx
- Carga nominal: 10A 250VAC/28VDC, 10A 125VAC/28VDC.
- Vida eléctrica: 100.000
- Vida mecánica: 10.000.000
- Bobina Tensión nominal: 3-48VDC
- Potencia de la bobina: 0.36W, 0.45W
- Temperatura ambiente: -25 °C a +70 °C



Figura 17. Relé SRD-05VDC-SL-C

Fuente (Songle)

CAPÍTULO 3

DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

3.1 REQUERIMIENTOS DEL SISTEMA

El sistema domótico a ser desarrollado debe cumplir con ciertos requisitos, los cuales obedecen generalmente a los requerimientos del usuario. También debe cumplir ciertas especificaciones para todos los dispositivos y sensores que intervienen en la implementación.

El sistema domótico debe cumplir con los siguientes requerimientos:

- Contar con un enrutador o coordinador para la interconexión de red de sensores inalámbrica.
- Capacidad para unir dispositivos Xbee y sensores a la red inalámbrica.
- Software de gestión remota para la supervisión.
- Capacidad de acceso a la nube Etherios de los dispositivos.

- Medición de los parámetros de temperatura y luminosidad almacenados en una base de datos.
- Control de iluminación mediante el encendido y apagado de luces de manera manual o automática.
- Detector de intrusión con la utilización de sensores de movimiento y magnético.
- Envío de alerta a través de correo electrónico sobre la activación de la alarma de intrusión.

3.2 DISEÑO DEL SISTEMA

3.2.1 Funcionalidad del sistema

En la Figura 18 se muestra el diagrama de bloques del sistema, en donde se puede apreciar las diferentes áreas y sus respectivos controles.

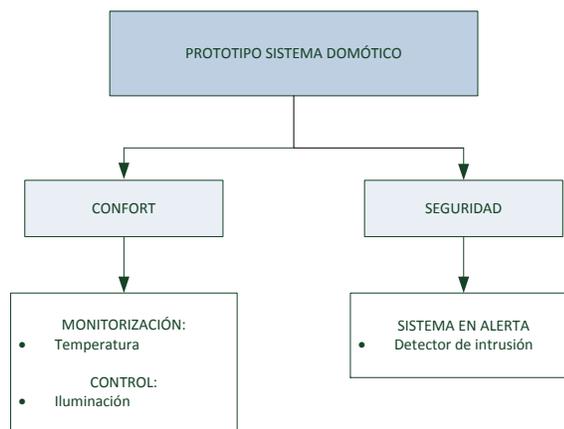


Figura 18. Diagrama de bloques del prototipo

CONFORT

Control de iluminación

El control del encendido y apagado de las luces se llevó a cabo mediante dos configuraciones: modalidad día y modalidad noche.

Para la modalidad día, el encendido y apagado de la luz depende de las señales generadas por el sensor de presencia y luminosidad. En cambio para la modalidad noche, el encendido y apagado de la luz depende directamente de la señal generada por el sensor de presencia.

Además se añadió un control manual y automático de la luz, en el control automático se debe ingresar la hora de encendido de la luz.

Monitorización de temperatura

La monitorización de temperatura se realiza por medio del dispositivo Xbee Wall Router, que tiene integrado un sensor de temperatura. Cuando se conecta el dispositivo se puede obtener la información del ambiente en tiempo real.

SEGURIDAD

Detector de intrusión

Una vez activado el detector de intrusión, el sistema recibe las señales del sensor magnético y de presencia, si se activan las dos señales, el sistema emitirá una alarma y a la vez se enviará un correo electrónico notificando la

activación de la alarma. Se debe ingresar la dirección de correo electrónico del destinatario que recibirá la notificación.

3.2.2 Arquitectura del sistema

El sistema domótico consiste en la monitorización y control de dispositivos. Cuenta con dos dispositivos finales para la toma de muestras de los sensores de presencia y activación de actuadores, un coordinador y un router de recolección de datos de temperatura y luminosidad.

La Figura 19 muestra la arquitectura del sistema, donde el dispositivo coordinador recolecta la información y envía los datos a la nube de dispositivos Etherios. La nube de dispositivos ofrece la integración de aplicaciones con las redes de los dispositivos en tiempo real. Creando una cuenta gratuita se puede conectar hasta 5 dispositivos coordinadores.

Para recopilar los datos de los dispositivos conectados y enviarlos a la nube se utiliza el software DIA de Digi ESP™ para Python. Además permite una fácil integración con aplicaciones de cliente que utilizan servicios web.

La aplicación web cliente se desarrolló en Google App Engine. La interfaz de usuario permite visualizar los datos de los sensores y controlar los dispositivos finales.

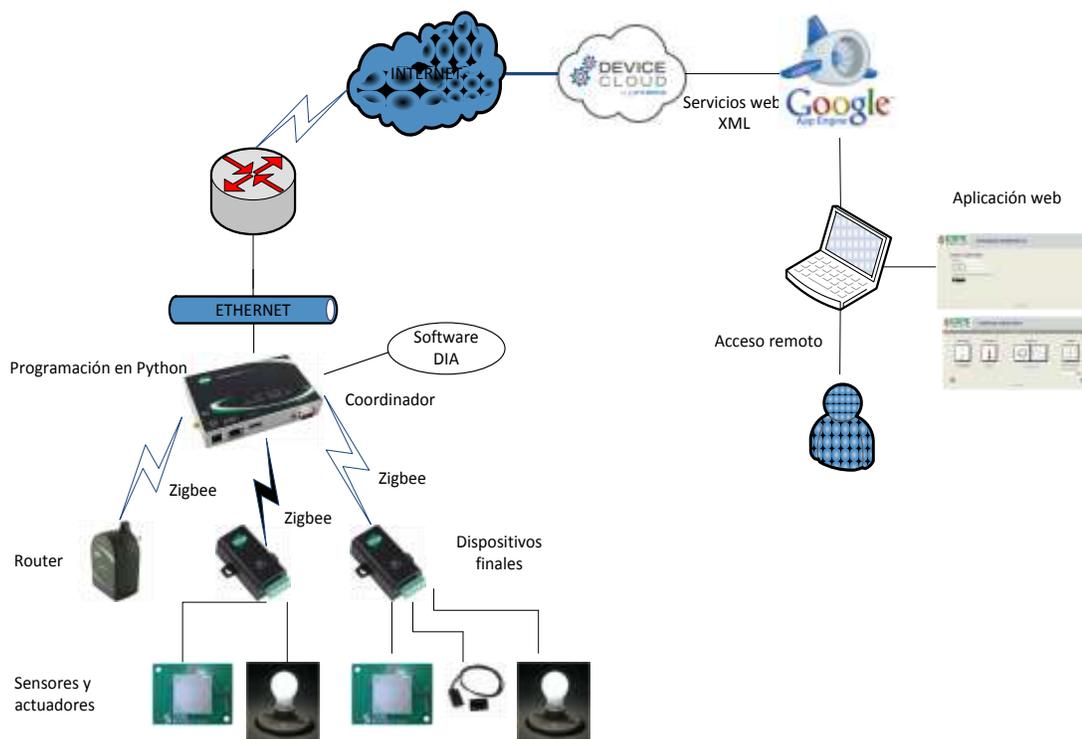


Figura 19. Arquitectura del sistema

3.2.3 Diseño de la red

El diseño de la red inalámbrica de sensores ZigBee se debe realizar de manera que cumpla la cobertura de las áreas elegidas y elegir adecuadamente los sensores, actuadores y dispositivos que intervienen en ella.

La red está compuesta por cuatro dispositivos, un coordinador (ZC), un router (ZR) y dos dispositivos finales (ZED). El router se encarga de la medición de los valores de temperatura y luminosidad del ambiente, y los dispositivos finales permiten recoger las señales de los sensores de presencia y sensor magnético y activar los actuadores.

La red está estructurada con topología tipo estrella como se muestra en la Figura 20, el dispositivo coordinador es el centro de la red, y el router y los dispositivos finales se comunican directamente con él.

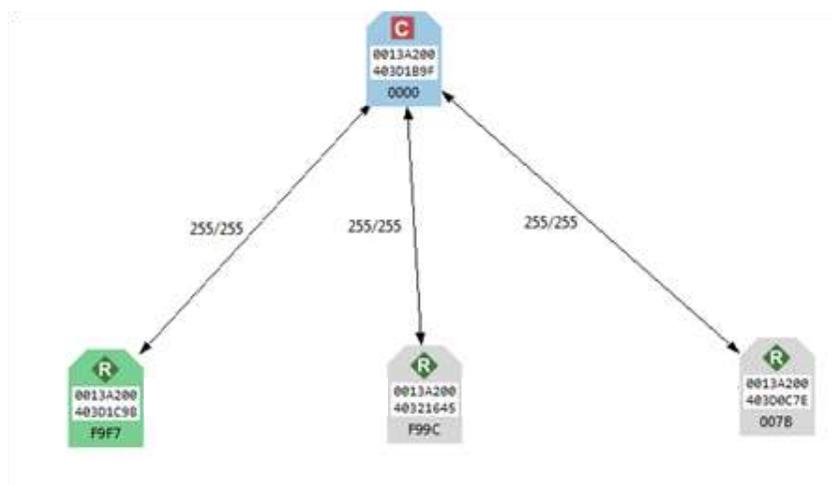


Figura 20. Red tipo estrella

Fuente (Software X-CTU)

Selección de componentes

- Dispositivo coordinador: ConnectPort X4 Gateway.
- Dispositivo router: Xbee Wall Router, que incluye sensores de temperatura y luminosidad.
- Dispositivos finales: 2 adaptadores Xbee DIO, en el primer adaptador se utilizan dos entradas digitales, una para el sensor de presencia y otra para el detector magnético, y una salida digital para la activación del actuador. En el otro adaptador se utiliza una entrada digital para el sensor de presencia y una salida para la activación del actuador.

3.2.4 Diseño de los circuitos de acondicionamiento

Acondicionamiento para los sensores

Para la selección de la fuente de alimentación del circuito se debe considerar el voltaje de alimentación de los sensores de presencia, el cual es de 5-24VDC. Por lo que se escogió una batería de 9VDC.

Para las señales de los sensores de presencia y magnético no fue necesario realizar circuitos de acondicionamiento, ya que se conecta la señal de salida directamente a los pines de entrada digital del adaptador Xbee DIO.

Acondicionamiento para los actuadores

Para la activación de los actuadores se debe realizar un circuito de acondicionamiento con las salidas digitales del adaptador Xbee DIO.

En la selección de los actuadores se tomó en cuenta que se tiene una fuente de alimentación de 9VDC, se utilizó la misma fuente de alimentación de los sensores para optimizar los recursos existentes. Los actuadores que se utilizaron son relés de 5VDC, por lo que se debe utilizar un regulador de voltaje para alimentar a los relés.

Se utilizó un transistor npn, el cual puede ser utilizado en amplificación o como switch o interruptor. En la Figura 21 se muestra el circuito de acondicionamiento, en este circuito se utilizó el transistor 2N3904 como switch para la activación del relé. Para evitar que el transistor pueda fundirse durante

el paso de la zona de corte a la de saturación, debido a la carga inductiva de la bobina del relé, se utilizó el diodo 1N4007 para proteger al transistor.

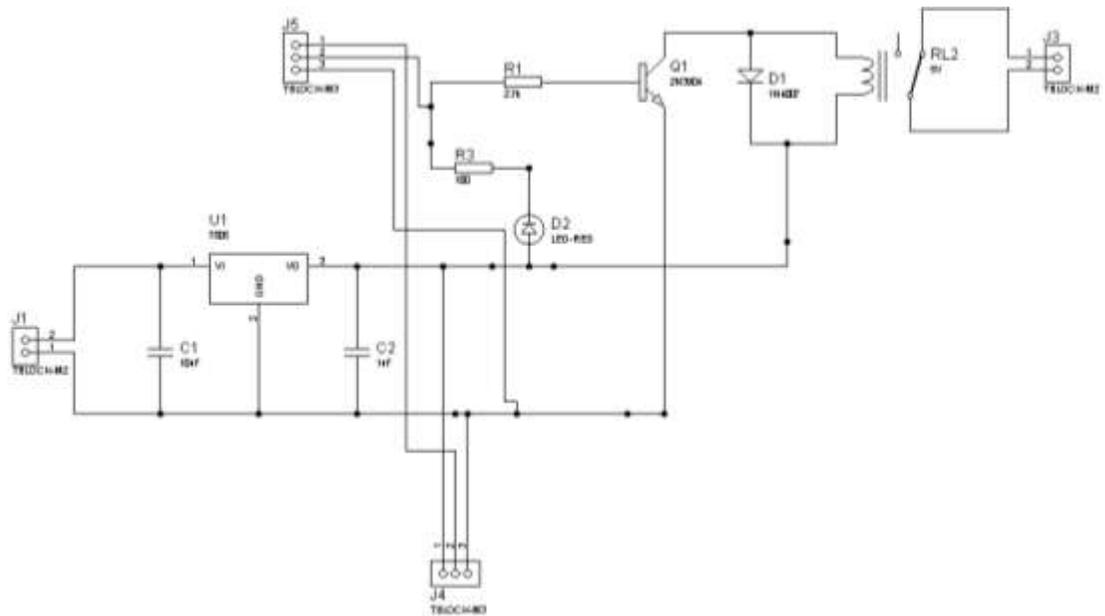


Figura 21. Circuito de acondicionamiento del actuador

3.3 CONFIGURACIÓN DE LOS DISPOSITIVOS DE LA RED

Para comenzar a configurar los dispositivos es necesario conectar y encender el ConnectPort X4 Gateway, se utiliza un cable Ethernet, el un extremo se conecta al ConnectPort X4 Gateway, y el otro extremo al módem que provee internet.

El Xbee Wall Router y los adaptadores Xbee DIO se conectan a un tomacorriente, y se espera a que el led verde se ilumine, esto indica que los dispositivos están asociados a la red Xbee. La configuración de los dispositivos

finales y del nodo coordinador se realiza mediante la interfaz web, el router no necesita configuración.

Configuración del dispositivo coordinador

El nodo coordinador es el ConnectPort X4 Gateway, se configura para que transmita hacia todos los dispositivos de la red a través del canal d. El identificador de la red o PAN ID es 1, los parámetros más importantes configurados en su interfaz web se muestran en la Figura 22.



Figura 22. Configuración del dispositivo coordinador

Fuente (Interfaz web Xbee)

Configuración de los dispositivos finales

Los dispositivos finales se configuran con el mismo identificador de red y canal de operación, además se debe configurar los parámetros DH y DL direccionándolos al nodo coordinador como se muestra en la Figura 23.

XBee Configuration Return to Network View Previous Next

Extended Address: 00:13:a2:00:40:3d:0c:7e!
 Product Type: Wall Router
 Firmware Version: 0x2241

Basic Settings

Advanced Settings

The following are advanced settings. Use caution - modifying some settings can make remote devices inaccessible from the gateway.

Aggregation route notification (AR): 255 x 10 sec (0-255)
 Associate LED blink time (LT): 0 x 10 msec (10-255)
 Broadcast radius (BR): 0 (0-32)
 Command sequence character (CC): * (char)
 Cluster identifier (CI): 0x11 (0x0-0xFF)
 Command mode timeout (CT): 100 x 100 msec (2-655)
 Destination address (DH/DL): 00:13:a2:00:40:3d:1b:9f! (address) **Dirección del coordinador**
 Destination endpoint (DE): 0xa8 (0x1-0xFF)
 AD0/DIO0 configuration (D0): 1 (0-3)
 AD1/DIO1 configuration (D1): 2 (0-3)
 AD2/DIO2 configuration (D2): 2 (0-3)
 AD3/DIO3 configuration (D3): 2 (0-3)
 AD4/DIO4 configuration (D4): 0 (0-3)
 DIO5/Assoc configuration (D5): 1 (0-3)
 DIO6 configuration (D6): 0 (0-3)

Figura 23. Configuración del dispositivo final

Fuente (Interfaz web Xbee)

Una vez terminada la configuración de todos los dispositivos, se puede visualizar los nodos asociados a la red en la Figura 24.

XBee Network

Gateway Device Details

PAN ID: 0xf06b - 0x0000000000000001
 Channel: 0x0d (2415 MHz)
 Gateway Address: 00:13:a2:00:40:3d:1b:9f!

Network View of the XBee Devices

Node ID	Network Address	Extended Address	Node Type	Product Type
[e704]!		00:13:a2:00:40:30:de:fd!	router	Digital IO Adapter
[f99c]!		00:13:a2:00:40:32:16:45!	router	Digital IO Adapter
[007b]!		00:13:a2:00:40:3d:0c:7e!	router	Wall Router
Prueba	[0000]!	00:13:a2:00:40:3d:1b:9f!	coordinator	X4 Gateway

Clear list before device discovery

Figura 24. Nodos asociados a la red.

Fuente (Interfaz web Xbee)

3.4 DEASARROLLO DEL SOFTWARE

3.4.1 Software de gestión remota

El software utilizado es Digi ESP™ para Python, es un entorno de desarrollo integrado (IDE) basado en Eclipse.

Para realizar la conexión de los dispositivos al Gateway se crea un proyecto DIA como se observa en la Figura 25.

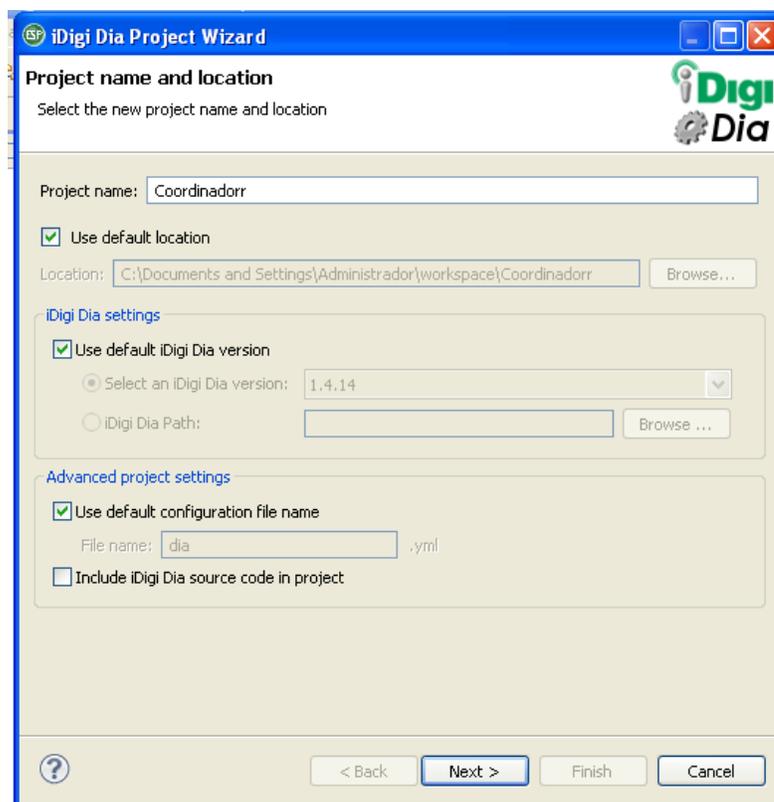


Figura 25. Creación proyecto DIA

Fuente (Software Digi ESP™ para Python)

La aplicación DIA proporciona las funciones básicas de adquisición remota de los datos de los dispositivos conectados al Gateway. Contiene los controladores de los adaptadores Xbee DIO, del Xbee Wall Router y de los canales de las variables del sistema.

En la aplicación también se debe configurar la presentación RCI Handler que permite el acceso a los Servicios Web como se muestra en la Figura 26.



Figura 26. Configuración proyecto DIA.

Fuente (Software Digi ESP™ para Python)

3.4.2 Configuración de los controladores de los dispositivos

Controlador del dispositivo Xbee Wall Router

En este dispositivo únicamente se debe configurar la dirección MAC como se muestra en la Figura 27.

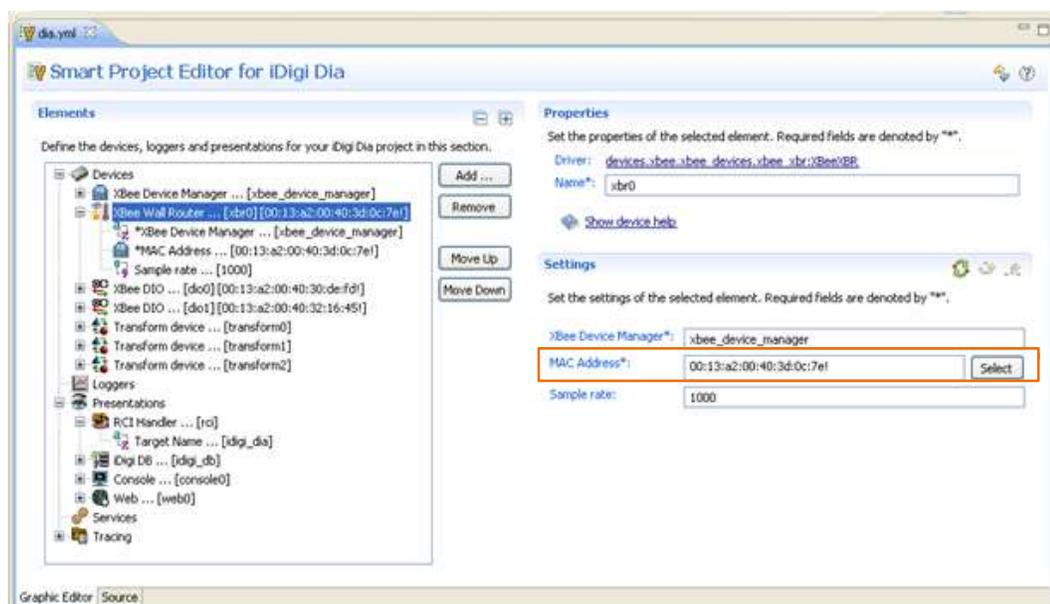


Figura 27. Controlador Xbee Wall Router

Fuente (Software Digi ESP™ para Python)

Controlador del dispositivo Xbee DIO

En este dispositivo se debe configurar la dirección MAC y el modo de los 4 canales, ya sea como entradas o salidas digitales. Se configura el canal 1 y 2 como entradas digitales, y el canal 3 y 4 como salidas digitales, tal como se muestra en la Figura 28.

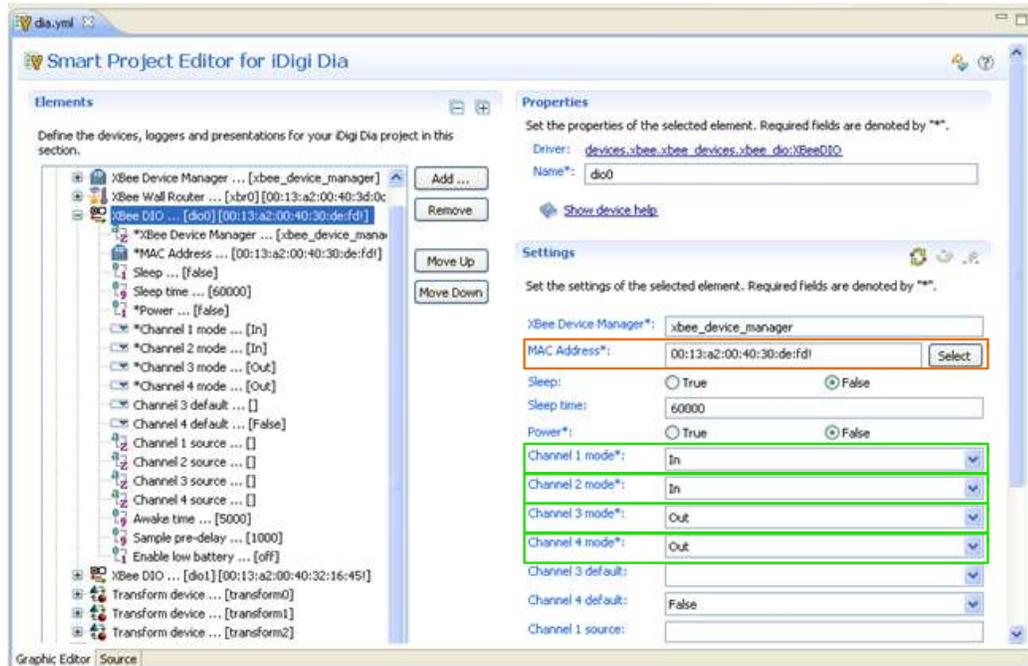


Figura 28. Controlador del adaptador Xbee DIO

Fuente (Software Digi ESP™ para Python)

Controlador de los canales de las variables del sistema

Para las variables se utiliza el controlador Transform Device, no es un dispositivo físico, pero permite crear un canal booleano cuyo valor se da a través de una expresión de comparación donde el resultado es verdadero o falso.

- **Variable control iluminación modalidad día**

Se debe configurar el nombre del nuevo canal, en este caso es *cid*, y agregar los canales que utilizamos como entradas de la expresión de comparación. Para la variable *cid* utilizamos los canales mostrados en la Tabla 4.

Tabla 4

Canales utilizados en la variable *cid*

NOMBRE DEL CANAL	DISPOSITIVO	EXPRESIÓN
xbr0.light	Xbee Wall router	c[0]<30 and c[1]==True
dio1.channel1_input	Adaptador Xbee DIO	

El valor de cada canal se conoce como “c[0]” en la expresión, si se añade un segundo canal es “c[1]” y así según el número de canales que se utilice.

El canal *xbr0.light* proporciona el valor de luminosidad, mientras que el canal *dio1.channel1_input* proporciona el valor de la señal del sensor de presencia. Una vez finalizada la configuración se crea un nuevo canal llamado “transform0.cid”.

En la Figura 29 se muestra la configuración final del controlador del canal.

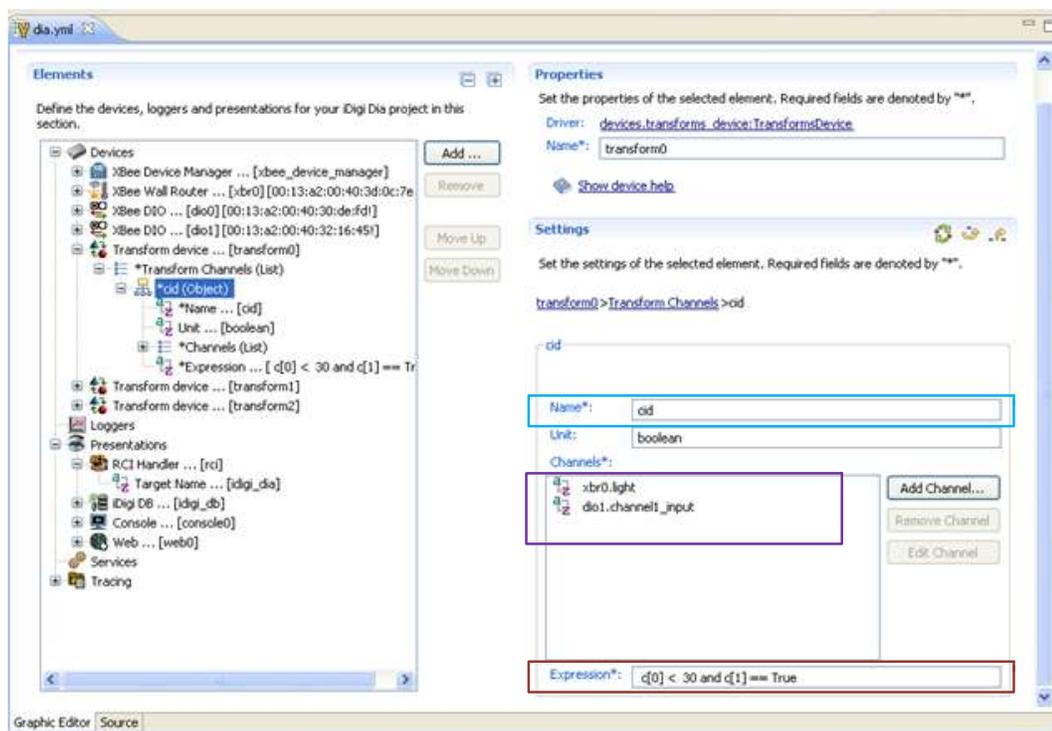


Figura 29. Controlador del canal transform0.cid

Fuente (Software Digi ESP™ para Python)

- **Variable control iluminación modalidad noche**

Se debe configurar el nombre del nuevo canal, en este caso es cin, y agregar el canal que utilizamos como entrada de la expresión de comparación.

Se añade el canal *dio1.channel1_input* que proporciona el valor de la señal del sensor se presencia. Una vez finalizada la configuración se crea un nuevo canal llamado “transform1.cin”.

En la Figura 30 se muestra la configuración final del controlador del canal.

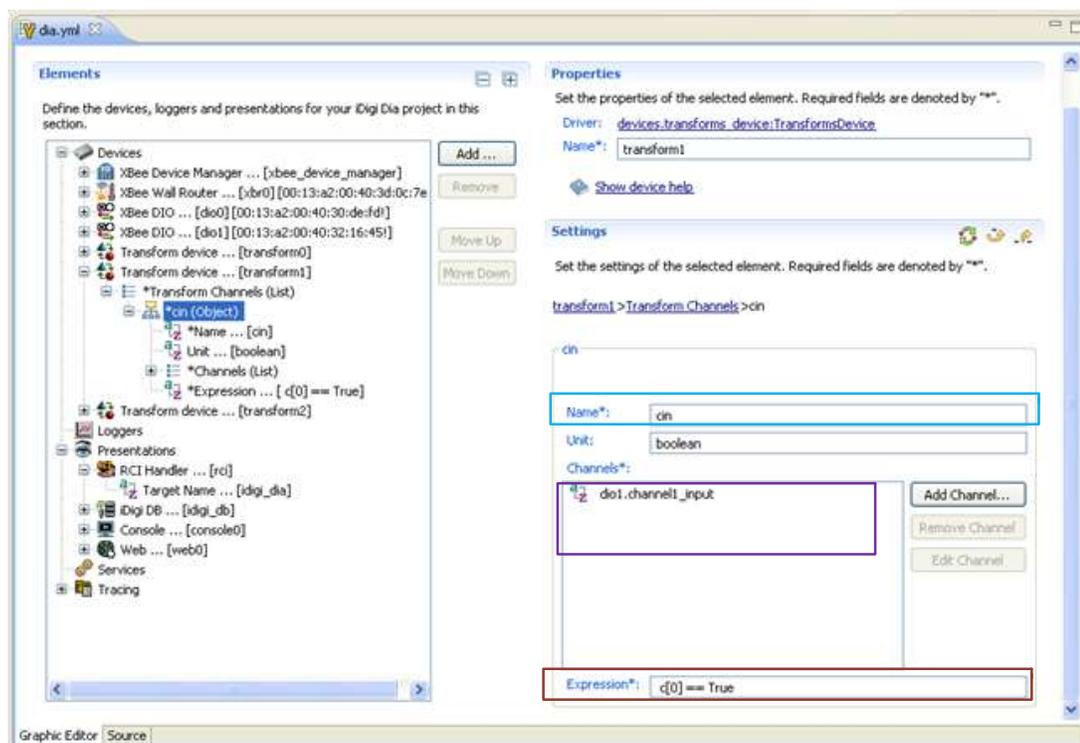


Figura 30. Controlador del canal transform1.cin

Fuente (Software Digi ESP™ para Python)

- **Variable detector de intrusión**

Se debe configurar el nombre del nuevo canal, en este caso es ddi, y agregar los canales que utilizamos como entradas de la expresión de comparación. Para la variable ddi se utiliza los canales mostrados en la Tabla 5.

Tabla 5

Canales utilizados en la variable ddi

NOMBRE DEL CANAL	DISPOSITIVO	EXPRESIÓN
dio0.channel1_input	Adaptador Xbee DIO	c[0]==True and c[1]==True
dio0.channel2_input	Adaptador Xbee DIO	

El canal *dio0.channel1_input* proporciona el valor de la señal del sensor de presencia, mientras que el canal *dio0.channel2_input* proporciona el valor de la señal del sensor magnético. Una vez finalizada la configuración se crea un nuevo canal llamado “transform2.ddi”.

En la Figura 31 se muestra la configuración final del controlador del canal.

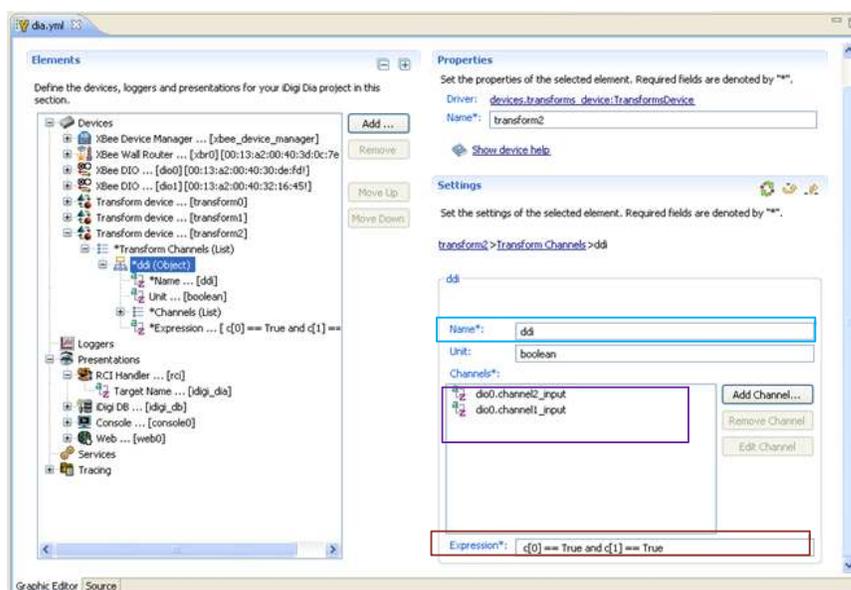


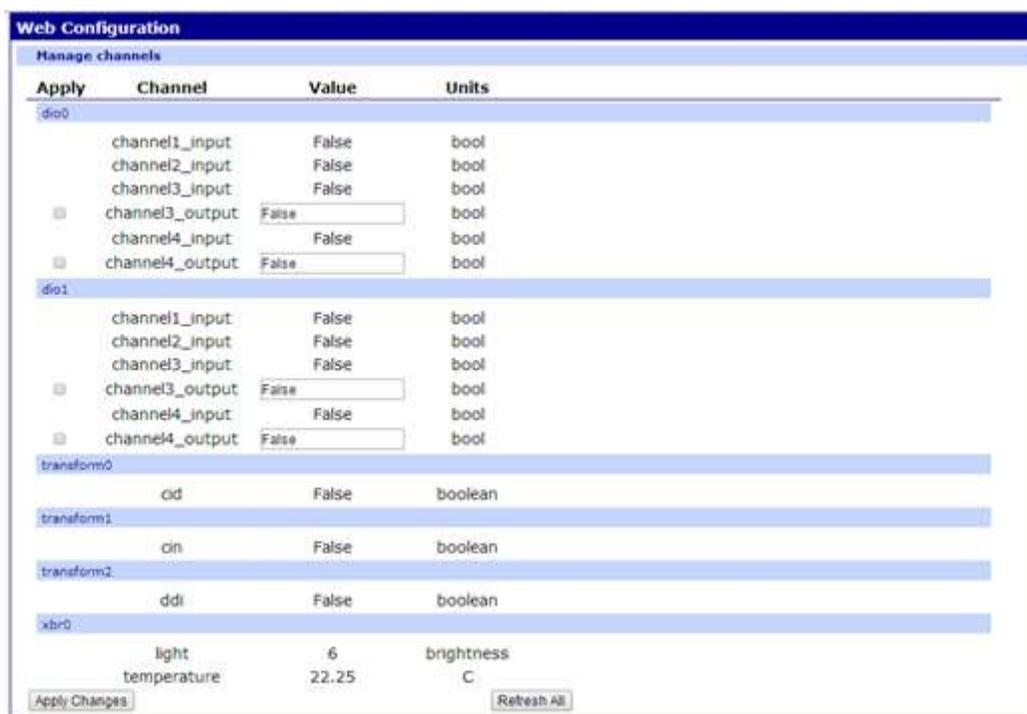
Figura 31. Controlador del canal transform2.ddi

Fuente (Software Digi ESP™ para Python)

3.4.3 Ejecución de la aplicación DIA

Una vez que la aplicación DIA está completamente configurada, dar un clic en el botón Run  y seleccionar la opción Remote iDigi Dia para ejecutar la aplicación en el ConnectPort X4 Gateway.

En la Figura 32, se muestra la ventana adicional que aparece una vez que se ejecuta la aplicación, esta ventana es la Presentación Web de la aplicación DIA, que permite interactuar con los dispositivos Xbee y visualizar las lecturas actuales de los sensores.



Apply	Channel	Value	Units
Manage channels			
dio0			
	channel1_input	False	bool
	channel2_input	False	bool
	channel3_input	False	bool
<input type="checkbox"/>	channel3_output	False	bool
	channel4_input	False	bool
<input type="checkbox"/>	channel4_output	False	bool
dio1			
	channel1_input	False	bool
	channel2_input	False	bool
	channel3_input	False	bool
<input type="checkbox"/>	channel3_output	False	bool
	channel4_input	False	bool
<input type="checkbox"/>	channel4_output	False	bool
transform0			
	od	False	boolean
transform1			
	on	False	boolean
transform2			
	ddi	False	boolean
xbr0			
	light	6	brightness
	temperature	22.25	C
<input type="button" value="Apply Changes"/>		<input type="button" value="Refresh All"/>	

Figura 32. Presentación web de la aplicación DIA

Fuente (Software Digi ESP™ para Python)

3.4.4 Conexión de los dispositivos Xbee al servidor

Para poder acceder y descargar los datos desde la nube de dispositivos Etherios se debe crear una cuenta de usuario. Al crear la cuenta gratuita se accede al Administrador de dispositivos y se puede conectar hasta 5 dispositivos coordinadores.

En la ventana Device Management se agrega el dispositivo ConnectPort X4 Gateway, seleccionando su dirección MAC. El Device Cloud añade el dispositivo al inventario, el cual aparecerá conectado como se muestra en la Figura 33.

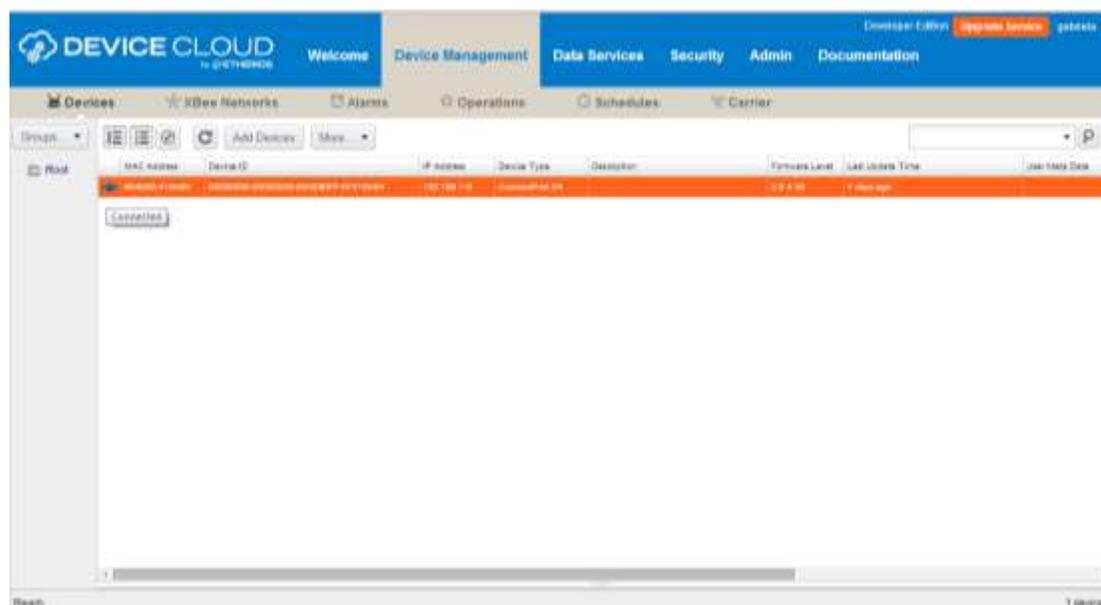


Figura 33. Interfaz Nube de dispositivos

Fuente (Nube de dispositivos Etherios)

La nube de dispositivos permite gestionar de forma segura los dispositivos en su inventario. Seleccionando la pestaña Xbee Networks se puede visualizar los dispositivos conectados a la red Xbee como se muestra en la Figura 34.

Device ID	Node ID	Module Type	Product Type	User Type	Role	Current Address	Power Address
M 13 42 38 44 38 2E 10	00000000000000000000000000000000	Xbee DS	High ID Address	000000	Master	0400	0400
M 13 42 38 44 38 2E 10	00000000000000000000000000000000	Xbee DS	Low Node	000000	Slave	0400	0400
M 13 42 38 44 38 2E 10	00000000000000000000000000000000	Xbee DS	Low Node	000000	Coordinator	0400	0400

Figura 34. Dispositivos conectados a la red Xbee

Fuente (Nube de dispositivos Etherios)

3.4.5 Almacenamiento de datos

Los datos que maneja la aplicación se almacenan en ficheros xml. Estos ficheros se guardan en la nube de dispositivos Etherios. Para observar los datos almacenados es necesario realizar una petición sci de servicios web.

La ventana API Explorer permite enviar solicitudes de servicios web al dispositivo registrado en la nube. Para recuperar las lecturas de los datos de los dispositivos se envía una solicitud HTTP POST como se muestra en la Figura 35.

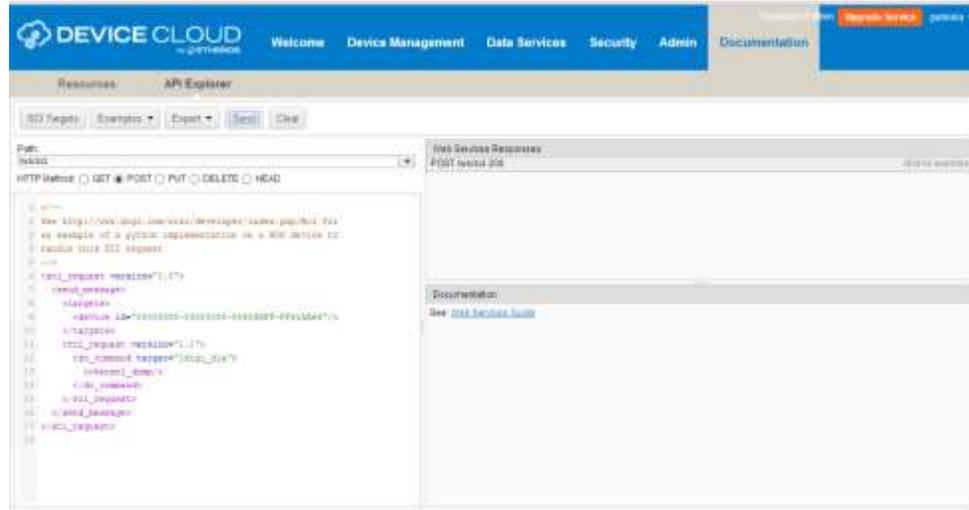


Figura 35. Solicitud HTTP POST

Fuente (Nube de dispositivos Etherios)

Una vez enviada la solicitud HTTP POST al dispositivo de la nube, este responderá con las lecturas actuales de los datos de los dispositivos como se muestra en la Figura 36.

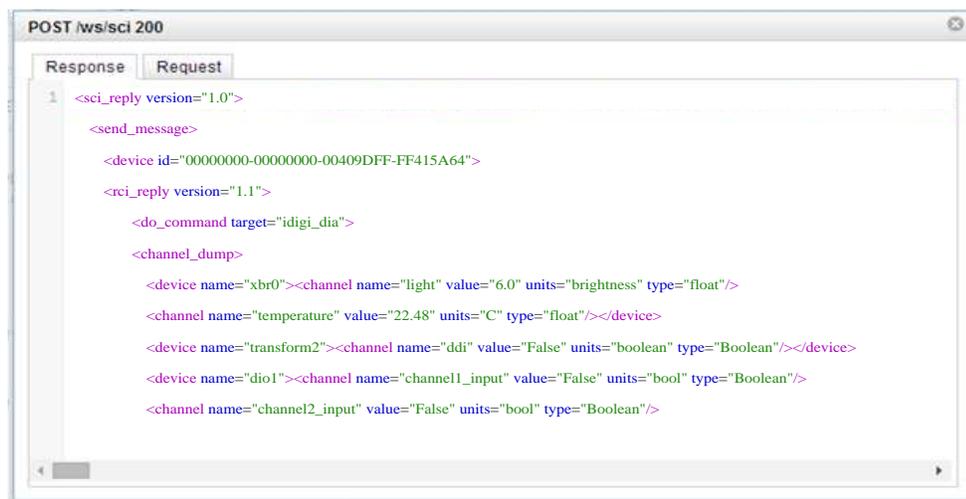


Figura 36. Fichero almacén de datos

Fuente (Nube de dispositivos Etherios)

3.5 DESARROLLO DE LA INTERFAZ DE USUARIO

Se ha desarrollado una aplicación web cliente en Google App Engine. Se escogió esta plataforma como servicio porque permite implementar aplicaciones web utilizando el lenguaje de programación Python. Además Google tiene integración con Eclipse y al tener el software Digi ESP™ para Python basado en Eclipse se lo puede utilizar para desarrollar la aplicación.

3.5.1 Descripción de la plataforma Google App Engine

Google App Engine es una plataforma que ofrece facilidades para el diseño, desarrollo, testing, despliegue y alojamiento de aplicaciones. Con Google App Engine no es necesario utilizar un servidor, solo hay que subir la aplicación para que los usuarios puedan empezar a utilizarla (Google Developers).

Para desarrollar aplicaciones en Google App Engine hay que crear una cuenta de usuario la misma que proporciona un conjunto básico de recursos de forma gratuita. Se puede ejecutar hasta 10 aplicaciones, cada aplicación puede utilizar hasta 500 MB de almacenamiento, 1GB de almacenamiento en el datastore, privilegios para enviar 2000 mensajes de correo electrónico por día y suficiente CPU y ancho de banda para permitir un servicio eficaz (Google Developers). En la Figura 37 se muestra la interfaz de la plataforma GAE.

Además desde la plataforma se puede acceder a los siguientes servicios:

- Almacén de datos (datastore), donde se almacenar toda la información que la aplicación necesite para funcionar.
- Extracción de URL, que permite acceder a recursos de internet.
- Correo, ya que las aplicaciones podrán enviar correos electrónicos.
- Programación de tareas, servicios que se vayan a ejecutar a intervalos regulares.

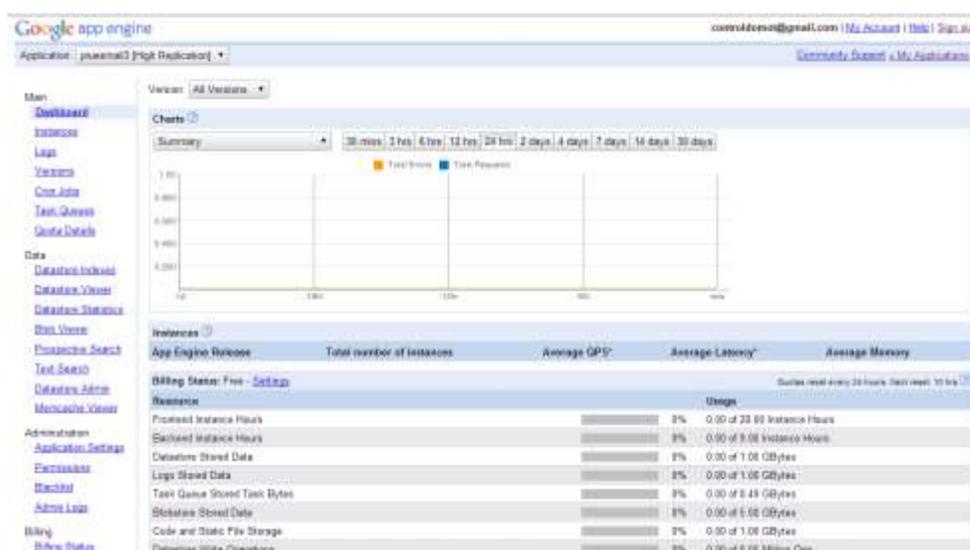


Figura 37. Interfaz de la plataforma Google App Engine

Fuente (Google App Engine)

3.5.2 Creación de cuenta en Google App Engine

Para obtener una cuenta en Google App Engine, es necesario tener una cuenta de correo electrónico en Gmail, para el desarrollo de la aplicación se ha creado la cuenta controldomot@gmail.com.

Con la cuenta creada se debe ingresar en <https://appengine.google.com/> para registrarse y poder crear las aplicaciones. Al iniciar la sesión se tiene acceso a la consola de administración de las aplicaciones como se observa en la Figura 38.



Figura 38. Consola de Administración de las aplicaciones GAE

Fuente (Google App Engine)

Para crear la aplicación hacer clic en el botón *Create Application*, la ventana de creación se muestra en la Figura 39. En el campo *Application Identifier*, se ingresa el nombre de la aplicación. Se debe verificar que el nombre esté disponible, y después hacer clic en el botón *Create Application*.

El dominio asignado a la aplicación es: <http://.....appspot.com>. Ahora se debe programar la aplicación y descargar el App Engine SDK para Python, cuyas librerías se utilizan para desarrollar la aplicación.

Google app engine

conradmoro@gmail.com | My Account | Help | Sign out

Create an Application

You have 23 applications remaining.

Application Identifier: [Check Availability](#)

All Google accounts names and options offered on independent services may not be used as Application Identifiers. You can map the application to your sub-domain like [LINK:000](#).

Application Title:

Default users access your application.

Authentication Options (Advanced) [Learn more](#)

Google App Engine provides an API for authenticating your users, including Google Accounts, Google Apps, and OpenID. If you choose to use the feature for some parts of your site, you'll need to specify what type of users you expect to use your application.

Open to all Google Accounts users (default)
If your application uses authentication, anyone with a valid Google Account may sign in.

Restricted to the following Google Apps domain:

e.g. foo.com
If your application uses authentication, only members of this Google Apps domain may sign in. If your application uses Google Apps, use the option to create an application (e.g. an API tracking tool) that is only accessible to accounts on your Google Apps domain. This option cannot be changed once it has been set.

(Experimental) Open to all users with an OpenID Provider
If your application uses authentication, anyone who has an account with an OpenID Provider may sign in.

© 2014 Google | Terms of Service | Privacy Policy | Help | Discussion Forum | Support | Docs

Figura 39. Página para la creación de aplicaciones en GAE

Fuente (Google App Engine)

3.5.3 Diseño de la aplicación web

La aplicación web muestra el funcionamiento de una red ZigBee. La aplicación es capaz de leer y escribir parámetros de los módulos Xbee de forma remota. Utiliza servicios web (XML) para comunicarse al Gateway de Digi usando la nube de dispositivos Etherios.

La aplicación web consta de una interfaz gráfica muy amigable con el usuario, en la cual se tiene el control y monitorización de los dispositivos de la red.

A continuación se explica el funcionamiento del entorno que consta básicamente de 5 partes que son:

- Ingreso de contraseña
- Página principal
- Pestaña de Monitoreo
- Pestaña de Control
- Página de Error

Ingreso de contraseña

Es la primera página que se despliega al tener acceso mediante un navegador web, esta página se muestra en la Figura 40. Solicita al administrador del sistema el nombre de usuario, la contraseña y los 4 últimos dígitos de la dirección MAC del ConnectPort X4 Gateway. Esta página proporciona la seguridad necesaria al usuario para el acceso al prototipo del sistema domótico.



Figura 40. Interfaz ingreso de contraseña

Página Principal

Al iniciar la sesión se carga la página principal, tiene una barra de menú que contiene tres pestañas: Inicio, Monitoreo y Control.

La pestaña Inicio muestra una rápida presentación del proyecto. Además en la parte inferior tiene dos botones, uno para recargar la página y otro para salir del sistema como se muestra en la Figura 41.



Figura 41. Interfaz página principal

Pestaña de Monitoreo

La pestaña Monitoreo contiene dos secciones como se muestra en la Figura 42, la primera corresponde al monitoreo del nivel de luminosidad y del valor de la temperatura del ambiente en grados centígrados. La segunda sección

permite seleccionar la modalidad para el control de iluminación ya sea Día o Noche. Además se puede activar o desactivar el detector de intrusión e ingresar el correo electrónico del destinatario que recibirá el correo de notificación cuando se active la alarma.



Figura 42. Interfaz pestaña Monitoreo

Pestaña de Control

La pestaña Control contiene dos secciones, la primera corresponde al control de iluminación y la segunda al detector de intrusión como se muestra en la Figura 43.

En la sección de Iluminación se visualiza el modo de configuración, el nivel de luminosidad y si el foco permanece prendido o apagado dependiendo del modo seleccionado. Además permite el control manual o automático de la luz.

Al seleccionar el control manual aparece dos botones ON, OFF, que se utilizan para encender o apagar el foco. Si se selecciona el control automático se debe ingresar la hora de encendido del foco y hacer clic en el botón Activar/Desactivar.



Figura 43. Interfaz pestaña Control

En la sección de Intrusión se visualiza si el detector de intrusión de encuentra activado o desactivado, además se muestra el estado de la señal de los sensores, cuando los sensores se activan el indicador cambia de color plomo a verde. Igualmente cuando la alarma se activa su indicador cambia de color plomo a rojo y se activa una ventana de alarma indicando la activación de la misma.

Página de Error

En esta ventana se visualiza los distintos errores por los cuales no se despliega la aplicación como se muestra en la Figura 44. Estos errores pueden ser:

- Falla de descarga de la aplicación DIA, compruebe que el Gateway está encendido.
- Verifique que la dirección de su Gateway es correcta.
- Compruebe que el archivo DIA está funcionando y configurado con la presentación RCI.



Figura 44. Interfaz página de Error

3.5.4 Programación de la aplicación web

3.5.4.1 Lenguajes de programación utilizados

Para la programación de la aplicación web se ha empleado cuatro lenguajes de programación distintos: Python, JavaScript, HTML y CSS.

Python

La aplicación GAE se desarrolla en Python 2.7. Python es un lenguaje de programación independiente de la plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones Windows a servidores de red o incluso páginas web (Van Rossum, 2009).

JavaScript

JavaScript es un lenguaje de programación que se utiliza para crear páginas web dinámicas. Incorpora efectos, animaciones y ventanas con mensajes de aviso para mejorar la experiencia del usuario. Se utilizó la librería jQuery para manejar eventos y agregar interacción con Ajax a la página web.

HTML y CSS

HTML es el lenguaje que permite dar estructura a la página web, se usa para añadir párrafos, encabezados, imágenes. CSS es un lenguaje de estilo que define la presentación del documento HTML.

3.5.4.2 Archivos de la aplicación

En la Figura 45 se muestra los archivos de la aplicación. Estos son:

- Archivos de configuración
- Archivos de Python
- Archivos de JavaScript
- Archivos HTML
- Archivo CSS
- Una carpeta con imágenes

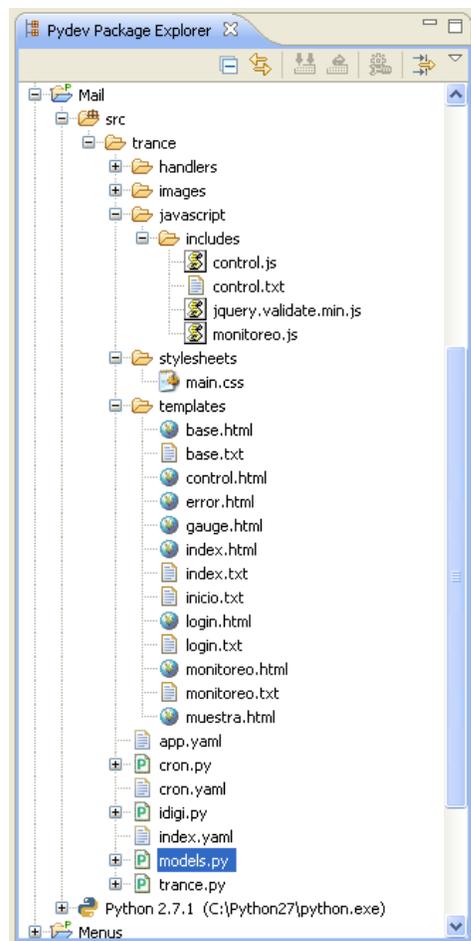


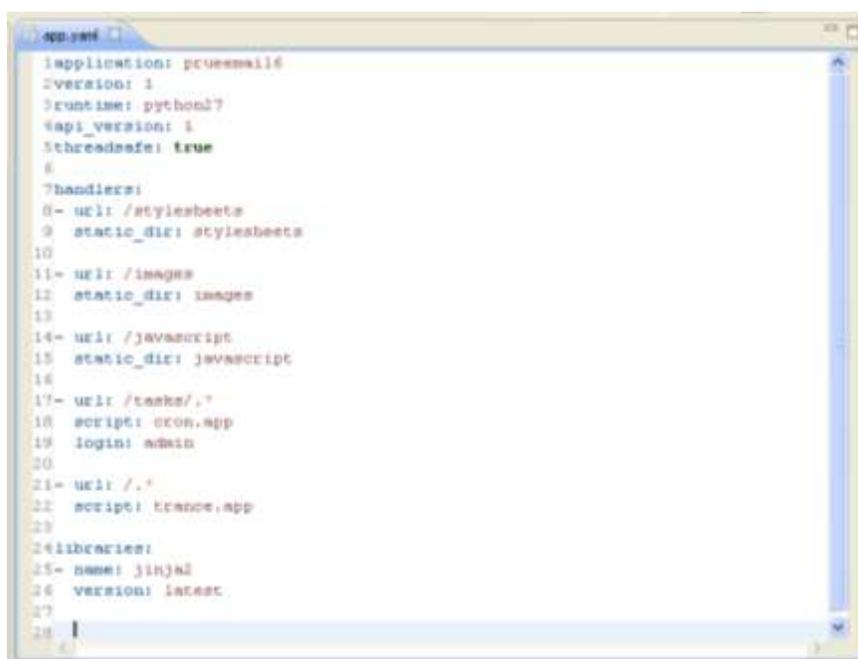
Figura 45. Archivos de la aplicación

Fuente (Software Digi ESP™ para Python)

Archivos de configuración

- Archivo `app.yaml`

La aplicación creada con *App Engine* tiene un archivo de configuración denominado `app.yaml`. Este archivo especifica las rutas URL que corresponden a los controladores de solicitudes y archivos estáticos. Contiene el nombre de la aplicación, handlers y librerías como se muestra en la Figura 46.

A screenshot of a text editor window titled 'app.yaml'. The window displays the following YAML configuration code:

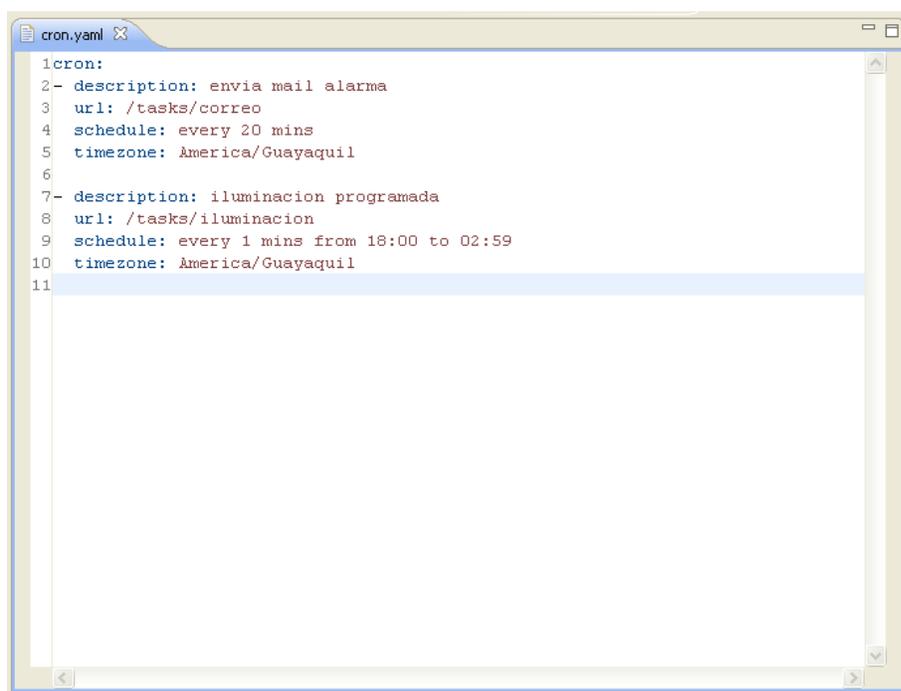
```
1 application: pcodemail6
2 version: 1
3 runtime: python27
4 api_version: 1
5 threadsafe: true
6
7 handlers:
8   0- url: /stylesheets
9     static_dir: stylesheets
10
11- url: /images
12   static_dir: images
13
14- url: /javascript
15   static_dir: javascript
16
17- url: /tasks/*
18   script: cron.app
19   login: admin
20
21- url: /*
22   script: trance.app
23
24 libraries:
25- name: jinja2
26   version: latest
27
28
```

Figura 46. Archivo `app.yaml`

Los handlers contienen las listas de direcciones URL a manejar durante la ejecución del código de la aplicación, o de los archivos estáticos que se cargan con el código, como imágenes, CSS y JavaScript. En las librerías se incluyó `jinja2` para el manejo de las plantillas en la aplicación (Google Developers).

- **Archivo cron.yaml**

El archivo cron.yaml permite configurar tareas programas con regularidad que operan en momentos definidos. Los trabajos de cron se activan automáticamente por el servicio *App Engine Cron*, sin necesidad de ingresar a la aplicación web (Google Developers).



```
1cron:
2- description: envia mail alarma
3  url: /tasks/correo
4  schedule: every 20 mins
5  timezone: America/Guayaquil
6
7- description: iluminacion programada
8  url: /tasks/iluminacion
9  schedule: every 1 mins from 18:00 to 02:59
10 timezone: America/Guayaquil
11
```

Figura 47. Archivo cron.yaml

En la aplicación se programó dos tareas cron como se muestra en la Figura 47. La primera tarea envía un correo electrónico notificando la activación de la alarma del detector de intrusión, esta tarea se realiza cada 5 minutos. La segunda tarea programa el control automático de la luz y se realiza cada minuto, desde las 18:00 hasta las 07:00.

Archivos de Python

Los archivos escritos en Python definen todas las funciones necesarias para que la aplicación web maneje de manera adecuada el intercambio de datos con el código HTML y la comunicación con la nube de dispositivos Etherios.

Para permitir la configuración remota, control e intercambio de información entre un servicio web y un dispositivo de Digi se utiliza el Remote Command Interface (RCI) (Etherios).

Se creó una carpeta handlers para manejar los controladores que permiten la autenticación de usuario con el servidor de la nube y el contenido que se muestra en la plantilla HTML. Estos archivos son:

- **base_handler.py**

El archivo base_handler.py permite mostrar el contenido en las plantillas. Se define la clase Basehandler, dentro de esta se encuentran las funciones get_user y do_render. La función get_user verifica la sesión de usuario, si no existe usuario se redirige a la página de inicio de sesión. La función do_render toma los valores que se definen en el diccionario y los escribe en la plantilla HTML.

- **auth.py**

El archivo auth.py maneja la autenticación con el servidor de la nube de dispositivos Etherios. Se define las clases LogoffHandler y LoginHandler. La clase LogoffHandler permite cerrar la sesión de usuario y redirigir a la página de

ingreso. La clase LoginHandler maneja la entrada de los administradores de la aplicación. Posee una función post que lee los datos introducidos en el formulario HTML, si los datos se autentifican correctamente en el filtro de Device Cloud se conecta con éxito al servidor, si no se encuentran los datos muestra un mensaje de error y redirige nuevamente a la página de ingreso.

Además de la carpeta handlers se definen otros archivos Python que son:

- **idigi.py**

El archivo idigi.py crea las constantes SCI_VERSION, RCI_VERSION y DEFAULT_IDIGI_URL. Estas constantes especifican la versión del SCI y RCI que se envían en las peticiones RCI a los dispositivos a través de la nube de dispositivos y se encolan en las peticiones del servidor llamado Server Command Interface (SCI). Una solicitud de SCI se compone de XML y envía una solicitud POST a `http(s):// {DEFAULT_IDIGI_URL} /ws/sci`.

Se define las clases iDigiRequestError, Sample y iDigiClient. La clase iDigiRequestError retorna el tipo de error que se produce cuando la aplicación no se puede comunicar con el Gateway. La clase Sample contiene los datos de las variables como el nombre del canal, valor y unidades. La clase iDigiClient permite realizar el intercambio de información entre la aplicación web y la nube de dispositivos Etherios utilizando peticiones RCI. En esta clase se encuentra las funciones `command_data`, `channel_get`, `channel_dump` y `channel_set`.

La función `command_data` retorna el error que se produce en el nivel de solicitud, esto se da cuando existe una solicitud XML inválida, cuando no se reconoce la solicitud o cuando se envía un comando desconocido.

La función `channel_get` retorna una lista con los valores de los canales al momento de realizar la solicitud. La función `channel_dump` retorna una lista con los valores más recientes de los canales y la función `channel_set` permite modificar el valor de un canal.

- **models.py**

El archivo `models.py` crea las constantes `REQUEST_RCI` y `SCI_TMPL`, además define las entidades creadas en la base de datos de la aplicación web. La clase `User` contiene los datos del usuario, como el nombre, contraseña y Gateway.

- **main.py**

El archivo `main.py` contiene las funciones necesarias para que la aplicación web maneje de manera adecuada todos los datos de los dispositivos conectados a la red ZigBee. Además carga los archivos HTML que se visualizarán en la aplicación web.

A continuación se explica cada una de las clases definidas en el archivo `main.py`.

GaugeHandler

La clase `GaugeHandler` accede a la base de datos `Sample` y muestra el nombre del canal con su respectivo valor y unidades.

Manual

La clase Manual define la función post, la cual recoge el parámetro de un radio button que permite encender o apagar el foco. Si el parámetro leído es *On* u *Off*, se envía este valor utilizando la función `channel_set` para modificar el estado del canal. Si el valor no corresponde a *On* u *Off* la función devuelve el error 403, que indica que se ha negado el acceso a la acción que se solicita, esto sirve para filtrar los valores que acepta el canal del Adaptador Xbee DIO. En la Figura 48 se muestra el diagrama de flujo de la clase.

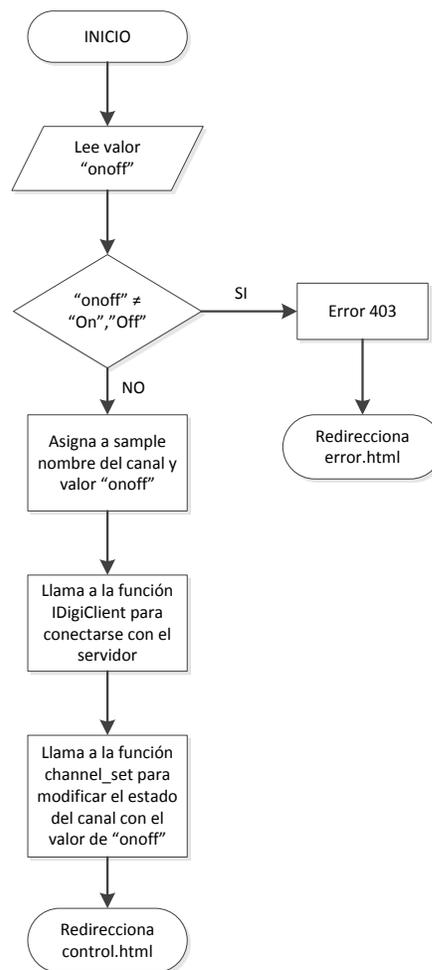


Figura 48. Diagrama de flujo de la clase Manual

Automa

La clase Automa define la función post, la cual recoge los valores de hora y minutos ingresados por el usuario. Se crea un objeto de la clase Hora, y se inicializa los atributos con los datos extraídos de los spinner. Se genera una consulta a la base de datos y se rescata la primera entidad. Se borra los datos de la entidad y se guarda los nuevos datos ingresados. En la Figura 49 se muestra el diagrama de flujo de la clase.

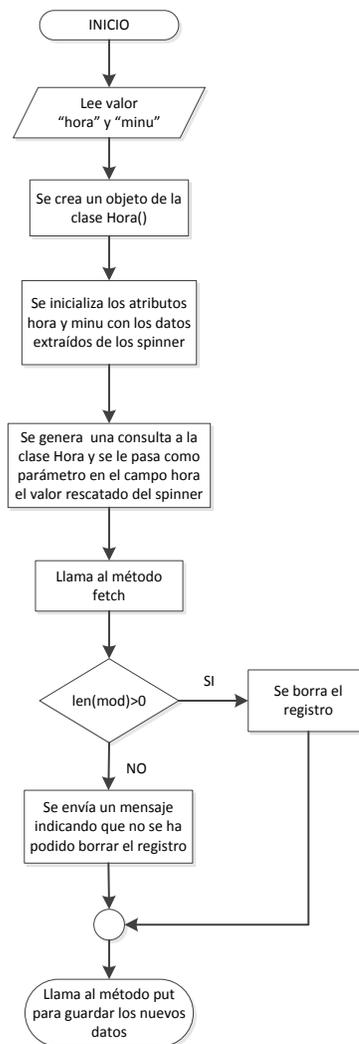


Figura 49. Diagrama de flujo de la clase Automa

Automatico

La clase Automatico define la función get, donde se llama a la función datetime() que devuelve la fecha y hora actual, se separa los valores de horas y minutos para compararlos con los valores almacenados en el objeto de la clase Hora, si los valores son iguales se envía On para activar el pin de salida del adaptador Xbee DIO caso contrario se envía Off. Se utiliza la función channel_set para modificar el estado del canal. En la Figura 50 se muestra el diagrama de flujo de la clase.

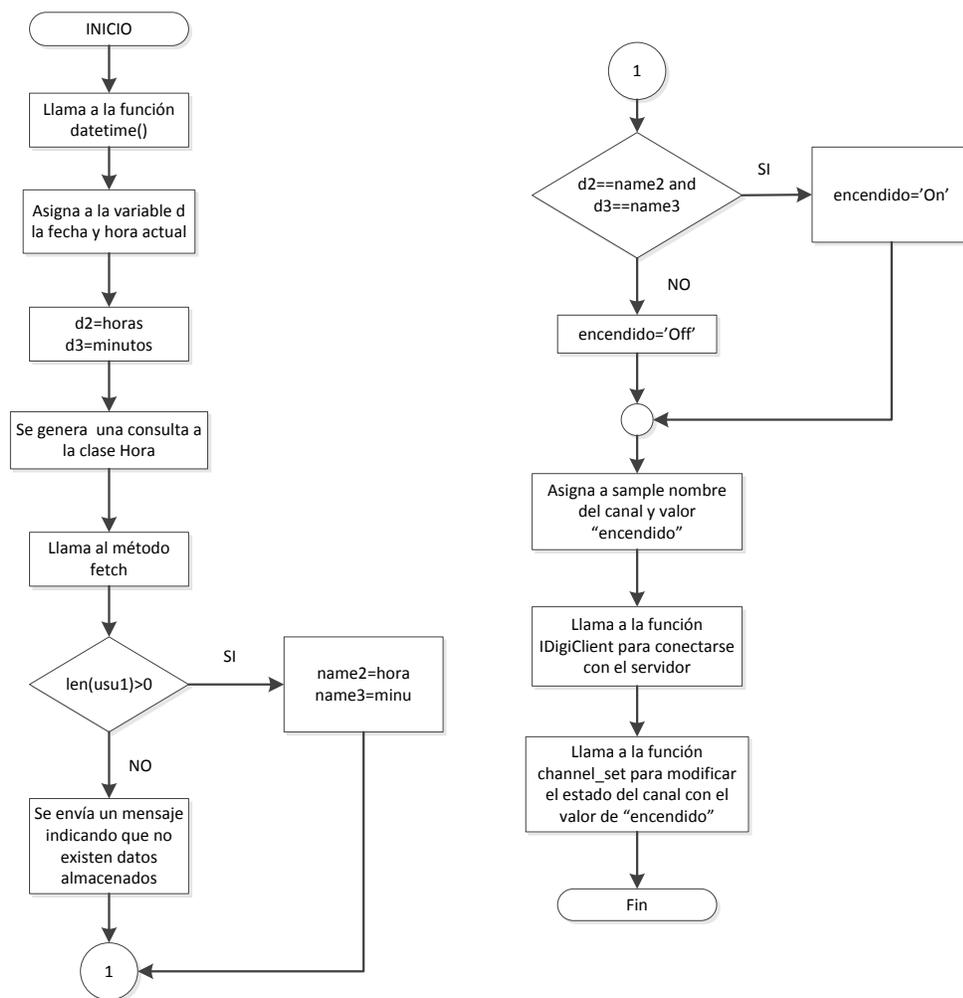


Figura 50. Diagrama de flujo de la clase Automatico

Modos

La clase Modos permite realizar una consulta en la base de datos para conocer la configuración que fue seleccionada día o noche, de acuerdo al valor recogido se lee el valor de los canales del adaptador Xbee DIO y del Wall Router para encender o apagar la luz. En la Figura 51 se muestra el diagrama de flujo de la clase.

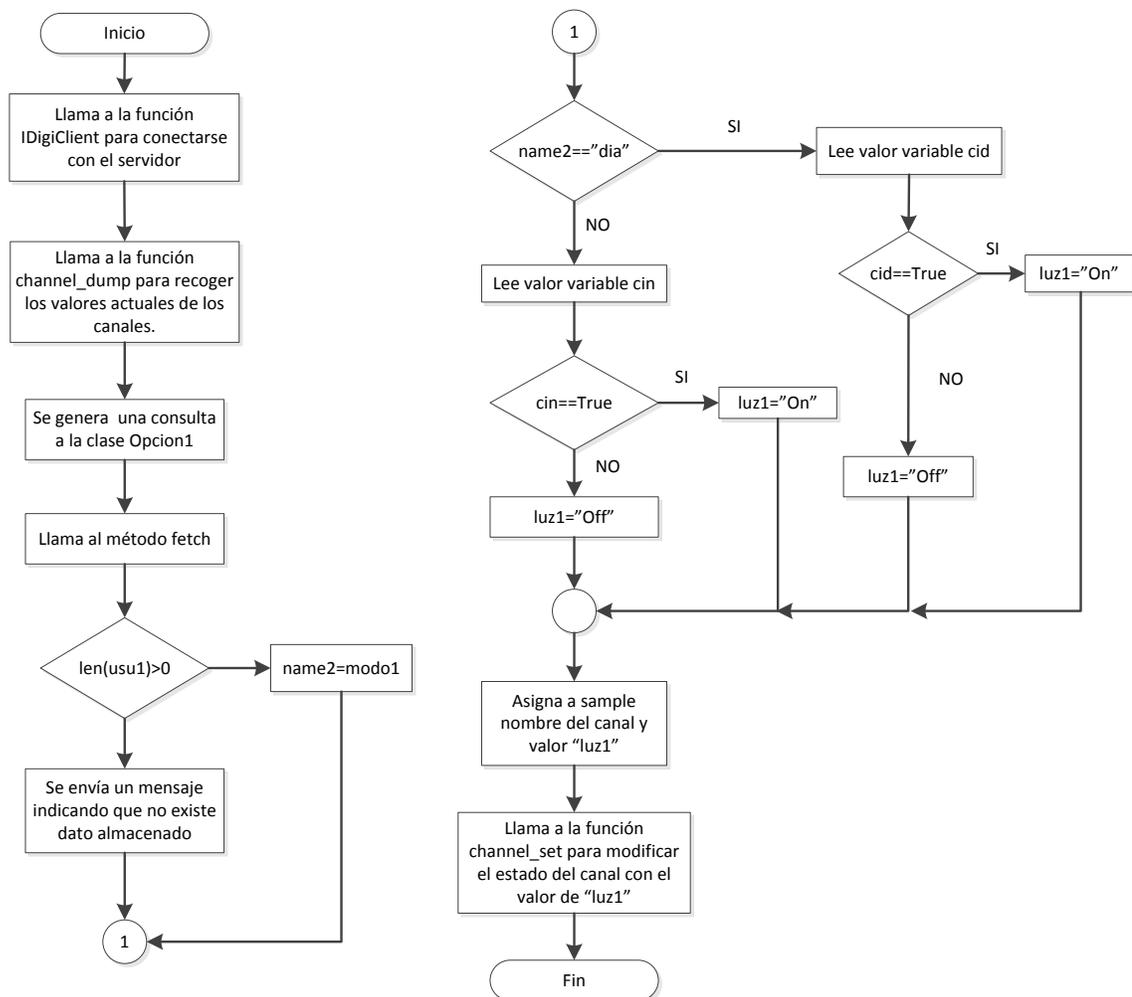


Figura 51. Diagrama de flujo de la clase Modos

Alarma

La clase Alarma realiza una consulta en la base de datos para conocer si el detector de intrusión está activado, si lo está, lee el valor de los canales del adaptador Xbee DIO. Si se activan las señales del sensor magnético y de presencia se llama a la función EnviarMail para enviar el correo de notificación de activación de alarma. En la Figura 52 se muestra el diagrama de flujo de la clase.

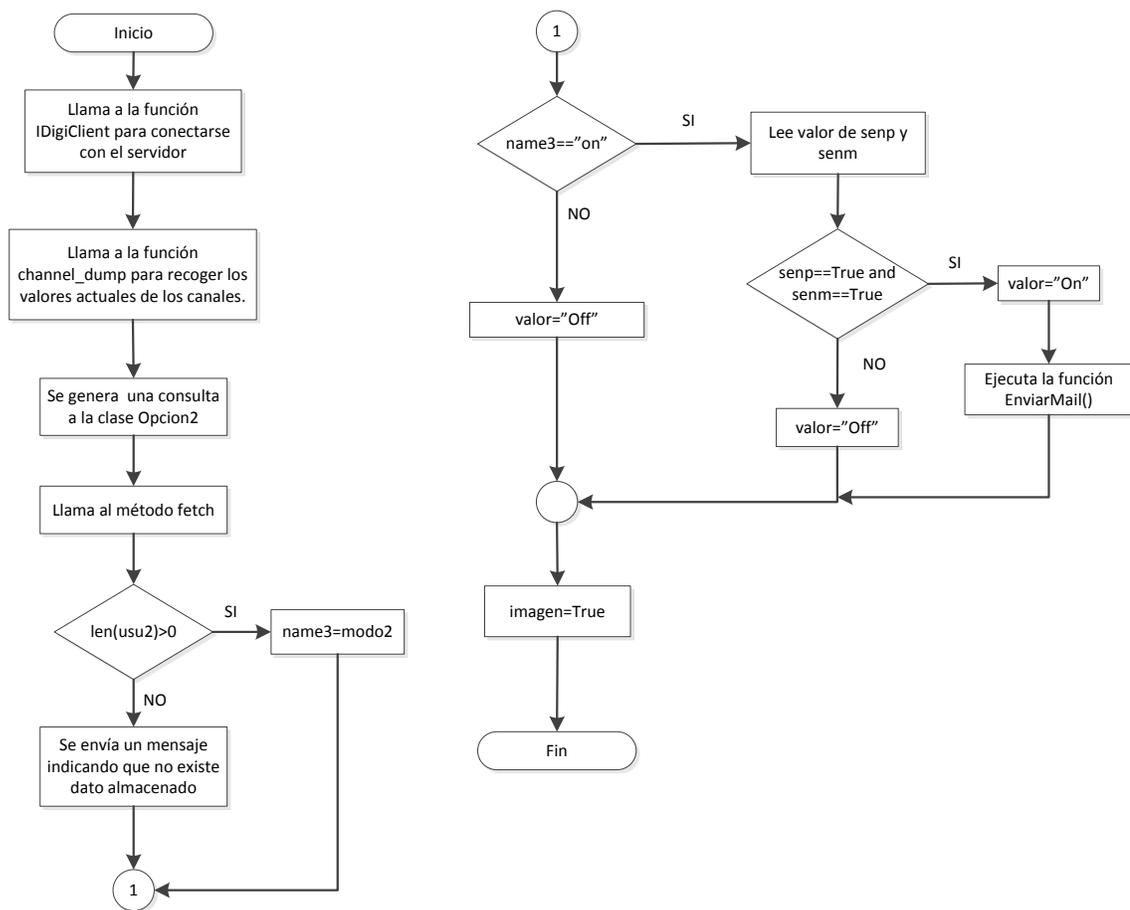


Figura 52. Diagrama de flujo de la clase Alarma

GuardaMail

La clase GuardaMail define la función post, la cual recoge la dirección de correo electrónico ingresado por el usuario. Se crea un objeto de la clase Mail, y se inicializa el atributo correo con el valor extraído del input email. Se genera una consulta a la base de datos y se rescata la primera entidad. Se borra el dato de la entidad y se guarda la nueva dirección de correo ingresada.

EnviarMail

La función EnviarMail genera una consulta a la clase Mail y asigna a name4 la dirección de correo electrónico almacenada en la base de datos. Para enviar el mensaje de correo electrónico se utilizó la clase EmailMessage. Se debe editar los campos destinatario, asunto y cuerpo del mensaje. Al campo destinatario se asigna el valor de la variable name4. Finalmente se utiliza el método send() para enviar el mensaje de correo electrónico.

LoggingHandler

La clase LoggingHandler verifica la existencia de usuario, si no existe usuario re-direcciona a la página login.html, caso contrario almacena el dato y re-direcciona a la página index.html. En la Figura 53 se muestra el diagrama de flujo de la clase.

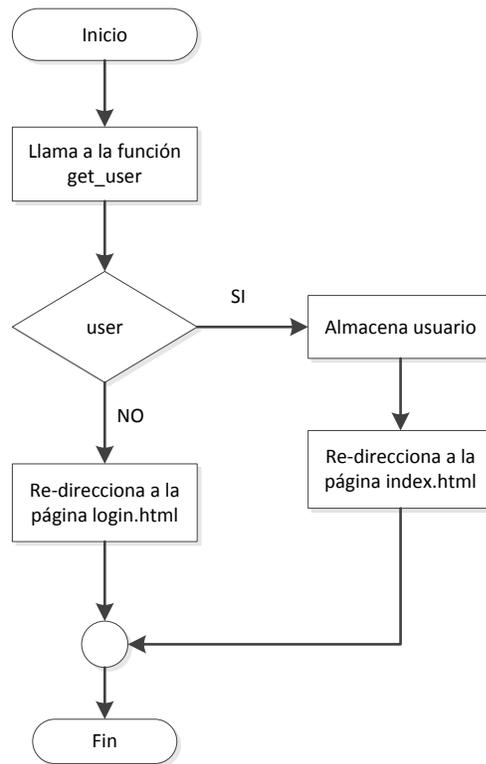


Figura 53. Diagrama de flujo de la clase LoggingHandler

MainPage

La clase MainPage pasa a la plantilla index.html la fecha y hora actualizada, además verifica si el Gateway esta encendido, si no lo está re-direcciona a la página error.html.

MonitoreoPage

La clase MonitoreoPage tiene las funciones render_front y post. La función render_front lee los datos de luminosidad y temperatura del Wall Router y los

pasa a la plantilla monitoreo.html para su visualización. La función post recoge los valores de los radio buttons, crea un objeto de la clase Opcion1 que tiene los datos de configuración de modalidad y de la clase Opcion2 que indica si se ha activado el detector de intrusión, se inicializa los atributos con los datos extraídos de los radio buttons. Se genera una consulta a la base de datos y se rescata la primera entidad. Se borra los datos de la entidad y se guarda los nuevos datos ingresados. En la Figura 54 se muestra el diagrama de flujo de la clase.

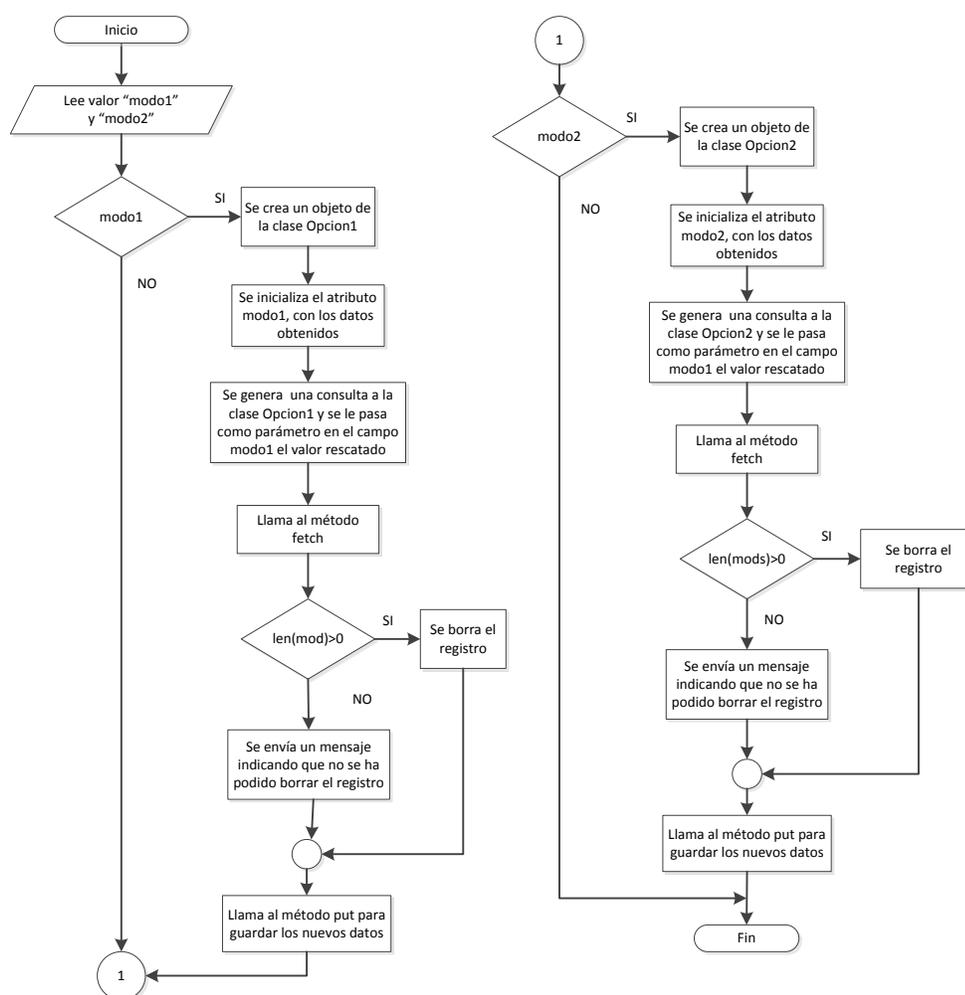


Figura 54. Diagrama de flujo de la clase MonitorPage función post

ControlPage

La clase ControlPage lee el valor de luminosidad proporcionado por el Wall Router, además se genera una consulta a la base de datos y se rescata la entidad asignando a name2 el valor de la configuración de modo de iluminación y a name3 el estado del detector de intrusión. Estos valores se pasa la plantilla control.html para su visualización. En la Figura 55 se muestra el diagrama de flujo de la clase.

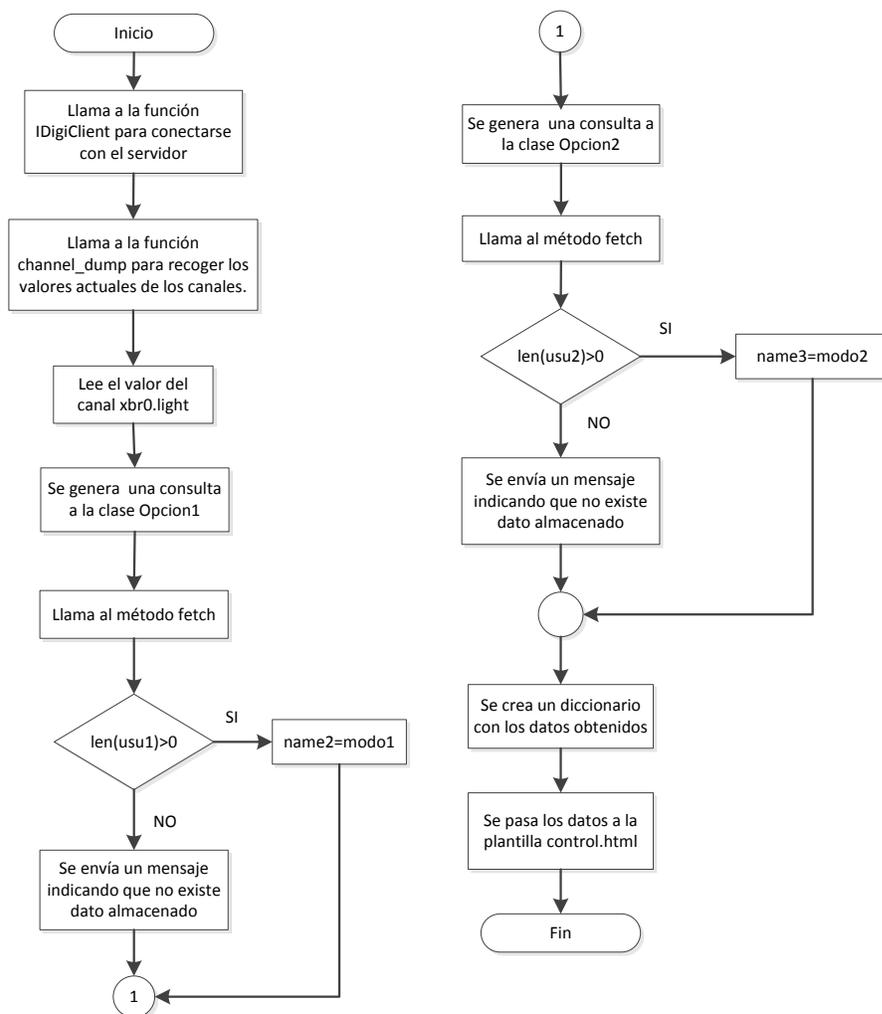


Figura 55. Diagrama de flujo de la clase ControlPage

- **cron.py**

En el archivo cron.py se define la función EnviarMail y las clases Enviacorreos e Iluminacion que permiten crear y configurar las tareas programadas que se ejecutan en determinado momento automáticamente. La función EnviarMail genera una consulta a la clase Mail y asigna a name4 la dirección de correo electrónico almacenada en la base de datos, este valor se asigna al campo destinatario para enviar el correo electrónico notificando la activación de la alarma.

La clase Enviacorreos genera una consulta a la clase Opcion2 y recoge el estado del detector de intrusión almacenado en la base de datos, si esta activado realiza la lectura de los sensores y envía la notificación cuando se active la alarma. La clase Iluminacion permite encender la luz a la hora programada por el usuario.

Archivos de JavaScript

- **control.js**

Este archivo lee los datos obtenidos de la modalidad de iluminación, luminosidad, detector de intrusión, estado de los sensores y alarma, permite actualizar su estado y declarar funciones para crear efectos que mejoren la experiencia del usuario. El intercambio de datos se realizó mediante peticiones Ajax para evitar la constante recarga de la página.

- **monitoreo.js**

Este archivo recoge el correo electrónico ingresado por el usuario, los datos de los radio buttons de la modalidad día, noche y de la activación o desactivación del detector de intrusión y los envía a las clases creadas en main.py a través de peticiones Ajax.

Archivos HTML

- **base.html**

Este archivo sirve como el modelo de los demás archivos html. Contiene la cabecera, el menú y pie de página que es igual para todos los archivos. También carga los archivos JavaScript y CSS que utiliza la aplicación web.

- **login.html**

Este archivo contiene el formulario para ingreso a la cuenta Device Cloud de Etherios. Tiene los campos nombre de usuario, contraseña y el Id del Gateway.

- **index.html**

Este archivo tiene un div que muestra una pequeña descripción de los servicios que brinda el prototipo domótico y la fecha y hora actualizada.

- **monitoreo.html**

Este archivo tiene dos divs que representan las dos secciones a ser visualizadas, la primera muestra el nivel de luminosidad y temperatura, y la segunda permite seleccionar la configuración de modalidad de iluminación y la activación o desactivación del detector de intrusión.

- **control.html**

Este archivo contiene los bloques que corresponden al control de iluminación y al detector de intrusión, donde se muestra el estado de los sensores y valores obtenidos de la nube de dispositivos Etherios.

- **error.html**

Este archivo muestra un párrafo con el contenido del error generado en la aplicación web.

Archivo CSS

El archivo main es una hoja de estilos que da formato a la página web, permitiendo cambiar color, tamaño y tipo de letra.

3.6 IMPLEMENTACIÓN DEL PROTOTIPO

Una vez seleccionados los equipos y elementos a ser utilizados en la red ZigBee, se procede a implementar el prototipo del sistema domótico a través de un módulo.

Se construyeron 2 circuitos impresos que serán conectados a los dispositivos finales, cada placa tiene entradas para los sensores, y el circuito de acoplamiento de los actuadores. A continuación se indican las placas que fueron construidas:

- Control de iluminación

En la Figura 56 se muestra la placa con las conexiones para el sensor de presencia y el adaptador Xbee DIO que permite el control de iluminación mediante las configuraciones de modalidad. Además tiene el circuito de acoplamiento para la activación del actuador.



Figura 56. Placa del control de iluminación

- Detector de intrusión

En la Figura 57 se muestra la placa con las conexiones para el sensor de presencia, magnético y el adaptador Xbee DIO que permite el control manual y automático de la luz y la activación del detector de intrusión. Además tiene el circuito de acoplamiento para la activación del actuador.

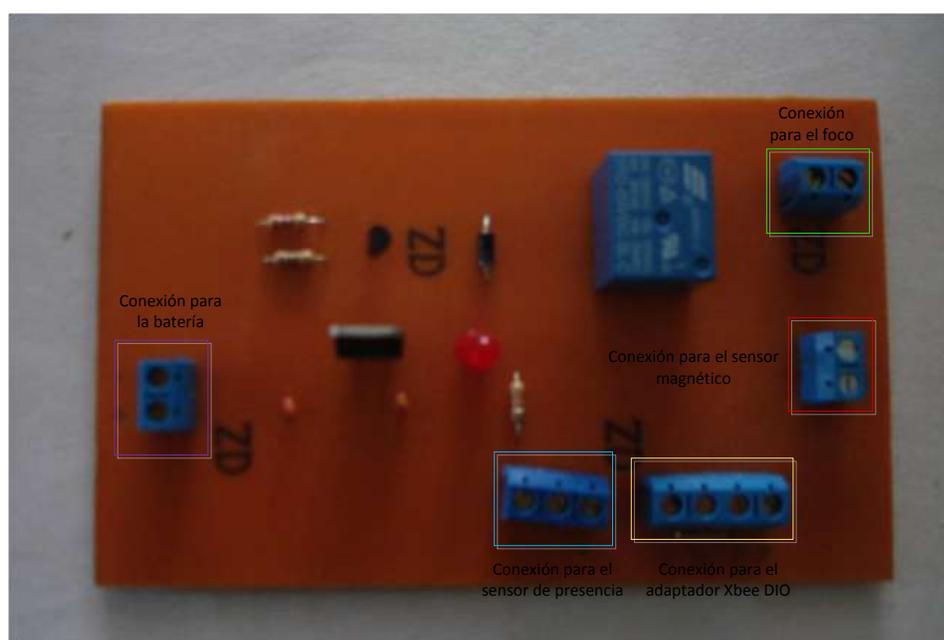


Figura 57. Placa del detector de intrusión

El módulo está hecho de acrílico blanco como se muestra en la Figura 58 y tiene la función de alojar todos los dispositivos utilizados en el prototipo.



Figura 58. Módulo del prototipo

Se realizó el montaje de los dispositivos y el cableado del sistema eléctrico para la alimentación del Wall router y los adaptadores Xbee DIO. En la Figura 59 se muestra el módulo terminado, la parte frontal esta de tal forma que se pueda desmontar en caso de ser necesario para revisión o reemplazo de dispositivos.



Figura 59. Módulo del prototipo terminado

CAPÍTULO 4

PRUEBAS Y RESULTADOS

Una vez terminada la programación e implementación de los componentes del prototipo del sistema domótico, es necesario evaluar el desempeño de la red inalámbrica de sensores y comprobar el correcto funcionamiento del hardware y software implementados.

4.1 MEDICIÓN DE COBERTURA Y SENSIBILIDAD DE LOS NODOS

Un parámetro importante para evaluar el desempeño de la red inalámbrica de sensores es la medición de cobertura que se realiza mediante pruebas de recepción de paquetes y nivel de potencia recibida a través del parámetro RSSI (Received Signal Strength Indicator) y LQI (Link Quality).

Digi proporciona el software X-CTU que permite medir el nivel de recepción entre el módulo coordinador y un nodo final en una topología punto a punto, mediante la utilización de la ventana *Radio Range Test* mostrada en la Figura 60.



Figura 60. Medición del parámetro RSSI en la ventana Radio Range Test

Fuente (Software X-CTU)

De esta forma, se presenta en la Tabla 6 los resultados obtenidos del parámetro RSSI en el software X-CTU.

Tabla 6**Resultados del parámetro RSSI**

Distancia de prueba	Nivel de RSSI	Cantidad de paquetes rechazados	Porcentaje de recepción de paquetes
0,5m	-39dBm	0	100%
1m	-40dBm	0	100%
2m	-42dBm	0	100%
3m	-42dBm	0	100%
4m	-42dBm	0	100%
6m	-45dBm	0	100%
7m	-48dBm	0	100%
8m	-53dBm	0	100%
10m	-54dBm	0	100%
12m	-57dBm	0	100%
14m	-61dBm	0	100%
17m	-71dBm	0	100%

Se observó el comportamiento de la red a diferentes distancias de conexión para determinar si el diseño y la comunicación ZigBee son válidos dentro de las distancias requeridas en un hogar domótico, en la Tabla 6 se observa que a una distancia máxima de 17m sigue funcionando correctamente y no se obtiene pérdidas de paquetes.

Seleccionando la opción red se puede obtener el indicador de calidad de enlace LQI, mostrado en la Figura 61.



Role	MAC	Network Address	Last scan	Connections
Router	0013A200403D1C98	F9F7	8	Show connections
Coordinator	0013A200403D189F	0000	8	
Router	0013A200403D0C7E	007B	6	3 [007B] 0013A200403D0C7E 255 Active
Router	0013A20040321645	F99C	6	2 [F99C] 0013A20040321645 255 Active 3 [F9F7] 0013A200403D1C98 255 Active

Figura 61. Medición del parámetro LQI

Fuente (Software X-CTU)

El valor del parámetro LQI se encuentra en el rango de 0 a 255, donde 255 representa una calidad de enlace óptima. En la Figura 62 se muestra los resultados obtenidos del indicador de calidad de la red.

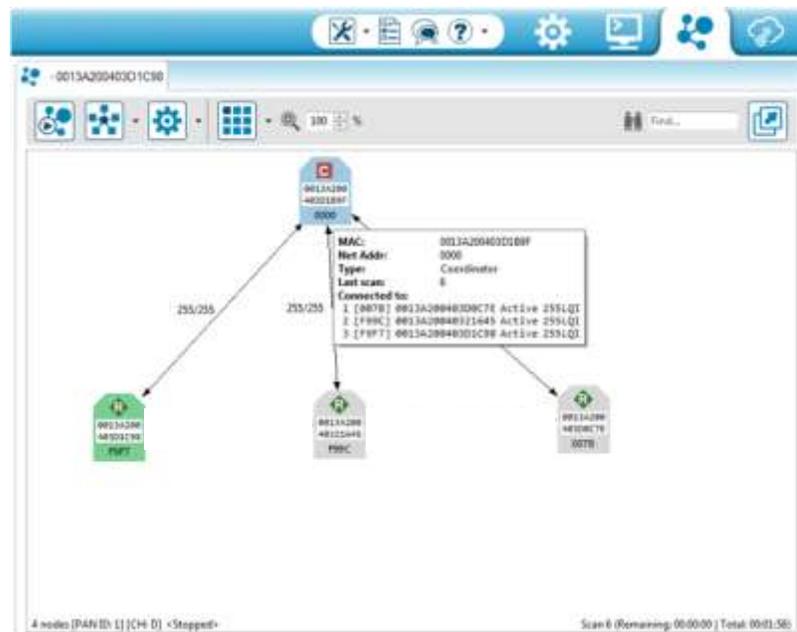


Figura 62. Resultados del parámetro LQI

Fuente (Software X-CTU)

4.2 TIEMPO DE RESPUESTA DE LA RED

Otro parámetro considerado al momento de evaluar una red inalámbrica de sensores es el tiempo de respuesta. Se realizó la medición del tiempo de respuesta del encendido y apagado de la luz desde una red local y una red externa como se muestra en la Tabla 7 y Tabla 8.

Prueba red local

Tabla 7

Resultados del tiempo de respuesta en red local

Encendido[seg]	Apagado[seg]
0.82	0.81
0.61	0.52
0.68	0.63
0.52	0.58
0.65	0.71
0.53	0.51
0.60	0.61
0.49	0.45
0.52	0.47
0.57	0.60
Promedio	
0.59	0.58

Prueba red externa

Tabla 8**Resultados del tiempo de respuesta en red externa**

Encendido[seg]	Apagado[seg]
2.42	2.10
3.14	4.15
2.15	6.17
5.27	4.28
3.54	2.41
6.12	4.75
4.31	5.10
2.18	3.23
2.56	2.14
3.13	5.35
Promedio	
3.5	3.9

4.3 CONSUMO DE ENERGÍA

El objetivo de esta prueba es determinar el consumo de potencia en los nodos, para lo cual se realizó la medición en los modos de operación: modo sleep y modo transmisión, los resultados se pueden observar en la Tabla 9.

Tabla 9

Resultados del consumo de corriente en los nodos

	Modo de operación	Voltaje [V]	Corriente [mA]	Potencia [W]
Nodo1	Modo sleep	8.60	26.6	0.22
	Modo transmisión	8.38	33.7	0.28
Nodo2	Modo sleep	8.49	28.5	0.24
	Modo transmisión	8.18	34.9	0.29
Consumo Total	P. máxima de consumo		0.56W	
	P. mínima de consumo		0.47W	

La eficiencia en la potencia consumida por los nodos se expresa en la Ecuación 1.

$$Eficiencia = \frac{Potencia\ en\ modo\ sleep}{Potencia\ en\ modo\ transmisión}$$

Ecuación 1.

El ahorro de energía es:

$$Eficiencia = \frac{0.46W}{0.57W}$$

$$Eficiencia = 80\%$$

$$Ahorro = 20\%$$

El ahorro de energía es del 20%, el cual representa un valor importante en un sistema inalámbrico.

4.4 PRUEBA GENERAL DEL PROTOTIPO

Para verificar el correcto funcionamiento del prototipo del sistema domótico se procedió a comprobar la funcionalidad de los servicios domóticos ofrecidos.

Confort

- Control de iluminación

Primero se seleccionó el modo de configuración Día en la página web y se verificó que mientras existe un nivel de luminosidad adecuado la luz permanece apagada así detecte presencia y cuando el nivel de luminosidad es menor a 30 la luz se enciende cuando detecta presencia.

De la misma manera se seleccionó el modo de configuración Noche y se verificó que el encendido y apagado de la luz depende únicamente de la señal recibida por el sensor de presencia. En la Figura 63 se muestra la prueba de control de iluminación.



Figura 63. Prueba de control de Iluminación por modos de configuración

Para el control manual se seleccionó la opción Manual y se verificó el encendido y apagado de la luz a través de los botones ON y OFF. Después se seleccionó la opción Automático y se ingresó la hora 20:55 como se muestra en la Figura 64, se seleccionó el botón Activar/Desactivar y se verificó que la luz se encendió a la hora programada.



Figura 64. Prueba del control Automático de iluminación

- Monitorización de temperatura

Para este caso, en la barra de menú se seleccionó la pestaña Monitoreo y se visualizó el valor de la temperatura del ambiente en grados centígrados como se muestra en la Figura 65.



Figura 65. Visualización de valores de temperatura y luminosidad

Seguridad

- Detector de intrusión

Para este caso, primero se seleccionó el botón Activar y se ingresó la dirección de correo electrónico que recibirá la notificación cuando se active la alarma como se muestra en la Figura 66.



Figura 66. Activación del detector de intrusión e ingreso de email

Se verificó que al activarse los sensores de presencia y magnético se activó la ventana de alarma en la página web y se recibió el correo de notificación como se muestra en la Figura 67.



Figura 67. Activación de los sensores y ventana de alarma

De manera general, todo el sistema respondió apropiadamente y con gran aproximación a lo que se planificó.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se determinó que el uso de la tecnología inalámbrica Zigbee es apropiada para la implementación del prototipo del sistema domótico con las aplicaciones especificadas en este proyecto como control de iluminación, monitorización de temperatura y detector de intrusión, debido a que estas requieren bajas tasas de transmisión de datos.
- Se logró identificar los motivos por los cuales los dispositivos Xbee son utilizados en aplicaciones domóticas, estos son flexibilidad, fácil implementación, interoperabilidad entre fabricantes y bajo consumo de potencia permitiendo un ahorro significativo de energía mientras permanecen en modo sleep, que los convierten en dispositivos eficaces para este tipo de aplicaciones.

- Se confirmó la factibilidad de la implementación de una red inalámbrica de sensores y actuadores para su aplicación en un sistema domótico, ya que su característica inalámbrica evita una instalación que dañe la estética de la vivienda.
- Se verificó que el ConnectPort X4 Gateway ofrece la robustez necesaria para el prototipo del sistema, ya que este operará continuamente permitiendo tener un control completo sobre los dispositivos conectados a la red.
- El desarrollo del software de gestión remota en Digi ESP permitió conectar todos los dispositivos al coordinador de la red y recolectar los datos de los canales de los dispositivos finales para almacenarlos en la nube de dispositivos Etherios.
- Se constató la gran ventaja que proporciona la nube de dispositivos Etherios en el almacenamiento de datos y gestión de dispositivos, ya que no se requiere de una infraestructura propia. Además los servicios web que ofrece permitió la creación de una aplicación web cliente para que el usuario pueda tener acceso a los datos y control de dispositivos de forma remota y en tiempo real. Esta funcionalidad requirió la implementación de una clave de acceso con el fin de brindar seguridad a los usuarios.

- El uso de la plataforma Google App Engine fue de gran ayuda, ya que facilitó el desarrollo, el despliegue y el alojamiento de la aplicación web. La aplicación web tiene su propio dominio por lo que el usuario puede ingresar sin ningún problema a la aplicación desde un dispositivo que esté conectado a internet.

5.2 RECOMENDACIONES

- Los dispositivos utilizados en el desarrollo del prototipo funcionan con baterías, con un tiempo de vida corto, por lo tanto es necesario optimizar los circuitos de acondicionamiento con elementos eficientes para minimizar el consumo de energía.
- Ya que el prototipo del sistema incluye una alarma, se podría agregar un modulo GSM para realizar la comunicación entre el sistema domótico y el celular del usuario, de tal forma que las notificaciones de alarma sean enviadas al teléfono móvil, permitiéndole al usuario tomar acciones de una manera eficaz.
- Se recomienda realizar cambios en la configuración de los dispositivos finales, para conseguir que los nodos entren en modo de bajo consumo.

- Se recomienda que la antena que posee el coordinador permanezca erguida para obtener resultados favorables tanto en la transmisión como en la recepción.
- Antes de tomar alguna decisión en cuanto a la manera como se van a conectar los elementos del prototipo del sistema domótico es importante realizar las pruebas necesarias con el fin de observar el comportamiento del diseño y determinar aquella que convenga más a los requerimientos del proyecto.

GLOSARIO

Ajax: *Asynchronous JavaScript and XML*. Es una técnica de desarrollo web para crear aplicaciones interactivas usando diferentes tecnologías web.

API: *Application Programming Interface*. Es un conjunto de funciones que permite al programador acceder a servicios de una aplicación a través del uso de un lenguaje de programación.

CSS: *Cascading Style Sheets*. Es un lenguaje de hojas de estilos creado para controlar la presentación de documentos definidos con HTML y XHTML.

DIA: *Device Integration Application*. Es un software escrito en Python que permite recoger la información de los dispositivos conectados y enviar los datos recogidos a la nube de dispositivos.

GAE: *Google App Engine*. Es una plataforma como servicio (PaaS) que permite crear y ejecutar aplicaciones en la infraestructura de Google.

Gateway: es un nodo en una red informática que sirve de punto de acceso a otra red.

IDE: *Integrated Development Environment*. Es un entorno de programación que ha sido empaquetado como un programa de aplicación.

M2M: *Machine to Machine*. Es la capacidad de intercambiar datos entre dos máquinas remotas.

RCI: *Remote Command Interface*. Es un mecanismo diseñado para permitir la configuración remota, el control y el intercambio de información entre el usuario y los dispositivos utilizando XML y HTTP.

SCI: *Server Command Interface*. Es un servicio web que permite a los usuarios acceder a la información y ejecutar comandos que se relacionan con los dispositivos.

WSN: *Wireless Sensor Network*

XML: *Extensible Markup Language*. Es un lenguaje de marcas utilizado para almacenar datos en forma legible.

Xbee: son dispositivos que pueden comunicarse entre sí de manera inalámbrica. Son fabricados por Digi International.

ZigBee: es un protocolo de comunicaciones inalámbrico basado en el estándar de comunicaciones para redes inalámbricas IEEE 802.15.4.

BIBLIOGRAFÍA

- Asencio, G., Maestre, J., Escaño, J., Macareno, C., Molina, M., & Camacho, E. (2011). Interoperabilidad en Sistemas Domóticos Mediante Pasarela Infrarrojos-ZigBee. En *Revista Iberoamericana de Automática e Informática industrial*. (págs. 397-404).
- Berlemann, L., Mangold, S., & Walke, B. (2007). *IEEE 802, Wireless Systems Protocols, Multi-hop Mesh-Relaying, Performance and Spectrum Coexistence*.
- Capa física*. (s.f.). Obtenido de http://es.wikipedia.org/wiki/Nivel_f%C3%ADsico
- Digi. (s.f.). *About us*. Obtenido de <http://www.digi.com/aboutus/>
- Digi. (s.f.). *ConnectPort® X4 Family*. Obtenido de http://www.digi.com/pdf/ds_connectportx4.pdf
- Digi. (s.f.). *XBee Wall Router*. Obtenido de http://www.digi.com/wiki/developer/index.php/XBee_Wall_Router
- Domínguez, H., & Sáez, F. (Junio de 2006). *Domótica: Un enfoque sociotécnico*. Obtenido de http://www.gsi.dit.upm.es/~fsaez/intl/libro_domotica.pdf
- Domodesk*. (s.f.). Obtenido de Introducción a Zigbee: <http://www.domodesk.com/a-fondo-zigbee>
- Domodesk*. (s.f.). *Domodesk*. Obtenido de <http://www.domodesk.com/a-fondo-infinite-sin-cables>
- Esquivel, R., Rangel, C., Guerra, V., Torres, G., González, M., & Pacheco, A. (Octubre de 2011). Obtenido de <http://dc651.4shared.com/doc/IMb0yS45/preview.html>
- Etherios, D. C. (s.f.). *RCI (Remote Command Interface)*. Obtenido de http://ftp1.digi.com/support/documentation/90000569_G.pdf
- Google Developers*. (s.f.). Obtenido de Google App Engine: <https://developers.google.com/appengine/docs/whatisgoogleappengine>
- Higuera, J., Kartsakli, E., & Valenzuela, J. (s.f.). *On the feasibility of Bluetooth, ZigBee and IEEE802.15.4 technologies on board high speed trains*. Obtenido de http://www.academia.edu/1160690/On_the_feasibility_of_Bluetooth_ZigBee_and_IEEE_802.15.4_technologies_on_board_high_speed_trains

- Inc., D. I. (21 de Mayo de 2014). *XBee DIN Accessories User's Guide*. Obtenido de http://ftp1.digi.com/support/documentation/90000891_H.pdf
- Inc., S. E. (2009). Obtenido de http://www.adrirobot.it/pir_sensor/pdf/DC-SS502_Ver1.0_EN.pdf
- Junestrand, S., Passaret, X., & Vázquez, D. (2005). En *Domótica y hogar digital* (pág. 65).
- Kepes, B. (2011). *UNDERSTANDING The Cloud Computing Stack SaaS, Paas, IaaS*. Obtenido de http://www.rackspace.com/knowledge_center/sites/default/files/whitepaper_pdf/Understanding-the-Cloud-Computing-Stack.pdf
- Longares, J. (s.f.). *Introducción a Zigbee y las redes de sensores inalámbricas*. Obtenido de <http://www.javierlongares.com/arte-en-8-bits/introduccion-a-zigbee-y-las-redes-de-sensores-inalambricas/>
- Mell, P., & Grance, T. (Septiembre de 2011). *The NIST Definition of Cloud Computing*. Obtenido de <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
- Ocampo, D., & Serna, S. (2010). Obtenido de http://www.microvirtual.org/images/Congreso/Ponencias_CVM2/Seguridad/SEG201_P.pdf
- Pallás, R. (s.f.). Sensores y Actuadores. En *Adquisición y distribución de señales* (págs. 27-44). Marcombo S.A.
- Redes inalámbricas*. (s.f.). Obtenido de <http://es.kioskea.net/contents/818-redes-inalambricas>
- Salgado, I. (2011). *ZigBee y sus aplicaciones*. Obtenido de <http://www.dea.icaei.upco.es/sadot/Comunicaciones/avanzadas/Zigbee%20y%20sus%20aplicaciones.pdf>
- Songle. (s.f.). Obtenido de <http://www.songle.com/en/pdf/20084141716341001.pdf>
- Top cloud computing providers*. (2010). Obtenido de <http://www.jansipke.nl/top-cloud-computing-providers/>
- Van Rossum, G. (2009). *Tutorial de Python*. Obtenido de <http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf>
- Vasco, D., Sintes, F., & Lagos, N. (Enero de 2003). *Domótica con el plc s7-200*. Obtenido de <http://ocw.upc.edu/sites/default/files/materials/15012628/40197-3452.pdf>

ACTA DE ENTREGA

El proyecto fue entregado al Departamento de Eléctrica y Electrónica y reposa en la Universidad de las Fuerzas Armadas – ESPE, desde:

Sangolquí, 02 DE SEPTIEMBRE de 2014

ELABORADO POR:

Gabriela

GABRIELA ALEXANDRA ALVARO SOTALIN

171883115-7

AUTORIDAD

Luis Orozco

Ing. Luis Orozco MSc.



DIRECTOR DE LA CARRERA DE ELÉCTRICA Y ELECTRÓNICA,

AUTOMATIZACIÓN Y CONTROL