



UNIVERSIDAD DE JAÉN
Centro de Estudios de Postgrado

Trabajo Fin de Máster

**DISEÑO Y
MONITORIZACIÓN A TRAVÉS
DE SENSORES DE
DIFERENTES ASPECTOS
AMBIENTALES Y DE
CONFORT EN UN EDIFICIO.**

Alumna: Moreno Cabrera, Mónica

Tutores: Prof. D. Manuel Ureña Cámara
Prof. D. Alejandro Sánchez García

Dptos: Ingeniería Cartográfica, Geodésica y
Fotogrametría
Ing. de Sistemas y Automática

Julio, 2016

Índice

1.	INTRODUCCIÓN	1
2.	ANTECEDENTES.....	3
2.1.	Redes de sensores y Arduino	3
2.2.	Inmótica.....	4
2.3.	Smart Buildings y eficiencia energética	5
2.4.	Escaneado láser 3D	7
2.5.	BIM	7
3.	OBJETIVOS	8
3.1.	Objetivos generales.	8
3.2.	Objetivos secundarios.....	9
4.	SITUACIÓN Y EMPLAZAMIENTO.....	9
4.1.	Situación	9
4.2.	Emplazamiento	10
5.	Instrumentación, hardware, software y lenguajes usados	11
5.1.	Instrumentación	11
5.1.1.	Láser escáner 3D	11
5.2.	Hardware.....	13
5.2.1.	Arduino UNO.....	13
5.2.2.	Shield Ethernet 2.....	16

5.2.3. Shield Wifi	17
5.2.4. Sensor de temperatura ds18b20.....	18
5.2.5. Fotorresistencia LDR.	18
5.3. Software utilizado	19
5.3.1. PosgreSQL	19
5.3.2. Geoserver.....	19
5.3.3. Revit.....	20
5.3.4. Maptek I-site	20
5.3.5. IDE Arduino	20
5.4. Lenguajes de programación	21
5.4.1. Lenguaje programación Arduino	21
5.4.2. HTML, Javascript, CSS y PHP.....	22
6. METODOLOGÍA	23
6.1. Elección de la sección de edificio a monitorizar	24
6.2. Elección del número y tipos de sensores	25
6.3. Diseño de la distribución de sensores	26
6.3.1. Zigbee	27
6.3.2. Bluetooth	28
6.3.3. WiFi.....	28
6.4. Captura y levantamiento de la zona.....	29

6.5. Puesta en marcha del sistema y captura de la información ..	32
6.6. Puesta en marcha de las bases de datos y el servidor	36
6.7. Análisis y visualización de los datos a través de un cliente ligero.	38
7. Desarrollo e implementación	40
7.1. Elección de la sección de edificio a monitorizar	40
7.2. Elección del número y tipos de sensores.	41
7.3. Diseño de la distribución de sensores	42
7.4. Captura y levantamiento de la zona.....	44
7.5. Puesta en marcha del sistema y captura de la información ..	48
7.6. Puesta en marcha de las bases de datos y el servidor	53
7.7. Análisis y visualización a través de un cliente ligero.....	56
8. RESULTADOS.....	63
9. CONCLUSIONES.....	65
Referencias:.....	67

Listado de figuras

Figura 1: Situación de Jaén en Andalucía [5]	10
Figura 2: Situación del edificio [6]	10
Figura 3: Emplazamiento del edificio [6]	11
Figura 4: Láser escáner 3D	12
Figura 5: Arduino UNO.....	13
Figura 6: Ethernet Shield 2	16
Figura 7: WiFi Shield	17
Figura 8: Sensor de temperatura Ds18b20.....	18
Figura 9: Fotorresistencia LDR	18
Figura 10: Icono posgreSQL	19
Figura 11: Geoserver.....	19
Figura 12: IDE Arduino	21
Figura 13: Sketch Arduino.....	21
Figura 14. Etapas del método aplicado	24
Figura 15: Esquema de conexión en Arduino	33
Figura 16: Librerías Arduino	34
Figura 17: Esquema de comunicación del sistema.....	35
Figura 18: Conectar servidor MySQL	36
Figura 19: Tablas creadas en MySQL	38
Figura 20: Esquema BBDD, servidor web y visor web.	39

Figura 21: Diseño web propuesto.....	39
Figura 22: Sección de edificio elegida.....	40
Figura 23: Esquema de conexión de los sensores final.	43
Figura 24: Distribución de los sensores.	44
Figura 25: Bases de estacionamiento.	45
Figura 26: Nube de puntos	46
Figura 27: Perfil del modelo en Revit.	46
Figura 28: Planta del edificio en Revit.	47
Figura 29: Modelo 3D de la sección de edificio	47
Figura 30: Esquema de conexión final	48
Figura 31: Esquema de conexiones	49
Figura 32: Montaje Arduino y sensores.....	50
Figura 33: Respuesta http	52
Figura 34: Respuesta monitor serial	52
Figura 35: Ejemplo de tabla con datos reales.....	53
Figura 36: Conexión Geoserver con PostgreSQL	54
Figura 37: SQL que crea las capas de los sensores	54
Figura 38: Resultado diseño de los sensores.....	56
Figura 39: Visualización de capas en la web	57
Figura 40: Control de capas.....	57
Figura 41: Leyenda.....	58

Figura 42: Formularios fecha y hora	58
Figura 43: Ejemplo de valores obtenidos al pinchar sobre el sensor	59
Figura 44: Botones para activar mapa interactivo	59
Figura 45: Tabla web.....	60
Figura 46: Gráfica de datos recabados	60
Figura 47: Modelo 3D incluido en la web	61
Figura 48: Estado sensores 1:00h-2:00h	61
Figura 49: Estado sensores 5:00h-6:00h	61
Figura 50: Estado sensores 11:00h-12:00h	62
Figura 51: Estado sensores 16:00h-17:00h.....	62
Figura 52: Estado sensores 21:00h-22:00h	62
Figura 53: Estado sensores 23:00h-1:00h	62
Figura 54: Datos recogidos	63
Figura 55: Pantalla Inicio de la web	64
Figura 56: Pantalla Modelo 3D	64

Listado de tablas

Tabla 1: Coordenadas del centro del edificio [7]	10
Tabla 2: Tipos de sensores	25

1. INTRODUCCIÓN

Desde hace años, la tecnología va avanzando a gran ritmo. Todo lo que nos rodea se va actualizando, modernizándose, informatizándose con el objetivo de ir mejorando los procesos o acciones que realizamos en el día a día, ya sea en el trabajo, en la calle o en el hogar para mejorar la calidad de vida de las personas o los procesos en el trabajo.

Este avance tecnológico permite la fabricación de dispositivos electrónicos con un tamaño cada vez más compacto, de baja potencia y económicos, con capacidad de detectar y procesar datos, tanto de forma inalámbrica como cableada. Una red de sensores, es un conjunto de estos dispositivos que reciben el nombre de nodos-sensores de bajo coste y bajo consumo que colaboran en una tarea común, formando una red, la cual puede ser inalámbrica, recibiendo el nombre WSN (red inalámbrica de sensores).

Esta clase de redes se caracterizan por su fácil despliegue y por ser auto configurables, registrando datos referentes a los sensores locales. Otra de sus características es su gestión eficiente de la energía, que les permite obtener una alta tasa de autonomía que las hacen plenamente operativas.

El desarrollo de tecnologías innovadoras ofrece nuevas posibilidades para actualizar las instalaciones eléctricas, núcleo de todos los edificios.

El control y monitorización de edificios puede suponer una serie de ventajas considerables para usuarios en aspectos de confort, ahorro energético y reducción de costes de mantenimiento. Para ello, se hace uso de las mencionadas redes de sensores.

Además, estas redes permiten interacciones inteligentes entre los usuarios y máquinas, lo cual proporciona un tremendo beneficio social. Pueden ayudar a evitar fallos de infraestructura, conservar valiosos recursos naturales, aumentar la productividad, mejorar la seguridad, y permitir nuevas aplicaciones para la recolección de datos en los edificios inteligentes. Este último tema es el que se va a abordar en este trabajo.

Al mismo tiempo, también se viene concienciando a la población de la importancia del ahorro energético, de la necesidad de luchar contra el cambio climático y de minimizar la dependencia de los combustibles fósiles.

Es un hecho cada vez más conocido que los edificios consumen un 40% de la energía total utilizada y son responsables del 36% de las emisiones de CO₂ en la Unión Europea [1], por lo que es tan importante como urgente mejorar la eficiencia energética en los edificios.

Ante nuestro objetivo de reducir las emisiones de CO₂ y asegurar un mundo sostenible a las futuras generaciones, podemos actuar invirtiendo en tecnologías renovables. Pero existe otra solución, que puede ser compatible con la anterior, si consumimos menos energía, demandaremos menos y al final supondrá un ahorro tanto económico como ecológico.

Para aumentar la eficiencia energética, la seguridad, la usabilidad y la accesibilidad cada vez más se construyen 'Smart Buildings' o Edificios Inteligentes. Los Edificios Inteligentes son aquellos cuyas instalaciones y sistemas (de climatización, iluminación, electricidad, seguridad, telecomunicaciones, multimedia, informáticas, control de acceso, etc.) permiten una gestión y control integrada y automatizada con sistemas inmóticos.

Desde esta perspectiva es desde donde se quiere abordar este proyecto. Se pretende instalar un prototipo de una red de sensores en el interior de un edificio, con el fin de disponer de los datos reales de consumo que pueden ser utilizados en un futuro para la mejora de la eficiencia de este edificio. Esto puede servir para que este edificio sea más inteligente, y por tanto, se conciencie del consumo de energía que produce para así mejorar la eficiencia si se prueba el mal uso de la energía consumida (aparatos de aire acondicionado, luces encendidas...).

Se aportarán estos datos que pueden ser de ayuda para futuros proyectos de automatización más eficaces que los ya existentes, aunque en este caso no se ejecutarán procesos de automatización o domóticos, para no aumentar la envergadura del proyecto.

Por otro lado, al realizarse un modelo 3D del edificio, con un láser escáner, y obtener datos ambientales y de confort del mismo, trataremos en que consiste la tecnología BIM (Building Information Modeling).

Cabe aclarar el significado del acrónimo BIM para comprender su aplicación y función. BIM es el Modelado Integrado de Información para la Edificación. El cual es una representación digital de las características físicas y funcionales de una instalación. Posee información total del edificio, es capaz de brindar los conocimientos sobre las

instalaciones y con ello forma una base fiable para las tomas de decisiones durante su ciclo de vida. [2]

En definitiva el presente trabajo estará relacionado con todos estos temas tratados en este apartado (redes de sensores, control energético, edificio inteligente, BIM...). Ya que se tratará el proceso de diseño e implementación de un pequeño sistema que permita el análisis de consumo energético en un edificio mediante el uso de una red de sensores (algunos de ellos inalámbricos). El procesamiento de los datos a través de una base de datos y la creación de una aplicación web para la posterior visualización de los datos y el análisis de los mismos. Por otro lado, se ha elaborado el modelo 3D, con el uso del escáner laser 3D, de la zona de estudio y planos para su visión gráfica en el cliente web creado.

2. ANTECEDENTES

En este apartado se verá la evolución de todos los contenidos que están relacionados con el trabajo de esta memoria. Como ya se ha explicado se va a realizar un prototipo de una red de sensores en el interior de un edificio, por ello los temas que se desarrollarán en este apartado son las redes de sensores y Arduino, al estar enlazado a este tema la automatización de edificios se hablará de Inmótica, Smart Building y eficiencia energética. Asimismo al realizarse un modelo 3D de parte del edificio se tratará sobre BIM y láser escáner.

2.1. Redes de sensores y Arduino

El desarrollo de las WSN fue inspirado por aplicaciones militares, especialmente en la vigilancia de las zonas de conflicto. La investigación sobre WSN se intensificó a principios de los años 1980, cuando la Agencia de Proyectos de Investigación Avanzados de Defensa de los Estados Unidos (DARPA), llevó a cabo el programa de redes de sensores distribuidos para los militares de Estados Unidos [3].

Aunque las redes de sensores fueron usadas inicialmente en aplicaciones militares, ahora se han extendido a muchas áreas industriales y de interés social como: monitorización ambiental, monitorización de la salud, gestión de aparcamiento o

control de tráfico, entre otros. De todas las aplicaciones se ha optado por la monitorización ambiental de un edificio para elaboración de este trabajo.

Para la realización del proyecto se podrían haber empleado módulos domóticos existentes en el mercado, pero como se ha comentado anteriormente, uno de los objetivos es realizar desde cero un sistema que permita obtener datos para la gestión del edificio.

Existen diferentes tipos de placas en el mercado como como son: Arduino, Raspberry Pi, Beagle One... Aunque ha optado por una placa Arduino que es, un microcontrolador con muchas ventajas y que cumple con todas las expectativas, como veremos a continuación:

- Precio. Existen multitud de modelos de placas originales, todos ellos de muy bajo coste, además de existir versiones de otros fabricantes.
- Sistema abierto. Existe la posibilidad de fabricar un modelo reducido más económico pues contamos con toda la información y ficheros de diseño.
- Sistema muy didáctico. Debido a su configuración de hardware y la sencillez del lenguaje, unido a la cantidad de información y ejemplos existentes (compartidos por una gran comunidad).
- Entradas y salidas disponibles. En función de las necesidades del proyecto, podremos elegir entre las distintas placas, que cuentan con multitud de entradas y salidas digitales, entradas analógicas, así como puertos de comunicaciones.
- Muy extendido y estandarizado. Existen infinidad de librerías de libre distribución para poder comunicarse con hardware y software de terceros.

2.2. Inmótica

La automatización y telecontrol no es un concepto nuevo. Estas tecnologías se pueden aplicar a diversos ámbitos: si nos referimos a una vivienda, estaríamos hablando de domótica, si se aplica a edificios, se conoce como inmótica, y si abarca a toda una ciudad podría definirse como urbótica.

En este trabajo nos hemos centrado en edificios, en este caso un edificio perteneciente a un campus universitario, aunque se ha diseñado un prototipo para que pueda ser utilizado en cualquier tipo de edificio, como una gran vivienda o cualquier tipo de edificio, público o privado.

En los años 70 ya existían soluciones que permitían la comunicación entre sensores y actuadores de una vivienda que se comunicaban a través de la línea eléctrica.

Con el paso de los años, han ido apareciendo protocolos y sistemas cada vez más complejos y a su vez robustos, que cuentan con unas características más adecuadas a cada campo. Por nombrar alguno, en automatización de edificios y viviendas, podemos encontrar KNX.

No es el objeto de este proyecto ahondar en los procesos de automatización y los buses de campo pues en este trabajo, el objetivo es el uso de otras tecnologías como las redes de datos para la monitorización de edificios.

Cuando el edificio cuenta con unas infraestructuras de redes de datos y otros sistemas informáticos que permiten combinar y tratar la información de todos los subsistemas integrados (automatización iluminación, seguridad, control de accesos, consumos eléctricos...) con otros sistemas informáticos, estaríamos hablando de un “edificio inteligente” o “Smart Building”. Esto solo es posible si partimos de un “edificio conectado” y apoyándonos en la tecnología IP.

2.3. Smart Buildings y eficiencia energética

Históricamente el hombre ha construido edificios para crear un entorno controlado para poder vivir y trabajar. Pero a lo largo de las últimas décadas han cambiado las prioridades en el diseño y la organización de edificios.

A principios de los años ochenta se comenzaron a construir en Estados Unidos y Japón los primeros edificios a los que se aplicaban la informática para mejorar su comodidad, habitabilidad y funcionalidad.

Para proporcionar un espacio cómodo para las personas dentro de los edificios de hoy nos encontramos con un complejo sistema de estructuras y tecnología. Con el tiempo, cada uno de los componentes dentro de un edificio ha sido desarrollado y

mejorado, permitiendo a los usuarios de edificios de hoy en día controlar los sistemas de iluminación, seguridad, calefacción, ventilación y aire acondicionado independiente.

Pero en la construcción actual se está empezando a mirar más allá y a considerar el impacto del edificio a la red eléctrica y al medio ambiente. Para cumplir estos objetivos, no es suficiente un edificio que simplemente contiene los sistemas que proporcionan comodidad, luz y seguridad. Los edificios del futuro deben conectar las distintas piezas de una manera integrada, dinámica y funcional. Esta visión es un edificio que cumple a la perfección su misión y reducir al mínimo los costos de energía, el apoyo a una red eléctrica robusta y mitigar el impacto ambiental.

Como ya se ha dicho, los edificios contabilizan un 40% de energía primaria consumida, más que los sectores del transporte y de la industria. De ahí la importancia de potenciar el uso de las nuevas tecnologías para mejorar la eficiencia de los inmuebles.

Haciendo uso de un sistema permita la monitorización y el control remoto es posible visualizar en tiempo real parámetros como consumos eléctricos y otras variables de la instalación (temperatura, estado de apertura de ventanas y compuertas, iluminación...) y a su vez actuar sobre los distintos sistemas controlados. En este caso nos centraremos en la visualización de los datos recogidos, no se incorporarán sistemas de actuación.

Analizando la información obtenida se pueden poner en marcha medidas de ahorro energético y comprobar, de forma inmediata, su efectividad.

Se ahorrará más energía en aquellas entidades que tengan una buena eficiencia energética frente a aquellas otras cuya eficiencia sea baja. Es decir, el mejor de los edificios, energéticamente hablando, será el que menos energía necesite para funcionar en relación con la envergadura de los servicios que deba cubrir.

La finalidad es satisfacer las necesidades del edificio al mínimo coste. En este control se pueden distinguir tres aspectos diferenciados:

- Regulación: con la que se pueda obtener la evolución del consumo energético de la vivienda o edificio.
- Programación: para programar distintos parámetros de la vivienda a nivel energético, como temperatura según horarios, días de la semana, mes, etc.

- Optimización: de modo que se minimice el consumo. El aprovechamiento de la energía y reducción de su consumo. Estos sistemas revierten a medio y largo plazo en su amortización, además de estar muy ligadas al concepto de confort. Las acciones destinadas a reducir el consumo están íntimamente ligadas a la integración de todos los dispositivos de la vivienda en el sistema.

2.4. Escaneado láser 3D

En la actualidad la incorporación de nuevas tecnologías, como el escáner láser 3D son, una alternativa y un complemento a los métodos de medición clásicos como la topografía y fotogrametría, ya que estos han mostrado mejor desempeño en términos de grado de automatización, precisión y velocidad, incapaces de ser obtenidas por el hombre en tan poco tiempo. Es decir el escáner láser es una tecnología que se utiliza para la creación de modelos 3D, reproduce la realidad directamente como puntos espaciales con alta densidad, en tiempo real, en un entorno digital, proporcionando información métrica y radiométrica.

La aparición en los últimos tiempos de aplicaciones informáticas capaces de gestionar con solvencia las nubes de puntos generadas por el escáner ha potenciado enormemente el rendimiento de esta herramienta en la obtención de levantamientos muy útiles, sobre todo, en el ámbito de la arquitectura.

2.5. BIM

Building Information Modeling está destinado a ser la siguiente evolución en el mundo de la construcción. Se compara el cambio al que se produjo cuando pasamos de dibujar en papel al diseño asistido por ordenador (CAD). [4]

Encontramos varios puntos de vista sobre el origen de este concepto. La empresa pionera en la aplicación del concepto BIM fue firma húngara Graphisoft, la cual lo implementó bajo el nombre de Virtual Building (Edificio Virtual). Autodesk comenzó utilizar el concepto BIM en 2002. [2]

BIM se refiere al conjunto de metodologías de trabajo y herramientas informáticas para la elaboración, gestión, ejecución y mantenimiento de proyectos de

edificación con un uso de la información continuamente coordinada y coherente, toda ella alrededor de una o más bases de datos compatibles en las que se almacena toda la información del edificio (teléfonos de proveedores, presupuestos, materiales, usos, etc.), éstas reciben el nombre de bases de datos paramétricas, las cuales podemos trabajar de forma única en un modelo 3D.

Uno de los objetivos principales de la tecnología BIM es el uso por múltiples usuarios de la información, ya sean arquitectos técnicos, arquitectos o ingenieros y que cada uno de ellos se ocupe de una tarea u otra, de ahí la necesidad de la interoperabilidad, de que la información esté completamente coordinada, sin limitaciones en número de usuarios y el número de archivos.

Con la aparición del sistema BIM, a la hora de diseñar un edificio ya no se representa los elementos arquitectónicamente, sino constructivamente, son elementos definidos según sus características, sus materiales, espesores, especificaciones, etc.

El sistema BIM pretende la idea de poder estudiar un edificio durante su ciclo de vida útil. Incluyendo la fase de diseño, ejecución, mantenimiento y demolición. Dentro de la fase de utilización, los usuarios de las edificaciones podrán acceder a toda la información para poder realizar correctamente el mantenimiento de la edificación.

3. OBJETIVOS

3.1. Objetivos generales.

El objetivo del Trabajo de Fin de Máster consiste en poner en práctica varios puntos diferenciados, pero que en este proyecto van a estar muy relacionados entre sí. Por un lado y como objetivo principal reside en determinar un conjunto de parámetros ambientales y de confort (temperatura e iluminación) Implementando para ello una red de sensores en el interior un edificio. Por otro lado, está la creación de un modelo tridimensional y bidimensional de la zona de actuación, que servirá para la posterior visualización espacial de los datos recogidos a través de un visor web, así como el análisis visual de la información proporcionada por dichos sensores.

3.2. Objetivos secundarios

Como objetivos secundarios del presente trabajo se abarcan también varios puntos:

- Uso de la plataforma Arduino para la recogida de datos de los sensores de temperatura y luminosidad.
- Conexión del servidor y los sensores.
- Puesta en funcionamiento de una pequeña infraestructura de datos espaciales.
- Elaboración de una aplicación web.
- Manejo del láser escáner.
- Elaboración de un modelo 3D con el software Revit.

4. SITUACIÓN Y EMPLAZAMIENTO.

4.1. Situación

El edificio elegido como ejemplo para la prueba de este trabajo se encuentra situado en la Comunidad Autónoma de Andalucía, concretamente en el municipio de Jaén. Se encuentra geográficamente al norte de la ciudad, dentro del Campus universitario de Las Lagunillas donde se ubica la Universidad de Jaén.



Figura 1: Situación de Jaén en Andalucía [5]



Figura 2: Situación del edificio [6]

4.2. Emplazamiento

Como ya se ha dicho el edificio está situado en el campus de las Lagunillas (Universidad de Jaén) (Figura 3). Concretamente, se han implantado la red de sensores en la tercera planta del edificio A3 (edificio de Ingeniería y Tecnología).

Las coordenadas UTM ETRS89 referidas al centro del edificio son las siguientes:

X (m)	Y (m)
431567,83	4182564,04

Tabla 1: Coordenadas del centro del edificio [7]



Figura 3: Emplazamiento del edificio [6]

5. INSTRUMENTACIÓN, HARDWARE, SOFTWARE Y LENGUAJES USADOS

5.1. Instrumentación

5.1.1. Láser escáner 3D

El escáner láser es un instrumento topográfico que permite obtener, en poco tiempo, gran cantidad de medidas en forma de millones de coordenadas espaciales en el sistema de referencia propio del instrumento. El uso del escáner láser terrestre presenta varias ventajas, entre las que destacan la precisión y cantidad de datos obtenidos en poco tiempo, y la posibilidad de realizar distintos análisis.

Para este trabajo se ha usado Leica scanstation modelo C10. Tiene un diseño compacto todo en una plataforma. Incluye escáner, sensor de inclinación, batería, controlador, almacenamiento de datos, auto-exposímetro de cámara de vídeo y plomada láser.

La estación láser C10 también cuenta con importantes avances de productividad, incluyendo el nuevo X-Mirror Smart que permite al usuario realizar escaneos de bóveda completa en muy pocos minutos usando un espejo giratorio.

El ScanStation C10 hace la Topografía de Alta Definición aún más rentable para los as-built, levantamientos topográficos y monitorizaciones. [8]



Las principales especificaciones técnicas son las que se describen a continuación:

- Tipo de instrumento: Escáner láser de muy alta velocidad, compacto, tiempo de vuelo y con compensador de doble eje, con precisión, alcance y campo de visión a nivel topográfico; cámara integrada y plomada láser.
- Interfaz de usuario: Control incorporado, notebook o tablet PC o control remoto.
- Almacenamiento de datos: Disco de estado sólido (SSD) integrado, PC externo o USB externo.

Figura 4: Láser escáner 3D

- Cámara: Cámara digital de alta resolución integrada con autoajuste y vídeo zoom.
- Alcance: 300 m a 90%; (alcance mínimo 0,1 m).
- Velocidad de escaneo: Hasta 50.000 puntos/seg, velocidad instantánea máxima.
- Tamaño de punto: De 0 – 50 m: 4,5 mm (basado en FWHH);
- Resolución: Horizontal y vertical totalmente seleccionable; espaciado mínimo de <1 mm, en todo el rango; capacidad de intervalo de punto único.
- Horizontal: 360° (máximo).
- Vertical: 270° (máximo).
- Puntería/visualización: Sin paralaje, vídeo zoom integrado.

- Pantalla incorporada Control de pantalla táctil con lápiz óptico, pantalla gráfica a todo color, QVGA (320 x 240 píxeles).

5.2. Hardware

El hardware usado para este TFM son las placas Arduino y sensores para la monitorización, cuyas características se describen en los siguientes puntos.

5.2.1. Arduino UNO

En primer lugar explicaremos la placa de Arduino UNO que se utilizará para recoger los datos de los sensores.

Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar.

La plataforma Arduino se programa mediante el uso de un lenguaje de programación intuitivo (IDE) de Arduino. Equipando Arduino con sensores, activadores, luces, altavoces, módulos complementarios (denominados shields), y otros circuitos integrados, es posible convertir la plataforma programable para casi cualquier sistema de control.

Existen varias versiones de las placas Arduino. Los fabricantes están lanzando constantemente placas nuevas con diferentes características. Para este TFM se ha elegido la placa Arduino UNO, por las particularidades que se detallan a continuación.

- Accesible.
- Multi-Plataforma.
- Entorno de programación sencilla y directo.
- Software ampliable y de código abierto.
- Hardware ampliable y de código abierto.

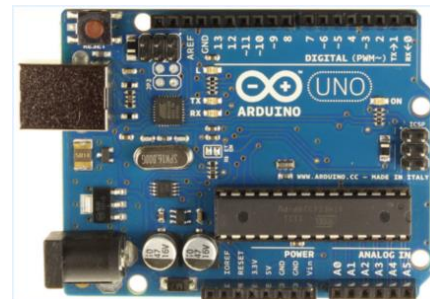


Figura 5: Arduino UNO

Arduino como es hardware de código abierto, todos los archivos de diseño, esquemas y código fuente están disponibles para todo el mundo de forma gratuita. Esto significa que es más fácil adaptarlo para un determinado fin.

Según la página web oficial de Arduino, estas son las principales características de Arduino Uno [9]:

- Micro controlador ATmega328.
- Voltaje de funcionamiento 5V.
- Pines E/S digitales 14.
- Pines de entrada analógica 6.
- Intensidad máxima por pin 40 mA.
- Memoria Flash 32 KB (ATmega328) de las cuales 0.5 KB las usa el gestor de arranque (boot loader).
- SRAM 2 KB (ATmega328).
- EEPROM 1 KB (ATmega328).
- Velocidad de reloj 16 MHz.

El Arduino UNO puede ser alimentado vía la conexión USB o con una fuente de alimentación externa. Algunos de los pines tienen funciones especializadas:

- Serie: Pin 0 (RX) y pin 1 (TX). Están disponibles para recibir datos (RX) y transmitir (TX) datos a través de puerto serie TTL.
- Interrupciones Externas: pin 2 y pin 3. Estos pines se pueden configurar para que interrumpan la ejecución del programa al detectar un flanco o un nivel.
- PWM: pines 3, 5, 6, 9, 10 y 11. Proporcionan una salida PWM (modulación por ancho de pulsos) con temporizadores de 8 bits de resolución.
- SPI: pines 10, 11, 12 Y 13. Estos pines proporcionan comunicación SPI (Serial Peripheral Interface).

- LED: pin 13. Hay un LED integrado en la placa conectado al pin digital 13, cuando este pin tiene un valor HIGH (5V) el LED se enciende y cuando este tiene un valor de LOW (0) el LED se apaga.
- 6 entradas analógicas, cada una de ellas proporciona una resolución de 10 bits (1024 valores). Por defecto se mide de tierra a 5 voltios.
- I2C: pin 4 (SDA) y pin 5 (SCL). Soporta del protocolo de comunicaciones I2C / TWI.
- AREF. Este pin está dos pines más arriba del digital pin 13, proporciona un voltaje de referencia para las entradas analógicas.
- Reset. Si en este pin se suministra un cero lógico, se reinicia el microcontrolador. El ATmega328 en las placas Arduino UNO viene precargado con un gestor de arranque (boot loader) que permite cargar nuevo código sin necesidad de un programador por hardware externo. La carga de un nuevo código se realiza a través del entorno de desarrollo Arduino y la conexión serie/USB.

Las placas de Arduino, como se ha comentado con anterioridad, cuentan con diversas formas y puertos de comunicación como puede ser el puerto serie (mediante USB o pines), el puerto I2C/TWI, SPI pero en este proyecto se pretende también aprovechar la comunicación mediante tecnología TCP/IP porque está presente en cualquier edificio y es una tendencia creciente. Actualmente están apareciendo modelos nuevos de Arduino que cuentan con puerto Ethernet ya integrado (por ejemplo Arduino YUN), pero en este proyecto se optó con anterioridad por los modelos UNO para los que existe la opción de instalarles un “Shield” que lo dote de comunicaciones, además de otras ventajas.

5.2.2. Shield Ethernet 2



Figura 6: Ethernet Shield 2

El Arduino Ethernet Shield permite a una placa Arduino conectarse a internet. Está basada en el chip Ethernet Wiznet W5100. El Wiznet W5100 provee de una pila de red IP capaz de soportar TCP y UDP. [10]

Con ésta placa y la ayuda de la librería proporcionada, podremos conectar nuestra placa Arduino a internet y también realizar tanto un pequeño servidor web, como un cliente. La configuración de red se realiza mediante software, por lo que podremos adaptar con facilidad la placa a nuestra red local.

Dispone de un zócalo para tarjetas de memoria micro-SD para poder almacenar ficheros o para utilizarlo como servidor web embebido. También incluye un controlador de reset automático para que el chip interno W5100 esté bien reiniciado y listo para utilizar al arranque.

Hay que tener en cuenta que el W5100 y la micro-SD comparten el bus SPI, por lo que sólo uno de ellos puede ser utilizado a la vez. Si se desea utilizar ambos simultáneamente, hay que tenerlo en cuenta al escribir el código.

Es compatible con el Arduino UNO y Arduino Mega.

La placa cuenta con los siguientes LED informativos:

- PWR: indica que la placa cuenta con alimentación.
- LINK: indica presencia de una red y parpadea cuando se envía o reciben datos.
- FULLD: indica que la conexión de red es full dúplex.
- 100M: avis de la presencia de una conexión a 100Mb/s.
- RX: parpadea cuando se recibe datos.
- TX: parpadea cuando envía datos.

- COLL: parpadea cuando detecta colisiones en la red.

5.2.3. Shield Wifi

La placa Arduino WiFi Shield permite conectar a Arduino a internet de forma inalámbrica mediante protocolo 802.11b/g (WiFi).

A continuación se resumen sus características principales [11]:

- Alimentación: 5V (proporcionado por Arduino).
- Red: 802.11b/g.
- Encriptaciones soportadas: WEP y WPA2.
- Conexión con Arduino por el puerto SPI.
- Zócalo para tarjeta Micro SD incorporado.
- Pines ICSP.
- Conexión FTDI para debug.
- Conexión Mini-USB para actualizaciones de Firmware.



Figura 7: WiFi Shield

La placa cuenta con cuatro indicadores en forma de LEDs:

- L9 (amarillo): conectado al pin 9 de Arduino.
- LINK (verde): indica la conexión a una red.
- ERROR (rojo): indica cuando hay un error de comunicación.
- DATA (azul): indica la transmisión/recepción de datos.

5.2.4. Sensor de temperatura ds18b20.

Los sensores utilizados nos permiten recibir información del entorno. Se han elegido buscando el compromiso entre las especificaciones y un bajo precio. Todos ellos cuentan con características suficientes para la realización del proyecto.

El sensor de temperatura DS18B20 es un dispositivo que se comunica de forma digital. Cuenta con tres terminales, los dos de alimentación y el pin “data”.

Con Arduino podemos leer la temperatura que registra este sensor que posee una característica muy peculiar. Utiliza la comunicación OneWire. Básicamente se trata de un protocolo especial que permite enviar y recibir datos utilizando un solo cable, a diferencia de la mayoría de los protocolos que requiere dos vías. De hecho, Arduino posee los pines RX y TX que son los encargados de enviar y

recibir información.



Figura 8:
Sensor de temperatura
Ds18b20

Sensor DS18S20 de la firma Dallas Semiconductor proporciona medidas de temperatura en grados centígrados con una resolución de 9 bits. Entre sus características cabe citar las siguientes [9]:

- Interface Onewire que precisa de una única línea de datos para la comunicación con el controlador.
- No necesita componentes externos.
- Se alimenta a través de la propia línea de datos o bien mediante una tensión de 3 a 5.5Vcc.
- Rango de temperatura de -55.00°C a +125°C con resolución de +/- 0.5°C.
- Tiempo de conversión de 750mS.

5.2.5. Fotorresistencia LDR.

Una LDR o fotorresistencia es básicamente un sensor de luz muy simple que cambia su valor resistivo con las variaciones de la iluminación ambiente. Las LDR no son muy precisas, por lo que no se puede obtener una lectura cuantitativa ni nada

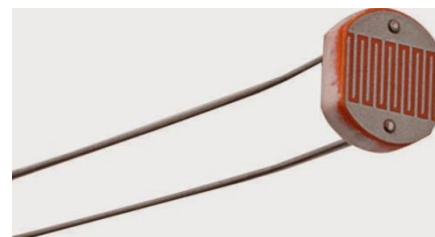


Figura 9: Fotorresistencia LDR

de eso. Sin embargo, son lo suficientemente buenas para diferenciar entre la luz y la sombra, o saber si la luz de una habitación está encendida o apagada.

La LDR cambia su resistencia con la luz por lo que podemos medir ese cambio usando alguno de los pines analógicos de una placa Arduino. Sus valores pueden oscilar desde 1023 que se encontraría cuando existe la oscuridad total 50 que se correspondería con luz brillante.

5.3. SOFTWARE UTILIZADO

Para la realización de este proyecto ha sido necesario el uso de diversos programas y herramientas para cada una de las fases y tareas. A continuación se enumeran los diferentes software utilizados con una breve descripción.

5.3.1. PostgreSQL

PostgreSQL



POSTGRESQL también es un sistema de gestión de base de datos relacional y libre, que agiliza la interacción de cliente, servidor y base de

Figura 10: Icono datos.
postgresql

5.3.2. Geoserver.



Figura 11: Geoserver

GeoServer es un servidor de código abierto escrito en Java, permite a los usuarios compartir y editar datos geoespaciales. Diseñado para la interoperabilidad, publica datos de las principales fuentes de datos espaciales usando estándares abiertos. GeoServer ha evolucionado hasta llegar a ser un método sencillo de conectar información existente a globos virtuales tales como Google Earth y NASA World Wind (véase así como mapas basados en web como OpenLayers, Google Maps y Bing Maps). GeoServer sirve de implementación de referencia del estándar Open Geospatial Consortium Web Feature Service, y también implementa las especificaciones de Web Map Service y Web Coverage Service. [13]

5.3.3. Revit

Autodesk Revit es un software de Modelado BIM, desarrollado actualmente por Autodesk. Permite al usuario diseñar con elementos de modelación y dibujo paramétrico. BIM es un paradigma del dibujo asistido por computador que permite un diseño basado en objetos inteligentes y en tercera dimensión. De este modo, Revit provee una asociatividad completa de orden bi-direccional. Un cambio en algún lugar significa un cambio en todos los lugares, instantáneamente, sin la intervención del usuario para cambiar manualmente todas las vistas. [14]

5.3.4. Maptek I-site

El software Maptek I-Site Studio es un paquete de procesamiento intuitivo de nubes de puntos. Diseñado para aplicaciones mineras, civiles, geológicas y otras de levantamiento, el I-Site Studio tiene una serie de funciones y herramientas potentes para convertir rápidamente conjuntos de datos detallados en resultados fáciles de entender. [15]

5.3.5. IDE Arduino

La programación de Arduino se realizó gracias al software gratuito que puede descargarse en su página oficial [16] (Arduino 1.6.5).

Para que nuestro Arduino pueda funcionar, primero se crea un programa, conocido como “sketch” en el editor de texto del IDE. Posteriormente, se compilará y almacenará en la memoria del microcontrolador.

Además del editor, el IDE cuenta con otras herramientas que nos facilitan la programación, el uso de librerías, la detección de errores, selección de puerto y modelo de placa a programar... Una de las herramientas más utilizadas a la hora de depurar el funcionamiento del programa es el Monitor Serial, que nos permite comunicarnos mediante el USB de nuestro PC y ver mensajes de la ejecución que hayamos programado.

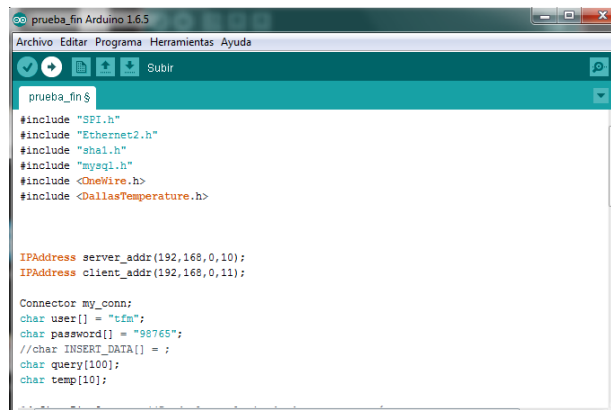


Figura 12: IDE Arduino

5.4. Lenguajes de programación

Los lenguajes utilizados son el de Arduino para programar las placas y en el entorno web, se usan: HTML para la estructura de la página, CSS3 para la edición de estilos y JavaScript para dar dinamismo a la página, por otro lado, el lenguaje PHP para la gestión de información en el servidor.

5.4.1. Lenguaje programación Arduino

El lenguaje de programación Arduino está basado en C/C++. Los programas, denominados comúnmente sketch, se dividen en tres partes principales: estructura, variables y funciones (véase la Figura 13). Además todo programa debe comprender dos funciones esenciales denominadas Setup() y Loop():

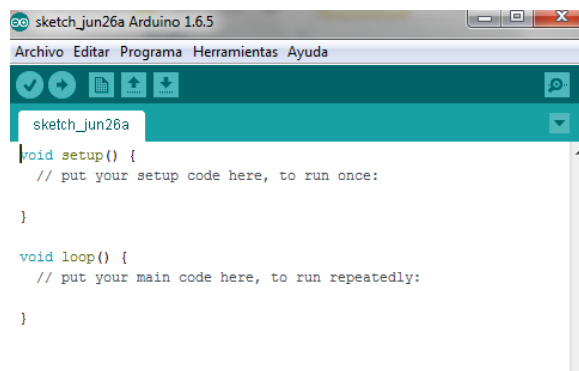


Figura 13: Sketch Arduino

- La función `Setup()`. La función `Setup` se llama cuando se inicia un sketch. Es un lugar para realizar tareas de configuración, como definir los pines o inicializar bibliotecas, etc. La función `Setup` sólo se ejecutará una vez, después de cada arranque o reinicio de la placa Arduino.
- La función `Loop()`. Después de crear una función `Setup()`, que inicializa y establece los valores iniciales, la función `Loop ()` hace exactamente lo que su nombre indica, ejecuta un bloque de código de forma cíclica.

El entorno de programación puede extenderse mediante el uso de librerías que proporcionan funcionalidades extra trabajando con hardware y manipulando información. El IDE trae instaladas de serie unas cuantas genéricas que son muy utilizadas como puede ser la conexión con placas Ethernet, WiFi y GPS, uso de puertos...

En otros casos, deberemos descargarnos las específicas que nos proporcione el fabricante de un determinado hardware o software, e importarlas a nuestro IDE para que puedan estar disponibles cuando realicemos la llamada en nuestro sketch.

5.4.2. HTML, Javascript, CSS y PHP

HTML es el lenguaje que se emplea para el desarrollo de páginas de internet. Está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla. HTML dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas, tablas, etc.



HTML es el lenguaje básico de la web, sirve para crear el contenido de la web y que se pueda usar en cualquier lugar. [25]

JavaScript es un lenguaje de programación, al igual que PHP, si bien tiene diferencias importantes con éste. JavaScript se utiliza principalmente del lado del cliente (es decir, se ejecuta en nuestro ordenador, no en el servidor) permitiendo crear efectos atractivos y dinámicos en las páginas web.

CSS es un lenguaje utilizado en la definición de estilos de documentos HTML. Un documento HTML viene siendo coloquialmente “una página web”. Entonces podemos decir que el lenguaje CSS sirve para organizar la presentación y aspecto de una página web. Este lenguaje es principalmente utilizado por parte de los navegadores web de internet y por los programadores web informáticos para elegir multitud de opciones de presentación como colores, tipos y tamaños de letra, etc. [17]

PHP (PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. Se utiliza para generar páginas web dinámicas. [17] Usado en este trabajo para la gestión de la información en el servidor.

6. METODOLOGÍA

El método general aplicado para la realización de este Trabajo Fin de Máster, el cual se puede aplicar al diseño y monitorización de cualquier edificio existente, independientemente de su uso y envergadura, puede dividirse en las etapas que se observan en el siguiente gráfico (Figura 13).

Se realizará una explicación detallada del método desde la elección del edificio para la monitorización, pasando por todas sus fases, como son la elección de los sensores que se van a usar, los factores a tener en cuenta para el diseño de la red, el levantamiento de la zona de estudio necesario para la representación espacial, los pasos a seguir para poner en funcionamiento todo el sistema (montaje y programación de Arduino), la base de datos y el servidor usado. Y por último, se indica cómo se debe crear el visor web para visualizar los datos y poder hacer un análisis visual de los mismos.

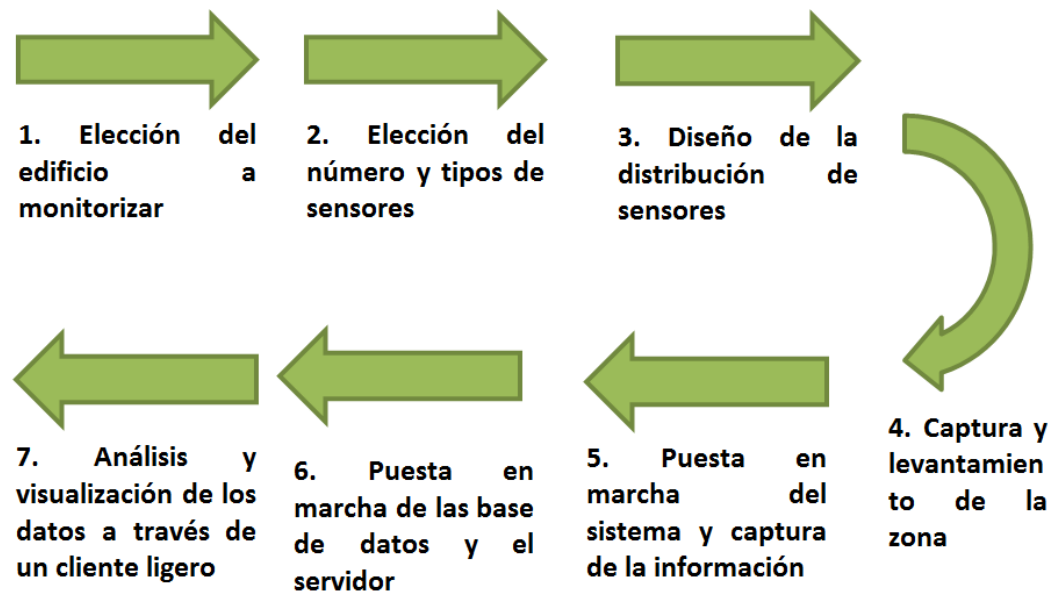


Figura 14. Etapas del método aplicado

6.1. Elección de la sección de edificio a monitorizar

El primer planteamiento es la elección de la sección del edificio en el cual se va a implantar la red de sensores para el monitoreo de los parámetros que se quiera recoger. Para lo que se tiene que tener diversos factores en cuenta para aplicar esta selección.

En función de la envergadura del proyecto y del edificio escogido, así como, del presupuesto del que se disponga para desarrollar este método, se podrá realizar la recogida de datos de todo el edificio o por otra parte se puede seccionar en relación al número de plantas que tenga. También se puede hacer la elección en la sección donde mayor accesibilidad o donde más interés se tenga para implantar la red por algún motivo.

Es importante tener en cuenta la orientación del edificio para ver las variaciones ambientales de los datos recogidos. Esto puede servir de ayuda a la hora de la elección de la que hablamos.

6.2. Elección del número y tipos de sensores

El segundo paso a seguir es escoger qué tipos de aparatos de medición se quieren usar. Para lo que se realiza un estudio de los diferentes tipos de sensores existentes compatibles con Arduino.

En la siguiente tabla (Tabla 2) se muestran un ejemplo de los tipos de sensores, más relevantes, existentes en la actualidad para Arduino.

Tipo	Ejemplo de sensores
Temperatura	Sensor de temperatura digital, Sensor de temperatura analógico, Termistor, Sensor de temperatura y humedad
Luminosidad	Foto-resistencia LDR, sensor digital de intensidad lumínica, sensor UV
Humedad	Sensor de temperatura y humedad DTH11, Sensor humedad del suelo
Movimiento	Sensor de movimiento PIR, Sensor de vibración, acelerómetro
Posición	Sensor de distancia por ultrasonidos, Sensor de proximidad por infrarrojos
Acústico	Micrófono, Sensor de sonido
Fuego/gas	Sensor de gas, sensor de llama

Tabla 2: Tipos de sensores

En principio, como la metodología es aplicable a un edificio, hay que seleccionar los que se adapten a las necesidades de interiores. Por ejemplo, no tendría sentido poner un sensor para medir la humedad del suelo o un sensor UV, por ello son descartados.

Se optan por dispositivos de bajo costo de los cuales hay una gran variedad [26]. No es necesario realizar un desembolso importante para capturar los datos, ya que este tipo sensores capturan datos de manera bastante precisa.

Es necesario saber cuál es el propósito del proyecto, puede ser la detección de presencia para un sistema de seguridad, o detectar el humo o gas en una cocina....Dependiendo del objetivo se precisará de un tipo de sensor u otro.

En el caso particular desarrollado en este TFM, como se quieren capturar datos ambientales y de confort en el interior de un edificio, se han elegido dos tipos de sensores, el primero de ellos es de temperatura, que además es un sensor digital, y el segundo recogerá la intensidad lumínica, sensor analógico. Además, la elección de estos dos sensores permite poner un ejemplo de cada una de las tipologías de captura de información.

Por consiguiente también será necesaria una placa que lea los datos capturados por el sensor. Habrá que elegir entre la diversidad de placas Arduino existentes (Uno, Genuino, mega, Leonardo...) cada uno con diferentes características y precios.

En este caso se ha usado Arduino Uno, cuyas características ya se han detallado en el apartado de instrumentación. Aunque cuenta con poca capacidad de memoria, es suficiente para cumplir con el objetivo de este trabajo. En otros casos habría que valorar la cantidad de datos a recoger para la elección del tipo de Arduino.

En conclusión la elección del tipo, número de sensores y la placa elegida se realizará en función de las necesidades a cubrir y del presupuesto que se disponga.

6.3. Diseño de la distribución de sensores

El diseño de la distribución de sensores dependerá del tipo de comunicación que utilicen. Es decir, cómo estarán conectadas a la red, si de manera inalámbrica o por cable. Por lo que para poder colocar los sensores primero habrá que conocer qué tipo de comunicaciones existen y cuál puede ser la más interesante en cada caso.

Según el tipo de conexión a internet se usará un tipo de dispositivo u otro, estos suelen llamarse Shield. Si es por cable, se usará un Shield Ethernet, el cual ya ha sido descrito en el apartado de hardware.

Por el contrario, las conexiones inalámbricas se implementan mediante el uso de otro tipo de Shield como son el Wifi o Wireless. Un Shield Wifi permite trabajar con comunicaciones inalámbricas usando el bus SPI para comunicarse con Arduino. Mientras que el Shield Wireless es un componente electrónico que permite adaptar diferentes módulos de comunicación, como pueden ser WiFi, Bluetooth, Zigbee... Pero por sí mismo no realiza ninguna conexión. A continuación se describirá cómo funcionan este tipo de módulos y sus principales características para poder hacer la elección del tipo de comunicación que se requiera en cada caso.

Los módulos Wireless necesitan enviar la información a un nodo coordinador o padre que tiene que ser uno de los dispositivos, los demás nodos son secundarios. El esquema de distribución de sensores está basado en comunicar cada nodo secundario con el coordinador. Entonces las topologías más usadas en estos casos son en forma de estrella, árbol o malla.

Por el contrario con Wifi o Ethernet no es preciso un coordinador, todas las placas pueden ser independientes, es decir que no se comuniquen entre ellas, aunque envíen información al mismo servidor.

El diseño de la distribución tiene que estar basado según el número de sensores del que se disponga, la zona que queramos cubrir con la red y el sistema de comunicación elegido entre los anteriores comentados.

6.3.1. Zigbee

Los módulos XBee utilizan el protocolo IEEE 802.15.4 mejor conocido como ZigBee. Este permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas entre sus diferentes módulos. Es especialmente útil para redes de sensores en entornos industriales, médicos y, sobre todo, domóticos. Tienen su propio protocolo de comunicación por radio frecuencia (en banda de radio libre), son robustos, de bajo costo, bajo consumo y tienen un alcance en sus distintos modelos entre 100 metros y los 10 kilómetros.

Cada módulo Zigbee, al igual que ocurre con las direcciones MAC de los dispositivos Ethernet, tiene una dirección única. Cada vez que un dispositivo se asocia a una red Zigbee, el Coordinador al cual se asocia le asigna una dirección única en toda la red de 16bits. [18]

6.3.2. Bluetooth

El Bluetooth es un estándar de comunicación inalámbrica que permite la transmisión de datos a través de radiofrecuencia en la banda de 2,4 GHz. Existen dos modelos de módulos Bluetooth: el HC-05 que puede ser maestro/esclavo (master/slave), y el HC-06 que solo puede actuar como esclavo (slave). La diferencia entre maestro y esclavo es que en modo esclavo es el dispositivo quien se conecta al módulo, mientras que en modo maestro es el módulo quien se conecta con un dispositivo. [19]

La tecnología Bluetooth es útil cuando la transferencia de información entre dos o más dispositivos que están cerca unos de otros cuando la velocidad no es un problema. Es el más adecuado para aplicaciones de bajo ancho de banda. Por lo que no es útil para redes de sensores.

6.3.3. WiFi

WiFi es más adecuado para el funcionamiento de las redes a gran escala, ya que permite una conexión más rápida, mejor rango de la estación base, y una mejor seguridad inalámbrica.

En resumen se podría decir que hay muchos tipos diferentes de redes de inalámbricas con diferentes ventajas y desventajas. Las principales características que se debe considerar al elegir un método de comunicación inalámbrica son topología de la red, velocidad de datos y el consumo de energía:

- Con WiFi se tiene alta velocidad de datos y también alto consumo de energía.

- Zigbee tiene baja tasa de datos y el bajo consumo de energía. Se utiliza para hacer redes de sensores de tipo malla.

- Con la tecnología Bluetooth la velocidad de datos y el consumo medio es de potencia media.

6.4. Captura y levantamiento de la zona

La determinación de la forma real de un edificio no es labor sencilla, y máxime cuando el objeto presenta dificultades intrínsecas como puede ser su tamaño o su propia forma. En general la adecuada definición de sus formas nos obliga a considerarlo como objeto tridimensional y a realizar su medición mediante sistemas que nos proporcionen la ubicación en el espacio de sus distintos elementos, es decir, que nos permita conocer las tres coordenadas cartesianas de los distintos puntos que definen su forma. La medición y levantamiento de edificios y construcciones ya existentes plantea generalmente dificultades debido a la inaccesibilidad de muchas partes.

Los métodos de levantamiento para el interior de un edificio se pueden dividir generalmente tres métodos de levantamiento gráfico: el levantamiento manual, el levantamiento instrumental o topográfico y el levantamiento fotogramétrico.

- El levantamiento manual consiste en tomar medidas directas utilizando los instrumentos de medida clásica: metro láser, cinta métrica, nivel de agua, jalón de alineamiento, etc.
- El levantamiento instrumental consiste en utilizar instrumentos topográficos: teodolito, estación total o láser escáner.
- El levantamiento fotogramétrico consiste en utilizar la fotografía y los programas informáticos en la elaboración del levantamiento. [20]

A continuación se realizará un estudio de los diferentes métodos de captura, ya comentados, para poder determinar la elección adecuada en cada situación.

Las técnicas basadas en el empleo de simples dispositivos de medición lineal (cintas métricas), combinadas con dispositivos para el control de la verticalidad (plomadas y niveles), además de no darnos información tridimensional clara de los elementos medidos, exigen poder acceder a los puntos extremos entre los que se realiza la medición. Por ello, en trabajos complejos y en los casos en que no es posible este acceso, se hace preciso acudir a técnicas instrumentales que mediante el uso de aparatos o equipos más o menos sofisticados. Estos nos permiten la medición a distancia sin tener necesidad de acceder físicamente a los puntos de estudio. Este proceso de medición es la forma más laboriosa y la que puede llevar a cometer más errores, aunque no por ello sigue sin ser válida. Sobre todo en el caso de pequeños

edificios, tipo viviendas o locales comerciales es una opción que se sigue usando en la actualidad.

En el caso de puntos inaccesibles, así como en el caso de no disponer de instrumentos de medida, la fotogrametría sirve de método fácil y rápido para levantamiento de un edificio. Basta con tomar dos fotos con una máquina fotográfica especial para un mismo objeto y revelarlas mediante un programa informático específico para obtener la recuperación de la perspectiva y restituir la “foto-elevación” en tres dimensiones. Gracias a estos métodos se trabaja en coordenadas espaciales con lo que se tiene siempre a mano datos del volumen y espacio en lugar de datos puramente planos. También si vamos a realizar trabajos fotogramétricos, generalmente, es necesaria la obtención de las coordenadas de algunos puntos de control, para su utilización en el proceso de orientación.

Las técnicas instrumentales de medición se basan fundamentalmente en la determinación de la dirección y la distancia a la que se encuentran los puntos del objeto con respecto a aquellos desde los que se realizan las mediciones. Posteriormente se realiza el trabajo de gabinete en el que se determinan las coordenadas de cada punto, que insertándolas en un programa CAD y con ayuda de los datos tomados en campo se realiza el plano del edificio. Para la determinación de estos datos se recurre al uso de aparatos topográficos, como pueden ser teodolitos o estaciones totales. [21]

Es de mención aparte el uso del láser escáner 3D. Es la solución tecnológica más avanzada y exhaustiva de las vistas en esta sección. Con él se puede desarrollar el levantamiento y obtener una nube de puntos con coordenadas tridimensionales que dibujen la realidad del edificio. También, si es necesario se toman unos puntos de control para la posterior orientación de las nubes de puntos.

Ni que decir tiene que esta última opción es la más cara, en cuanto al costo del alquiler del instrumento, y es posible que no se dispongan de los medios económicos para realizarla. Aunque un factor muy importante es que reduce el tiempo de captura de datos considerablemente, por lo que se abaratan los costos por otro lado. Suele ser muy útil en el caso de edificios de difícil medición como pueden ser edificios históricos, es una opción muy a tener en cuenta, ya que simplifica en gran medida el proceso de captura de datos.

Una vez visto los métodos existentes para poder realizar la captura, se explicará en detalle las fases de ejecución del levantamiento, en este caso aplicado al láser escáner, ya que es la técnica usada en este trabajo.

Como ya se ha explicado, en el apartado de instrumentación el funcionamiento, la tecnología y la utilidad de esta herramienta de medición, ahora se hará enfoque en su método de empleo.

Existe un principio común en la metodología básica para la captura de datos y el post-proceso en la mayoría de los instrumentos de escáner laser 3D, es por ello que la técnica utilizada se divide en tres etapas.

La primera etapa consiste en la planificación, donde se hará un diseño de las bases de estacionamiento del instrumento, para evitar zonas en sombra. La segunda etapa es el trabajo de campo en la toma de datos del escáner, aquí se seleccionará la densidad de puntos de escaneo, esto permitirá aumentar el área de captura, y con ello minimizar las interferencias posibles.

El escáner es una herramienta que documenta una superficie por medio de láser, y la presencia de obstáculos puede impedir que se registren algunas partes del elemento, creando sombras en la nube de puntos. Es por ello necesario posicionar en varios sitios el escáner, logrando con ello completar la zona oculta y abarcar la totalidad del edificio.

Una vez ubicado el Escáner laser 3D sobre un trípode en sus puntos de análisis se procederá al escaneo del elemento.

La tercera etapa es el trabajo de gabinete, donde la información se procesa con el fin de obtener resultados, y evaluar su importancia y calidad.

La primera fase del procesado de esta, es la unión de los distintos escaneos que se hayan realizado. Luego se deberán eliminar, a partir de un proceso de filtración de manera manual y controlada, todos los elementos que interfieran o distorsionen los resultados de representación.

Una vez procesada la nube de puntos es posible exportarla, a otro software de digitalización virtual como, por ejemplo, los sistemas BIM. Allí se podrá modelar y dar forma al elemento escaneado obteniendo un modelo en tres dimensiones.

6.5. Puesta en marcha del sistema y captura de la información

En este punto se detallará los pasos a seguir hasta poder capturar la información de los sensores, para ello se requiere el montaje, programación y ejecución del sistema.

Cada elemento, cuenta con unas características que deberán ser tenidas en cuenta a la hora del cableado y su comunicación.

Para montar el sistema se necesita la instrumentación elegida, en este caso la placa Arduino Uno, una Shield Ethernet, los sensores elegidos en cada caso, las resistencias apropiadas y una protoboard para la conexión de los cables correspondientes.

Todo circuito en Arduino, debe estar alimentado a través del pin 5v. Y por medio de la resistencia, los pines VCC y DATA van conectados entre sí y a tierra. Para el correcto montaje de cada sistema se puede encontrar mucha información de manera fácil y rápida a través de la web, ya que Arduino es un programa de código abierto, como ya se sabe. De manera general, la conexión de los sensores suelen ser muy sencillas.

Para la correcta lectura de valores existen librerías que permiten simplificar el proceso de captura de datos y arranque del dispositivo. Como se comentará posteriormente.

Una de las grandes ventajas de usar Shields con Arduino es que no es necesario cablear, sino que se montan apilándolos aprovechando los pines de los conectores que incorporan. Al contar con una forma idéntica a la placa Arduino, no surgen dudas a la hora de realizar la conexión pues encajan perfectamente uno sobre otro.

Un ejemplo de conexión de sensores, en este caso de temperatura y luminosidad, que son los que se han usado para realizar este prototipo es el que se puede observar en la siguiente imagen.

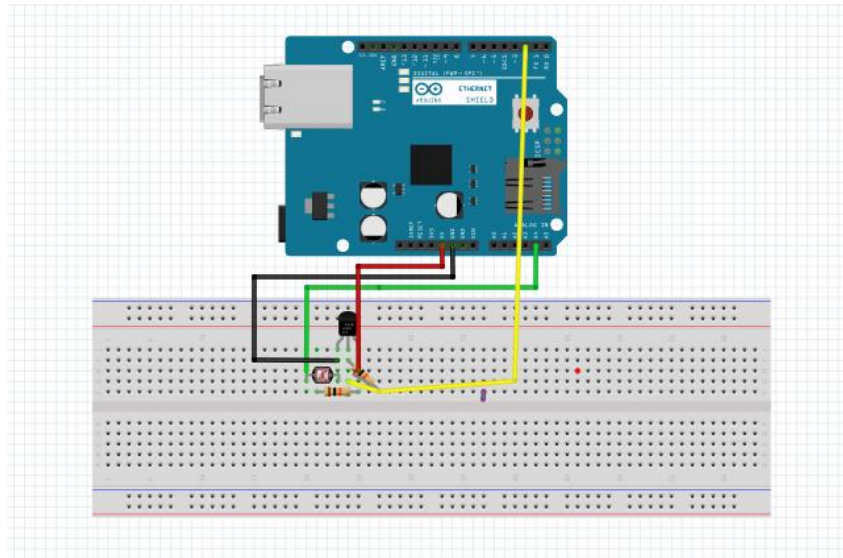


Figura 15: Esquema de conexión en Arduino

La programación es el siguiente paso para poder capturar los datos. Como se sabe, un sketch es el código que programaremos en el microcontrolador para marcarle su funcionamiento.

Como norma, en cada sketch de Arduino, y como no podía ser de otra forma, en los de este trabajo, podremos encontrar estas partes:

- Comentarios. No afectan al código ni se cargan en la placa, por lo que no ocupan memoria, pero son recomendables para poder entender lo que se ha programado y añadir información útil como dónde se deberán conectar ciertos dispositivos.
- Llamada a librerías y declaración de variables.
- Y las funciones Setup y Loop, como ya se explicó, se diferencian en que la primera se ejecuta una sola vez y la segunda es un bucle donde que ejecutará el Arduino, una y otra vez.

Aunque el loop sea la parte donde se ejecuten las órdenes que definimos para el funcionamiento del sistema, es posible el uso de funciones como en la mayoría de lenguajes de programación. Éstas se crean fuera del bucle, pero al ser llamadas dentro del Loop se ejecutan también de forma ininterrumpida.

Los sketch de programación de cada placa se adjuntan completos en la carpeta “Arduino”. Por su extensión, se ha preferido no incluirlos en esta memoria pero sí se nombrarán las principales funciones y librerías utilizadas.

La programación de Arduino en este caso, de forma general, consigue que el microcontrolador realice las siguientes tareas:

- Activar la comunicación con Ethernet Shield, el sensor de temperatura y el sensor LDR.
- Conectar el dispositivo a la red cableada. Para ello se deben definir la dirección IP que queremos fijar.
- Conectar con el servidor a través de la librería para recibir los datos en la base de datos, que se establecerá en el siguiente punto. Para lo que se necesitarán los parámetros de conexión a la misma, como el usuario y la contraseña.

A continuación se enumeran las utilizadas en este sketch y se describe brevemente para que sirve cada una de ellas:

```
#include "SPI.h"
#include "Ethernet2.h"
#include "sha1.h"
#include "mysql.h"
#include <OneWire.h>
#include <DallasTemperature.h>
```

Figura 16: Librerías Arduino

- **Ethernet2.h:** Conexión del shield Ethernet a internet. Permite aceptar conexiones entrantes y salientes.
- **SPI.h:** Gestiona la comunicación con dispositivos SPI (Serial Peripheral Interface). Se usan para la comunicación entre el Shield y la placa Arduino.
- **Sha1.h y mysql.h:** conectan Arduino con el servidor mysql.
- **oneWire.h:** Activa el sistema de comunicación OneWire, en el que se envía y recibe información a través de un único cable.
- **DallasTemperature.h:** interpreta los datos de temperatura.

Las funciones más destacables de este código son las siguientes:

- **sensors.requestTemperatures();** Prepara los datos para su lectura.
- **sensors.getTempC(direccionDigitalSensor);** obtiene la temperatura.
- **dtostrf(temperatura, 1, 1, temp);** convierte la variable float de temperatura en char para ser introducida en mysql.
- **sprintf(query, "INSERT INTO tfm.datos (id_s, T, Luz) VALUES (%d,%s,%d)", 1, temp, analogRead(LDR_pin));** con esta función se graban los datos en la tabla "datos" del servidor.

En definitiva lo que se consigue con este código es que Arduino se conecte a internet mediante la Shield Ethernet, recoja los datos de los sensores y los introduzca en una tabla alojada en una base de datos del servidor.

Siguiendo los pasos de montaje y programación anteriores, el sistema estaría listo para funcionar con solo proporcionar la alimentación y ejecutar cada sketch. Podremos visualizar los datos en el monitor serial. Y una vez programado comprobar si se graban los datos correctamente en el servidor.

Una vez concluidos los anteriores pasos, se procede a la captura de información de los datos recogidos por los sensores de temperatura y luminosidad.

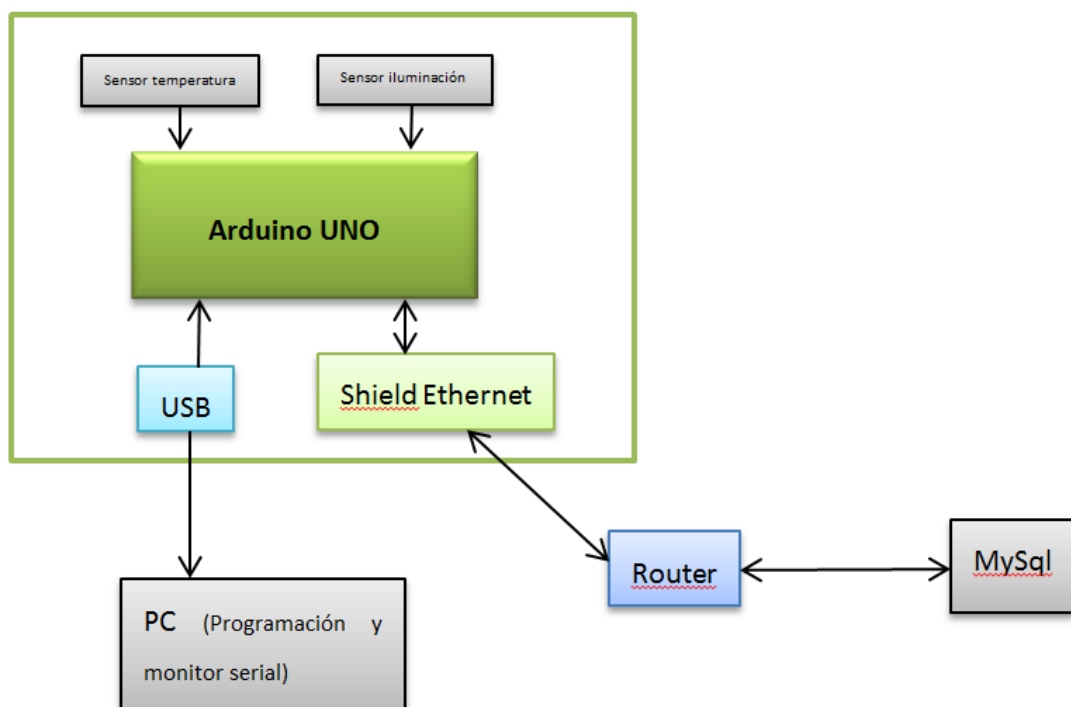


Figura 17: Esquema de comunicación del sistema

A modo de resumen, en el esquema anterior se puede observar la comunicación entre los diferentes elementos del sistema para su correcto funcionamiento.

En definitiva, los sensores y la Shield son conectados a Arduino para recibir las lecturas de los mismos. En esto consiste en montaje de la placa. Los anteriores elementos se conectarán a través del puerto USB al ordenador para programar y mostrar la información por el monitor serial. Y por último, se conecta a la red por cable, gracias al Ethernet Shield, y envía los datos al servidor, tal como se ha programado.

6.6. Puesta en marcha de las bases de datos y el servidor

A continuación se pone en marcha las bases de datos y el servidor para poder consultar la información obtenida.

Se crea una base de datos en el software Mysql, en las que irán incluidas las tablas donde se grabarán los datos obtenidos por los sensores. Ya que el código de Arduino está dispuesto para grabar en Mysql con la librería conector mysql-Arduino.

Para la creación de la base de datos se debe crear una nueva conexión en Mysql en la que se establezca la IP del servidor, un usuario y una contraseña (Figura 16).

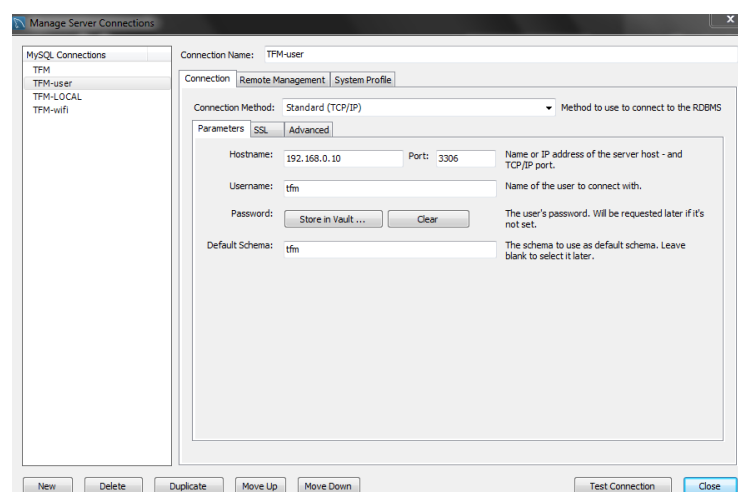


Figura 18: Conectar servidor MySQL

Más tarde se crea la base de datos y las tablas alojadas en la misma. Para la creación del diseño de las tablas se ha optado por crear dos tablas independientes, en la primera de ellas estarán incluidas las coordenadas en las que se han posicionado los sensores. Mientras que en la segunda el contenido será los datos recogidos por el sensor. Ambas estarán unidas por un identificador común, como se puede observar en las siguientes sentencias SQL que se han usado para la creación de las mismas, en esta ocasión.

SQL tabla 1:

```
Create table coordenadas (id_s int, geom GEOMETRY);
```

```
insert into coordenadas values (1, st_pointfromtext('POINT(431531.41 4182505.05)'));
```

```
insert into coordenadas values (2, st_pointfromtext('POINT(431515.12 4182506.80)'));
```

```
insert into coordenadas values (3, st_pointfromtext('POINT(431497.22 4182506.59)'));
```

```
insert into coordenadas values (4, st_pointfromtext('POINT(431476.19 4182506.82)'));
```

```
insert into coordenadas values (5, st_pointfromtext('POINT(431460.17 4182505.55)'));
```

SQL tabla 2:

```
CREATE TABLE datos (
```

```
    num serial primary key,
```

```
    id_s integer,
```

```
    T float,
```

```
    Luz integer,
```

```
    f_h timestamp );
```

	num	id_s	t	luz	f_h			id_s	geom
	integer	integer	double precision	integer	timestamp without time zone			integer	geometry

Figura 19: Tablas creadas en MySQL

Acabada la generación de las tablas entramos en el servidor web, Geoserver. Para poder importar los datos incluidos en dichas tablas en dicho servidor y poder visualizar los mismos, se debe de instalar un plugin de Mysql en el mismo.

Terminado el proceso de instalación se puede crear un espacio de trabajo y un almacén de datos en el servidor web con los parámetros para conectar con la base de datos deseada. Y se implementarán las capas configurando con una vista SQL en la que se unen las tablas que anteriormente hemos hecho mediante el identificador para que tengan la geometría y los datos recogidos. En esta consulta también se efectúa la media aritmética de la temperatura y luminosidad obtenida con el sensor.

Creadas las capas queda darle un estilo visual lo más adecuado posible. Para ello, se pueden editar los archivos sld del software Qgis, guardándolos con la configuración más adecuada para poder visualizarlos en Geoserver.

Se puede optar por tener diferentes tamaños para cada punto que contendrá los valores medios del sensor, o diferentes colores para los mismos, etc. si hay dos o más sensores diferentes, se puede optar por opciones combinadas o por la misma opción.

Para diferenciarlos se elige o bien simbologías diferentes para cada uno, en la que se pueden incluir las opciones anteriores de cambio de tamaño. O bien la misma simbología diferenciándose por el color, por ejemplo.

6.7. Análisis y visualización de los datos a través de un cliente ligero.

Es necesario que exista un interfaz que permita al usuario visualizar la información recogida. Para ello contamos con el entorno web que consta de tres partes bien diferenciadas, pero todas ellas interconectadas.

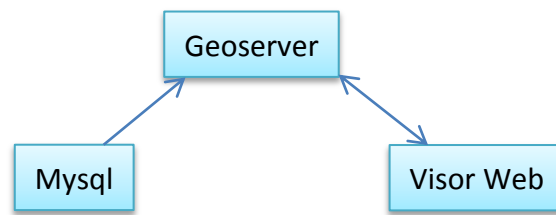


Figura 20: Esquema BBDD, servidor web y visor web.

Para poder visualizar los datos que proporcionan los sensores en la web es necesario cargar las capas que se han creado en el servidor web, anteriormente.

En la elaboración de la web se usan tres lenguajes: html para la elaboración de la web, JavaScript para ejecutar las funciones que realizará la página y css para crear el estilo de la misma.

El formato que deberá tener el cliente será el siguiente:

- Una cabecera.
- Un mapa donde se visualizarán las diferentes plantas del edificio, o las de estudio.
- Las capas de los diferentes sensores situadas en el plano anterior.
- Una leyenda en la que aparezcan los diferentes valores del sensor con el formato establecido.
- Por último, un apartado donde poder actuar sobre la información recogida. En este habrá formularios donde poder seleccionar la fecha y hora para poder ver los días que nos interesen. En los que se pueda realizar un análisis visual de los datos elegidos.

El diseño final quedará como se puede visualizar en la Figura 19.

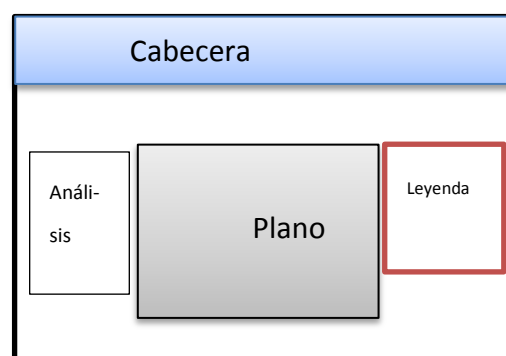


Figura 21: Diseño web propuesto

7. DESARROLLO E IMPLEMENTACIÓN

7.1. Elección de la sección de edificio a monitorizar

En primer lugar, se ha optado por la elección de una parte del edificio que es la zona de estudio para este trabajo. Se ha designado parte de la tercera planta del edificio A3 del Campus de las Lagunillas, como ya se ha mencionado en el apartado situación y emplazamiento. De esta planta se estudiará principalmente el pasillo que se encuentra más al norte. Se ha escogido el mismo, ya que es donde se encuentran los despachos del profesorado del departamento de Ingeniería geodésica y cartográfica. Y esto lo hace ser el lugar más accesible e idóneo, al poder situarse los instrumentos dentro de algunos despachos, en caso de ser necesario, a lo largo del pasillo. También se ha tenido en cuenta la orientación del mismo de este a oeste.

En la siguiente imagen se puede apreciar la sección descrita anteriormente (Figura 20).

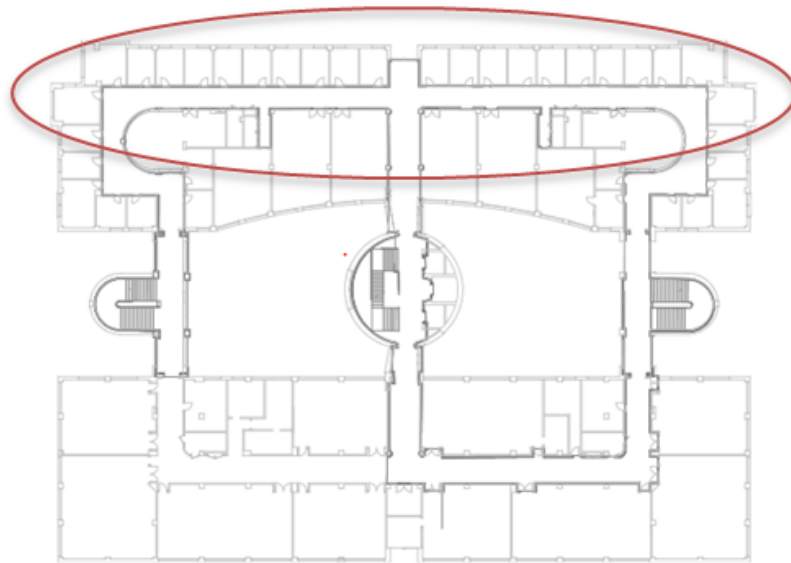


Figura 22: Sección de edificio elegida.

Para el propósito de este TFM, la elección de una zona pequeña permite probar tanto la captura 3D, como la creación del modelo, además del montaje y prueba del sistema de sensores con diferentes componentes y metodologías.

7.2. Elección del número y tipos de sensores.

En segundo lugar se ha escogido el número de sensores a incorporar, así como el tipo de los mismos. Dado que uno de los objetivos de este proyecto es recoger datos ambientales y de confort del edificio se ha optado por la elección del uso de sensores de temperatura y luminosidad. Como se ha comentado en la metodología, se han seguido los pasos para seleccionar qué sensores necesitamos que en ella se describieran. Finalmente, se han preferido los modelos Ds18b20 para el sensor de temperatura y una fotorresistencia LDR para medir la intensidad de luz.

Se ha optado por este modelo de sensor de temperatura entre otros porque es un dispositivo que se comunica de forma digital. Cuenta con tres terminales, los dos de alimentación y el pin “data”.

Con Arduino se lee la temperatura que registra este sensor que posee una característica muy peculiar, el uso de la comunicación OneWire. Básicamente se trata de un protocolo especial que permite enviar y recibir datos utilizando un solo cable, a diferencia de la mayoría de los protocolos que requiere dos vías.

Las ventajas son varias:

- Mayor precisión.
- Inmunidad frente a interferencias.
- Identificación de sensores por dirección.
- Con una sola línea de datos que en Arduino se gestionará con un solo pin, se puede gestionar todo un bus con varios sensores conectados a él.
- No necesitan calibración.
- El DS18B20, tiene un rango de temperaturas de funcionamiento de -55 °C a 125 °C.
- La resolución del sensor de temperatura es configurable por el usuario a 9, 10, 11, o 12 bits, correspondiente a incrementos de 0,5 ° C, 0,25 ° C, 0.125 ° C y 0,0625 ° C.

El otro sensor empleado es una fotorresistencia LDR es un componente electrónico cuya resistencia disminuye con el aumento de intensidad de luz incidente.

Ventajas de las fotorresistencias:

- Alta sensibilidad (debido a la gran superficie que puede abarcar).
- Fácil empleo.
- Bajo costo.
- No hay potencial de unión.
- Alta relación resistencia luz-oscuridad.

Pese a que su precisión no es muy elevada, es suficiente para el propósito de este trabajo. Ya que puede apreciarse con facilidad si una luz está encendida o apagada.

Uno de los principales motivos de haber seleccionado estos dos tipos de sensores, además de los que ya se han comentado, es para poder trabajar de forma simultánea con dos tipos de comunicación distinta, la digital con el sensor de temperatura y por otro lado la analógica con la LDR.

Una vez decidido qué sensores y modelos de los mismos se iban a usar. Se delibera cuántos de ellos son necesarios para crear la red.

Al disponer de cinco placas Arduino con sus correspondientes Shields, en cada placa se incorpora un sensor de temperatura más una fotorresistencia. Por lo tanto, habrá cinco sensores de cada tipo.

7.3. Diseño de la distribución de sensores

El diseño para la red de sensores elegido finalmente es una conexión mixta: dos de los nodos estarán conectados a la red por cable y los tres restantes tendrán una conexión Wifi.

El diseño que en principio se había propuesto, como se explica en la metodología, todos los Arduinos estarían conectados por cable a la red, ya que de esta forma la conexión es más sencilla.

Ha variado debido a que en los pasillos se habían retirado los routers instalados en un principio por motivo unas las modificaciones que se han producido en la red integrada en el campus. Por ello, ha sido imposible tener una conexión por cable y ha habido que diseñarla de manera mixta, como ya se ha comentado.

Por tanto, las tres placas conectadas por Wifi son las que se han establecido en los pasillos, mientras que las otras dos siguen conectadas por la red cableada ya que se encuentran en los despachos del profesorado en los que sí ha sido posible conectarse como en un principio se había diseñado.

El diseño de la red contempla un total de diez sensores distribuidos por toda el área, cinco de temperatura y otros tantos de luminosidad, así como un coordinador, que es el servidor. Cada uno de los nodos es independiente, no se comunican entre ellos, enviando los datos al coordinador. Como se puede apreciar en la Figura 21.

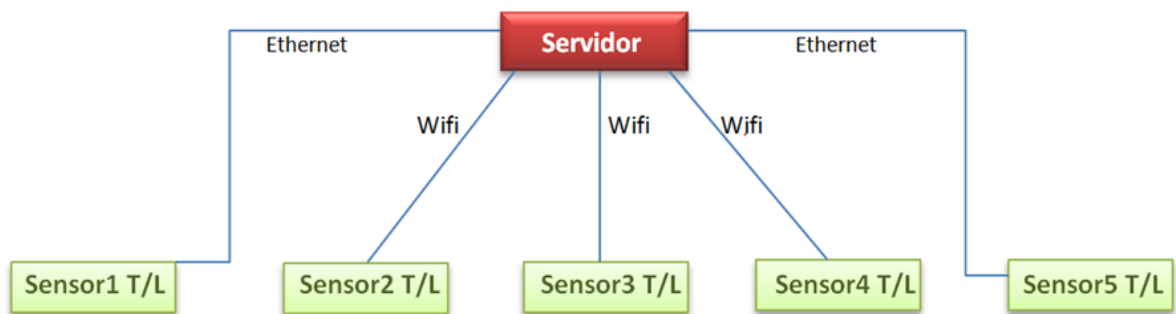


Figura 23: Esquema de conexión de los sensores final.

La distribución de los de sensores dentro de la sección de edificio por la que se ha optado anteriormente. Están dispuestos de manera lógica para que no se falseen los datos, por ejemplo, no ponerlos al lado de una ventana en la que inciden de manera directa los rayos solares , muy cerca de un aire acondicionado, o situaciones similares son las que se han tenido en cuenta a la hora de distribuir los nodos.

La situación de los mismos finalmente ha sido la siguiente: los dos sensores instalados en los despachos se colocan sobre una mesa en la que no le da la luz directa del sol, ni el aire acondicionado, para que los datos sean reales. Por otro lado, los ubicados en los pasillos se han colocado sobre un falso techo siguiendo el mismo protocolo que los anteriores.

El diseño del sistema se ha realizado como un prototipo de gestión real de datos recogidos en un edificio.

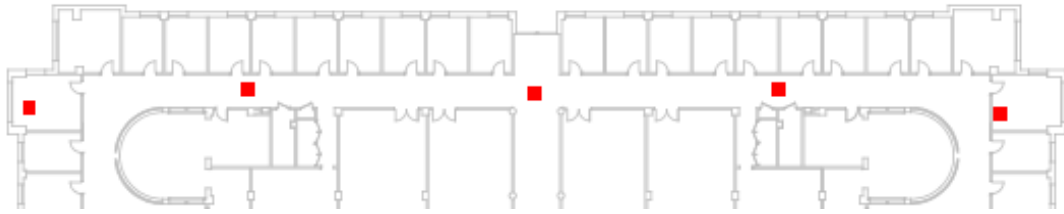


Figura 24: Distribución de los sensores.

7.4. Captura y levantamiento de la zona

Una vez elegida la ubicación de cada dispositivo se procede a la captura y levantamiento de la sección elegida. El primer paso ha sido la elección del método para llevar a cabo la captura de la zona que ha sido el método instrumental. Para ello se ha usado el láser escáner Leica, modelo C10.

Frente al uso de Estación total y dibujo a mano, se obtiene una reducción significativa de tiempo de medición y un aumento aún más significativo del detalle y resolución del registro volumétrico.

Cabe resaltar que algo que siempre se ha señalado como un problema (obtener muchos puntos pero no saber qué hacer con ellos) es realmente la más inmediata ventaja del escáner: aunque no le saquemos de forma inmediata todo el partido posible a estos millones de puntos, obtenemos una base de datos bruta de enorme precisión y detalle, que es en sí un producto altamente útil como tal.

Anteriormente, se ha estudiado el mejor lugar para establecer los puntos de estacionamiento, para así no tener zonas en sombra en la nube de puntos que obtendremos.

En la siguiente imagen se puede observar la situación final de los estacionamientos (Figura 23). Como se puede comprobar hay un total de quince bases en las que se ha situado el escáner. Se ha llevado a cabo este número de estacionamientos, en primer lugar, para reducir la densidad de puntos, y en segundo, por la estrechez de los elementos a capturar. De esta forma el número de puntos total es algo menor, más fácil de procesar pero con densidad suficiente para capturar no sólo la información general si no información de detalle.

Una vez estacionado y configurado el aparato se toman los datos en cada una de las estaciones.

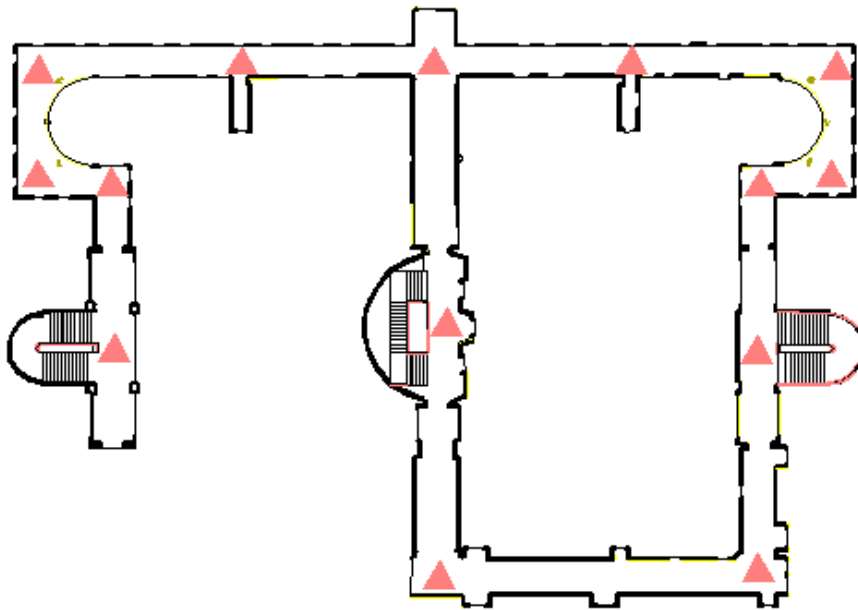


Figura 25: Bases de estacionamiento.

Una vez realizada la captura de los datos, se obtiene una nube de puntos de cada base, las cuales hay que ajustarlas y encajarlas, mediante giros y traslaciones, unas con otras para que tengan el mínimo error de cierre. Hay que aclarar, que al no haber definido puntos de control, las coordenadas resultantes de cada punto son coordenadas locales, Ya que no era necesario disponer de una orientación y posicionamiento absoluto de la representación 3D. Todo ello se realiza con el software Maptek I-site.

El siguiente paso sería la exportación de la nube de puntos a Autodesk Revit.

Los datos de dicha nube son utilizados en Revit como modelo de información de construcción, capaz de representar la geometría del edificio existente Figura 24.

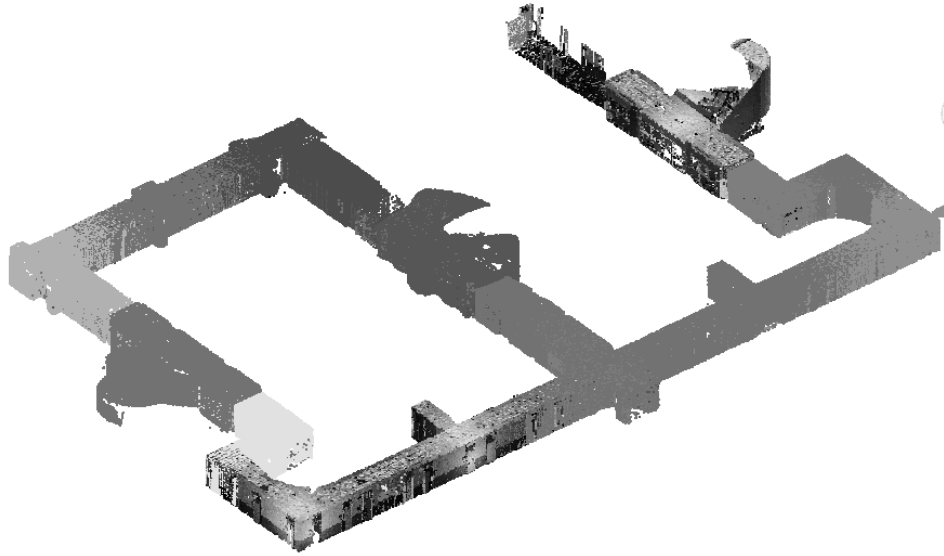


Figura 26: Nube de puntos

Una vez implantada la nube de puntos será posible crear de manera inversa un proyecto virtual, compuesto por los elementos paramétricos que nos ofrece la citada plataforma Revit.

Una vez posicionados los niveles correctamente, será posible generar en el modelo 3D creando los suelos, muros, techos, escaleras, etc. Y se le añaden los materiales constructivos correspondientes, editando cada elemento para poder reproducir la realidad. Se pueden obtener diferentes vistas, verlo tanto en 3D, como en planta, perfiles, etc. Figuras 25 y 26.

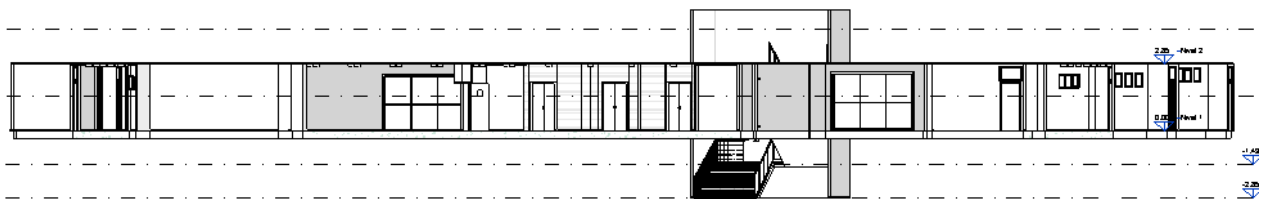


Figura 27: Perfil del modelo en Revit.

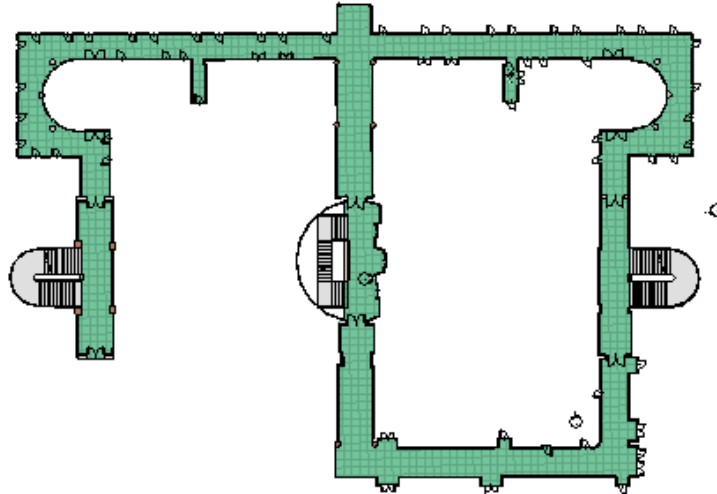


Figura 28: Planta del edificio en Revit.

Se genera un modelo 3D virtual capaz de emular el edificio real y del que se pueda extraer información métrica, composición elementos, a su vez con la ventaja de que puede ser visualizado fácilmente. Los visualizadores permiten explorar el modelo y generar las vistas necesarias para una mejor comprensión del objeto de estudio. Además se han añadido el posicionamiento de los sensores para poder visualizar su ubicación en el modelo BIM. Figura 27. Tanto el modelo bidimensional como el tridimensional, están incluidos en el visor web, como se verá en el punto 6.7.

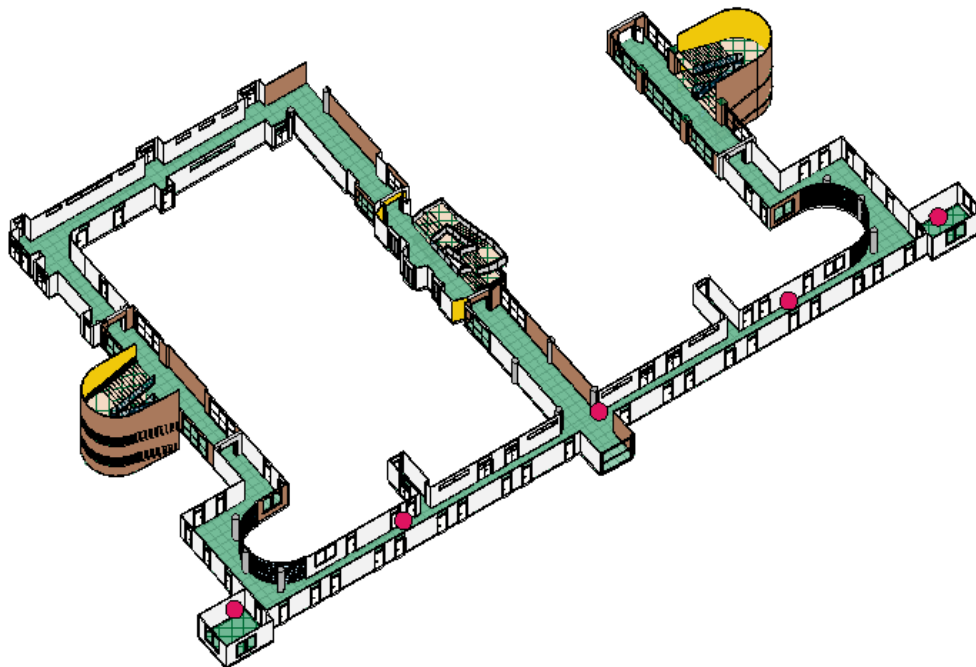


Figura 29: Modelo 3D de la sección de edificio

7.5. Puesta en marcha del sistema y captura de la información

Para la puesta en marcha del sistema, se han realizado algunos cambios con respecto a la metodología, por problemas que se han producido con MySQL, que serán detallados más adelante. Por lo que se ha cambiado de software usando en este caso PostgreSQL.

En el esquema que podremos observar a continuación, se muestra un ejemplo de la conexión de las placas Arduino donde irán conectados los distintos sensores y actuadores. También vemos el servidor local, Geoserver, la base de datos de Postgresql y el visor web. (Figura 28)

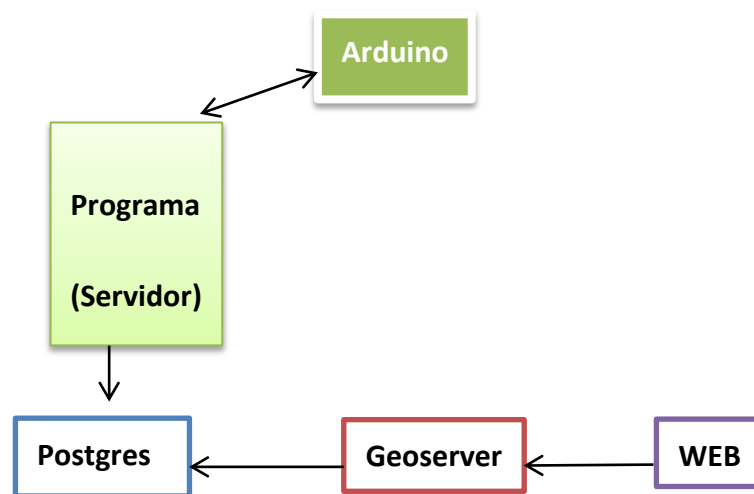


Figura 30: Esquema de conexión final

Como se puede apreciar en el esquema general de conexiones, el elemento central es el programa de Arduino que actúa de servidor web y servidor JSON, enviando una respuesta http. El servidor de PostgreSQL y Apache ejecuta periódicamente una conexión con los Arduinos, recaba la información y la almacena en

la base de datos de PostgreSQL. Asimismo, en Geoserver se podrá también ver estos datos al estar conectada a ella. Por tanto, en el visor web se realizarán peticiones al servidor web para poder visualizar toda esta información recabada. Inmediatamente se explicarán los pasos a seguir para poder tener todas estas conexiones.

En primer lugar se explicará el conexionado de la placa, el cual se puede ver en la siguiente imagen (Figura 29 y 30).

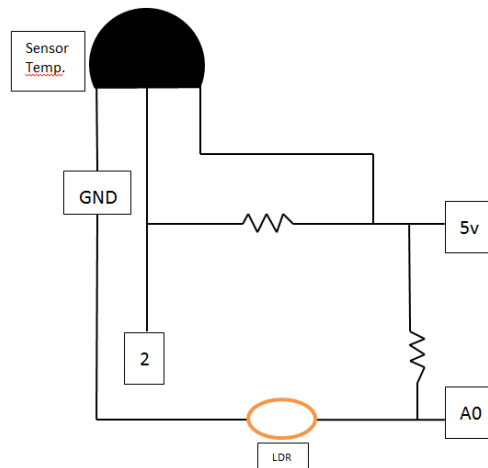


Figura 31: Esquema de conexiones

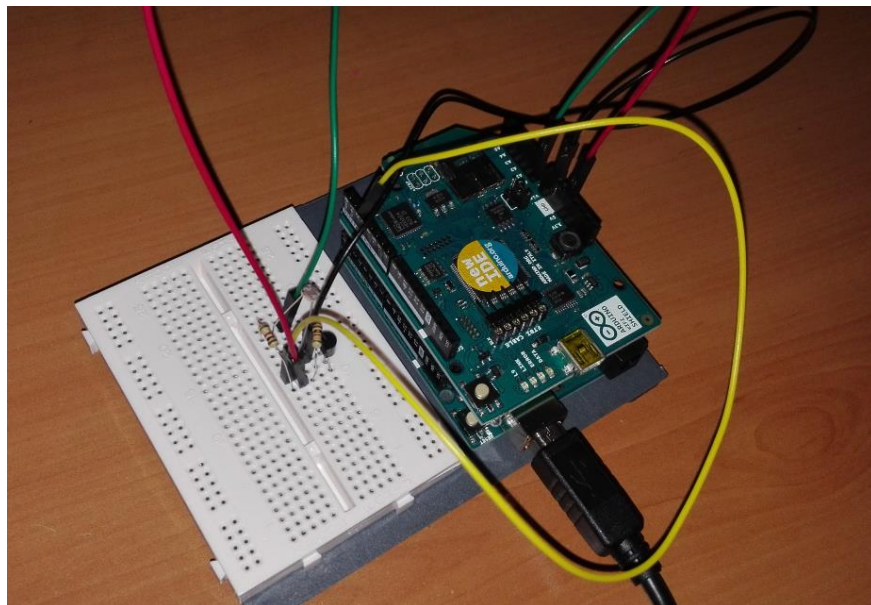


Figura 32: Montaje Arduino y sensores.

Se mostrará a continuación cómo se han conectado los diversos dispositivos para conseguir su funcionamiento.

En cada placa de Arduino se ha conectado un sensor de temperatura conectado a un conector a un pin digital, incluyendo la resistencia.

Como se puede observar el esquema, tiene cuatro cables: uno rojo para el voltaje, otro negro para el GND y uno amarillo para la entrada digital y verde para la entrada analógica.

En este proyecto tomaremos las entradas analógicas A0 de nuestra placa Arduino para leer los valores de la fotorresistencia y el pin digital 2 en el caso de la lectura de temperatura. La resistencia LDR nos marcará unos valores entre 0 y 1023 en función de la luz que capte. El caso del sensor de temperatura se puede recoger datos que oscilen entre -55 °C a 125 °C, pero dada la época en la que se realizan las mediciones, se podrán obtener mediciones entre 10°C de mínima temperatura y 40°C de máxima.

Las placas Arduino van anexas a cada Shield correspondiente, como ya se ha explicado anteriormente. Hay que mencionar que se ha usado el mismo circuito de conexión para todas las placas.

Cuando el sistema está montado se procede a la programación con el IDE de Arduino. En este caso se modifican varios aspectos con respecto a la explicada en la metodología al cambiar de servidor de base de datos como se ha visto en el esquema.

Además habrá dos sketch uno para la conexión con el Shield Ethernet y otra para el Shield WiFi, los cuales serán muy similares además de un código PHP que ha servido de conector con la base de datos en PostgreSQL. Todos los ficheros se podrán encontrar en la carpeta Arduino.

En ambos códigos se va a sustituir las librerías del conector Mysql-Arduino, ya que se tenían problemas por la reducida memoria del microcontrolador, y sobre todo, por las incompatibilidades del servidor con Geoserver que se explicarán en el siguiente punto.

En primer lugar se configuran las direcciones IP necesarias y se declaran las variables globales que en este caso serán las referentes a la identificación del sensor, la temperatura y la luz, asignándole los pines de conexión, que en este caso son el cero para la entrada analógica y el 2 para la digital usando el bus onewire. En este primer paso también se declarará la dirección mac del Arduino.

Después se aplican varias funciones, además de las normalmente usadas con la IDE Arduino (void setup y void loop), también se usarán otras diferentes, como se va a comentar. Se va a explicar brevemente para qué sirven cada una de estas funciones:

- Void setup: Se inicializa la Shield y el sensor digital de temperatura.
- Void loop: Se crea una conexión con el cliente, preparando los sensores para la lectura.
- Void respuestaHTML: si la conexión se ha realizado se envía una respuesta http con los datos de temperatura, iluminación, dirección del sensor y la identificación del mismo.
- Void determinarDirSensorOneWire(): determina la dirección digital del sensor de temperatura.
- void printAddress(DeviceAddress deviceAddress): imprime la dirección del dispositivo en el monitor serial.
- void printAddress(DeviceAddress deviceAddress,EthernetClient client): misma función que la anterior pero en este caso se imprime en el cliente.

En el caso de la conexión con Wifi Shield, se utiliza básicamente la misma estructura, añadiendo, en lugar de la librería de Ethernet, la librería para Wifi (WiFi.h). En el primer paso solo habría que cambiar las direcciones IP por el SSDI y el password de la red Wifi. Sólo se va a explicar las funciones que no aparecen en el caso de Ethernet, ya que son prácticamente iguales y sería repetitivo. La variación más importante es que la dirección mac que se calcula directamente en la función setup.

- void printMacAddress(): imprime la dirección digital del dispositivo.
- void listNetworks(): escanea las redes e imprime una lista con las disponibles.
- void printWifiStatus(): Imprime el estado de la Wifi (SSDI, password, subnet y gateway).

En resumen lo que hacen ambos códigos es leer el valor de los sensores y los datos de conexión, realizar la conexión a la red, bien Wifi o por cable, y enviar esos datos para ser visualizados como una respuesta http (Figura 30) o en el monitor serial (Figura 31). Se pueden visualizar los datos introduciendo la dirección IP asignada a cada dispositivo en un navegador o en el IDE de Arduino ejecutando el monitor serial.

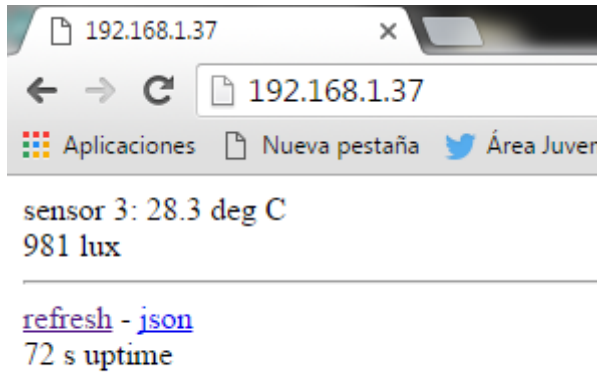


Figura 33: Respuesta http

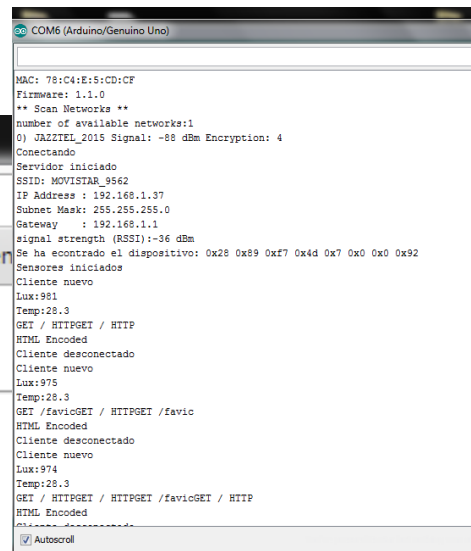


Figura 34: Respuesta monitor serial

Al realizar el proceso de programación de Arduino, conectándolo por medio del USB al ordenador y ejecutando el programa correspondiente, cada placa tendrá su correspondiente identificador, por lo que será lo único que habrá que cambiar para realizar este paso en cada dispositivo.

Llegados a este punto, se está recibiendo datos de los sensores, pero necesitamos que esos datos vayan a parar a nuestra base de datos para poder realizar el visor web que es donde se va a mostrar los resultados obtenidos.

Para poner en marcha el procedimiento, se ha elaborado un código PHP, como antes se ha expuesto, se puede ver en el fichero “Arduino/PHP”. En él se introducen todos los datos necesarios para conectar con la base de datos que se creará en el siguiente punto. Y básicamente la función que realiza leer la respuesta http que realiza cada placa y grabar dicha información, incluyendo la fecha del servidor, en la tabla elegida en PostgreSQL.

Es importante saber que con estos cambios realizados respecto a los pasos a seguir en la metodología con el conector a MySQL, se han producido importantes mejoras, por un lado el problema de la capacidad de memoria que era necesaria para utilizar la mencionada librería, y por otro lado la capacidad de obtener datos en tiempo real, lo cual es muy importante ya que se puede observar desde cualquier lugar los índices de temperatura y luminosidad del edificio.

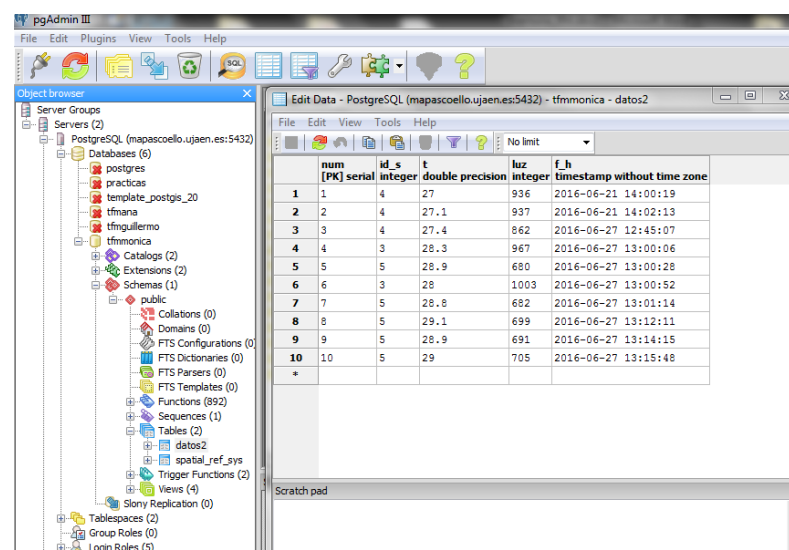
7.6. Puesta en marcha de las bases de datos y el servidor

En este punto se pone en marcha la base de datos para la conexión con Arduino, en la que solo habrá que ejecutar los programas antes vistos, y posteriormente, enlazar los datos recogidos al servidor web, Geoserver.

Hay que mencionar que se han producido cambios con respecto a lo establecido en la metodología, como ya se ha dicho, debido a que al realizar las pruebas con el plugin Mysql para Geoserver, se producían diversos errores. Tal como que Geoserver no reconocía las geometrías al cargar las tablas con las coordenadas de Mysql, además de otros errores que se producían continuamente al crear las capas. Es por ello por lo que se decidió cambiar de software y utilizar PostgreSQL, ya que en Geoserver incluye una conexión directa con el mismo, por lo que no se necesitan plugin adicionales. Y funcionan perfectamente las conexiones con la base de datos.

Una vez cambiado el servidor se siguen los pasos explicados en la metodología. Primero, se crea una nueva conexión que de acceso a la red a la que se conectan los microcontroladores. Para después incluir una base de datos dentro y crear dos tablas dentro de la misma, una de ellas para la recogida de datos (figura 32) y la siguiente contiene las coordenadas X e Y de la posición de los sensores.

Para establecer las tablas se usan las mismas sentencias SQL que se tenían en Mysql, vistas con anterioridad.



The screenshot shows the pgAdmin III interface. On the left, the 'Object browser' displays the database structure for 'PostgreSQL (mapascoello.ujen.es:5432)'. The 'Databases (6)' folder is expanded, showing 'postgres', 'practicas', 'template_postgis_20', 'tfrmonica', and 'tfrmonica'. The 'tfrmonica' database is selected, showing its 'Catalogs (2)', 'Schemas (1)', 'public' schema, and various objects like 'Collations (0)', 'Domains (0)', 'FTS Configurations (0)', 'FTS Dictionaries (0)', 'FTS Parsers (0)', 'FTS Templates (0)', 'Functions (892)', 'Sequences (1)', 'Tables (2)', 'Trigger Functions (2)', and 'Views (4)'. The 'Tables (2)' folder is expanded, showing 'datos2' and 'spatial_ref_sys'. The 'datos2' table is selected, and its data is displayed in the 'Edit Data' window.

	num [PK]	serial	id_s integer	t double precision	luz integer	f_h timestamp without time zone
1	1	4	27	936	2016-06-21 14:00:19	
2	2	4	27.1	937	2016-06-21 14:02:13	
3	3	4	27.4	862	2016-06-27 12:45:07	
4	4	3	28.3	967	2016-06-27 13:00:06	
5	5	5	28.9	680	2016-06-27 13:00:28	
6	6	3	28	1003	2016-06-27 13:00:52	
7	7	5	28.8	682	2016-06-27 13:01:14	
8	8	5	29.1	699	2016-06-27 13:12:11	
9	9	5	28.9	691	2016-06-27 13:14:15	
10	10	5	29	705	2016-06-27 13:15:48	
*						

Figura 35: Ejemplo de tabla con datos reales

En segundo lugar, queremos conectar dichas tablas con el servidor web, para lo que se agrega un nuevo almacén de datos en Geoserver para conectar con Postgis
Figura 33.

The screenshot shows the 'New Data Store' configuration page in the Geoserver web interface. The left sidebar contains a navigation menu with categories like 'Almacenes de datos', 'Servicios', 'Settings', 'Cacheado de Teselas', and 'Seguridad'. The main content area is titled 'Nombre del origen de datos *' and includes fields for 'Descripción', 'Habilitado' (checked), 'dbtype' (set to 'postgis'), 'host' (set to 'mapas.coello.ujain.es'), 'port' (set to '5432'), 'database' (set to 'tfmmonica'), 'schema' (set to 'public'), 'user' (set to 'tfmmonica'), and 'passwd' (masked with dots).

Figura 36: Conexión Geoserver con PostgreSQL

También se necesitan las capas para poder visualizar los datos. Se tendrán cuatro capas diferentes dos de ellas serán de la planta del edificio, el formato de las mismas es shp. Para poder obtener este formato se importan los archivos con las plantas en dxf al programa Qgis, y desde aquí se graban en el formato deseado que será incluido en Geoserver. Es importante destacar que una de ellas contiene la información del escaneo realizado, y la otra es el plano de arquitectura con toda la planta completa, que sirve de comparación con el plano del escaneo. Las otras dos capas restantes se refieren a la información recogida por cada sensor, habrá una correspondiente a los valores de temperatura y otra a los de luminosidad. Se han creado a partir de una sentencia sql (figura 34) en la que se llaman a los datos contenidos en las tablas creadas en Postgres.

The screenshot shows the 'Edit SQL View' configuration page in the Geoserver web interface. The left sidebar contains a navigation menu with categories like 'Servidor', 'Datos', 'Servicios', 'Settings', 'Cacheado de Teselas', and 'Seguridad'. The main content area is titled 'Editar vista SQL' and includes a text area for the 'Sentencia SQL' (SQL statement) and a table for 'Parámetros de la vista SQL' (SQL View Parameters).

Sentencia SQL

```
select coordenadas.id_s, geom, tm, lm
from coordenadas,
(select AVG(t) as tm, AVG(luz) as lm, id_s
from datos2
where datos2.f_h >= '%final%' and datos2.f_h
<= '%ffinal%')
group by id_s) as tagr
where coordenadas.id_s = tagr.id_s
```

Parámetros de la vista SQL

Nombre	Valor por defecto	Validar la expresión regular
final	2015-11-14 14:18:11	
ffinal	2015-11-14 13:57:11	

Figura 37: SQL que crea las capas de los sensores

Como se puede observar esta vista SQL lo que se pretende es crear los valores medios de temperatura e iluminación, uniendo con el identificador las tablas con las coordenadas y los datos recogidos. Además de agregar los parámetros de la fecha y hora inicial y final. Esto sirve para que posteriormente en la web se puedan elegir la fecha y hora deseada y obtener el valor medio de cada sensor.

Una vez configuradas ambas capas se le da un estilo SLD. Entre las opciones que se recogían en la metodología se ha preferido usar un mismo símbolo, un rombo, dividido en dos triángulos para diferenciarlos. Se ha creído que esta era la mejor opción porque se ve de manera más clara a qué símbolo corresponde cada sensor. Ha sido un símbolo en el que se podía apreciar el color a simple vista, ya que en las pruebas se habían realizado se ha certificado que el uso de otros símbolos con muchas líneas, tipo torre de control, no se diferenciaba cada rango de valores de manera rápida y clara. Y al tener dos símbolos en unas mismas coordenadas tampoco era óptimo el uso de otras opciones como el cambio de tamaño en función de su valor.

El triángulo que representa los valores de luminosidad se le ha incorporado una escala de grises desde el blanco, valor más iluminado hasta el negro, que representa la oscuridad. El fichero SLD de esta capa se puede ver en “anexos\geoserver\estilos\iluminacion”.

Los valores de temperatura se representan con un degradado desde azul hasta el rojo, siendo este último el que más grados centígrados tiene 40°C, y el azul corresponde al indicador que menos temperatura posee, en este caso 10°C. Aclarar que se han elegido estos valores debido a que al estar en verano es difícil que se alcancen temperaturas fuera de este rango en el interior de un edificio. Se puede ver el estilo SLD escrito en el “anexos\geoserver\estilos\temperatura”.

El resultado obtenido después de todo este proceso es el que se puede observar en la Figura 35.

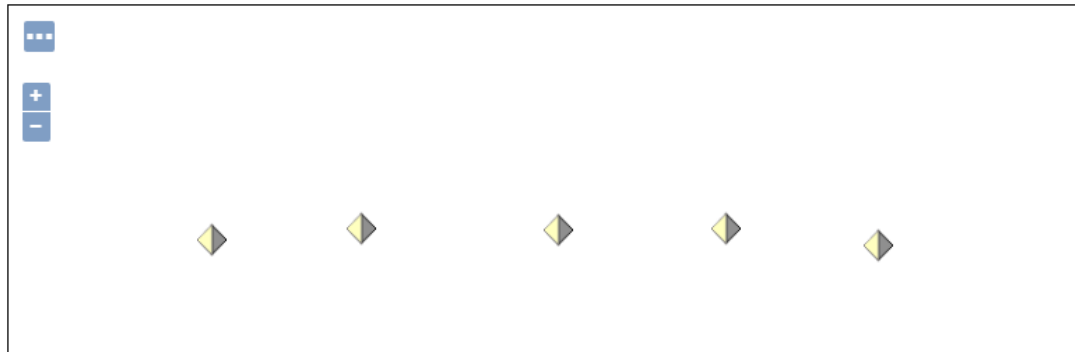


Figura 38: Resultado diseño de los sensores

Ya se tienen todos los requerimientos para poder visualizar a través de la web todas estas capas con los valores de los sensores.

7.7. Análisis y visualización a través de un cliente ligero

Al llegar a este punto, están instaladas las placas Arduino con sus sensores y todas las conexiones de base de datos y los trabajos del servidor web a punto.

Para la parte visual, se ha creado un sencillo interfaz basado en entorno web que permita visualizar los elementos desde un navegador.

Aunque exista un hardware que realice las tareas de monitorización, es necesario que exista un interfaz que permita al usuario manejar la información forma sencilla. El fichero html del mismo se puede consultar en “web”.

Ahora, se describirá como se ha diseñado la página web, así como las funciones que realiza y la justificación de las mismas.

La página cuenta con una pantalla de inicio que es la más importante, donde se puede visualizar y realizar un análisis de todos los datos obtenidos.

En la cabecera se tiene una imagen, y en la primera línea, están los enlaces que llevan a cada página: Inicio, Modelo 3D y Contacto. Pinchando en cada una de ellas se dirigirá a la página siguiente. Esta cabecera será común para cada pantalla.

Siguiendo con la página inicial, se encuentra el cuadro en el que se aloja el plano de la planta, tanto el capturado con el escáner láser 3D, como el de la planta

completa, se diferenciarán porque el primero de ellos tiene un color gris más oscuro que el anterior. Además, se incluirán las capas correspondientes a cada sensor. Esto se ha hecho cargando las capas que anteriormente introducimos en Geoserver.(Figura 36)

Para lo anterior, en JavaScript se ha creado un grupo guardado en la variable “map”, con estas capas wms realizando la llamada a la url de Geoserver donde se alojan los datos. El grupo tiene una capa base que es la tomada con el láser escáner. Las siguientes son subcapas que pueden activarse o desactivarse, en función de las necesidades, gracias al control que proporciona la función layerSwitcher. Como se puede comprobar en la siguiente (Figura 37).

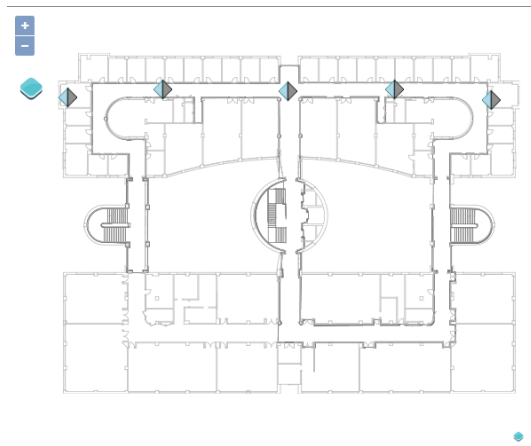


Figura 39: Visualización de capas en la web

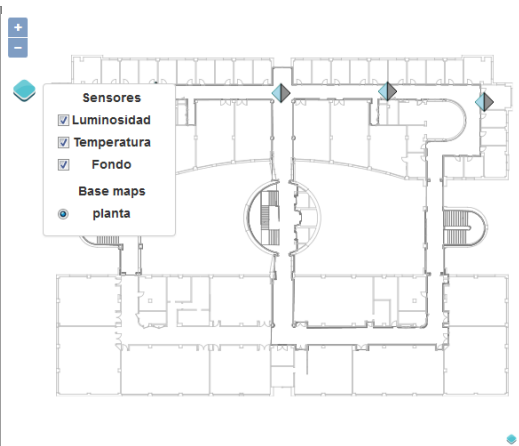


Figura 40: Control de capas

Se ha incluido una leyenda, para entender las variaciones de color de los sensores. El caso del sensor de temperatura se podrá obtener mediciones entre 10°C de mínima temperatura y 40 de máxima, como ya se ha explicado. Creando los rangos cada 6 grados centígrados, asociando a cada rango un color correspondiente desde el rojo que sería el más caliente, al azul que sería los valores más fríos. En la columna donde se analiza la luminosidad la intensidad de luz puede tomar valores de 0 a 1024, los valores se han dividido en una escala de grises donde el índice más alto es la oscuridad total, representado en negro y el más bajo muy luminoso teniendo un color blanco. Lo anterior se ha creado en formato html incluyendo las imágenes de cada triangulo con su valor asignado. Se puede observar el resultado en la figura 38.

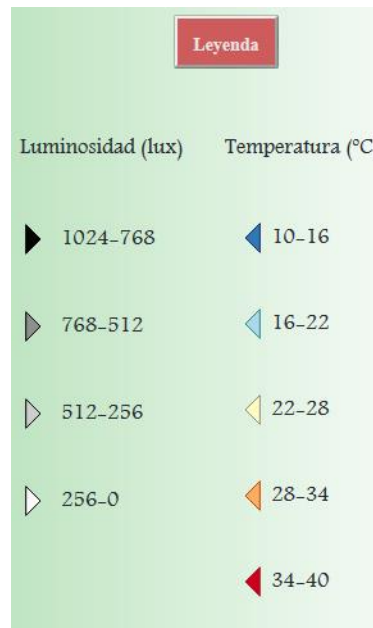


Figura 41: Leyenda

Otra de las aplicaciones de la web es poder elegir entre la fecha y hora que se desea ver la información. Se ha creado para ello un formulario que contiene la fecha de inicio y la final con un botón para enviar la información. Pulsando este se volverán a crear las capas de los sensores con los datos actualizados de fecha y hora, gracias a la función “crearsource”. Por lo que podrá variar el color de cada capa si el rango de valores de estas han cambiado.

Fecha inicio:
2016-05-23 01:00:00

Fecha fin:
2016-05-24 01:00:00

Enviar

Figura 42: Formularios fecha y hora

Otra función implementada es la llamada “GetFeatureInfo”, en la que pinchando sobre el sensor se puede obtener los valores medios tanto del sensor de temperatura, como el de luminosidad, en la fecha elegida, así como el identificador del mismo y la fecha inicial.

```
geometry:[object Object], id_s:5,  
tm:20.083333333333332, lm:554.5,  
f_h:2016-05-23 01:15:00
```

Figura 43: Ejemplo de valores obtenidos al pinchar sobre el sensor

También se puede realizar un análisis visual de los datos usando los botones incorporados que se pueden observar en la siguiente Figura 41.



Figura 44: Botones para activar mapa interactivo

Pulsando el primero de ellos se podrá ver en el cuadro del mapa una animación cambiando el valor medio de los sensores cada hora. Por lo que se pueden observar a simple vista las variaciones de temperatura o luz producidas a lo largo de las horas.

Pulsando los siguientes botones las acciones que se realizan son parecidas, pero en lugar de hora en hora, varían de día en día, comprendiendo este las horas de sol, de noche en noche, o de semana en semana. También, como en el caso anterior, se pueden apreciar visualmente la variación en estos tramos de tiempo.

Todo ello se ha creado con las funciones de JavaScript “horas”, “día”, “noche” y “semana” para lo que se ha necesitado también la función “addhoras”, que consiste en devolver una fecha sumándole un número determinado de horas. Al pulsar cada botón creado en HTML se llama a la función correspondiente ejecutándose cada cinco segundos, por ejemplo, en dos minutos podremos ver la variación de los sensores cada hora a lo largo de todo un día completo.

En el caso de los botones tabla y gráfica, su misión es crear una tabla con un determinado número de datos grabados en un array al pulsar sobre los sensores.

Sucede lo mismo con la gráfica estos mismos arrays son los que se usan la hora de hacer la representación. Las funciones usadas para ello son “addtable” y “gráfica”.

Fecha	Temp.	Luz
2016-05-22T23:00:00Z	20	50
2016-05-23T01:00:00Z	21	50
2016-05-23T03:00:00Z	21	50
2016-05-23T05:00:00Z	21	280
2016-05-23T07:00:00Z	24	800
2016-05-23T09:00:00Z	24	960
2016-05-23T11:00:00Z	20	960
2016-05-23T13:00:00Z	23	700

Figura 45: Tabla web

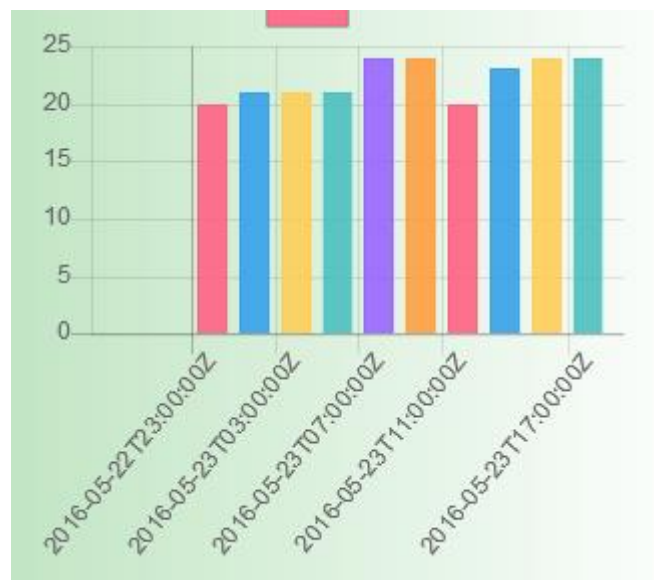


Figura 46: Gráfica de datos recabados

En la siguiente página de modelado 3D, se ha incluido el resultado final del modelo realizado en Revit de parte de la tercera planta del edificio. Esto ha sido posible al uso de una plataforma llamada A360 en la que se pueden subir proyectos de Revit directamente a la nube y visualizarlos desde cualquier dispositivo. [24] Generando una dirección donde aparecen los datos del proyecto, por lo que esta ha sido incluida en la página de modelado para poder ver el modelo 3D, Figura 42. Se puede consultar el código html en [modelado/modelo3d.html](#).



Figura 47: Modelo 3D incluido en la web

Es necesaria para la presentación de la página web tener un estilo, para poder editar, mover, rotar, cambiar, colores, formato de textos, etc. Para realizarlo se crea una página de estilo, asociada a cada elemento de la web en la que se le da forma para hacerla lo más atractiva posible. Este fichero se llama contacto.html.

Con todos los resultados obtenidos incluidos en la web, se puede realizar un pequeño análisis visual de los datos. Por ejemplo, se van a observar los valores que se obtienen en las diferentes horas del día para poder entender para qué serviría.

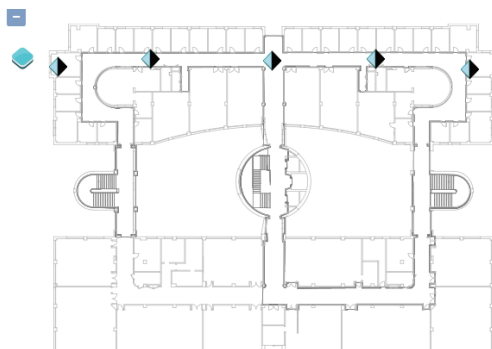


Figura 48: Estado sensores 1:00h-2:00h

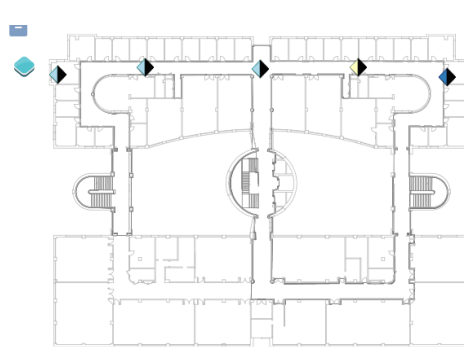


Figura 49: Estado sensores 5:00h-6:00h

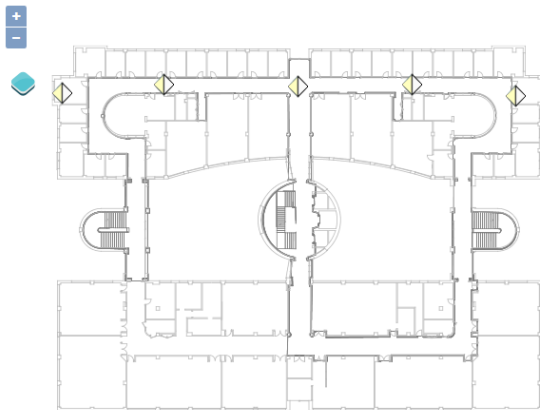


Figura 50: Estado sensores 11:00h-12:00h

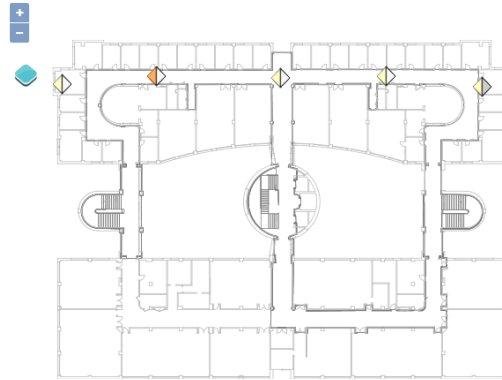


Figura 51: Estado sensores 16:00h-17:00h

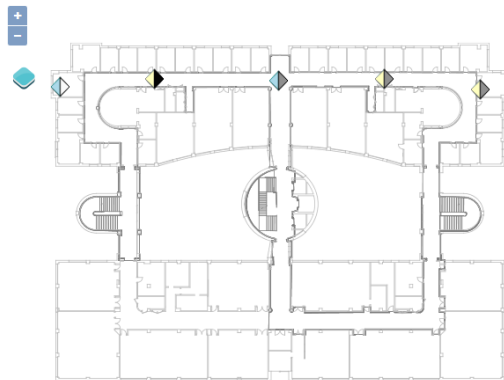


Figura 52: Estado sensores 21:00h-22:00h

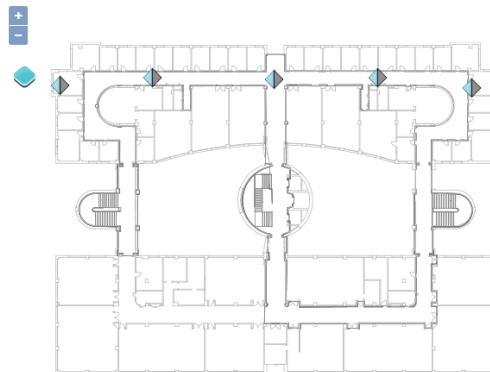


Figura 53: Estado sensores 23:00h-1:00h

Con estos datos visuales se puede actuar en función de los resultados obtenidos. Se observa el evidente cambio de temperatura del día a la noche teniendo durante el día unos valores de temperatura más cálidos que en la noche. Vemos también que durante el medio día la intensidad lumínica recogida es muy alta por lo que los sensores aparecen en blanco.

En el caso de las 16:00h-17:00h horas observamos que hay un punto donde la temperatura supera al resto. Esto puede ser debido a que se haya estropeado el climatizador y no esté realizando correctamente su función. Por lo que se podría tomar acciones sobre ello. Ya que esto influye en que las demás máquinas deberán actuar

con más potencia debido a que en esa zona la temperatura no está dentro de los valores de confort.

En el caso de las 21:00-22:00h, en el que se observa que hay una luz encendida ya que la intensidad es superior a la del resto, puede ser debido a que se haya olvidado apagar esta luz, ya que el edificio está prácticamente desalojado a partir de estas horas. También se puede avisar para que no vuelva a ocurrir. Y con la colaboración de todo el personal se pueda mejorar la eficiencia en este edificio.

8. RESULTADOS

Los resultados obtenidos son varios en función de las partes. En primer lugar la prueba de que el sistema funciona y está en marcha, pudiéndose capturar los datos ambientales y de confort. Los datos recogidos, un ejemplo de los mismos se muestra en la siguiente Figura 49.

	num [PK] serial	id_s integer	t double precision	luz integer	f_h timestamp without time zone
1	1	4	27	936	2016-06-21 14:00:19
2	2	4	27.1	937	2016-06-21 14:02:13
3	3	4	27.4	862	2016-06-27 12:45:07
4	4	3	28.3	967	2016-06-27 13:00:06
5	5	5	28.9	680	2016-06-27 13:00:28
6	6	3	28	1003	2016-06-27 13:00:52
7	7	5	28.8	682	2016-06-27 13:01:14
8	8	5	29.1	699	2016-06-27 13:12:11
9	9	5	28.9	691	2016-06-27 13:14:15
10	10	5	29	705	2016-06-27 13:15:48
11	11	3	27.3	1014	2016-06-28 09:37:04

Figura 54: Datos recogidos

Otro de los productos es el modelo 3D finalizado con todos los detalles elaborados en Revit. El cual se va a visualizar en la web, como se podrá apreciar más tarde.

Por último, se logra crear un cliente web, con todas las funcionalidades que se han descrito en el apartado anterior. Y que permite realizar un análisis visual de los datos con el que se pueden tomar futuras decisiones a la hora de mejorar la eficiencia del edificio. Por ejemplo, si se ha quedado una luz o el aire acondicionado encendido en un edificio en el que no hay nadie de noche, como es el caso, pues se puede

proceder a llamar la atención a la persona que tuvo este fallo, para que no se vuelva a cometer.

El diseño del visor web ha resultado de la siguiente manera:

❖ Pantalla Inicio.

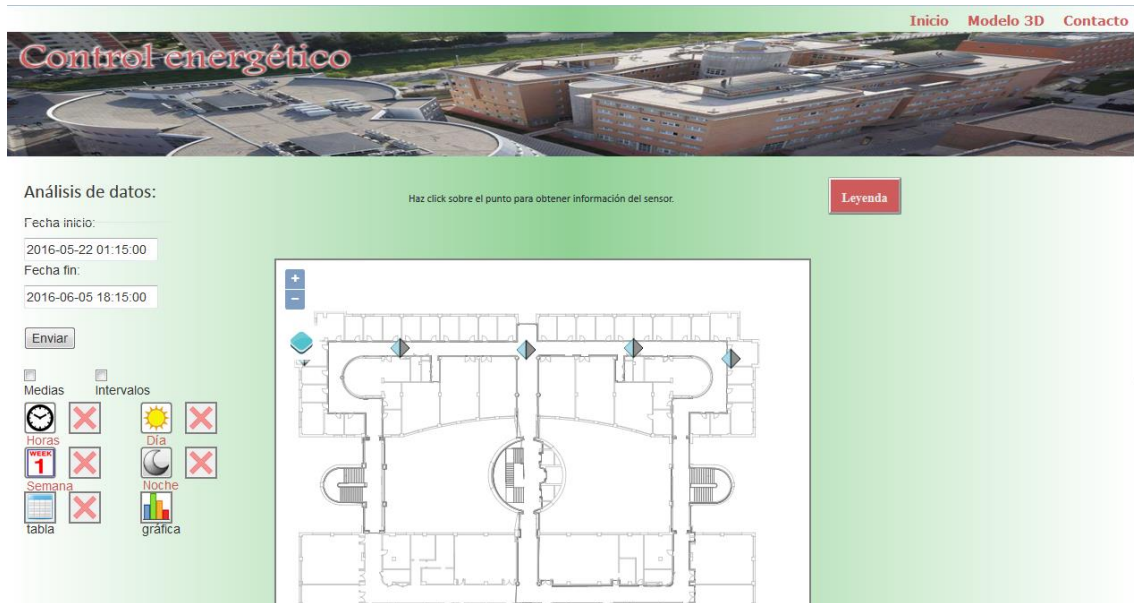


Figura 55: Pantalla Inicio de la web

❖ Pantalla modelo 3D

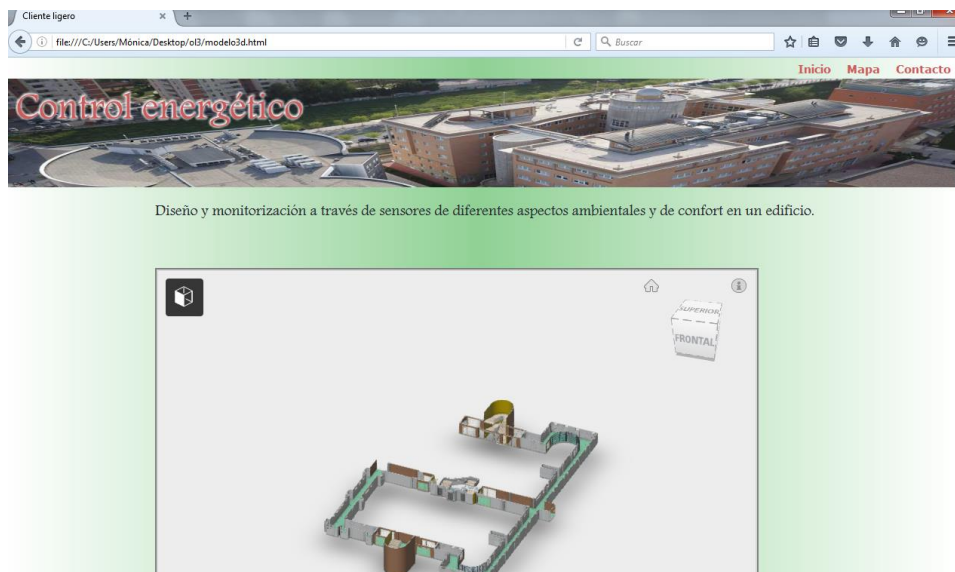


Figura 56: Pantalla Modelo 3D

9. CONCLUSIONES

Como ya se sabe, el proyecto se puede dividir en tres partes: el diseño y monitorización de la red, el levantamiento del edificio para la creación de un modelo tridimensional y la elaboración de un visor web. Por lo que se va a analizar cada parte para sacar las conclusiones finales.

En relación al diseño y monitorización, se ha cumplido con el objetivo principal del proyecto que es implementar un prototipo referente a cómo instalar una red con diferentes sensores y que esa red esté en funcionamiento. Es decir, se han podido capturar los datos de temperatura y luminosidad mediante la placa Arduino. Al comprobar que este método funciona es posible instalar este tipo de red en cualquier edificio existente.

También se ha elaborado el modelo 3D de la sección de edificio que se monitorizó, gracias al uso del láser escáner y del software Revit, se ha podido trabajar con la tecnología BIM. Este modelo también puede ser utilizado para futuros análisis energéticos, de iluminación, etc.

Respecto al visor web también se ha generado exitosamente, pudiendo visualizar la información de los sensores y realizar un análisis visual del mismo.

Por lo que se puede determinar, que a pesar de las limitaciones y contratiempos que se han producido a lo largo de la puesta en marcha del proyecto, que se comentarán a continuación, se ha cumplido con los objetivos ejecutando las diferentes partes de este TFM.

Inmediatamente se analizarán los diferentes inconvenientes que se han producido, y las conclusiones sacadas en relación a los mismos.

Al realizar las pruebas para montar la red se han dado algunos problemas como se ha comentado. Se ha llegado a la conclusión que el uso de la librería “conector mysql-arduino” no es recomendable implementarla en placas que tengan poca memoria, como es el caso del dispositivo usado Arduino UNO, ya que genera errores. Sería ideal poder haber usado otro modelo de microcontrolador con más capacidad, por ejemplo, Arduino Mega.

Por otro lado, tampoco es aconsejable el uso del plugin de Mysql para Geoserver. Al menos en la versión usada, requiere de muchas mejoras que no generen tantos problemas. Tales como el no reconocimiento de las geometrías, errores al calcular las coordenadas, etc.

La parte positiva, es que al no usar Mysql, se ha generado un prototipo que recoge los datos en tiempo real, mejorando el inicialmente planteado.

Si se dispusiese de un presupuesto más elevado sería conveniente el uso de baterías más potentes para las placas que utilicen Shield Wifi, ya que las usadas había que recargarlas cada día y la duración del proceso de carga era muy elevado, por lo que restaba horas en las que no se han podido obtener datos reales. En relación al tema de presupuesto, hubiera sido más adecuado contar con más dispositivos de medición para haber obtenido más datos y poder realizar un mejor análisis con datos reales.

La plataforma Arduino como tecnología para el monitoreo climático, ofrece resultados confiables. Sería recomendable la evaluación de más variables ambientales sobre esta placa con mirar a mejorar la escalabilidad del prototipo utilizado.

Este trabajo puede servir para futuras líneas de investigación y desarrollo en este campo. Con los datos que se obtengan, si se prolongaran las mediciones en el tiempo, se podría realizar un estudio exhaustivo del uso de la energía consumida y poder hacer un edificio más eficiente, energéticamente hablando. Que es uno de los motivos por lo que se ha realizado este trabajo fin de Máster. También se podrían ampliar estas mediciones abarcando toda la planta o edificio para una mayor eficiencia.

Para la mejora del sistema se ha propuesto la conexión al sistema eléctrico para no depender de baterías que limitan la toma de datos.

Así como, en un futuro, diseñar procesos de automatización en base a la información que se obtenga.

La conclusión final de este Trabajo Fin de Máster es que se han cumplido con los objetivos propuestos en este trabajo. Y además, con la finalidad del Máster en Tecnologías Geoespaciales para la Gestión Inteligente del Territorio, que consistía en acercar la geomática a otras ramas. En este caso se ha cumplido, ya que se ha usado la electrónica y la visualización de datos espaciales para un mismo fin.

Referencias:

- [1] Agencia Andaluza de energía. (2010). Ahorro y eficiencia energética. Consultada el 22/04/2016, en <https://www.agenciaandaluzadelaenergia.es/empresas/ahorro/edificios>
- [2] Wikipedia. (2008). Modelado de la información de construcción. Consultada el 22/04/2016, en [https://es.wikipedia.org/wiki/Modelado de informaci3n de construcci3n](https://es.wikipedia.org/wiki/Modelado_de_informaci3n_de_construcci3n)
- [3] Wikipedia. (2008). Redes de sensores. Consultada el 22/04/2016, en [https://es.wikipedia.org/wiki/Red de sensores 2016](https://es.wikipedia.org/wiki/Red_de_sensores_2016)
- [4] Carlos Cámara. (2009). Del cad al BIM. Consultada el 23/04/2016, en carloscamara.es/blog/2009/01/08/del-cad-al-bim
- [5] Participa en Andalucía. (2012). Consultada el 23/04/2016, en <http://www.participaenandalucia.net/index.html> 2016
- [6] Google maps.(2014). Consultada el 04/05/2016, en <https://www.google.es/maps>
- [7] Iberpix. Consultada el 04/05/2016, en www.ign.es/iberpix2/visor/
- [8] Leica Geosystems AG. Leica ScanStation C10. Consultada el 30/05/2016, en [http://hds.leica-geosystems.com/downloads123/hds/hds/ScanStation%20C10/brochures-datasheet/Leica ScanStation C10 DS es.pdf](http://hds.leica-geosystems.com/downloads123/hds/hds/ScanStation%20C10/brochures-datasheet/Leica_ScanStation_C10_DS_es.pdf)
- [9] Gus C. (2013). Monitor de temperatura y luz. Consultada el 10/06/2016, en <http://www.klinamen.net/index.php/proyectos/arduino/30-monitor-de-temperatura-y-luz>
- [10] Página oficial de Arduino. Arduino+Ethernet Shield. Consultada el 10/06/2016, en <https://www.arduino.cc/en/Main/ArduinoEthernetShield>
- [11] Página oficial de Arduino. Arduino+Wifi Shield. Consultada el 10/06/2016, en <https://www.arduino.cc/en/Main/ArduinoWifiShield>

[12] Margaret Rouse. Definición MySQL Consultada el 17/06/2016, en searchdatacenter.techtarget.com/es/definicion/MySQL

[13] Osgeo. Geoserver. Consultada el 10/06/2016, en live.osgeo.org/es/overview/geoserver_overview.html

[14] Wikipedia. (2016). Revit. Consultada el 18/06/2016, en <https://es.wikipedia.org/wiki/Revit>

[15] Maptek Pty Ltd. (2016). I-Site. Consultada el 18/06/2016, en www.maptek.com › Cl › Productos › I-Site

[16] Página oficial de Arduino. (2015). Consultada el 23/06/2016, en <https://www.arduino.cc>

[17] Elbert Green Hubbard. (2013). CSS. Consultada el 25/06/2016, en <http://aprenderaprogramar.es>

[18] Andrés Duarte. Arduino y Xbee. Consultada el 25/06/2016, en <http://www.andresduarte.com/arduino-y-xbee>

[19] Diymaker. (2014). ARDUINO + BLUETOOTH. Consultada el 25/06/2016, en <http://diymakers.es/arduino-bluetooth/>

[20] Autor: desconocido. Levantar geométricamente los edificios existentes. Consultada el 25/06/2016, en http://www.rehabimed.net/Publicacions/Metode_Rehabimed/II.%20Rehabilitacio_Ledifici/ES/2a%20Parte.%20Herramienta%204.pdf

[21] Antonio Almagro Gorbea y Gerardo López Hernández . Sistemas topográficos para mediciones en arquitectura. Consultada el 25/06/2016, en <http://digital.csic.es/bitstream/10261/29543/1/ManualCALCO.pdf>

[22] Electronica embajadores. (2015). Módulo arduino wireless sd SHIELD. Consultada el 27/06/2016, en <http://www.electronicaembajadores.com/Productos/Detalle/-1/LCA1SCC/modulo-arduino-wireless-sd-shield 10>

[23] Autor: desconocido. (2014). Diferencia entre opciones RF. Consultada el 27/06/2016, en <http://arduino.stackexchange.com/questions/3270/whats-the-difference-between-rf-options-wifi-xbee-nrf24l01> 15

[24] Autodesk (2015). Características A360. Consultada el 22/06/2016, en <https://a360.autodesk.com/features/index.html>

[25] W3 Consortium. (2016). ¿Qué es html? Consultada el 28/06/2016, en <https://www.w3.org/html/>

[26] Ardafruit. (2016). Sensor. Consultada el 01/07/2016, en <https://www.adafruit.com/category/35>