



|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Estudio de las plataformas software existentes para la Internet de las cosas

|   |    |
|---|----|
| Objetivo .....  | 2  |
| Introducción.....   | 2  |
| El origen de la IoT.....  | 2  |
| ¿Qué es y qué nos ofrece exactamente el Internet de las Cosas?..... | 3  |
| ¿Por qué ahora? .....   | 3  |
| Progresos de la conectividad inalámbrica.....                       | 5  |
| Protocolos y estándares.....  | 14 |
| La IoT y el Big Data .....  | 15 |
| El futuro del Marketing.....  | 15 |
| Privacidad y seguridad en la IoT.....                               | 16 |
| Retos de la IoT.....  | 17 |
| Estudio plataformas IoT.....  | 19 |
| Plataformas hardware .....  | 19 |
| Microcontroladores .....  | 19 |
| Microprocesadores .....   | 26 |
| Plataformas Software.....   | 30 |

## Diseño y desarrollo prototipo IoT

|   |    |
|---|----|
| Descripción del prototipo.....                          | 45 |
| Componentes utilizados y costes. ....                   | 45 |
| Hardware .....  | 45 |
| Descripción componentes.....                            | 46 |
| Software.....   | 48 |
| Desarrollo del prototipo.....                           | 48 |
| Fase 1: Montaje del circuito y código para Arduino..... | 48 |
| Fase 2: Creación y estructura de la base de datos. .... | 50 |
| Fase 3: Página web y creación de gráficas .....         | 51 |
| Detalle de la página web .....                          | 52 |
| Instalación de archivos y código fuente. ....           | 54 |
| Carga de código en la placa Arduino. ....               | 54 |
| Instalación aplicación web.....                         | 55 |
| Conclusiones.....                                       | 56 |
| Bibliografía.....                                       | 57 |
| Apéndice.....   | 58 |

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Objetivo

El estudio realizado en este proyecto tiene como finalidad presentar el concepto de IoT, así como la de dar a conocer las plataformas software y hardware más significativas que existen en la actualidad.

Al mismo tiempo, se muestra cómo esta nueva tecnología queda lejos de presentarse como una tendencia para el futuro, puesto que actualmente es ya una realidad, donde las grandes corporaciones mundiales del sector TIC la perciben como un objetivo estratégico efectuando notables inversiones con el fin de lanzar los dispositivos y servicios más novedosos al mercado.

Asimismo, observaremos como esta nueva tendencia puede transformar la sociedad y nuestra forma de interactuar con ella.

Para finalizar, se diseñará un caso práctico basado en la plataforma Arduino, donde se pondrá en práctica alguna de las funcionalidades expuestas en el estudio.

## Introducción.

La humanidad está en constante progresión y uno de los pilares que más ha contribuido a ello ha sido la tecnología.

A lo largo de la historia las innovaciones tecnológicas han provocado cambios cualitativos de gran envergadura en la estructura socioeconómica de los países, y es quizás, a partir de la finalización de la segunda guerra mundial, cuando se origina una evolución tecnológica que se ha visto incrementada exponencialmente hasta nuestros días.

Esta evolución, se está forjando en todos los campos (medicina, transporte, energía), pero es sobre todo en el campo de las telecomunicaciones donde durante los últimos años estamos viviendo los mayores cambios. En este sentido, hace relativamente poco tiempo que la aparición de internet y la telefonía móvil han supuesto una revolución en nuestra forma de comunicarnos.

Antes del nacimiento del móvil, cuando queríamos conversar con alguien llamábamos a un lugar, con la llegada del móvil pasamos directamente a llamar al individuo con el que queremos comunicar, pero la tecnología ha continuado evolucionado y gracias a internet y a las redes sociales ahora es posible ponernos en contacto con varias personas a la vez de y desde cualquier parte del mundo.

Desde hace pocos años se ha dado un paso más, que consiste en la comunicación de las personas con objetos de la vida cotidiana y es aquí donde surge el concepto de la internet de las Cosas (IoT o Internet of the Things en inglés).

## El origen de la IoT.

Puede parecer lo contrario, pero la idea de internet de las cosas no es un término nuevo. El primer dispositivo de Internet, fue una máquina de Coca-Cola en la Universidad Carnegie Melon a principios de 1980, a la que los programadores podían conectarse a través de internet y comprobar el estado de la máquina y verificar si había o no había una bebida fría antes de decidirse a hacer el viaje a la máquina.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Posteriormente, en 1990, Simon Hackett y John Romkey crearon una tostadora con conexión a Internet por TCP / IP, y que controlada remotamente por el protocolo (SNMP MIB) permitía a través de cualquier ordenador encenderla o apagarla así como determinar el tiempo de tostado.

El término IoT, fue introducido inicialmente por el ingeniero **Bill Joy** en el año 1999 cuando realizó un estudio sobre el potencial de aplicaciones resultantes de la comunicación establecida entre dos dispositivos conectados mediante Internet, permitiendo automatizar y controlar millones de procesos y actividades que llevamos a cabo diariamente.

Una década después, el 12 de julio de 2009, un investigador británico, **Kevin Ashton**, que por aquellos años trabajaba en el Massachusetts Institute of Technology (MIT) como cofundador y director ejecutivo del Centro de Auto-ID desarrollando unos sistemas de sensores e identificadores de radio frecuencia (RFID), quién empleó por primera vez el nombre de internet de las cosas en un artículo publicado en el RFID Journal.

En dicho artículo introducía el concepto de conectar todas las cosas que nos rodean con la finalidad de poder contarlas, saber su posición o su estado en cualquier momento así como aportarnos información sobre el entorno que les rodea.

Finalmente, en noviembre de 2012, **Sanjay Sarma** anunció el lanzamiento de la iniciativa "Nube de las cosas" en el MIT, en la que el campo de las investigaciones basadas en la RFID se amplió para integrar la informática en nube y datos voluminosos.

## ¿Qué es y qué nos ofrece exactamente el Internet de las Cosas?

Una vez que ya conocemos los orígenes de la IoT, el siguiente paso es definirlo y presentar sus servicios.

Aunque carecemos de una definición estandarizada para este término, podríamos decir que se trata de una red global de información y comunicación en donde todos los objetos que nos rodean se encuentra identificados y conectados permanentemente a Internet y que gracias a la fuente de datos que proporcionan sensores y actuadores embebidos, permiten la captura, almacenamiento y gestión de toda la información emitida por dichos objetos con el propósito de automatizar actividades y procesos diarios en nuestra vida cotidiana, así como la de analizar toda los datos generados aportándonos información útil que nos ayude a la correcta toma de decisión frente las situaciones que se nos presentan diariamente.

Es significativo destacar que la IoT no sólo radica en conectar objetos entre ellos y manejarlos desde un dispositivo remoto, sino que **la información precisa, automatizada y en tiempo real** es una característica clave de las aplicaciones de la IoT, por lo tanto, la interpretación de estos datos desde el mundo físico probablemente dará lugar a la incursión de diversos servicios de nuevos negocios que pueden ofrecer importantes beneficios económicos y sociales.

## ¿Por qué ahora?

No existe un único factor que por sí mismo haya originado este interés por la IoT, más bien deberíamos señalar que ha sido la convergencia de muchas tecnologías la que ha logrado encaminar este sector.

Varios de estos factores son:

- **Smartphones y tablets.**

Son uno de los principales impulsores de la IoT, hay que señalar como la adopción masiva por parte de la sociedad

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

de estos aparatos han supuesto una verdadera revolución, permitiendo estar conectados permanentemente y poder interactuar con dispositivos desde cualquier lugar permitiendo ofrecer nuevos modelos de negocio.

- **Cloud computing**

La computación en la nube, es otro elemento destacado, siendo este nuevo modelo de prestación de servicios de negocio y tecnología determinante para poder consolidar la interconexión entre dispositivos.

Ya existen en el mercado empresas ofreciendo plataformas específicas para IoT como puede ser Carriots o Xively.

- **Disminución del consumo de energía, tamaño y coste de sensores y dispositivos**

El progreso evolutivo de la tecnología ha supuesto una disminución de consumo de energía, tamaño y coste de sensores y dispositivos, dando como resultado la aparición de nuevas plataformas hardware destinadas tanto al sector profesional como para aficionados interesados en crear entornos u objetos interactivos.

Ejemplos de estas plataformas tenemos a Arduino, Raspberry Pi o Mbed.

- **Protocolo IPv6**

En la IoT por concepto, las cosas deben comportarse como nodos de internet, es decir cada objeto debe poseer una dirección ip única con el fin de poder ser identificados. Según estimaciones del Internet Business Solutions Group de Cisco, unos 25.000 millones de dispositivos estarán conectados a Internet en 2015, y 50.000 millones en 2020.

Lógicamente el principal inconveniente para alcanzar estas dimensiones es la progresiva merma de la cantidad de direcciones IPv4 disponibles, donde sería imposible interconectar tal cantidad de dispositivos y objetos.

Este contratiempo ha quedado solventado con la llegada del nuevo protocolo de direccionamiento IPv6 y que permite identificar un total de  $2^{128}$  direcciones, para que nos hagamos una idea, con la tecnología IPv6 podríamos identificar cada uno de los átomos que componen nuestro planeta tierra.

También es interesante mencionar **6LowPan** (Low Power Area Networks), que es un estándar creado por **Internet Engineering Task Force** publicado por primera vez en 2007. Este estándar optimiza la conexión IPv6 para usos de baja potencia y tecnologías de comunicación que precisan poco ancho de banda, es por tanto una versión reducida de IPv6 que permite una mayor escalabilidad y que básicamente incluye los mecanismos necesarios para comprimir direcciones IPv6 sobre IEEE 802.15.4.

- **Progreso de M2M**

Aunque se tiende a confundir ambos términos, Internet de las cosas (IoT) no es lo mismo que Máquina a Máquina (M2M).

La comunicación M2M consiste en el intercambio automático de datos entre máquinas y equipos sin ningún tipo de intervención humana y ofrece a la Internet de las Cosas la conectividad necesaria para desarrollar sus capacidades o como señala Matt Hatton (Director de Machina Research), M2M sería “la tubería de la Internet de las Cosas”.

Por lo general, estas comunicaciones se basan en la transmisión de mensajes cortos (SMS). Así, las M2M son máquinas que envían y reciben información a través de servicios de telemetría y/o telecontrol.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

La mayor promoción que está recibiendo el M2M es por parte del sector industrial, puesto que permite obtener información precisa en tiempo real y/o consultar los históricos que se almacenan en registros de los servidores, sin para ello tener que desplazarse al punto de medida. Esto permite reducir tiempos y esfuerzos que al final se trasladará en un aumento de la productividad y de la competitividad.

En los próximos años el M2M también recibirá un gran impulso desde las ciudades inteligentes (**Smartcities**).

- **Incremento de las redes sensores inalámbricos (WSN)**

Los sensores inalámbricos son dispositivos de bajo coste y consumo (nodos) que son capaces de obtener información de su entorno, procesarla localmente, y comunicarla a través de enlaces inalámbricos hasta un nodo central de coordinación.

Los nodos actúan como elementos de la infraestructura de comunicaciones al reenviar los mensajes transmitidos por nodos más lejanos hacia al centro de coordinación.

Las redes de sensores inalámbricas son un componente esencial de Internet de las cosas. Este sector tiene un fuerte apoyo científico, tecnológico e industrial, y el vínculo con una Internet de las cosas es inmediato.

Las áreas de aplicación pueden ser muy diversas, como por ejemplo la eficiencia energética, entornos de alta seguridad, sensores ambientales, sensores industriales, automoción, medicina y la domótica.

### Progresos de la conectividad inalámbrica.

Como hemos comentado en el apartado anterior, en los últimos años se han producido una serie de adelantos técnicos que ha impulsado significativamente la penetración de la IoT en la sociedad, entre estos adelantos hay que destacar los que están directamente relacionados con la interconexión de los objetos y más concretamente la **conectividad inalámbrica**.

La interconexión de objetos con Internet representa un gran reto tecnológico, principalmente por el gran volumen y variedad de dispositivos, además de la necesidad de disponer de conexión inalámbrica de bajo consumo.

Hoy en día, ninguna tecnología inalámbrica tiene una cuota de mercado dominante en las aplicaciones de la IoT. Aunque al principio la IoT estaba solo enfocado a la tecnología RFID, ha ido evolucionando y en la actualidad se combinan varias tecnologías con el fin proporcionar la conectividad a dispositivos tan dispares. De hecho, dependiendo de la funcionalidad y características se adopta uno u otro sistema.

A continuación se destacan algunos de estos sistemas así como su utilidad, ventajas y desventajas.

### WIFI

Basada en el estándar 802.11, es una de las tecnologías de comunicación inalámbrica mediante ondas más utilizada hoy en día. WIFI, también llamada WLAN (wireless lan, red inalámbrica), posibilita el acceso móvil de lado ancha a Internet y opera sobre las frecuencias de 2,4 Ghz y 5 Ghz.

#### Ventajas:

- Universalidad de su conexión para todo tipo de smartphones o tablets.
- El coste es muy bajo.
- Ofrece alta velocidad.
- Cuenta en la actualidad con un buen nivel de seguridad.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

**Desventajas:**

- Interferencias de las señales.
- Distancia limitada para la recepción de la señal.
- Consumo relativamente alto de energía.

### Zigbee

El estándar 802.15.4, llamado también ZigBee, trabaja en 2,4 GHz, la misma banda que Bluetooth, pero también utiliza otras bandas libres como la banda ISM para usos industriales, científicos y médicos; en concreto, 868 MHz en Europa, 915 en Estados Unidos. Está orientado al control remoto y es un estándar ideal en aplicaciones domóticas ya que los volúmenes de información que deben transmitirse son pequeños.

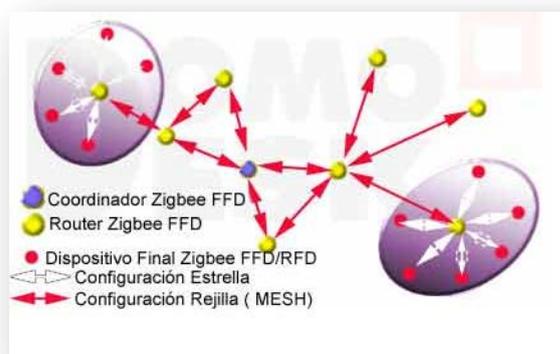
Trabaja con topología en forma de malla para poder aumentar su alcance. Se pueden definir hasta 255 nodos y el estándar incorpora un mecanismo pregunta-respuesta para determinar en cada momento cuáles son las rutas óptimas.

Las redes ZigBee pueden usar el entorno ‘con balizas’ o ‘sin balizas’. Las balizas son usadas para sincronizar los dispositivos de la red, identificando la red domótica, y describiendo la estructura de la ‘supertrama’. Los intervalos de las balizas son determinados por el coordinador de red y pueden variar desde los 15 microsegundos hasta los 4 minutos.

**El modo ‘sin balizas’:** se usa el acceso múltiple al sistema en una red punto a punto cercano. Funciona como una red de dos caminos, donde cada dispositivo es autónomo y puede iniciar una conversación en donde los otros pueden interferir. El dispositivo destino puede no oír la petición o el canal puede estar ocupado., es típicamente usado en sistemas de seguridad, donde los dispositivos, por ejemplo, sensores, detectores de movimiento o de rotura de cristales, duermen el 99,999% del tiempo. Estos elementos ‘despiertan’ de manera regular para anunciar que siguen en la red. Cuando un evento tiene lugar (se detecta algo), el sensor se ‘despierta’ instantáneamente y transmite la alarma. El coordinador de red, alimentado de la red principal todo el tiempo, recibe el mensaje y activa la alarma respectiva.

**El modo ‘baliza’** es un mecanismo de control del consumo de potencia en la red. Este modo permite a todos los dispositivos saber cuándo pueden transmitir. Aquí, los dos caminos de la red tienen un distribuidor que controla el canal y dirige las transmisiones. La principal ventaja de este método de trabajo es que se reduce el consumo de la potencia.

#### Modelo de red ZigBee



|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

### Ventajas

- Proporciona larga duración de la batería.
- Dispositivos baratos y de construcción sencilla.

### Desventajas

- Para conectar los dispositivos a internet es necesario un hub central.
- La tasa de transferencia es muy baja.

## Z-Wave

Es una especificación propietaria de la compañía Sigma Designs.

Trabaja en la banda de los 908.42MHz evitando la gran cantidad de emisoras en la banda de los 2,4GHZ y puede llegar a trabajar a 40 kbit/s pudiendo operar en rangos de hasta 30 metros en condiciones ideales.

La topología de red es tipo malla y cada elemento se comporta como un nodo que puede ser receptor o emisor reenviando el mensaje.

Al igual que Zigbee, está orientado al control remoto y es ideal en aplicaciones domóticas.

### Ventajas

- Z-Wave utiliza un protocolo mucho más sencillo que Zigbee, por lo que el desarrollo puede ser más rápido y más simple.
- Necesita poca energía y poco ancho de banda.
- Utiliza frecuencias de radio en la banda inferior a 1Mhz con lo cual no colisiona con las redes WiFi.

### Desventajas:

- La tasa de transferencia es muy baja.

## Insteon

Insteon es una robusta y redundante red de doble malla que combina la radio frecuencia inalámbrica (RF) en la banda de 915MHz con señales PLC del cableado eléctrico existente de la casa para comunicar unos módulos con otros.

Se confirma cada mensaje que se recibe, y si se detectan errores, el mensaje se reenvía automáticamente.

Cada dispositivo Insteon actúa como un repetidor de 2 vías, y a medida que aumentan la red se vuelve más fuerte y fiable.

Todos los dispositivos INSTEON tienen su propia y única ID, y también permite que los mensajes se cifren con una norma como AES-256.

### Ventajas

- Es menos susceptible que otras redes de su clase a las interferencias
- Emite señales X10 para que permitir aprovechar los módulos X10 ya instalados.
- Todos los dispositivos tienen su propia y única ID.

### Desventajas

- Menor gama de productos en comparación a Zigbee o Z-Wave.
- Aunque no es necesario un dispositivo central de control, es conveniente añadir un Hub.

## RFID

El estándar de identificación por frecuencia radial (RFID) fue inventado por el ruso Léon Theremin como una herramienta de espionaje para la Unión Soviética en 1945.

Son pequeños dispositivos, similares a una pegatina, que pueden ser adheridos a un producto, podríamos decir que

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

es un código de barras evolucionado. Que a diferencia del código de barras tradicional donde el rayo debe apuntar al código sin obstáculos en el medio, en RFID las ondas no son direccionales y pueden atravesar los obstáculos.

Se utiliza principalmente para identificar objetos a una distancia de pocos metros, con un lector normalmente estacionario que permite comunicarse inalámbricamente con pequeñas baterías transpondedores (etiquetas) conectados a los objetos. También se puede emplear para determinar la ubicación aproximada de objetos proporcionando la posición del lector cuando es conocida, para estas aplicaciones, las velocidades de 10 kbps que da esta tecnología suelen ser suficientes.

Los objetos que utilizan RFID no se comportan exactamente igual a los nodos de Internet, ya que la escasez de recursos evita la transmisión de paquetes de datos de gran tamaño y no permite basarse en los protocolos establecidos para Internet, es por ello que el lector RFID actúa como una puerta de enlace.

Los protocolos basados en HTTP y TCP han desarrollado para su uso entornos de RFID, éstos se utilizan para configurar los lectores y distribuir los datos capturados a través de Internet.

Puede trabajar en varias bandas de frecuencias:

- **Frecuencia Baja (9 – 135 kHz):** su principal ventaja es su aceptación en todo el mundo, funciona cerca de los metales y está ampliamente difundida. La distancia de lectura es inferior a 1,5 metros, por lo que las aplicaciones más habituales son la identificación de animales, o en las llaves del coche, como a sistema de seguridad antes de arrancar.
- **Frecuencia Alta (13,56 MHz):** esta frecuencia también está muy difundida, pero a diferencia de la frecuencia baja, la alta no funciona cerca de los metales. Normalmente se utiliza en aplicaciones tales como la trazabilidad de los productos, movimientos de equipajes de avión o acceso a edificios.
- **Frecuencia Ultra-alta (433 MHz y 860-960 MHz):** los equipos que operan a estas frecuencias UHF ( Ultra High Frequency ) no pueden ser utilizados de forma global porque no existen regulaciones globales para su uso y su aplicación depende de la legalidad del país. Este tipo de frecuencia se usa para aplicaciones de trazabilidad con tags activos.
- **Frecuencia de Microondas (2,45 GHz y 5,8 GHz):** estas frecuencias son las más habituales para los tags activos, pero no tienen el problema de la falta de regulaciones globales, además ofrecen largas distancias de lectura y altas velocidades de transmisión. Los tags activos que operan en el rango de las microondas son muy usados para seguimiento y trazabilidad de personas u objetos.

Existen dos Tipos de etiquetas RFID:

**Pasivas:** Requieren una fuente externa de energía que sólo se activa al reflejar la de las ondas del lector:

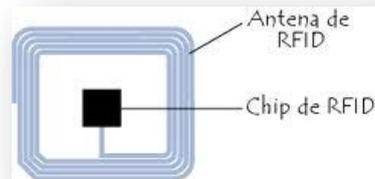
- Vida ligada a su durabilidad
- Menor coste
- Menor rango de lectura
- Desechables en su caso

**Activas:** Incluyen una fuente de energía interna y pueden leer y emitir por sí mismas.

- Actividad ligada a su batería
- Mayor coste
- Mayor rango de lectura y capacidad de almacenamiento
- Mayores funcionalidades (sensores de temperatura, humedad, luz, etc.)

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Esquema básico de una etiqueta RFID pasiva.



#### Ventajas:

- No requiere alimentación
- Tecnología establecida y ampliamente utilizada.
- Las etiquetas son baratas.

#### Desventajas:

- Altamente inseguro
- Coste por tarjeta
- No es compatible con los teléfonos inteligentes

### Near Field Communication (NFC)

El estándar NFC se trata de una tecnología inalámbrica que funciona en la banda de los 13.56 MHz , y no es necesario licenciarla para usarla, deriva de las etiquetas RFID. Se basa en el acoplamiento de dos circuitos inductivos.

Su tasa de transferencia puede alcanzar los 424 kbit/s, por lo que su enfoque es para comunicación instantánea y el alcance se mueve como máximo en un rango de los 20 cm.

La tecnología NFC puede funcionar en dos modos:

- **Activo**, en el que ambos equipos con chip NFC generan un campo electromagnético e intercambian datos.
- **Pasivo**, en el que solo hay un dispositivo activo y el otro aprovecha ese campo para intercambiar la información.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Uno de los principales usos de NFC es el pago a través del móvil



#### Ventajas:

- Ofrece una conexión de baja velocidad con la configuración extremadamente simple.
- El tiempo de conexión entre los dos dispositivos es extremadamente rápido.
- Se puede utilizar para arrancar conexiones inalámbricas con mayor capacidad.

#### Desventajas:

- Únicamente la conexión entre los artefactos es de **modo punto a punto**, no hay posibilidad de crear una red inalámbrica entre varios dispositivos.
- La tecnología NFC tiene menor presencia en smartphones.

### Bluetooth

Bluetooth es una interfaz de radio (trabaja en 2,4 GHz) para distancias cortas y servicios multimedia, como la transmisión sin hilos de imágenes entre cámaras de fotos digitales y móviles.

Para IoT existen dos variantes que usan esta tecnología

#### ➤ **Bluetooth Low Energy (BLE, Bluetooth 4.0, Bluetooth Smart).**

Es una nueva tecnología digital de radio (inalámbrica) interoperable para pequeños dispositivos desarrollada por Bluetooth.

Se fundamenta en la reducción del consumo y, por tanto, en minimizar la potencia de transmisión de la señal radio utilizada y el radio de cobertura, dispone de conexiones cifradas usando AES de 128 bits y códigos de redundancia para minimizar las transmisiones erróneas.

Añade mejoras en la velocidad de transmisión de hasta 24 Mbps y mayor capacidad de ahorro de energía. Es compatible con NFC, cuyo chip habilita a los dispositivos para interactuar entre sí por medio de la proximidad.

La última actualización es la versión 4.2, que incorpora las siguientes mejoras:

- **Mejor coexistencia con la nueva conexión móvil 4G LTE.** La banda de frecuencia de Bluetooth se sitúa entre las bandas 40 y 41 del 4G LTE, lo que puede provocar interferencias. En Bluetooth 4.1 estas interferencias, si existen, se corrigen.
- **Reconexiones automáticas.** Mejoras en los mecanismos para que un periférico se reconecte

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

automáticamente cuando ha salido y vuelve a entrar dentro del campo de alcance de una conexión BT.

- **Mejor transferencia de datos por lotes.** Se están poniendo de moda los sensores que capturan datos (por ejemplo, las mencionadas pulsaciones cuando sales a correr) y los almacenan todos de golpe cuando llegas a casa. Con Bluetooth 4.1 estas transmisiones en bloque son más rápidas y eficaces.
- **Conexión simultánea.** Un mismo dispositivo puede usarse como receptor y como emisor de datos al mismo tiempo.
- **Internet de las Cosas.** Se han añadido nuevos estándares y modos de funcionamiento para que todo tipo de objetos puedan conectarse entre sí, o a Internet, a través de Bluetooth.
- **Soporte extendido de IPv6.**

Bluetooth LE se está posicionando como el estándar clave para dar soporte a la nueva ola de wearable devices que estamos viendo llegar al mercado.

### ➤ **iBeacons.**

Es propietaria de Apple y viene integrado en iOS 7, se basa en el uso de Bluetooth Low Energy.

Los iBeacon se entienden con dispositivos que funcionen con cualquier SO, siempre con el requisito de que estos dispongan de Bluetooth 4.0.

**Los sensores pueden funcionar durante dos años** con una simple pila de botón e integran un acelerómetro, memoria Flash, procesador ARM y la citada conectividad Bluetooth.

Existen multitud de empresas que comercializan estas balizas, los precios varían, pero por lo general están en torno a los 30€ por unidad.

Cada baliza tiene al menos 3 parámetros configurables.

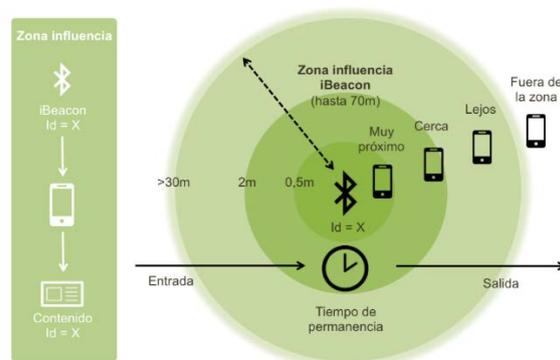
**UUID:** se utiliza el mismo para identificar todas las balizas de un recinto.

**Mayor ID:** identifica las balizas de una misma habitación, sección...

**Menor ID:** identifica un objeto.

La zona que la señal de un iBeacon cubre puede llegar a configurarse hasta 70m de radio, según el espacio en que se coloque y la marca de la baliza.

A esta zona, se le denomina “zona de influencia” y puede configurarse en tres tramos, según la distancia:



Pero el potencial de los iBeacons no está en las balizas en sí, sino en cómo se programa el receptor, es decir; el dispositivo móvil. Esta tecnología no envía contenido, sino que únicamente emite una señal que funcionará como trigger. Es necesario desarrollar una app que entienda estas señales y que sepa qué hacer cuando entra en cada una de las zonas de influencia de los iBeacons.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

En la siguiente imagen se muestra que aspecto tienen los iBeacons



**Ventajas:**

- Tecnología establecida y ampliamente utilizada.
- Frente a otras tecnologías como la NFC su alcance llega hasta los 50 metros, lo que permite que el usuario no tenga que “pasar” su terminal por un lector para recibir la información.

**Desventajas:**

- La tecnología Bluetooth está evolucionando muy rápido y obliga a cambios de hardware.
- La duración de las baterías de los iBeacons.

**Redes de telefonía móvil**

Este método de conexión es muy utilizado en M2M y permite mediante el uso de una SIM (subscriber identity module), el acceso a las distintas redes de telefonía móvil (GSM, GPRS, CDMA, GSM, 2G / GSM, 3G, 4G / LTE, EDGE), asimismo, este tipo de conexión puede alcanzar el 100% de cobertura en caso de que la señal de la estación base sea muy limitada puesto que se puede realizar la conexión vía satélite.

**Ventajas:**

- Conexión Estable
- Compatibilidad universal
- Máxima cobertura

**Desventajas:**

- No existe conexión directa con el dispositivo (satélite)
- En algunos casos el coste puede ser alto.
- Retardos (satélite)
- Alto consumo de energía

Otro estándar de conectividad inalámbrica en crecimiento es el protocolo **Weightless** que utiliza frecuencias de televisión no usadas habitualmente y se ha desarrollado específicamente para aplicaciones M2M.

**Tabla comparativa de las diferentes tecnologías Inalámbricas**

|                  | Wifi  | Zigbee  | Z-Wave               | Insteon   | RFID  | NFC   | Bluetooth Low Energy  | iBeacons   |
|------------------|---|---|----------------------|---|---|---|---|--|
| <b>Estándar</b>  | IEEE 802.11 a,b,g,n,ac                                      | IEEE 802.15.4   | IEEE 802.15.4        | N/A   | ISO 14443   | ISO 14443                                   | IEEE 802.15.1   | IEEE 802.15.1  |
| <b>Topología</b> | Estrella  | Malla, estrella y árbol   | Malla                | Doble malla   | Malla y estrella  | Malla y estrella                            | Piconet Arquitectura maestro-esclavos   | Ibeacons Arquitectura maestro-esclavos   |
| <b>Banda RF</b>  | 2,4-5 GHz   | 868 MHz en Europa, 915 en Estados Unidos y 2,4 GHz en todo el mundo | 908.42MHz            | 915MHz  | Frecuencia Baja ( 9 – 135 kHz ):<br>Frecuencia Alta ( 13,56 MHz )<br>Frecuencia Ultra-alta ( 433 MHz y 860-960 MHz )<br>Frecuencia de Microondas ( 2,45 GHz y 5,8 GHz | 13.56 MHz                                   | 2,4 GHz   | 2,4 GHz  |
| <b>Velocidad</b> | Hasta 300 Mbit/s  | 250 kbit/s  | 40 kbit/s            | 38.4 kbit/s   | 10 kbit/s   | 424 kbit/s                                  | 24 Mbps   | 24 Mbps  |
| <b>Alcance</b>   | 300 metros  | 10 metros   | 30 metros            | 50 m  | 7 m   | 20 cm.                                      | 10 metros   | 50 metros  |
| <b>consumo</b>   | Alto  | Muy bajo  | Muy bajo             | Muy bajo  | Muy Bajo  | Muy bajo                                    | Bajo  | Bajo   |
| <b>Nodos</b>     | 32  | 65535   | 232                  | 256   | N/A   | N/A   | 8   | 8  |
| <b>Uso</b>       | Aplicaciones que requieran una alta transferencia de datos. | Control remoto. Domótica.   | Domótica. Seguridad. | Domótica. Ideal para reutilización de módulos X10 ya instalados | Identificación de objetos, sobre todo para empresas de logística  | Indicado para realizar pagos desde el móvil | Transferencia de información multimedia. Conexión de cascos y altavoces Inalámbricos. | Crear eventos de entrada y salida para realizar recorridos guiados, tanto en rutas turísticas como en rutas de entrada o salida a distintas partes de un edificio. |

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Protocolos y estándares

A pesar de que en la actualidad se cumplen las condiciones oportunas para su despegue, como ya hemos comprobado por ejemplo con las tecnologías de conexión inalámbricas, uno de los retos a los que nos enfrentamos se da en la comunicación de los dispositivos entre sí, puesto que en muchas ocasiones se trata de pequeños sensores alimentados por baterías y enlazados con pasarelas más potentes a través de redes radio de baja potencia (redes WSN o LowPAN). Estas redes radio tienen mayor probabilidad de pérdida de mensajes y para su funcionamiento óptimo precisan que los mensajes sean lo más cortos posibles, además hay que añadir el problema de rendimiento que ofrece el protocolo HTTP, que si a favor tiene su potencia y sencillez, en contra nos encontramos con un alto costo en el tamaño del código de una implementación y en consumo de recursos de red.

Como consecuencia, el uso de HTTP-REST sobre TCP/IP resulta ineficiente y desaconsejable en estos entornos y por el momento **no existe ningún protocolo aún para la IoT que cuente con el apoyo mayoritario de la comunidad**, y que por tanto pueda considerarse un estándar. Los protocolos más utilizados se presentan en el siguiente cuadro resumen:

| MQTT  | MQTT-S  | CoAP  | XMPP   | REST API  |
|---|---|---|--|---|
| <ul style="list-style-type: none"> <li>• Uso de mensajes "broadcast" para suscripción y publicación de datos con independencia de la aplicación.</li> <li>• Transporte de mensajes transparente y con un flujo de datos optimizado lo cual permite reducir el tráfico en la red.</li> <li>• Soporte bidireccional y sin pérdidas de datos.</li> <li>• Los dispositivos pueden permanecer un 95 % en estado de inactividad operativa o "sleep mode"</li> </ul> | <ul style="list-style-type: none"> <li>• Es similar a MQTT pero los sensores pueden permanecer un 100% en su estado de inactividad operativa.</li> <li>• Es diez veces superior en escalabilidad de dispositivos que MQTT.</li> <li>• Trabaja igualmente bajo UDP que sobre TCP.</li> </ul> | <ul style="list-style-type: none"> <li>• Protocolo de aplicación de transferencia web para funcionar con recursos muy limitados.</li> <li>• Trabaja bajo UDP.</li> <li>• Permite que dispositivos y redes limitadas se conecten a Internet, además de integrarse con la arquitectura Web y el HTTP</li> </ul> | <ul style="list-style-type: none"> <li>• Escalabilidad en tiempo real.</li> <li>• Aconsejable para cuando se requiere seguridad extra y el tráfico es alto y complejo por dispositivo</li> </ul> | <ul style="list-style-type: none"> <li>• Comunicación unidireccional desde los dispositivos a la nube.</li> <li>• No hay limitaciones específicas para el tráfico.</li> </ul> |

El problema es que no se ve muy claro el panorama en cuanto a estándares comunes de conectividad entre proveedores, y que por lo tanto que una verdadera coordinación entre múltiples dispositivos nunca llegue a acontecer puesto que los fabricantes, crearán entornos cerrados de IoT que solo trabajarán con sus productos.

Para evitar esta fractura, una solución estaría en llegar un acuerdo para la utilización de protocolos y estándares abiertos. En este sentido la Fundación Linux, creó el **AllSeen Alliance**, que lanzó una pila de código a finales del 2013, la cual puede ser utilizada por cualquier fabricante de sistemas electrónicos o de electrodomésticos para conectarse con otros productos. La alianza espera que el enorme peso de la adopción de esta pila, llamada **AllJoyn**, ayude a empujar la IoT hacia los estándares abiertos.

Otra alternativa es la que ofrece la **UIT** (Unión Internacional de Telecomunicaciones) para el **desarrollo de un estándar nuevo** en el que pueden colaborar las entidades que lo deseen. A esta proposición se la conoce como **Globals Standards Initiative (IoT-GSI)**, y pretende dar una respuesta abierta a las cuestiones comunes de IoT que deben ser estandarizadas para poder trabajar a nivel global.

En poco tiempo hemos tenido varias novedades referentes a las apuestas que grandes compañías están haciendo

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

por estandarizar el internet de las cosas, hoy surge una nueva iniciativa a la que han llamado **Thread Group**, donde se suman más gigantes tecnológicos como Nest Labs (que pertenece ahora a Google), Samsung y ARM.

Este nuevo estándar estaría enfocado en conectar productos diseñados exclusivamente para el hogar, prometen que cualquier dispositivo que tenga conectividad inalámbrica con el estándar IEEE 802.15.4 podrá ser utilizado añadiendo una simple capa de software.

## La IoT y el Big Data

En este apartado presentaremos el concepto de Big Data y como está estrechamente relacionado con la IoT.

Big Data hace referencia al **control, gestión y análisis de gigantes cantidades de datos que no pueden ser analizados de forma tradicional** y las herramientas usadas para dicho análisis.

El objetivo del Big Data es poder convertir toda esta información en datos relevantes para la toma de decisiones. Las empresas que sean capaces de realizar un análisis del Big Data, capturar analizar y explotar eficazmente esta ingente cantidad de datos, mejorarán su competitividad y productividad. Las preferencias de los consumidores varían con frecuencia y hay que estar atentos para proporcionarles lo que demandan en cada momento si queremos **mantener la relevancia y competitividad** de nuestro negocio.

El gran crecimiento de IoT en los próximos años permitirá que cualquier objeto dotado de sensores o funcionalidades sea una fuente de datos. Estos datos debidamente procesados proporcionarán información relevante que nos ayudaran a comprender nuestro entorno. Para lograrlo, se habla de la asociación de la IoT con el Big Data.

Según afirma la empresa IBM, en el año 2015 la primera fuente de información para el Big Data será la internet de las cosas, superando incluso a las redes sociales.

Los dos conceptos, Big Data y la IoT aplicadas a un entorno empresarial proporcionan una **herramienta de análisis del consumidor**, donde las compañías podrán utilizar la información que se irá almacenando sobre nosotros en la red para ofrecernos realmente sólo aquello que nos interesa y por supuesto vender de la forma más óptima y personalizada.

Según la consultora ABI Research, **la integración, análisis y almacenamiento de datos de la IoT**, tendrá un valor de **5.700 millones de dólares en el año 2015** y esta cifra seguirá en aumento en los próximos cinco años, es por ello que muchos expertos consideran los datos como el oro del mañana y por lo tanto, será imprescindible realizar una buena gestión del Big Data obtenido de millones de “conversaciones” entre las cosas y los seres humanos con el fin de ofrecer a cada persona exclusivamente la información que necesite.

## El futuro del Marketing

Si hay un sector que va salir beneficiado de esta asociación entre la IoT y el Big Data, este va a ser el sector del Marketing.

En el informe “Big Data, Big Impact: New Possibilities for International Development” realizado por el World Economic Forum, señala que **los datos son un activo económico comparable a las divisas de oro**.

Para los profesionales del marketing la primera consecuencia es obtener una fuente muy potente de datos, una fuente de información en la que las preguntas correctas tienen respuestas precisas sobre los deseos y comportamientos de los consumidores.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Hay dos factores que van a permitir que el sector del marketing y publicidad digital crezca exponencialmente: el uso de la información de los dispositivos conectados a la IoT y el aumento de espacios publicitarios.

**El creciente número de objetos conectados a internet** generan una gran cantidad de información de nuestros usos que tras ser procesados y analizados en detalle optimizará las campañas de marketing enfocándolas de forma individualizada en función de las necesidades de los consumidores.

**Las posibilidades del marketing digital crecerán exponencialmente** ya que, el número de pantallas se incrementarán y por consiguiente el número de espacios publicitarios se incrementará también.

Este crecimiento de las capacidades del marketing online obligará a la realización de procesos analíticos previos y posteriores más detallados, todo ello con el fin de poder optimizar las campañas y poder apuntar al público objetivo de una forma más exacta.

Con esta evolución del marketing digital, los datos analíticos que nos ofrecerán las campañas de publicidad digital se acercarán cada vez más a los estándares de “perfección”.

A pesar de los problemas que surgirán en torno a la legalidad o no de la forma de recopilar datos, los cuales serán imprescindibles para la publicidad, podemos afirmar que **el futuro del marketing digital será muy prometedor.**

## Privacidad y seguridad en la IoT

Uno de los riesgos que se corre con toda esta información que generan los dispositivos IoT es que pueden revelar detalles íntimos personales acerca de nuestras vidas, tales como nuestras necesidades médicas, interacciones con otras personas y los hábitos personales. Estos datos son analizados por empresas que posteriormente son utilizados o cedidos a terceros, como pueden ser desarrolladores de aplicaciones o empresas de marketing para ofrecer publicidad.

El problema es que las normas de protección de estos datos no avanzan al mismo ritmo, y **ante la expansión social que se producirá en los próximos años de la IoT**, se hace necesaria una normativa para asegurar la privacidad de los usuarios.

En este sentido, la actual Ley de Protección de Datos de Carácter Personal (LOPD) regula ciertos aspectos como la limitación explícita al cruce de datos con el fin de evitar que una determinada información se pueda relacionar con un determinado usuario, pero no existe un apartado concreto para la IoT y el Big Data, es por ello que son muchas las empresas están vulnerando la privacidad de los datos y poniendo en jaque la privacidad del usuario. Un ejemplo de estas técnicas de explotación, es la empresa responsable del videojuego de estrategia Clash of Clans, que ha utilizado la tecnología Big Data para predecir el desenganche de los clientes.

**Estos procedimientos pueden ser muy peligrosos para el ciudadano**, puesto que un análisis de los patrones de su uso podría revelar detalles de su estilo de vida, hábitos o presencia. Un ejemplo significativo de estas prácticas sucedió hace unos años cuando la cadena Target envió a una adolescente de Minneapolis cupones de descuento para productos premamá y de bebé. El padre acudió indignado a la compañía. Sin embargo, a los pocos días la chica confesó que, efectivamente, estaba embarazada. Había comprado complementos vitamínicos, toallitas sin perfume y otro tipo de productos que llevaron al programa informático de la tienda a considerarla una madre en potencia, se puede considerar como una anécdota pero si en lugar de una empresa de ese sector fuese una aseguradora, con los datos obtenidos podría subirnos las cuotas o directamente ser rechazados.

Tal y como asegura Scott Taylor, responsable de HP sobre privacidad y protección de datos “el uso inadecuado de la información, podría favorecer la discriminación de niños, enfermos, personas socialmente desfavorecidas o de cualquier tipo de ciudadano”.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

En cuanto a lo que respecta a la seguridad, se evidencia su patente falta en este entorno. La tendencia hacia la conectividad de todos los objetos cotidianos incrementa los riesgos para la seguridad de datos y dispositivos, por lo tanto, los ciberdelincuentes tendrán muchos más objetivos donde poder realizar sus fechorías bien robando información concreta de un usuario o hackeando los dispositivos de los fabricantes. Según los expertos, hay varios ataques que se podrían volver comunes:

- **La denegación del servicio.** La paralización de un servicio se agrava si todos los dispositivos están conectados.
- **Los ataques de malware.** Mediante un código malicioso, se podrían infectar cientos de ordenadores para controlar una red de dispositivos inteligentes o poner en peligro su software. Durante las pasadas Navidades, más de 100.000 dispositivos de consumo, incluyendo frigoríficos conectados a Internet y televisores inteligentes, ayudaron a enviar más de 750.000 emails con malware en todo el mundo.
- **Violaciones de datos.** Espiar las comunicaciones y recopilar información sobre estos dispositivos (que además pueden almacenar información en la nube) se convertirá en otro de los ataques más frecuentes, comprometiendo nuestra privacidad. **Infracciones inadvertidas.** Nuestra información confidencial no solo puede sufrir ataques específicos, sino que también puede perderse o verse expuesta accidentalmente si los dispositivos no protegen debidamente la privacidad.
- **Los ataques a la seguridad de nuestro propio hogar.** La mayoría de fabricantes de estos dispositivos no han tenido en cuenta que la seguridad era necesaria y muchos carecen de mecanismos para proteger la información debidamente.

Todo ello pone de relieve la necesidad de que las empresas, fabricantes y usuarios empleen los mecanismos necesarios para mitigar estos ataques, proporcionado por ejemplo mecanismos de actualización de firmware y métodos de encriptación más robustos por parte de las compañías y recurriendo a fuertes **políticas de contraseñas por parte de los usuarios.**

## Retos de la IoT

A pesar de la evolución que está viviendo, la IoT sigue siendo una tendencia tecnológica que no está exenta de problemas y desafíos y que por el momento, tendremos que esperar para ver cómo avanza el desarrollo y producción de estos nuevos aparatos y su interconexión.

Si no queremos que sea otra tecnología que pertenezca a una amplia lista de fracasos tecnológicos, es indispensable encontrar soluciones a varios aspectos que necesariamente deben ser adecuadas para brindar los niveles de servicio y seguridad que se van a requerir.

Los principales desafíos con los que se encuentra el ecosistema de IoT son:

**Duración de las baterías:** Todas las tecnologías relacionadas con la IoT están avanzando a pasos agigantados, excepto la relacionada con las baterías. La última gran innovación en este sentido fue la batería Li-Ión, y estamos viendo como ya en muchos casos es insuficiente para proporcionar la autonomía demandada por los nuevos dispositivos. Hoy en día existen varios campos de investigación, como por ejemplo Apple con las baterías de hidrógeno, IBM con la batería de litio-aire o Toshiba con la pila de metanol, pero sin vistas a que estén disponibles a corto plazo. Debido a las limitaciones de la vida y duración de la batería, los nodos se construyen teniendo presente la conservación de la energía, y generalmente pasan mucho tiempo en modo ‘durmiente’ (sleep) de bajo consumo de potencia.

**La falta de una infraestructura compartida:** La Internet de las Cosas es un complejo de interconexión de hardware, tales como sensores y actuadores, y el software que funciona en el nivel de ensamblado. Juntos, constituyen una plataforma para desarrolladores y empresas. La naturaleza vertical de esta plataforma ha

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

contribuido a la fragmentación de su infraestructura y actualmente hay pocas plataformas de código abierto en Internet de las Cosas y cada dispositivo IoT actualmente está instalado en su plataforma y ecosistema.

En este sentido, el operador de infraestructuras de telecomunicaciones **Abertis Telecom** ha presentado en el **MWC** el despliegue de la primera red IoT puesta en marcha en España y diseñada para ofrecer soluciones de conectividad. El sistema '**Ultra Narrow Band**' (UNB) desplegado es de tecnología Sigfox, y cuenta con más de 1.300 estaciones base para dar cobertura a dispositivos y objetos conectados a lo largo de la geografía española

**La falta de estándares:** No existe una organización mundial para estandarizar protocolos, hardware, software en el Internet de las cosas. Hay algunas iniciativas dentro de la UIT (Unión Telegráfica Internacional) y especialmente importante el trabajo que está realizando el Comité Técnico de las Comunicaciones Máquina a Máquina (M2M) de la ETSI tratando de agrupar las múltiples tecnologías y estándares ya existentes en un escenario muy fragmentado, para emprender su armonización y completarlas en los ámbitos en que se detecten vacíos normativos.

**Ausencia de nuevos e innovadores modelos de negocio rentables:** De momento son pocos los sectores que apuestan por la diversidad de posibilidades que ofrece la IoT, en el caso de las compañías de seguros de automóviles ya han surgido algunos servicios como “Pay as You Drive”, que se basa en que se paga más o menos por el seguro del coche en función de la forma de conducir, los lugares por los que se conduce o las horas en las que se conduce. En España lo ofertan las empresas Generali y Mapfre, pero de momento sin grandes resultados.

**Control de los datos:** Desde la perspectiva del usuario, esta es una de las barreras más importantes para la adopción a gran escala de la tecnología. El control de los datos es comúnmente confundido con la propiedad de los datos. No se trata de quién es dueño de la información sino de controlar y decidir sobre quién tiene acceso a mis datos.

**Intercambio de información:** En el paradigma de la Internet de las Cosas, los datos son oro. Las empresas ofrecen un servicio gratuito o de precio nominal a cambio de los datos personales de un consumidor. Estos datos se venden bien a los anunciantes o son utilizados para desarrollar otros productos o servicios útiles para los consumidores. Los clientes huyen por lo general de este tipo de aplicaciones.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Estudio plataformas IoT

Una vez presentado el concepto de IoT en profundidad, en la siguiente parte del proyecto, se realiza un estudio y comparativa de las distintas plataformas de la IoT existentes actualmente en el mercado, tanto a nivel hardware como software, no obstante, en algunos casos encontraremos empresas que ofrecen un servicio integral.

Por otro lado, el auge que está experimentando la IoT está provocando que cada pocos meses nos topemos con la aparición de nuevas plataformas H/S que en muchos casos son meros clones o incluyen pequeñas variantes con respecto de las ya existentes, es por ello, que se ha seleccionado un número de plataformas lo más representativo posible, descartando estas últimas.

### Plataformas hardware

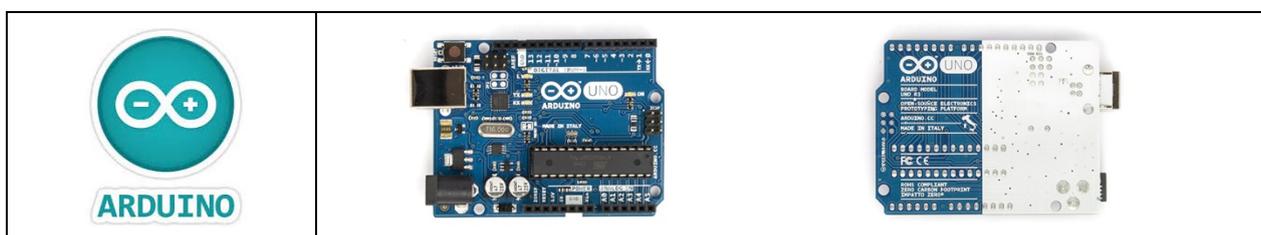
La gran diversidad existente de plataformas permite realizar diferentes clasificaciones ateniéndose a sus características y funcionalidades, como por ejemplo, por código abierto, por tipo de conectividad, uso profesional. Sin embargo, en este estudio he optado por organizarlas por placas basadas en un microcontrolador y placas basadas en microprocesador.

Las basadas en **microcontroladores**, tienen la capacidad para trabajar en tiempo real, ofreciendo así, una mayor flexibilidad a la hora de utilizar casi cualquier tipo de sensor, es por ello que son una buena apuesta para proyectos de hardware puro, por otro lado, pueden ser encendidas y apagadas sin ocasionar daños, cosa que no ocurre con los microordenadores.

Las basadas en **microprocesador**: son un computador independiente que ejecutan un sistema operativo, ofrecen multitarea, y dada su mayor potencia son aconsejables para aplicaciones que utilizan una interfaz gráfica de usuario, presentan problemas en la lectura de los sensores analógicos requiriendo en algunos casos asistencia hardware extra.

Por lo tanto, ambas son totalmente válidas para conectar nuestros objetos a internet, donde la elección de una u otra plataforma, dependerá del tipo de proyecto a desarrollar.

### Microcontroladores



<http://www.arduino.cc/>

Es una plataforma de electrónica de código y hardware abierto, destinada principalmente a la creación de prototipos para entornos interactivos. Es una placa basada en los microcontroladores **Atmel AVR** y **CortexM3 de 32 bits** que dispone de varios puertos de entrada/salida para actuar sobre el entorno, es una plataforma potente y muy económica apta para todo tipo de usuarios y donde no existe la necesidad de tener grandes conocimientos previos, además dispone de herramientas sencillas de utilizar para el diseño y de aplicaciones electrónicas de control.

Cuenta con las denominadas placas **Shields Arduino**, que se pueden conectar en la parte superior de la placa Arduino y extender sus capacidades, como por ejemplo, dotarle de conectividad wifi, Zigbee o Bluetooth. Los

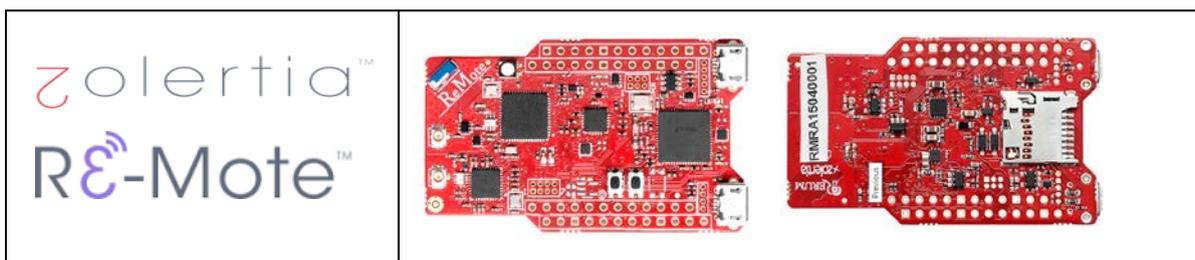
|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

diferentes Shields siguen la misma filosofía que el conjunto de herramientas originales: son fáciles de montar, y baratos de producir.

Dado su carácter educativo y aprendizaje, las placas pueden ser hechas a mano o compradas montadas de fábrica. Los ficheros de diseño de referencia (CAD) están disponibles bajo una licencia abierta.

El lenguaje de programación de alto nivel se conoce con el nombre de **Arduino**, está basado en Wiring y cuenta con unas sentencias y sintaxis muy similares a las del lenguaje C, también cabe la posibilidad de utilizar el lenguaje **Processing** que está basado en Java. Todo este software puede ser descargado de forma gratuita y viene con multitud de ejemplos. Además, disponen de librerías para prácticamente cualquier componente externo que se le quiera acoplar. Una vez grabado un programa sobre la memoria del controlador, éste se ejecuta sin necesidad de estar conectado a un ordenador. Disfrutamos así de un proyecto hardware/software totalmente autónomo e independiente.

Asimismo, cuenta con una **gran comunidad de apoyo** y una considerable documentación, que sumado a los incontables tutoriales que existen en la red sobre esta plataforma, la hacen una de las más fáciles de desarrollar. Está orientada especialmente para aprendizaje, educación y experimentación, aunque también cabe la posibilidad de realizar proyectos profesionales.



<http://remote.zolertia.com/solution-as-you-play/>

Es una plataforma hardware de código abierto, que permite conectar cualquier idea a Internet, proporciona una conectividad perfecta para la mayoría de aplicaciones ya sean para interior como para exterior y está alimentada por baterías.

La Re-Mote fue desarrollada en conjunto con varias universidades y socios industriales de distintos países en el marco de un proyecto europeo para crear una IoT potente destinada a la educación, las ciudades inteligentes, la seguridad, la robótica y proyectos.

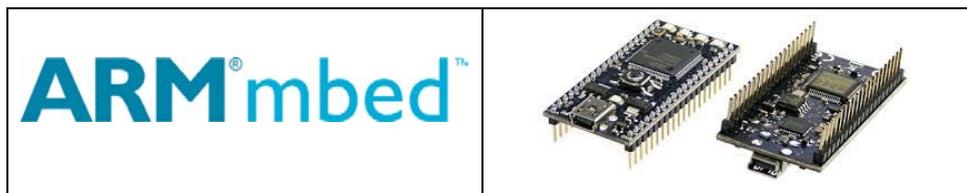
Se basa en el CC2538 ARM Cortex-M3 con una radio IEEE 802.15.4 Zigbee 2.4GHz, corriendo hasta 32Mhz con 512 KB de flash programable y 32 KB de RAM, incluye un transceptor CC1120 RF 868 / 915Mhz. También cuenta con un sensor de temperatura TMP102, microSD sobre SPI, cargador de batería y maneja voltajes que van desde 2 a 28 V CC sin necesidad de transformadores adicionales.

Se pueden añadir distintos sensores, como por ejemplo, de gas, lluvia o acelerómetros.

Es compatible con Sigfox, la primera y única sociedad que ofrece una conectividad celular mundial para la Internet de los objetos, totalmente dedicada a las comunicaciones de baja velocidad.

Soporta los sistemas operativos, TinyOS, RIOT y Contiki. Asimismo es compatible con Arduino y Raspberri Pi. La programación se realiza remotamente desde una aplicación en la nube llamada Thingsquare.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



<https://mbed.org/>

Antes conocida como Sensinode, Mbed es una plataforma propietaria para prototipos y experimentación basada en microcontroladores ARM Cortex-M3 y ARM Cortex-M0 de 32 bits. La velocidad es de 96MHz y dispone de 32KB de memoria RAM y 512KB de memoria FLASH. Se alimenta mediante una conexión USB de 5V o una fuente de alimentación externa de 4.5-9V. Dispone de 40 pines que le añaden conexión a Ethernet, lectura/escritura a dispositivos USB, y comunicación con otros dispositivos a través de bus SPI, I2C, UART, CAN. Además, estos pines se pueden configurar de diferentes maneras.

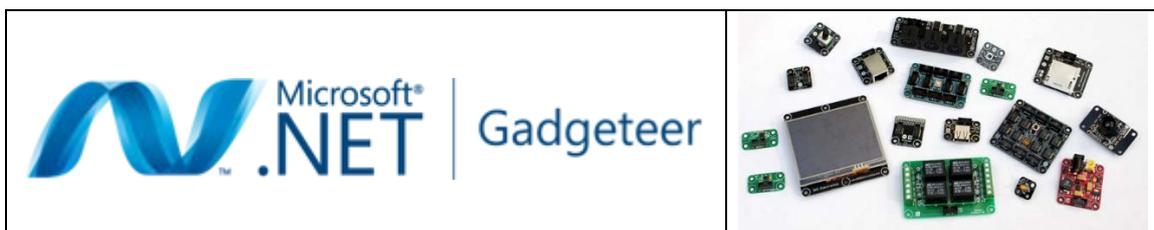
Es ideal para desarrolladores experimentados ofreciendo una plataforma productiva para realizar pruebas conceptuales y prototipos, mientras que a los principiantes, les provee una forma accesible de realizar proyectos con microcontroladores de 32 bits mediante el acceso a librerías (de lenguaje C), tutoriales y ejemplos, además de contar con la comunidad online de mbed.

El usuario crea sus programas de manera online, los compila y posteriormente descarga el ejecutable de la plataforma MBED, aunque es posible también, programarlo en con un IDE offline.

El entorno permite de una manera sencilla desarrollar código, importar librerías y ejemplos y publicar nuestro código en colaboración con otros desarrolladores y permite exportar proyectos a IDE's "offline" como: uVision, Code Red y CodeSourcery.

Una de las características más novedosas de estas herramientas es la facilidad de programación, ya que conectando el dispositivo al ordenador a través de un cable USB, la placa MBED aparece en el host como una unidad de memoria en la cual solamente debemos copiar el fichero ejecutable, falta pulsar el botón reset de la placa y el programa se ejecutará.

Esta integración ofrece a los usuarios un entorno de desarrollo de sistemas embebidos para desarrollar prototipos de una manera simple, eficiente y muy rápida.

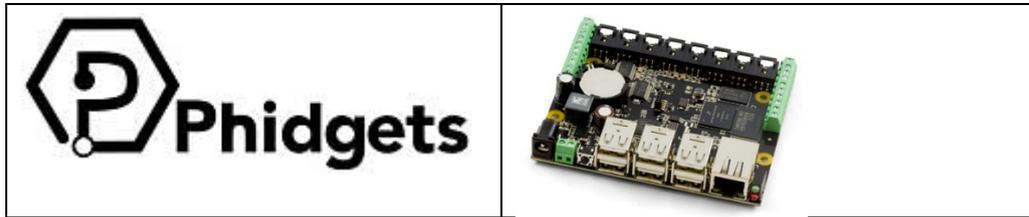


<http://www.netmf.com/gadgeteer/>

Microsoft .NET Gadgeteer es un conjunto de herramientas de código abierto para la creación de pequeños dispositivos electrónicos que utilizan .NET Micro Framework. Combina las ventajas de la programación orientada a objetos, montaje sin soldadura de la electrónica, y el apoyo para el diseño físico personalizable, no tiene capacidad de proceso en tiempo real.

El Sistema Gadgeteer .NET se compone de una placa base que contiene un procesador incorporado y una variedad de módulos que se conectan a la placa base a través de una sencilla interfaz de plug-and-play. Hay una gran variedad de módulos .NET Gadgeteer disponibles, incluyendo: pantalla, cámara, redes, almacenamiento y gran diversidad de sensores y controles de entrada. Las tomas de placas base Gadgeteer .NET están numeradas y marcadas con una o más letras que indican qué módulos se pueden conectar en él.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



<http://www.phidgets.com/>

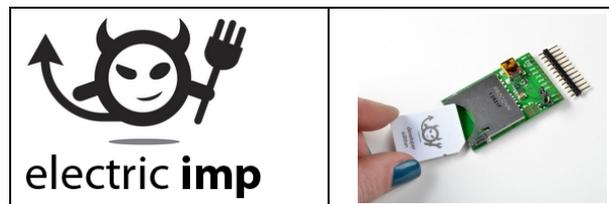
Es un plataforma propietaria, que cuenta con una CPU **Freescale i.MX28** que corre a 454 MHz y dispone de un puerto Ethernet.

Phidgets sólo necesita (y sólo permite) una conexión a un ordenador, además debe estar conectado al PC permanentemente, pero permite gracias a los lenguajes de alto nivel, desarrollar de una forma más rápida y fácil. El coste es de 3 a 5 veces superior a Arduino.

Phidgets tiene la misma filosofía de diseño que .NET Gadgeteer: **componentes modulares** y un sistema de software único que le permite ponerse en marcha rápidamente, la construcción de elementos hardware es muy rápida ya que no es necesario contar con conocimientos de electrónica.

La mayor diferencia entre Phidgets y otros kits de creación de prototipos es que el controlador Phidgets en sí mismo no es programable.

En su lugar, se conecta Phidgets a sensores (patentados) para la E/S y que se conectan luego a un PC. Puede utilizar los lenguajes de programación de alto nivel (Java / C ++ / etc) para controlar directamente los sensores / actuadores.



<https://electricimp.com/>

La plataforma eléctrica Imp cuenta con hardware, software, sistema operativo, APIs, servicios en la nube y la seguridad totalmente integrado, proporcionando a sus dispositivos de inteligencia, escalabilidad y flexibilidad.

Son **pequeñas tarjetas Wi-Fi** con forma de tarjeta SD, que pueden acoplarse a casi cualquier dispositivo, contiene un chip Wi-Fi compatible con WEP, WPA y WPA2, junto a una antena. Además, tiene un procesador integrado y seis salidas (entre ellas UART, I2C y SPI), lo que hace que **se pueda programar** para funcionar con dispositivos que utilicen este tipo de tecnologías.

La propia compañía ofrece circuitos de adaptación para otros aparatos existentes, y están en conversaciones con los fabricantes de varias marcas para que **incluyan una ranura para esta tarjeta en sus dispositivos**.

Cuentan con un procesador **Cortex-M3** que tiene un consumo bajo de energía y un buen rendimiento, los seis conectores de la tarjeta pueden utilizarse como UART, I2C, SPI, entrada y salida analógica, PWMs, GPIOs... y todo ello seleccionable por software.

En cuanto al desarrollo, no hay que instalar ningún SDK, ni JTAG, simplemente se abre una ventana de un navegador y se compila y se ejecuta el código directamente en el Imp, por lo se podría estar compilando y ejecutando código sobre un Electric Imp que se encuentre en cualquier parte del mundo. El software que ejecuta

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  | <b>Fecha:</b> 13 de junio del 2015 |
|  | <b>Versión del documento:</b> V3.0  |                                    |

Imp está escrito en Squirrel, un lenguaje parecido a C con extensiones para hablar con las interfaces de hardware y el servicio.

La empresa ofrece en diferentes formatos en función de las necesidades de cada fabricante y que incorporan, dependiendo del modelo, todo tipo de **sensores, acelerómetros, detectores de presencia, de sonido**, etc.

Cada módulo imp está optimizado para funcionar en modo de “sueño profundo” para minimizar el consumo de energía, permitiendo que los dispositivos funcionen durante años con sólo dos pilas AA.



<http://www.mediatek.com/en/about/company-overview/>

La compañía taiwanesa de semiconductores, nos trae una plataforma de desarrollo **propietaria** para la IoT y que bajo la iniciativa global **MediaTek Labs proporciona** un kit de desarrollo software (SDK), un kit de desarrollo hardware (HDK), y documentación técnica.

La base es la plataforma de desarrollo **LinkIt** y se centra en el **chipset MediaTek Aster (MT2502)** que es hasta el momento el **System-on-Chip (SoC)** más pequeño del mercado (5.4mm\*6.2mm).

Gracias a su modularidad en la arquitectura del software ofrece a los desarrolladores un alto grado de flexibilidad. También se le pueden unir módulos de **conectividad WiFi o un receptor GPS**.

MediaTek ofrece los componentes base de la plataforma en forma de la **placa de desarrollo LinkIt ONE por 79 dólares, que cuenta con el SoC Aster**, los chipsets WiFi y GPS, y las herramientas hardware y software con las que ponerse a trabajar.

Proporciona una API para acceder a las características de la SoC Aster, que es similar a la de Arduino; además con **The LinkIt ONE SDK** (para Arduino), es posible crear aplicaciones de dispositivos LinkIt ONE utilizando las herramientas propias de Arduino.



<http://www.thinkingthings.telefonica.com/>

Los Thinking Things son diferentes cubos de plástico, basados en placas Arduino, que se acoplan como si fueran piezas de lego. Cada cubo tiene una función distinta. Se denomina stack a un conjunto de cubos conectados. Se pueden poner todos los cubos que deseemos en el mismo stack.

Es por tanto, una solución modular para la creación de productos inteligentes y conectados a internet. Accediendo a una web podemos ver los datos y actuar definiendo unas series de reglas.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Dispone de varios módulos:

**El módulo de ambiente**, mide la temperatura del aire, la humedad del aire y la luz ambiente.

**El módulo sensor de presencia**, detecta movimiento de personas delante de él.

**El módulo de energía**, es la batería del stack. Puede funcionar y cargarse conectado a un adaptador microUSB o incluso a un ordenador. Tiene una batería que puede alimentar el stack independientemente. Su duración depende de la cobertura móvil y del tiempo establecido entre conexiones. Las baterías actuales pueden alimentar un stack durante un mes conectándose cada hora. Puedes poner más de un módulo de energía para alargar la vida.

**Módulo de conectividad**, utiliza la red móvil, por lo que funciona en cualquier sitio que funcione el móvil ya que tienen incorporada una SIM.

Mediante **Thinking Things Open**, se accede a la electrónica y al firmware, permitiendo a los desarrolladores, construir e incluso manufacturar masivamente su propio producto.

Este pack básico se puede adquirir en la tienda de Thinking Things de Telefónica **por un precio de 90 euros**, y en ese coste se incluye un año de conectividad que permite consultar y gestionar los módulos Thinking Things a través de una sencilla interfaz web que se permite dejar en manos de Telefónica o instalar otros servidores.

| Plataformas Hardware   | MICROCONTROLADORES   |  |  |   |   |  |  |
|------------------------|--|--|--|---|---|--|--|
|                        | Arduino  | Zolertia   | ARM  | .NET Gadgeteer                                  | Phidgets  | Electric Imp   | MediaTek                                     |
| Modelo                 | UNO R3   | Re-mote  | mbed LPC1768   | FEZ Raptor                                      | PhidgetSBC3   | imp001   | MediaTek Aster (MT2502)                      |
| Open Source            | SI   | SI   | NO   | SI  | NO  | SI   | NO   |
| Precio                 | 29,95 €  | 46,15 €  | 49 €   | 99,95 €   | 150 €   | 29,95 €  | 79 €   |
| Dimensiones placa      | 6,86 cm x 5,3 cm   | 5,7x3,5 cm.  | 44mm x 26mm  | 102 x 87 x 7.5 mm                               | N/A   | 5,1cm x 3,2mm x 0,6 cm   | 8,39 x 5,3 cm                                |
| Procesador             | ATmega328  | ARM Cortex-M3  | ARM Cortex-M3  | Atmel SAM9X35                                   | Freescall i.MX28                                    | Cortex-M3  | MediaTek Aster (MT2502)                      |
| Velocidad              | 16 MHz   | 96MHz  | 96MHz  | 400 MHz   | 454 MHz   |  | 260MHz                                       |
| Tamaño bus direcciones | 32-bits  | 32-bits  | 32-bits  | 32-bits   | 32-bits   | 32-bits  | 32-bits                                      |
| Reloj en tiempo real   | si   | si   | si   | No  |   |  |  |
| SO                     |  |  |  |   | Debian 7.0  |  |  |
| HDMI                   | NO   | NO   | NO   | NO  | NO  | NO   | NO   |
| RAM                    | 2 Kb   | 32 Kb  | 32 Kb  | 128 Mg  | 128 Mg  | 128 Kb   | 4 Mg   |
| FLASH Memory           | 32 Kb  | 512 Kb   | 512 KB   | 4 Mb  | 1 Gb  | N/A  | 16 Mg  |
| Conectividad           | NO   | IEEE 802.15.4<br>GSM   | Ethernet   | NO  | Ethernet  | 802.11 b/g/n   | GPRS, GSM,WIFI<br>Dual Bluetooth 2.1 y 4.0   |
| IDE                    | Herramientas Arduino   | ThingSquare  | Lightweight Online<br>Compiler<br>High level C/C++ SDK | Visual Studio                                   | C y C++<br>Java, .NET(Mono)<br>Python               | Squirrel   | ONE IDE                                      |
| GPU                    | NO   | NO   | NO   | NO  | NO  | NO   | NO   |
| USB                    | NO   | NO   | 1 USB 2.0  | NO  | 6 USB 2.0   | NO   | NO   |
| PROS                   | Facilidad de programación.<br>Comunidad numerosa<br>Mucha Información disponible | Compatibilidad con Sigfox<br>Conexión por Zigbee y sensor de<br>temperatura ya entregado | Facilidad de<br>programación.                          | Interface para conectar módulos<br>muy sencilla | Los componentes modulares y de conexión<br>sencilla | Se programa desde un<br>navegador y se compila y se<br>ejecuta el código directamente<br>en el Imp | Alto grado de flexibilidad                   |
| CONTRAS                | Poca potencia del<br>microcontrolador  | Existe poca documtación e información.   | No es open source                                      | Coste alto.                                     | No tiene posibilidad de autonomía propia            | Sólo orientado a un uso<br>profesional   | No es open source.<br>Kit de iniciación caro |

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  | <b>Fecha:</b> 13 de junio del 2015 |
|  | <b>Versión del documento:</b> V3.0  |                                    |

## Microprocesadores



<http://www.intel.es/content/www/es/es/do-it-yourself/galileo-maker-quark-board.html>

Galileo une la capacidad de **Linux con Arduino** en una misma placa.

La Intel Galileo es una placa open hardware basado en el procesador **Quark SoC X1000** de 32bits de Intel con una velocidad de 400MHz. Cuenta con conector Ethernet, un zócalo para microSD, puerto serie RS-232, USB Host, un puerto mini PCI Express y 8 MByte NOR Flash.

La placa de desarrollo ejecuta un sistema operativo Linux libre que contiene las librerías de software de Arduino, lo que le permite ofrecer una mayor escalabilidad y reutilizar el software ya existente, llamados “bocetos”. Además la placa es compatible con el IDE de Arduino y con las Arduino Shields. También incluye pines compatibles con una amplia gama de Shields Arduino Uno R3.

En cuanto al entorno de programación, utiliza cualquier versión de Visual Studio 2013 incluida la versión Express (aunque será necesario instalar Nuget Package Manager).

Dispone de múltiples interfaces de E/S, de las más extendidas en el sector, entre las que se incluyen ACPI, PCI Express, Ethernet 10/100 Mb, lector SD, puertos para dispositivos USB 2.0 y EHCI/OHCI USB, UART de alta velocidad, puerto serie RS-232, flashNOR programable de 8 MB y un puerto JTAG para mayor comodidad en la depuración de software.

Al igual que Arduino, Galileo es una plataforma **destinada a la educación** que cuenta con mayor potencia pero también con un precio superior.



<https://www.raspberrypi.org/>

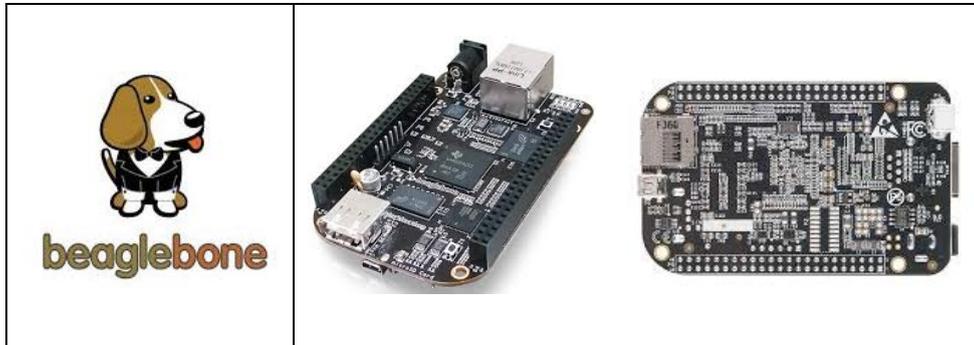
Raspberry Pi, es un miniordenador con el tamaño de una tarjeta de crédito, **desarrollado en Reino Unido por la Fundación Raspberry Pi**, con el objetivo principal de estimular la enseñanza de ciencias de la computación en las escuelas. Al ser un miniordenador podemos utilizarlo para desarrollar cosas bastante más complejas que con Arduino y puede utilizar lenguajes de programación de alto nivel como Python, C++ y Java.

Los pines GPIO (General Purpose Input Output) de la Raspberry PI nos permiten trabajar con electrónica de una forma fácil como lo podríamos hacer también con Arduino.

En la segunda revisión de la PI disponemos de dos pines +5V, cinco de GROUND (toma de tierra), dos de +3,3V, ocho pines GPIO de propósito general, cinco SP10, dos UART, y dos pines I2C1.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  | <b>Fecha:</b> 13 de junio del 2015 |
|  | <b>Versión del documento:</b> V3.0  |                                    |

Estos pines los podemos configurar y controlar directamente desde el intérprete de ordenes Bash, o desde lenguajes de programación como C o Python, gracias a las librerías GPIO que se han implementado para ellos.



<http://beagleboard.org/bone>

BeagleBone, es una plataforma hardware de **código abierto**, que consiste en un ordenador pequeño del tamaño de una tarjeta de crédito, donde se puede ejecutar un sistema operativo, como puede ser **Linux/Android 4.0**, es prácticamente un miniordenador donde se pueden ejecutar programas sobre estos sistemas operativos.

BeagleBone, está diseñado para funcionar a un nivel mucho más alto y tiene mucha más capacidad de proceso que Arduino. olo funciona con dispositivos de 3.3V

El BeagleBonees una gran combinación de parte de la flexibilidad de la interfaz Arduino con el procesador rápido y entorno Linux completo de la Raspberry Pi. Cuenta con varios puertos I/O y la **potencia de procesamiento para el análisis en tiempo real** proporcionada por un procesador ARM ® MHz AM335x 720, 256MB DDR2

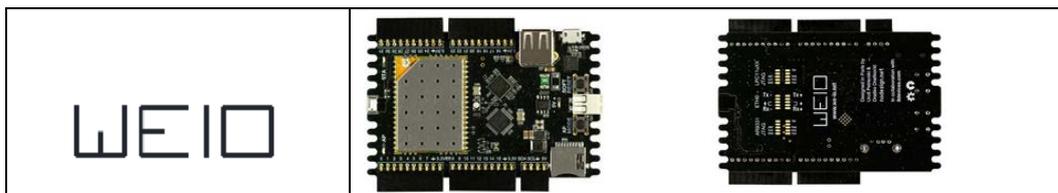


<http://cubieboard.org/>

Otra plataforma mini-ordenador, que permite conectar un disco duro SATA, incrementando enormemente las posibilidades de nuestro proyecto, asimismo, ofrece mayor capacidad de hardware respecto a sus competidores pero a un precio ligeramente superior.

Es de código abierto y dispone de un procesador **Allwinner A10** a 1000MHz, 1GB DDR3 a 480MHz de RAM, tarjeta gráfica Mali 400, salida HDMI y tarjeta Ethernet 10/100.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.<br><b>Versión del documento:</b> V3.0                      | <b>Fecha:</b> 13 de junio del 2015 |

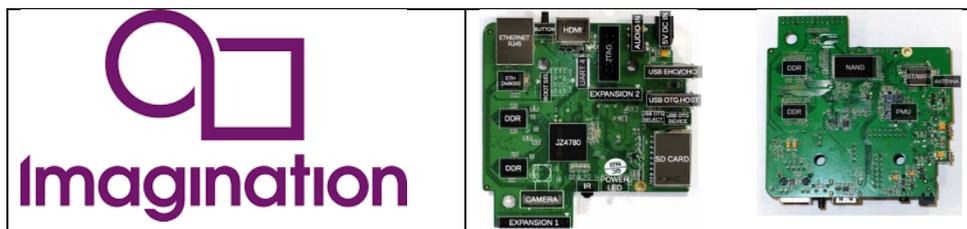


<http://we-io.net/hardware/>

WeIO es una plataforma hardware de **código abierto** para el Internet de las cosas con conectividad Wi-Fi. La placa está basada en un microprocesador Atheros AR9331 MIPS 24K Wireless SoC a 400 MHz y ejecuta la distribución Linux OpenWRT, también posee una MCU NXP LPC que optimiza los recursos de E / S y da soporte para las E/S analógicas. Permite programar a través de un navegador web directamente usando HTML5 o Python puesto que no es necesario instalar ningún IDE porque ya viene integrado en la propia placa, utiliza como editor utiliza ACE que está escrito en Javascript.

WeIO presenta una tecnología novedosa, denominada **WebSocket**, que permite enviar notificaciones desde el mundo físico a todos los clientes (teléfonos, tabletas, PCs) en tiempo real, de manera que podemos estar al tanto de todos los acontecimientos que están ocurriendo.

Todas las nuevas conexiones a WeIO se representan directamente en la consola con informaciones adicionales, como la dirección IP, el tipo y la versión del operativo sistema.



<http://store.imgtec.com/us/product/mips-creator-ci20/>

Plataforma hardware de código abierto, es una placa destinada a fabricantes, estudiantes y desarrolladores, está basada en una CPU MIPS32 de doble núcleo y la GPU PowerVR SGX540 GPU, el SoC Ingenix JZ4780 trabaja a 1.2 GHz e incorpora el Sistema Debian 7 ya preinstalado, aunque también es posible también descargar e instalar Android 4.4, Gentoo y Yocto.

Ofrece una resolución por HDMI de 2K y conectividad 10/100M Ethernet, 802.11 b/g/n, Bluetooth 4.0.

De todas las plataformas presentadas, las más conocidas son Arduino y Raspberry Pi, seguramente porque fueron las primeras plataformas en aparecer en sus respectivas categorías, pero basándonos en el estudio realizado, hay otras plataformas compiten de tú a tú e incluso las superan.

A este respecto, resaltaría la creada por Intel, **Galileo**, cuyas especificaciones técnicas superan en mucho a las ofrecidas por Arduino, sobre todo en velocidad de proceso, además al ser totalmente compatible con dicha plataforma también obtenemos uno de los puntos fuertes de Arduino que es su gran comunidad de usuarios y la extensa documentación técnica disponible. Todo ello a un precio muy equilibrado.

También destacaría la placa **Creator CI20** de Imagination Technologies, con sus 1,2 Ghz de velocidad de procesador y conectividad inalámbrica de serie, es una opción muy destacable con una buena relación calidad precio.

Para finalizar y a modo resumen, se adjunta una tabla comparativa con los modelos comentados

| Plataformas Hardware   | MICRO-ORDENADORES  |  |                                       |  |  |   |
|------------------------|--|--|---------------------------------------|--|--|---|
|                        | Galileo  | Raspberry Pi 2   | BeagleBone                            | Cubieboard   | WEIO   | Imagination Technologies  |
| Modelo                 | Galileo Gen 1  | Model B  | <b>BeagleBone Black</b>               | Cubieboard1  | Atheros AR9331   | Creator CI20  |
| Open Source            | SI   | SI   | <b>SI</b>                             | SI   | SI   | SI  |
| Precio                 | 35 €   | 35 €   | 89 €                                  | 49 €   | 69 €   | 65 €  |
| Dimensiones placa      | 10cm x 7 cm  | 8,7 x 5,7 cm   | 8,62 cm x5,3 cm                       | 10x6cm   | 91 x 68 mm   | 9 cm x 9,5 cm   |
| Procesador             | Intel® Quark X1000                                       | Quad-Core Cortex A7  | Cortex-A8 + 2xPRU(200Mhz)             | ARM Cortex A8  | AR9331 y Coprocesador LPC11xx ARM Cortex-M0  | Ingenic JZ4780  |
| Velocidad              | 400MHz   | 900 MHz  | 1000 MHz                              | 1000 Mhz   | 400 MHz  | 1.2 GHz   |
| Tamaño bus direcciones | 32-bits  | 32-bits  | 32-bits                               | 32-bits  | 32-bits  | 32-bits   |
| Reloj en tiempo real   | si   | No   | si                                    |  | no   |   |
| SO                     | Yocto<br>VxWorks (RTOS)<br>Microsoft Windows             | NOOBS,RASPBIAN<br>Snappy, Ubuntu<br>OPENELEC,OSMC<br>PIDORA,RISC OS  | Angstrom,<br>Ubuntu                   | GNU/Linux, Berryboot,<br>Ubuntu 12.04,<br>OpenElec,<br>Android con XBMC incluido | OpenWRT  | Debian, Android,<br>Gentoo, Yocto,<br>Arch, OpenWRT, Buildroot  |
| HDMI                   | NO   | SI   | SI                                    | SI   | NO   | SI  |
| RAM                    | 512KB en-chip SRAM, y 256MB DRAM                         | 1 Gb   | 512 Mg DDR3                           | 1Gb DDR3   | 64 Mb  | 1 Gb  |
| FLASH Memory           | 8 MB   | No permanente en la placa  | N/A                                   | 4 Gb   | 16 Mb  | 8 Gb  |
| Conectividad           | Ethernet   | Ethernet   | Ethernet                              | Ethernet   | Ethernet<br>802.11 b/g/n   | Ethernet<br>802.11 b/g/n<br>Bluetooth 4.0   |
| IDE                    | EPython, C, C++, Java, Scratch, Ruby, Arduino ID         | IDLE, Scratch, Squeak/Linux  | Python, Scratch, Squeak, Cloud9/Linux | Android SDK,Sun JDK, Eclipse,NDK....   | ACE<br>HTML5<br>Python   | Android SDK,Sun JDK, Eclipse,NDK...   |
| GPU                    | NO   | VideoCore IV 3D  | PowerVR SGX530(200 MHz)               | Mali 400   | NO   | PowerVR SGX540  |
| USB                    | 2 USB 2.0  | 4 USB 2.0  | 1 USB 2.0                             | 1 MINI USB 2.0<br>2 USB 2.0  | 6 USB 2.0  | 2 USB 2.0   |
| PROS                   | Compatible con Arduino<br>Potencia a un precio razonable | Gran comunidad y proyectos basados en él<br>Accesorios de fácil disponibilidad                                   | Gran capacidad de procesamiento       | Incorpora una conexión SATA nativa   | IDE integrado en la propia placa.<br>Tecnología <b>WebSocket</b> , que permite enviar notificaciones desde el mundo físico a todos los clientes(teléfonos, tabletas, PCs) en tiempo real | Resolución por HDMI de 2K, conectividad 10/100M Ethernet, 802.11 b/g/n, Bluetooth 4.0 integrados.   |
| CONTRAS                |  | No es capaz de alimentar discos duros por USB.<br>Comparte un mismo bus para los 2 puertos USB y la conexión LAN | Comunidad reducida.                   | Comunidad reducida   | Comunidad reducida<br>Precio alto  | Quizás para algunos proyectos no sean necesarias todas las opciones que traen serie y no existe posibilidad de escoger una placa más básica |

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Plataformas Software



<https://xively.com/>

Xively se define como una “Plataforma como un Servicio” (PaaS), funciona bajo arquitectura cloud pública para desarrollar y gestionar productos comerciales y soluciones para el Internet de las Cosas, originalmente era conocida Pachube, más tarde pasó a Cosm y finalmente a Xively, pertenece a la empresa Logmein.

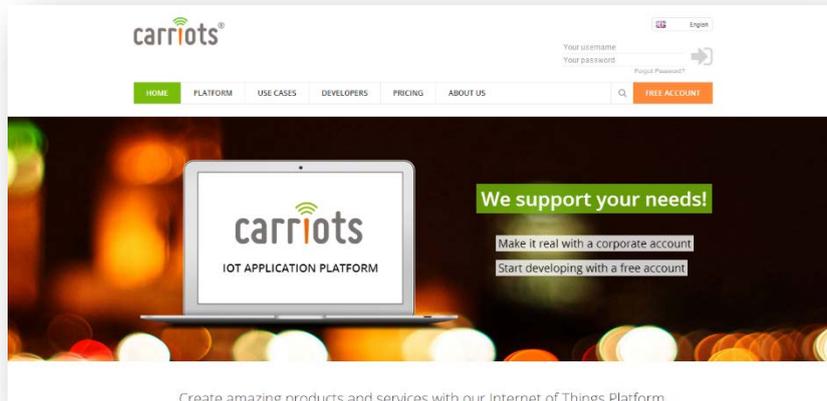
Consiste en un servicio online desarrollado específicamente para el Internet de las Cosas y permite publicar los datos recogidos por distintos sensores (como pueden ser sensores de humedad, temperatura, gases, luminosidad, radiación, etc.) mediante gráficas en tiempo real y widgets.

La plataforma es propietaria, pero las librerías son de código abierto (bajo la licencia BSD-3) que aprovechan la API basada en los estándares MQTT, WebSockets y HTTP

Características:

- Un intuitivo Developer Workbench centraliza todo lo necesario para crear, depurar y desplegar servicios conectados.
- Un completo centro de desarrollo con guías de API, tutoriales, vídeos y bibliotecas que soportan cientos de plataformas, millones de gateways y miles de millones de dispositivos.
- Un servicio de provisión y gestión altamente escalables que pueden apoyar la comunicación “one to one” de millones de dispositivos
- Las fuentes de datos definidas como públicas estarán indexadas y disponibles en Google y Bing.
- Tiene soporte heterogéneo de dispositivos
- La API está basada en una arquitectura de desarrollo REST.
- Las bibliotecas aprovechan la API basada en estándares sobre MQTT, WebSockets y HTTP para hacer la conexión a la Internet de las cosas de forma simple, intuitiva y rápida.
- Soporta múltiples formatos de datos, incluyendo JSON, XML y CSV.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



<https://www.carriots.com/>

Carriots es una empresa española que ofrece una plataforma en la nube que actúa como servicio (PaaS) diseñado para la Internet de las Cosas (IoT) y proyectos máquina a máquina (M2M).

Características:

- Permite a cualquier usuario conectar y gestionar hasta 10 dispositivos de forma gratuita, recopilar datos, y construir Apps a través de código Groovy junto a las librerías de la SDK.
- Los datos de los diferentes dispositivos pueden ser enviados a la plataforma en XML o JSON, y utilizar protocolos seguros de comunicación como HTTPS además de otras medidas adicionales de seguridad como cifrado personalizado o checksums.
- Dispone de un módulo de gestión de dispositivos que junto a los protocolos de comunicación en los dos sentidos permite controlar los dispositivos por completo, como por ejemplo comprobar el estado de tus dispositivos, cambiar configuraciones o actualizar el firmware.
- Capacidad de gestión de proyectos.
- Integración con otros sistemas, gracias a su API REST y el sistema de acceso desde los scripts de Groovy. Incluso permite crear crear panel de control personalizado con gestión completa de los dispositivos.
- Ofrece herramientas para facilitar el desarrollo, como un sistema de trazas (logs), un terminal para ejecutar código Groovy y el acceso directo y por programación a las alarmas.
- Prácticamente no hay limitación de cuánto pueden escalar los proyectos tanto en volumen como en rapidez.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



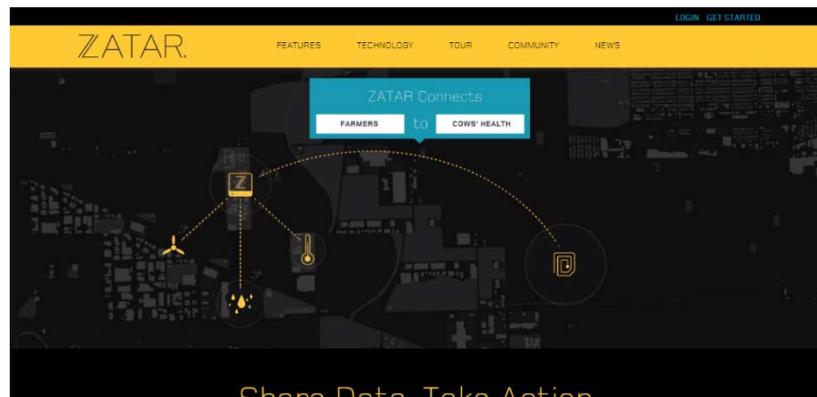
<http://es.ptc.com/axeda>

Axeda ofrece una completa plataforma propietaria basada en la nube para la gestión de productos IoT y M2M.

#### Características:

- Es necesario el uso de un gateway para el manejo de los diferentes dispositivos y también utiliza agentes software
- Para la conexión de los dispositivos, utiliza agentes software y un Gateway.
- Capacidad para procesar, transformar, y organizar los datos de las máquinas y sensores.
- Herramientas de desarrollo basadas en la nube que permiten que incluyen la gestión de datos, un motor de scripting, SDKs y servicios web para acceder los datos de las aplicaciones en la nube.
- Gestión, las aplicaciones web permiten monitorizar de forma remota, gestionar el servicio, y controlan los productos y dispositivos conectados.
- Encriptación SSL/TLS.
- Soporta el protocolo MQTT
- API abierta; y servicios REST y SOAP
- Comunicación bidireccional cloud-to-cloud
- Alertas y notificaciones

|   |  |   |
|---|--|---|
| <p><b>TFC</b></p> <p><b>Integración redes telemáticas</b></p> | <p><b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas</p> |   |
|   | <p><b>Tipo de documento:</b> Memoria.</p>  | <p><b>Fecha:</b> 13 de junio del 2015</p> |
|   | <p><b>Versión del documento:</b> V3.0</p>  |   |



<http://www.zatar.com/>

Es una infraestructura basada en la nube, que tiene la capacidad de detectar dispositivos conectarlos a internet, es una plataforma open source, también permite a los diferentes usuarios, compartir datos y colaborar entre ellos de una forma sencilla

Características:

- Interface optimizada para tablets
- No requiere de desarrollo.
- Permite generar perfiles específicos para cada dispositivo
- Visualización de datos en tiempo real
- Trabaja con cualquier tipo de dispositivos
- La API está basada en REST y JSON
- Escalable
- Posibilidad de personalizar las aplicaciones
- Seguridad HTTPS y certificados SSL.
- Intercambio de datos con encriptación RSA.
- Alta eficiencia de rendimiento de los dispositivos en la red mediante el uso del protocolo CoAP.



<http://www.thingworx.com/>

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

ThingWorx facilita la rápida creación, simplificada de extremo a extremo de aplicaciones inteligentes para la agricultura, las ciudades, rejilla, el agua, la construcción y la telemática. Las industrias tradicionales se transforman y equipadas con conectividad de hoy en día y las soluciones más inteligentes a través de dispositivos conectados que proporcionan amplia recopilación de datos y análisis para la toma de decisiones basadas en datos. ThingWorx reduce el tiempo, el costo y los riesgos de la construcción de aplicaciones M2M y la IO. Los usuarios pueden construir interfaces móviles integrales con codificación cero, aprovechar ThingWorx Compositor para el modelado de aplicaciones, así como cuadros de mando en tiempo real y espacios de trabajo colaborativos – todo con la escalabilidad para soportar millones de dispositivos.

### Características principales:

- Diseño basado en modelos con ThingWorx Compositor
- SQUEAL (Búsqueda, consulta, análisis) para la inteligencia basada en búsquedas
- Diseño completo, tiempo de ejecución y entorno de inteligencia
- Crear cuadros de mando en tiempo real y espacios de trabajo colaborativos
- Crear interfaces móviles sin necesidad de programación
- Motor de ejecución dirigida por eventos
- 3-dimensional de almacenamiento
- Soporta los requisitos de escala para millones de dispositivos
- Soporta nubes de dispositivos 3ª parte, conexiones de red directos



<http://www.openremote.org/display/HOME/OpenRemote>

Una solución de middleware de código abierto para la Internet de las cosas, OpenRemote le permite integrar cualquier dispositivo – sin importar la marca o protocolo – y diseñar cualquier interfaz de usuario para iOS, Android o navegadores web.

El uso de herramientas de diseño basados en la nube de OpenRemote para el desarrollo de soluciones completamente personalizadas, las actualizaciones son aerodinámico, lo que significa sus dispositivos son,

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

literalmente, a prueba de futuro.

### Características principales:

- Se integra una variedad de protocolos
- Soluciones personalizadas que se adaptan a sus necesidades
- De cuentas únicas soluciones para marca totalmente
- Nube basada en herramientas de diseño

### Costo:

- Diseñador: GRATIS
- Diseñador profesional: 150 €
- Diseñador profesional con eBox: €350
- Diseñador profesional con eBox con libro: €375



<http://www.nimbits.com/index.jsp>

Nimbits es APAAS que se puede descargar en un Raspberry Pi, Servidor Web, Amazon EC2 o Google App Engine. La plataforma se utiliza para el desarrollo de soluciones de hardware y software que pueden conectarse a la nube o entre sí, la tala y recuperar grandes cantidades de datos desde dispositivos físicos, desencadenando eventos o alertas, o iniciar el análisis complejo.

### Características principales:

- Descarga servidores Nimbits en chips, servidores o la nube
- Plataforma de código abierto
- Disparadores de eventos y alertas
- Grabar y proceso geo y datos con fecha y hora
- Construir previsto Google App Engine y Linux Sistemas
- Compatible con la mayoría de los servidores J2EE (Apache Tomcat, Espolón Server)

**Coste:** GRATIS

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



<http://www.marvell.com/solutions/internet-of-things/>

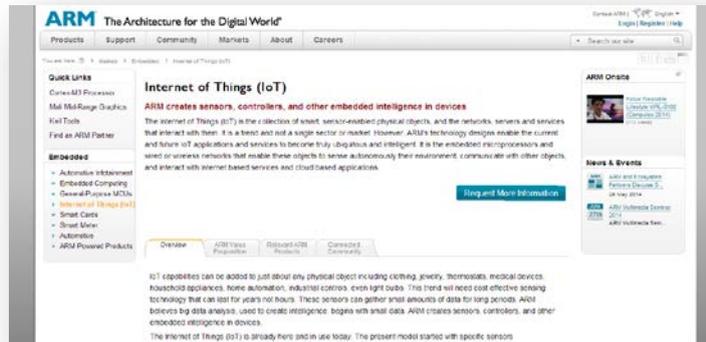
Marvell agiliza el proceso de llevar los dispositivos conectados al mercado con soluciones flexibles y rentables para los OEMs y ODMs con soluciones tanto de hardware como de software. De Marvell Wi-Fi microcontrolador Internet de las Cosas Plataforma es una plataforma de hardware / software que permite la conectividad sin fisuras con los clientes móviles, servicios en la nube y otros dispositivos. Avastar Wi-Fi SoC Marvell disponen de firmware que realiza gran parte del manejo del protocolo Wi-Fi necesario y el software Easy-Connect es una pila de software rico en funciones para el desarrollo rápido a un costo menor.

### Características principales:

- Sistema operativo FreeRTOS
- Marco de gestión de energía para aplicaciones de baja potencia.
- Habilitar streaming de audio y visual en cualquier dispositivo IO
- Componentes de middleware de redes avanzadas
- Incorporado en las bibliotecas para acelerar el desarrollo
- Construido en la nube y las integraciones de los clientes móviles
- Modelos de comunicación flexibles para P2P, de dispositivo a dispositivo
- Apoyo Over-the-air de actualización de firmware con la codificación mínima
- Soporte de la plataforma de desarrollo de Linux, Windows y MacOS

Coste: N/A

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



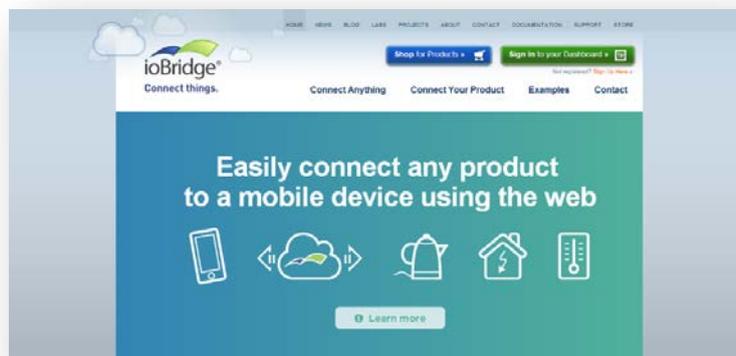
<http://www.arm.com/markets/internet-of-things-iot.php>

ARM crea sensores, controladores, microprocesadores y otros tipos de inteligencia integrada para la IO, permitiendo objetos ordinarios para detectar automáticamente las variables en el entorno, comunicarse con otros dispositivos y objetos e interactuar con aplicaciones basadas en la nube y otras redes. ARM licencias de tecnología a varios socios, permitiendo a las organizaciones de agregar valor y diferenciarse de sus competidores.

#### Características principales:

- Procesadores Cortex
- Desarrollar alto rendimiento, plataformas de bajo costo
- Procesadores de manejar múltiples canales de E / S y las normas de protocolo
- Herramientas Keli
- Bibliotecas de baja fuga de ultra PIPD
- Los procesadores gráficos y motores de vídeo
- Las tarjetas inteligentes, contadores inteligentes y otros productos integrados
- Periféricos y herramientas de desarrollo; DS-5 Development Studio

Coste: N/A



<https://www.iobridge.com/>

ioBridge permite conectar cualquier producto a un dispositivo móvil a través de la web con la ioBridge plataforma RealTime.io la IOy la tecnología RealTime.io Iota, Además acelera el tiempo de comercialización y reduce el coste por producto conectado.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

### Características principales:

- Iota disponibles como módulos o firmware
- Instalación sencilla
- Web API para aplicaciones personalizadas
- Integración con los sistemas de propiedad o de terceros
- Libre interfaz web y usuario del panel configurable
- Los clientes pueden crear cuadros de mando sin necesidad de escribir software
- Soporta 100.000 a 1.000.000 de dispositivos por servidor
- No hay suscripciones necesarias para los usuarios finales
- Puerta de entrada a base de Iota para la ejecución off-the-shelf

### Coste:

- Iota Wi-Fi kit de evaluación: \$ 199
- IO-204 Web Gateway Kit de evaluación: \$ 199
- Cuadros de mando personales gratuitos, Android y aplicaciones de iOS



<http://go.sap.com/solution/platform-technology.html>

Las soluciones de IO de SAP facilitan la conectividad y la comunicación multidireccional para que los usuarios puedan interactuar con sus dispositivos de nuevas. La transformación de las operaciones de servicio de campo y gestión de activos a distancia, proporcionando visibilidad de la cadena de suministro y predecir y subsanar cuellos de botella logísticos son sólo algunos de los desafíos que resuelve el mantenimiento de SAP remoto.

### Características principales:

- Integra los datos en tiempo real con análisis de distribución
- Seguimiento y monitoreo de activos
- Actualiza el software de forma remota
- En tiempo real, visibilidad de extremo a extremo en carga, contenedores, mercancías enviadas
- Identifica y corrige los cuellos de botella logísticos
- El control del tráfico en tiempo real
- Adquirir conocimientos sobre la ejecución por menor
- Responder a las señales de demanda up-to-the-minute
- Utiliza algoritmos predictivos para la optimización del producto-ubicación

Coste: N/A

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



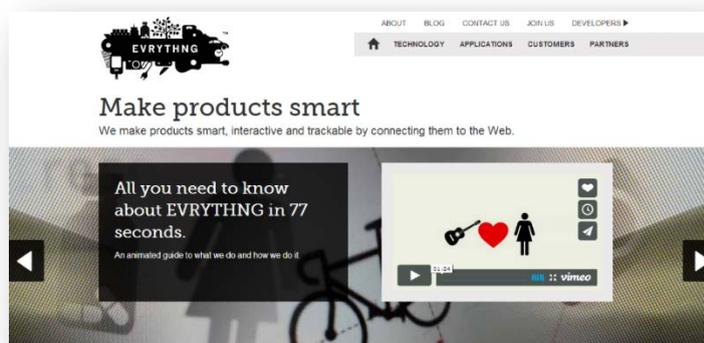
<http://www.arkessa.com/>

Proporciona soluciones de conectividad y gestión de la IO, Arkessa permite la monitorización, gestión y control de los dispositivos conectados con el mismo nivel de funcionalidad aerodinámica como si los dispositivos se conectan directamente a su escritorio a través de su plataforma de mosaico la IO. Las soluciones empresariales de Arkessa permiten a las empresas aprovechar el IoT para desarrollar nuevas fuentes de ingresos, mejorar la satisfacción del cliente y proporcionar valor añadido a partir de dispositivos remotos. La plataforma de Arkessa se demuestra a través de más de 500 redes móviles que abarcan más de 200 países de todo el mundo.

#### Características principales:

- Disponible como una solución PaaS, arrendado o licenciado
- Alojado en la nube, en el centro de datos o en las instalaciones
- Compatible con los nuevos dispositivos IO o dispositivos M2M legado
- Control en tiempo real y datos consolidada
- Portal de gestión de patrimonio único para la gestión de dispositivos
- Oracle certificada como Exastack Ready
- Integrar datos de la máquina con el CRM, ERP, los grandes datos y los sistemas de análisis
- Datos totalmente encriptada cumple con las regulaciones internacionales
- Soporta GSM, GPRS, 3G, 4G, y el satélite

**Coste:** N/A



<https://evrythng.com/>

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

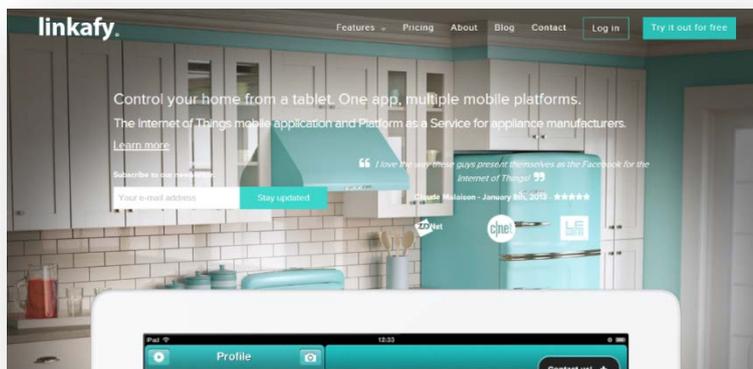
Ofrece a cualquier producto u objeto su propia identidad digital activa con el motor evrythng, Dando API, acceso seguro y una presencia en línea persistente. Con evrythng, los clientes pueden personalizar los productos con un contenido único, generado por los usuarios digitales, las marcas pueden recompensar a los clientes por las acciones o las compras, a su vez productos en objetos virtuales para permitir el intercambio social para impulsar las ventas, y el análisis de acceso en tiempo real para capitalizar y mejorar las características y capacidades que demandan los consumidores. Una plataforma de extremo a extremo, evrythng ofrece todo lo necesario para la fabricación de productos inteligente, interactivo y rastreado.

**Características principales:**

- Servir a millones de identidades digitales Activo
- Persistente presencia digital único, para cualquier objeto físico
- El intercambio de datos de objetos con CRMs o ERPs
- Soporta una gran variedad de tecnologías
- Control conectada productos directamente desde la web
- URL corta único y reprogramable para cada objeto
- Potente motor de reglas para aplicaciones personalizadas
- Organizar productos con colecciones, etiquetas o lotes
- Búsqueda Funcional
- HTTPS y tokens de API-cripto seguro
- Administrar miles de millones de objetos sin sacrificar el rendimiento

**Coste:**

Portal Desarrollador: registro Beta GRATIS



<http://www.linkafy.com/>

Linkafy faculta a la casa inteligente con un PaaS para fabricantes de electrodomésticos para introducir fácilmente los aparatos conectados al mercado. Linkafy sirve como una única aplicación que puede controlar todos los aparatos en el domicilio del usuario final o el medio ambiente, en lugar de la compleja multitud de aplicaciones en general, necesarios para el funcionamiento de varios dispositivos de diferentes fabricantes. Los fabricantes utilizan API o SDK de Linkafy para integrar con los dispositivos IoT-, permitiendo a los clientes controlar, supervisar y programar tareas en varios electrodomésticos desde una interfaz central.

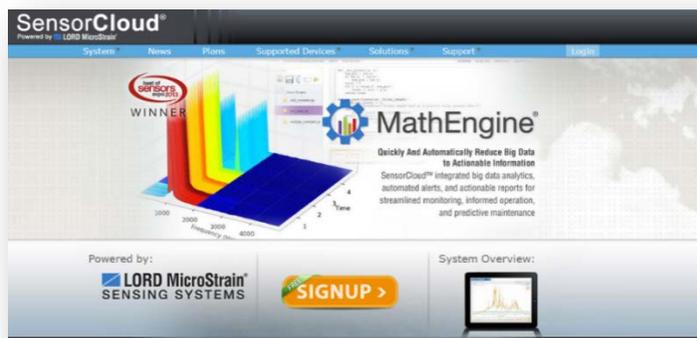
**Características principales:**

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

- Ver notificaciones de electrodomésticos en una interfaz central
- Maneje notificaciones y programar tareas de forma remota
- Comparte control remoto y monitoreo con otros usuarios
- Supervisar el consumo de servicios públicos con el tiempo
- Notificaciones basados en eventos y disparadores
- Configure los horarios diarios para la supervisión de seguridad u otras acciones
- Funciona en todas las plataformas móviles

**Coste:**

- LIBRE: 1 electrodoméstico, 10 casos
- Micro Fabricante: \$ 79 / mes – 1 aparato, 1.000 casos
- Pequeño Fabricante: \$ 349 / mes – 3 aparatos, 5.000 instancias
- Grande Fabricante: \$ 789 / mes – 5 aparatos, 10.000 instancias



<http://www.sensorcloud.com/>

SensorCloud es una solución de Jehová MicroStrain, una empresa que produce, transductores integrados inteligentes, sensores y redes de sensores. SensorCloud ofrece análisis Big Data integrados, alertas automáticas e informes accionables para el mantenimiento predictivo y seguimiento ágil de dispositivos conectados. Un almacenamiento de datos única, la virtualización y la plataforma de gestión remota, SensorCloud apoya cualquier dispositivo de red, sensor o sensor a través de una API OpenData.

**Características principales:**

- Almacenamiento de datos ilimitado
- Fiabilidad Triple-redundante
- Recolectar y preservar los flujos de datos del sensor a largo plazo
- Tiempo de visualización serie de herramientas / gráfica
- Fácil de navegar a través de grandes cantidades de datos
- Analizar a puntos singulares de interés
- Desplegar rápidamente de procesamiento y análisis de datos de aplicaciones
- Flexible SMS / email scripting alerta

**Coste:** GRATIS: 3 millones de puntos de datos / mes

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



<https://thingspeak.com/>

Es una plataforma de datos abierta para el Internet de las cosas, ThingSpeak permite la recogida de datos en tiempo real con una API abierta y sofisticadas capacidades de procesamiento de datos y análisis. Con plugins, visualizaciones de datos y la capacidad de integrar los datos con una variedad de plataformas de terceros, sistemas y tecnologías, incluyendo otras plataformas líderes de la IO como ioBridge y Arduino, ThingSpeak es el complemento perfecto para un sistema empresarial existente para aprovechar el Internet de las Cosas.

**Características principales:**

- API abierta
- Integración con Raspberry Pi, Arduino, ioBridge / RealTime.io y más
- Datos de localización geográfica
- Procesamiento de datos y visualizaciones
- Mensajes de estado del dispositivo
- Integración con redes sociales
- Integraciones con el análisis de datos y servicios de terceros
- ThingSpeak API impulsado por Phusion Empresa de Pasajeros

**Coste:** GRATIS

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |



<http://www.xobxob.com/>

Xobxob permite añadir cualquier proyecto a la IO con servicio en la nube.

Los dispositivos conectados pueden enviar y recibir mensajes de Xobs, que funcionan como pequeños buzones. Al compartir Xobs, los dispositivos pueden comunicarse e interactuar entre sí, lo que permite acciones como el control remoto de la puerta del garaje usando un teléfono inteligente. Diseñado para ser simple con la programación mínima requerida, XobXob ofrece una amplia gama de proyectos de la muestra y las herramientas para simplificar el proceso.

#### **Características principales:**

- Conecta dispositivos físicos y virtuales
- API REST
- Funciona con cualquier plataforma conectada a Internet
- Bibliotecas Arduino
- Actualmente en fase Alfa
- Bibliotecas adicionales

Coste: GRATIS

De entre todas ellas me gustaría destacar a **Zatar**, por su filosofía de código libre, y porque promueve la colaboración y compartición de datos entre sus usuarios, además de que permite conectar cualquier clase de dispositivo.

Estas cualidades son bajo mi punto de vista indispensables para que al final tenga éxito el paradigma de la IoT, puesto que otras opciones propietarias que sólo permiten la conectividad de sus dispositivos crean pequeñas "islas" que acaban con el espíritu de la conectividad total entre los objetos.

| Plataformas     | COMPARATIVA                         |          |                 |   |      |                                    |                         |
|-----------------|-------------------------------------|----------|-----------------|---|------|------------------------------------|-------------------------|
|                 | Soporte heterogéneo de dispositivos | Tipo     | Cloud           | Open Source   | REST | Acceso datos                       | Servicio descubrimiento |
| Xively          | SI                                  | PaaS     | SI              | Plataforma propietaria. Las librerías son de código abierto | SI   | Acceso seguro                      | SI                      |
| Carriots        | SI                                  | PaaS     | SI              | No  | SI   | Acceso seguro                      | NO                      |
| Axeda           | Necesita Gateway                    | M2M PaaS | SI              | No  | SI   | Basado en privilegios de usuarios  | NO                      |
| Zatar           | SI                                  | PaaS     | SI              | SI  | SI   | Acceso seguro                      | SI                      |
| ThingWorx       | SI                                  | M2M PaaS | SI              | No  | SI   | N/A                                | No                      |
| Open Remote pro | SI                                  | Hub      | NO              | LGPLv3  | NO   | Basado en privilegios de usuarios  | SI                      |
| Nimbits         | SI                                  | Servidor | SI/Centralizado | Licencia Apache   | SI   | 3 niveles                          | NO                      |
| Marvell         | SI                                  | PaaS     | SI              | No  | SI   | Acceso seguro                      | NO                      |
| ARM Mbed        | Necesita Gateway                    | M2M PaaS | SI/Centralizado | NO  | CoAP | A elección del usuario             | NO                      |
| ioBridge        | SI                                  | PaaS     | SI              | No  | SI   | Acceso seguro                      | SI                      |
| SAPTechnology   | SI                                  | PaaS     | SI              | SI  | SI   | Acceso seguro                      | SI                      |
| Arkessa         | SI                                  | M2M PaaS | SI              | NO  | N/A  | Seguridad similar a la de Facebook | NO                      |
| EvryThng        | SI                                  | M2M PaaS | Centralizado    | NO  | SI   | N/A                                | NO                      |
| Linkafy         | SI                                  | PaaS     | SI              | No  | SI   | Acceso seguro                      | SI                      |
| SensorCloud     | No                                  | PaaS     | SI              | NO  | SI   | N/A                                | NO                      |
| ThingSpeak      | SI                                  | Servidor | SI/Centralizado | GNU GPLv3   | SI   | 2 niveles                          | Limitado                |
| XobXob          | SI                                  | PaaS     | SI              | SI  | SI   | Acceso seguro                      | SI                      |

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Diseño y desarrollo prototipo IoT

### Descripción del prototipo

La segunda parte de este proyecto se trata de poner en práctica y trasladar varios de los conceptos e ideas vistas en la primera parte del estudio sobre la IoT.

Para ello, se realiza un sencillo diseño donde la plataforma hardware elegida es Arduino. Los motivos para esta elección es que a su filosofía de código abierto hay que sumarle las siguientes características:

- Está dirigido al desarrollo de prototipos y la educación.
- Precio asequible de la mayoría de sus componentes.
- Entorno de programación simple y directo.
- Destinado tanto a principiantes como a usuarios avanzados.
- Numerosa documentación y tutoriales disponibles de forma gratuita.
- Cuenta con una gran comunidad de usuarios, foros y webs dedicadas.

Se trata de emular un **entorno doméstico** controlado desde una página web, donde seremos capaces de conectar y desconectar objetos, así como gracias a los datos recogidos a través de un sensor de temperatura y una fotorresistencia que se almacenan en una base de datos, lograr realizar estadísticas en tiempo real. La decisión de no utilizar ninguna plataforma software IoT, y realizar su desarrollo está motivado principalmente para que de esta forma se pueda ver su integración y comprender mejor su funcionamiento.

Aunque se trata de un prototipo, con la ayuda de las impresoras 3d, un módulo WIFI y la gran variedad de sensores disponibles hace muy factible su comercialización, ya que a pesar del auge de esta tecnología, en España no es un mercado que esté por el momento muy saturado, así por ejemplo, un cliente potencial son los compradores de una vivienda de nueva construcción, donde se puede ofrecer como un valor añadido y la posibilidad de personalizarlo en base a las pretensiones del cliente.

### Componentes utilizados y costes.

La arquitectura de la aplicación se sustenta equitativamente en una parte software y en una parte hardware, el software utilizado no lleva ningún coste, pero sí que habría que sumar el tiempo dedicado a la personalización y programación de la aplicación, aproximadamente 50 €/hora.

#### Hardware

| COMPONENTES                            | PRECIO        |
|--|---------------|
| Arduino UNO R3                         | 11 €          |
| Ethernet W5100 SHIELD para Arduino UNO | 12 €          |
| Protoboard                             | 3 €           |
| Sensor de temperatura analógico LM35DZ | 2,55€         |
| Fotorresistencia GL5549                | 0,40 €        |
| 5 Resistencias 220 omh                 | 0.60 €        |
| 4 leds Diodos LED 5mm                  | 0.25 €        |
| <b>TOTAL</b>                           | <b>29.8 €</b> |

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  | <b>Fecha:</b> 13 de junio del 2015 |
|  | <b>Versión del documento:</b> V3.0  |                                    |

## Descripción componentes

### Arduino UNO R3



Es el componente principal del prototipo. Consta de una placa electrónica basada en el microprocesador Atmega328.

Cuenta con 14 pines digitales de entrada / salida (de los cuales 6 pueden ser utilizados como salidas PWM), 6 entradas analógicas, una conexión USB, un conector de alimentación, un header ICSP, y un botón de reinicio.

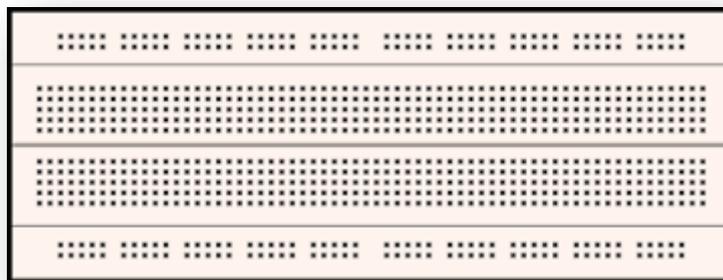
### Ethernet W5100 SHIELD



Basado en el chip de ethernet Wiznet W5100 cuenta con funcionalidades de IP tanto para TCP como UDP, posee un conector RJ45 estándar para ethernet y soporta hasta 4 conexiones simultáneas, incluye un slot para tarjetas micro-SD.

El Arduino UNO utiliza los pines digitales 11, 12 y 13 (SPI) para comunicarse con este shield y recurre a la librería Ethernet para escribir programas que se conecten a Internet.

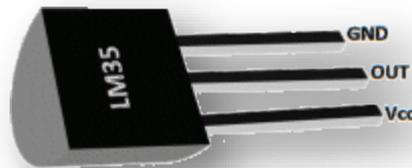
### Protoboard



Es una tarjeta para prototipos, muy utilizada para ensamblar y probar circuitos electrónicos sin la necesidad de realizar soldaduras.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  | <b>Fecha:</b> 13 de junio del 2015 |
|  | <b>Versión del documento:</b> V3.0  |                                    |

### Sensor de temperatura analógico LM35

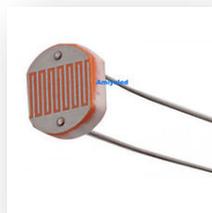


El LM35 es un sensor de temperatura, que viene calibrado con una precisión de 1°C y es capaz de medir entre -55°C y 150°C. Tiene 3 pines de conexión. Uno es para la alimentación, otro para masa (0 Voltios) y otro para la salida de datos.

Para realizar la conversión del valor analógico leído a temperatura en grados centígrados, debemos utilizar la fórmula la siguiente:

$$\text{Valor } ^\circ\text{C} = (5.0 * \text{analogRead}(\text{PIN\_SENSOR}) * 100.0) / 1024.0$$

### Fotorresistencia GL5549



Es un componente cuya resistencia varía sensiblemente con la cantidad de luz percibida. La relación entre la intensidad lumínica y el valor de la resistencia no es lineal, y su comportamiento es el siguiente:

- Más luz = menor resistencia eléctrica
- Menos luz = mayor resistencia eléctrica

### Resistencia 220 Ohm



|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Como Arduino maneja 5 voltios y para encender el LED necesitamos entre 5 y 20 mA, necesitamos conectar una resistencia con el fin de no quemar el LED.

Para elegir la resistencia, aplicamos la ley de ohm, donde el voltaje es igual a la corriente por la resistencia

$$V = I * R$$

La diferencia de potencial en los leds es aproximadamente 2 voltios.

$$\text{Entonces: } R = V/I \quad R = 5v - 2v / 15mA \quad R = 3 v / 0.015 A \quad R = 200 \text{ ohms}$$

La resistencia que debemos ocupar para encender el led y no sobrecargarlo debe estar alrededor de 200 ohms.

## LED



Es un componente optoelectrónico pasivo y, más concretamente, un diodo que emite luz. La pata más larga es el ánodo positivo y la más corta es el cátodo negativo.

## Software

- **IDE Arduino**, es el entorno de desarrollo integrado que utilizaremos para programar nuestra placa.
- **XAMPP** Necesario para instalar un servidor web con Apache, PHP y MySQL.
- **Highcharts** es una biblioteca de gráficos escritos en JavaScript que ofrece una forma fácil de añadir gráficos interactivos a nuestro sitio o aplicación web, además, no se necesita el permiso de sus autores para su implementación en sitios web personales o sin fines de lucro.

## Desarrollo del prototipo

El proyecto se divide en tres fases:

1. Montaje del circuito y código Arduino.
2. Captura y almacenamiento de datos.
3. Diseño de página web y creación de gráficas.

### Fase 1: Montaje del circuito y código para Arduino

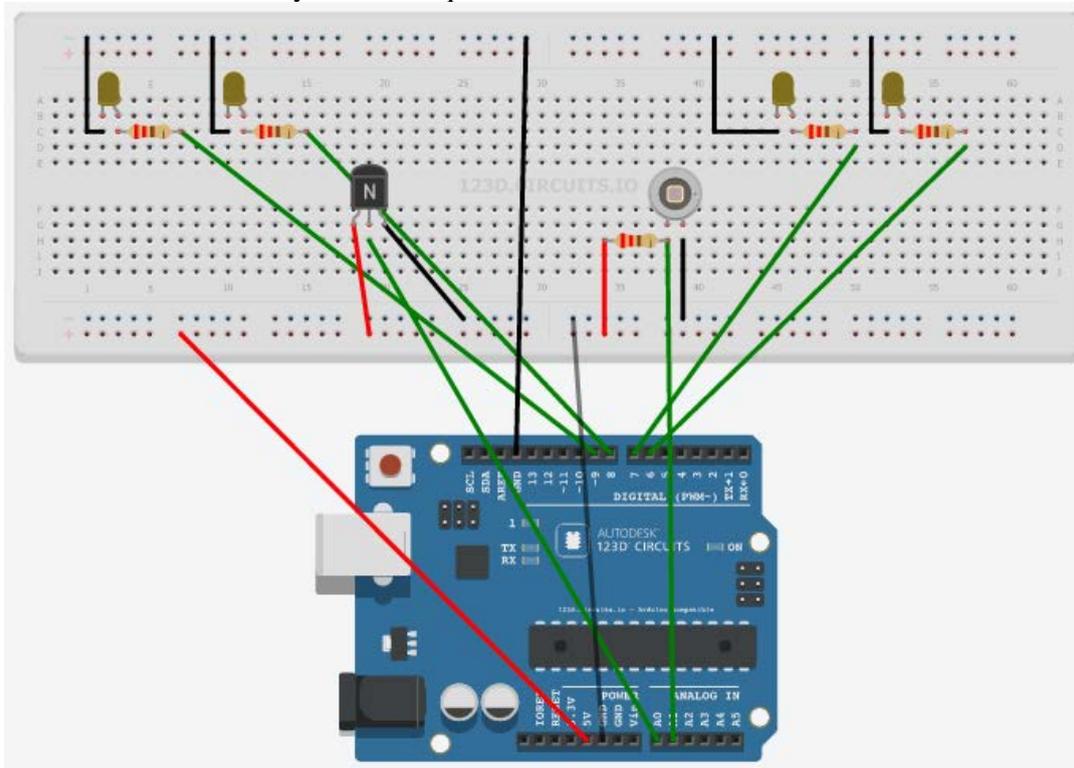
El diseño del circuito constará de dos sensores, uno capturará las variaciones de temperatura y el otro las de luz, estos valores nos servirán para que dependiendo de unos límites establecidos, podremos conectar o desconectar dos leds que simulan aparatos eléctricos (un ventilador y una lámpara), además estos valores se almacenarán en una base de datos para ser tratados posteriormente para realizar las estadísticas.

También conectaremos otros dos leds que nos servirán para demostrar que es posible encenderlos y apagarlos desde nuestra aplicación web.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

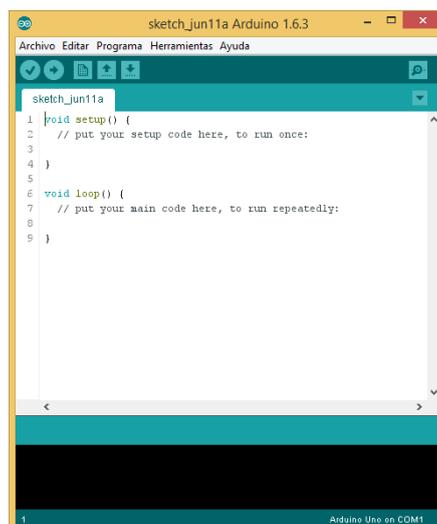
Gracias al Ethernet Shield y a la librería Ethernet, podremos utilizar nuestra placa Arduino **tanto como cliente como servidor web**, y en la aplicación diseñada se emplean ambas opciones.

El circuito ya montado quedaría como se muestra a continuación



Una vez descargado e instalado el IDE de Arduino, debemos seleccionar nuestro modelo de placa y puerto COM.

Todo programa escrito en Arduino consta de dos funciones principales. Ambas son del tipo void (no devuelven un valor) y se denominan **void setup()** y **void loop()**. La función setup se ejecuta una sola vez mientras que el loop se ejecuta un número infinito de veces.



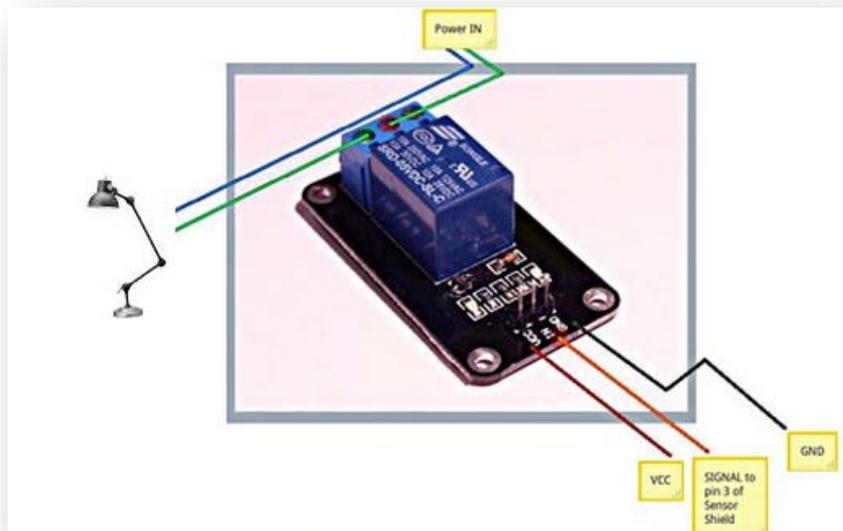
|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

**Nota:** El código fuente comentado, tanto de Arduino como de la aplicación web se encuentra disponible en el apartado de apéndice.

La ip de nuestra placa es la 192.168.0.50 y la de nuestro servidor web la 192.168.0.18

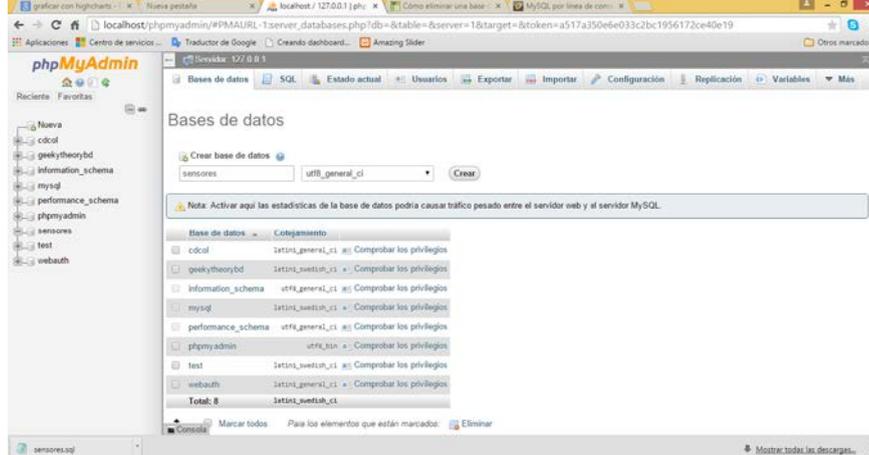
En caso de querer sustituir el control de los leds para manejar un aparato eléctrico, no es necesario modificar el código, con cambiar cada led por un relé bastaría.

La conexión quedaría como se muestra en la imagen



## Fase 2: Creación y estructura de la base de datos.

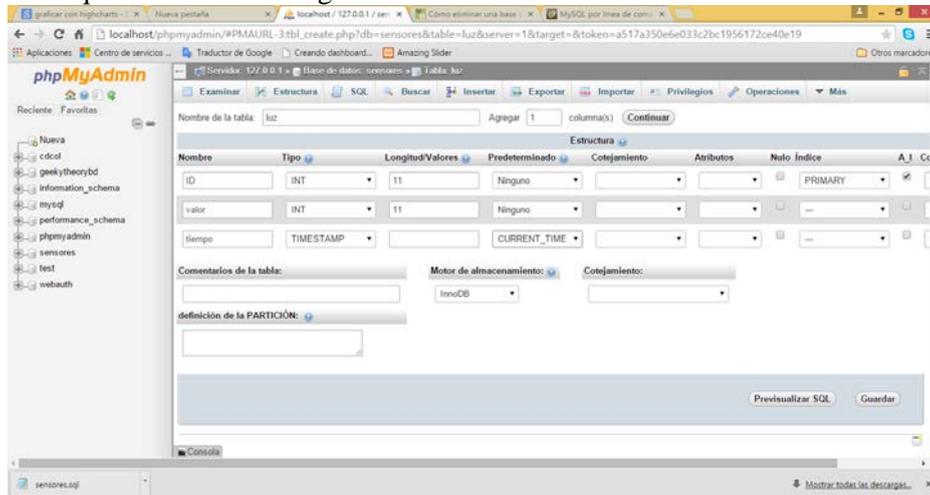
Una vez instalado el XAMPP, generamos una base de datos llamada sensores, que contendrá dos tablas: una llamada **luz** y otra llamada **temperatura**, y que almacenarán los datos de luz y temperatura respectivamente.



|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Cada una de ellas tendrá tres columnas: **ID, valor y tiempo**

Los parámetros son los que vemos en la imagen



En **ID** guardamos el número de registro, en **valor** almacenamos los datos leídos y en **tiempo** guardamos la fecha de creación del registro.

Para insertar los datos leídos por nuestros sensores a nuestra página web, creamos tres archivos PHP: **config.php, iot.php, iot2.php**

En config.php definimos nuestros datos de conexión a nuestra base de datos llamada sensores

Iot.php lo utilizaremos para insertar los valores de nuestro sensor de temperatura en la **tabla temperatura.**

Iot2.php, se usará para insertar los valores de nuestro sensor de luz en la **tabla luz.**

### Fase 3: Página web y creación de gráficas

Para realizar las estadísticas, nos apoyaremos en la librería **Highcharts**

Para guardar los valores de la tabla luz, generamos un fichero llamado **funciones.php**, que creará una **clase llamada funciones** y que será la encargada de gestionar nuestra tabla luz mediante las funciones siguientes:

**ConectarBD(), DesconectarBD(), getArraySQL, getAllInfo(), getDia()**

El archivo **barras.php** llamamos a la funciones de **funciones.php** para cargar los datos insertados en Mysql y que son necesarios para mostrar las gráficas.

En los valores de temperatura, el funcionamiento es similar al descrito para la luz. **funciones.php** realiza las mismas operaciones que su homónimo del sensor de luz, **mediatemp.php**, nos facilita la temperatura media por días, y **vartemp.php**, donde veremos la variación de temperatura por días.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

### Detalle de la página web

Para acceder a la web introducimos en nuestro navegador la dirección:

**http://localhost/main/index.html**

En la siguiente imagen vemos el aspecto que tendrá el menú principal de web



Si pulsamos en el botón “**controlar lámpara y ventilador a distancia**”, accedemos al servidor web instalado en nuestro Arduino, y desde donde seremos capaces de controlar dos leds independientes que simulan una lámpara y un ventilador.

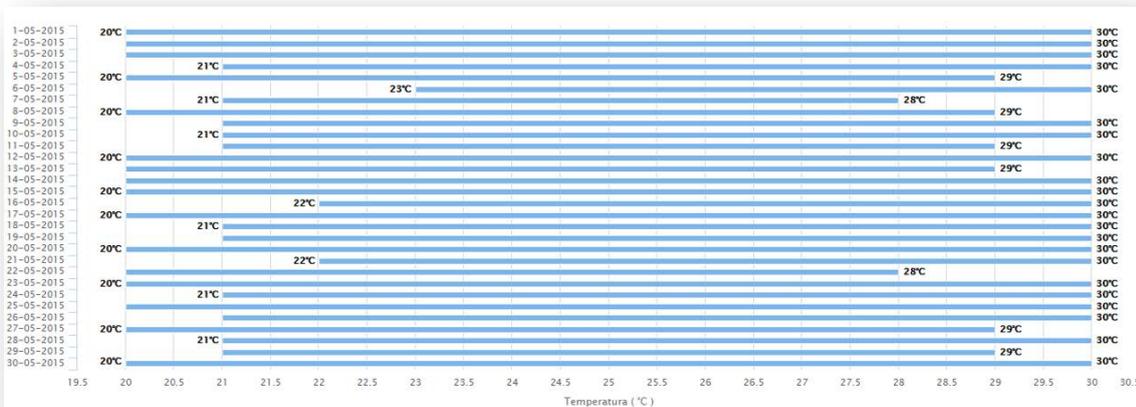


|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

El botón **“temperatura media por días”** nos genera una gráfica con la media de las temperaturas tomadas durante el día:



El botón **“variación temperatura por días”**, nos muestra los valores máximos y mínimos de temperatura que hemos tenido a lo largo de los días.



|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

En “estadística sensor luz”, vemos cuantas veces se ha conectado la luz a lo largo del día.



### Instalación de archivos y código fuente.

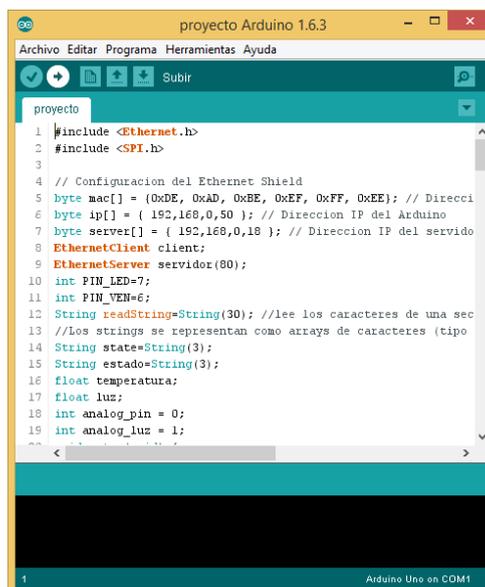
Para comprobar el funcionamiento del proyecto, se facilitan los códigos fuente tanto de la placa Arduino, como de los directorios que hay que copiar en nuestro servidor web, dichos archivos se adjuntan con el proyecto en una carpeta llamada **CODIGOS\_FUENTE.ZIP**

Además, como para utilizar las gráficas es necesario recopilar la información de los sensores durante varios días, se adjunta una base de datos con información aleatoria que simula la captura realizada durante un mes.

Para configurarlo hay que seguir estos pasos:

### Carga de código en la placa Arduino.

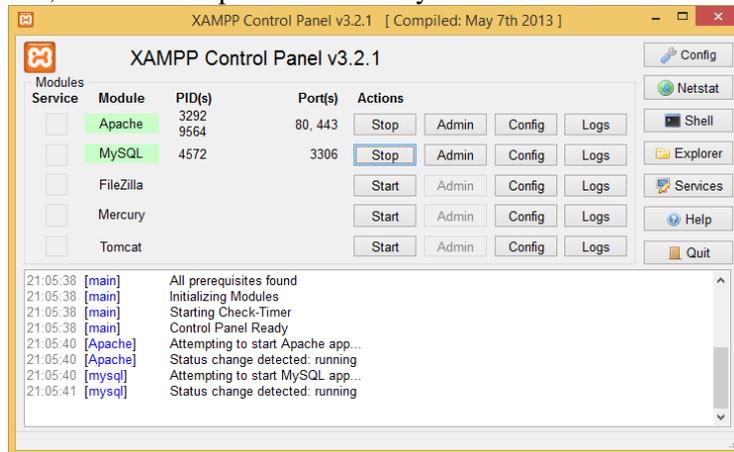
Abrimos el archivo **proyecto.ino** facilitado y lo subimos a la placa Arduino pulsando en el icono de la flecha



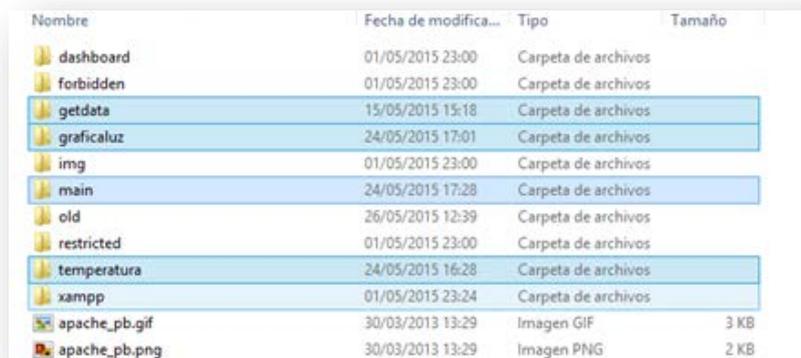
|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Instalación aplicación web.

Una vez instalado el XAMPP, iniciamos su panel de control y arrancamos los servicios Apache y MySQL



Seguidamente debemos copiar los directorios **graficaluz**, **temperatura**, **getdata** y **main** en la ruta `c:\xampp\C:\xampp\htdocs`, y que están incluidos en la carpeta **CODIGOS\_FUENTE.ZIP**.

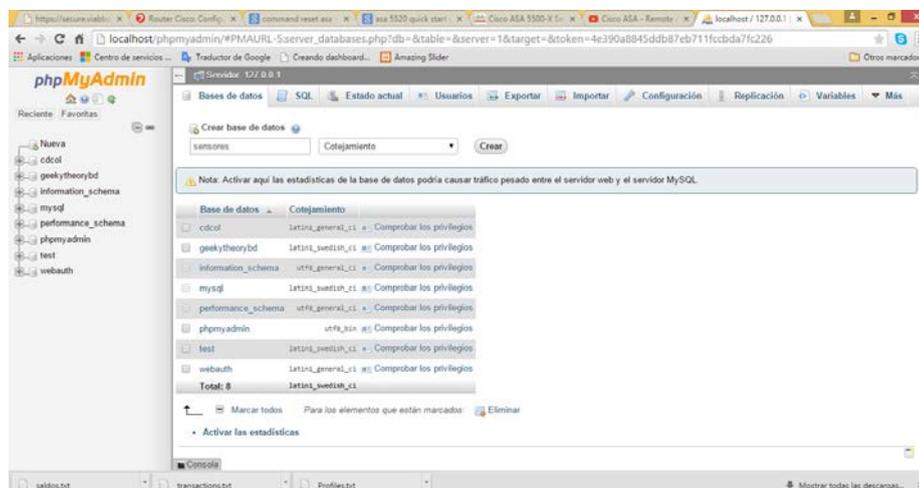


## Importación datos a MySql

Para importar las tablas de la base de datos escribimos en nuestro navegador la ruta:

**<http://localhost/phpmyadmin/>**

Accedemos a la administración de MySQL, y nos creamos una base de datos que se llamará **sensores**.



|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

Una vez que se ha generado aparece en la parte de la izquierda, la marcamos y nos vamos a **importar**, seleccionamos el archivo **sensores.sql** y pulsamos en **continuar**.

Con estos pasos ya tendremos los datos cargados y la aplicación lista para su uso.

## Conclusiones

Entre los diversos campos y temas a escoger para realizar este proyecto de fin de carrera me decidí finalmente por la elaboración de un **estudio sobre las plataformas software existentes para la Internet de las cosas**. Hasta hace unos meses que se trató este tema en una de las asignaturas estudiadas en esta ingeniería, era un concepto tecnológico totalmente desconocido para mí, pero he de confesar que desde un principio me llamó la atención en cuanto a que supone un nuevo salto en la evolución de las telecomunicaciones, sus grandes posibilidades comerciales y una oportunidad de poner en práctica varios de los conocimientos adquiridos durante estos años de aprendizaje.

El resultado final, cumple con los requisitos demandados, y si bien para el proyecto se reclamaba una comparativa sobre las diferentes plataformas software y la realización de algún tipo prototipo, he creído conveniente profundizar algo más, para ello, se ha elaborado una comparativa de las plataformas hardware, se han analizado los orígenes de la IoT, las diversas tecnologías en la que se apoya, y su impacto en la sociedad.

Destacar que si bien hay mucha labor de búsqueda, selección y síntesis de información, el mayor esfuerzo se lo he dedicado a la elaboración del prototipo, especialmente en la obtención de los datos de los sensores y la confección de las estadísticas, dada mi escasa experiencia en Mysql, PhP y las librerías Highchars, se ha convertido en un interesante reto su aprendizaje.

Para concluir, me gustaría agradecer el apoyo y consejos recibidos por mi consultor y desear que este estudio sirva para dar a conocer y despertar el interés por la IoT, un nuevo paradigma tecnológico que a buen seguro dará mucho de hablar en los próximos años.

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Bibliografía

¿Qué es eso del big data?

[http://elpais.com/elpais/2015/03/26/buenavida/1427382655\\_646798.html](http://elpais.com/elpais/2015/03/26/buenavida/1427382655_646798.html)

¿Que nos aportará el concepto IoT?

<http://www.iomando.com/blog/2015/2/16/que-nos-aportar-el-concepto-iot-entrevista-a-marc-collado-ceo-founder-de-iomando>

6 COMMUNICATION PROTOCOLS USED BY THE INTERNET OF THINGS

<https://getkisi.com/kisiblog/2015/2/10/6-communication-protocols-used-by-the-internet-of-things-and-who-the-winner-will-be>

¿Qué es IoT (el Internet de las Cosas)?

<http://www.domodesk.com/a-fondo-que-es-el-internet-de-las-cosas>

A Primer On The Internet of Things & RFID

<http://blog.atlasrfidstore.com/internet-of-things-and-rfid>

De cómo afectarán el Internet de las Cosas y el Big Data al trabajo de los medios

<http://www.media-tics.com/noticia/4291/Medios-de-Comunicacion/De-como-afectaran-el-Internet-de-las-Cosas-y-el-Big-Data-al-trabajo-de-los-medios.html>

El ABC de la Internet de las cosas

<http://cioperu.pe/articulo/16055/el-abc-de-la-internet-de-las-cosas/?p=5>

Internet de las cosas — Máquinas, empresas, personas, todo

<https://itunews.itu.int/es/4503-Internet-de-las-cosas-Maquinas-empresas-personas-todo.note.aspx>

Desde la Internet de los equipos hacia la Internet de las cosas

[http://www.academia.edu/1745098/Internet\\_de\\_las\\_cosas](http://www.academia.edu/1745098/Internet_de_las_cosas)

El impacto en la privacidad de Internet de las cosas y el Big Data

<http://legitec.com/la-aepd-analiza-el-impacto-en-la-privacidad-de-internet-de-las-cosas-y-el-big-data/>

Tecnología innovadora para comunicar los sistemas inteligentes de las ciudades, buscando una gestión más eficaz del tráfico

<https://www.tecnocarreteras.es/2012/05/16/tecnologia-innovadora-para-comunicar-los-sistemas-inteligentes-de-las-ciudades-buscando-una-gestion-mas-eficaz-del-trafico/>

Wireless Sensor Network

<http://www.mfbarcell.es/conferencias/wsn.pdf>

Implantación de Arduino en las redes Ethernet

[http://unicarlos.com/ARDUINO/Arduino%20+%20Ethernet%20Shield%20\(1\).pdf](http://unicarlos.com/ARDUINO/Arduino%20+%20Ethernet%20Shield%20(1).pdf)

HighCharts: Librería para creación de gráficos

<https://enboliviacom.wordpress.com/2013/03/01/highcharts-libreria-para-creacion-de-graficos/>

Web oficial Arduino

<http://www.arduino.cc/>

Web oficial Highcharts

<http://www.highcharts.com/>

Web oficial MySQL

<http://www.mysql.com/>

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

## Apéndice

```

Projecto.ino

// cargamos las librerías
#include <Ethernet.h>
#include <SPI.h>

// Configuración del Ethernet Shield
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFF, 0xEE}; // Dirección MAC
byte ip[] = { 192,168,0,50 }; // Dirección IP del Arduino
byte server[] = { 192,168,0,18 }; // Dirección IP del servidor

// Utilizaremos nuestra placa tanto de cliente como de servidor web
EthernetClient client;
EthernetServer servidor(80);

// definimos los pins que manejarán los leds y las variables
// Pin del LED que simula una lámpara
int PIN_LED=7;

// Pin del LED que simula un ventilador
int PIN_VEN=6;

String state=String(3);
String estado=String(3);
float temperatura;
float luz;
int analog_pin = 0;
int analog_luz = 1;

// lee los caracteres de una secuencia en una cadena.
// Los strings se representan como arrays de caracteres (tipo char)

String readString=String(30);
void setup(void) {
  pinMode(PIN_LED,OUTPUT);
  digitalWrite(PIN_LED,LOW);
  state="OFF";
  pinMode(PIN_VEN,OUTPUT);
  digitalWrite(PIN_VEN,LOW);
  estado="OFF";pinMode(PIN_LED,OUTPUT);
  digitalWrite(PIN_LED,LOW);
  state="OFF";
  pinMode(PIN_VEN,OUTPUT);
  digitalWrite(PIN_VEN,LOW);
  estado="OFF";

  pinMode(8,OUTPUT); //pin sensor temperatura
  pinMode(9,OUTPUT); //pin sensor luz.
  Ethernet.begin(mac, ip);

  // Inicializamos el Ethernet Shield
  delay(1000); // Esperamos 1 segundo
}

void loop(void) {
  temperatura = analogRead(analog_pin);

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

temperatura = 5.0*temperatura*100.0/1024.0;
luz=analogRead(analog_luz);

//Cuando la temperatura pasa de 28 grados, se enciende el led que simula el ventilador

if (temperatura < 28){
digitalWrite(8, LOW);

}else
digitalWrite(8,HIGH);

//Display en Serial Monitor
Serial.print(temperatura);
Serial.println(" oC");

// Proceso de envío de muestras de temperatura al servidor web
Serial.println("Connecting...");
if (client.connect(server, 80)>0) { // Conexion con el servidor
  client.print("GET /getdata/iot.php?valor="); // Enviamos los datos por GET
  client.print(temperatura);
  client.println(" HTTP/1.0");
  client.println("User-Agent: Arduino 1.0");
  client.println();
  Serial.println("Conectado");
} else {
  Serial.println("Fallo en la conexión");
}
if (!client.connected()) {
  Serial.println("Desconectado!");
}
//definimos el límite para conectar el led que simula un ventilador
if (luz>=400){

digitalWrite(9, HIGH);
}else
digitalWrite(9, LOW);
client.stop();
client.flush();

// Proceso de envío de muestras al servidor web
Serial.println("Connecting...");
if (client.connect(server, 80)>0) { // Conexion con el servidor
  client.print("GET /getdata/iot2.php?valor="); // Enviamos los datos por GET
  client.print(luz);
  client.println(" HTTP/1.0");
  client.println("User-Agent: Arduino 1.0");
  client.println();
  Serial.println("Conectado");
} else {
  Serial.println("Fallo en la conexión");
}
if (!client.connected()) {
  Serial.println("Desconectado!");
}

client.stop();
client.flush();

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

//Configuramos nuestro Arduino como servidor web
EthernetClient cliente= servidor.available();
if(cliente) {
boolean lineaenblanco=true;
while(cliente.connected()) {
if(cliente.available()) {
char c=cliente.read();
if(readString.length()<30) {
readString.concat(c);
//Leemos petición HTTP carácter a carácter
//Almacenamos los caracteres en la variable readString
}
if(c=='\n' && lineaenblanco) //Si la petición HTTP ha finalizado
{
int LED = readString.indexOf("LED=");
int VEN = readString.indexOf("VEN=");
if(readString.substring(LED,LED+5)=="LED=T") {
digitalWrite(PIN_LED,HIGH);
state="ON"; }
else if (readString.substring(LED,LED+5)=="LED=F") {
digitalWrite(PIN_LED,LOW);
state="OFF";
}
if(readString.substring(VEN,VEN+5)=="VEN=T") {
digitalWrite(PIN_VEN,HIGH);
estado="ON"; }
else if (readString.substring(VEN,VEN+5)=="VEN=F") {
digitalWrite(PIN_VEN,LOW);
estado="OFF";
}
}
//Cabecera HTTP estándar y diseño de nuestra web
cliente.println("HTTP/1.1 200 OK");
cliente.println("Content-Type: text/html");
cliente.println(); //Página Web en HTML
cliente.println("<html>");
cliente.println("<head>");
cliente.println("<title>LUZ ON/OFF</title>");
cliente.println("</head>");
cliente.println("<body width=100% height=100%>");
cliente.println("<center>");
cliente.println("<h1>luz ON/OFF</h1>");
cliente.print("<br><br>");
cliente.print("Estado de la luz: ");
cliente.print(state);
cliente.print("<br><br><br><br>");
cliente.println("<input type=submit value=ON style=width:200px;height:75px
onClick=location.href='./?LED=T\>");
cliente.println("<input type=submit value=OFF style=width:200px;height:75px
onClick=location.href='./?LED=F\>");
cliente.println("<html>");
cliente.println("<head>");
cliente.println("<title>LAMPARA ON/OFF</title>");
cliente.println("</head>");
cliente.println("<body width=100% height=100%>");
cliente.println("<center>");
cliente.println("<h1>ventilador ON/OFF</h1>");
cliente.print("<br><br>");

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

cliente.print("Estado de la ventilador: ");
cliente.print(estado);
cliente.print("<br><br><br><br>");
cliente.println("<input type=submit value=ON style=width:200px;height:75px
onClick=location.href='./?VEN=T\>");
cliente.println("<input type=submit value=OFF style=width:200px;height:75px
onClick=location.href='./?VEN=F\>");
cliente.println("</center>");
cliente.println("<a href=javascript:history.back(1)>Volver<font face=webdings>20</a></font>");
cliente.println("</body>");
cliente.println("</html>");
cliente.stop();
//Cierro conexión con el cliente
readString="";
}
}
}
}

delay(3000); // Tiempo de espera antes de tomar otra muestra
}

```

```

config.php
<?php
// config.php
// Credenciales
$dbhost = "localhost";
$dbuser = "root";
$dbpass = "";
$dbname = "sensores";
// Conexión con la base de datos
$con = mysqli_connect($dbhost, $dbuser, $dbpass, $dbname);
?>

```

```

iot.php
<?php
// iot.php
// Importamos la configuración
require("config.php");
// Leemos los valores que nos llegan por GET
$val = mysqli_real_escape_string($con, $_GET['valor']);
// Esta es la instrucción para insertar los valores
$query = "INSERT INTO temperatura(valor) VALUES('".$val."')";
// Ejecutamos la instrucción
mysqli_query($con, $query);
mysqli_close($con);
?>

```

```

iot2.php
<?php
// iot.php
// Importamos la configuración
require("config.php");
// Leemos los valores que nos llegan por GET
$val = mysqli_real_escape_string($con, $_GET['valor']);
// Esta es la instrucción para insertar los valores

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```
$query = "INSERT INTO luz(valor) VALUES('".$svalor."");
// Ejecutamos la instrucción
mysqli_query($con, $query);
mysqli_close($con);
?>
```

```
funciones.php
<?php
class funciones{

    public $IDr = 0 ;
    //Función que crea y devuelve un objeto de conexión a la base de datos y chequea el estado de la misma.
    function conectarBD(){
        $server = "localhost";
        $usuario = "root";
        $pass = "";
        $BD = "sensores";
        //variable que guarda la conexión de la base de datos
        $conexion = mysqli_connect($server, $usuario, $pass, $BD);
        //Comprobamos si la conexión ha tenido éxito
        if(!$conexion){
            echo 'Ha sucedido un error inesperado en la conexión de la base de datos<br>';
        }
        //devolvemos el objeto de conexión para usarlo en las consultas
        return $conexion;
    }
    /*Desconectar la conexión a la base de datos*/
    function desconectarBD($conexion){
        //Cierra la conexión y guarda el estado de la operación en una variable
        $close = mysqli_close($conexion);
        //Comprobamos si se ha cerrado la conexión correctamente
        if(!$close){
            echo 'Ha sucedido un error inesperado en la desconexión de la base de datos<br>';
        }
        //devuelve el estado del cierre de conexión
        return $close;
    }
}

//Devuelve un array multidimensional con el resultado de la consulta
function getArraySQL($sql){
    //Creamos la conexión
    $conexion = $this->conectarBD();
    //generamos la consulta
    if(!$result = mysqli_query($conexion, $sql)) die();

    $rawdata = array();
    //guardamos en un array multidimensional todos los datos de la consulta
    $i=0;
    while($row = mysqli_fetch_array($result))
    {

        //guardamos en rawdata todos los vectores/filas que nos devuelve la consulta
        $rawdata[$i] = $row;

        $i++;
    }
    //Cerramos la base de datos
```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

$this->desconectarBD($conexion);
//devolvemos rawdata
return $rawdata;
}

function getAllInfo(){
//Creamos la consulta
$sql = "SELECT count(*) FROM luz where valor > 400 GROUP BY DAY(tiempo);";
//obtenemos el array con toda la información donde se suman los valores que superen el umbral
de conexión que hemos definido.
return $this->getArraySQL($sql);
}

function getDia(){
//Creamos la consulta
$sql = "SELECT tiempo FROM luz where valor > 400 GROUP BY DAY(tiempo);";
//obtenemos la fecha del primer valor introducido
return $this->getArraySQL($sql);
}
}
?>

```

#### Barras.php

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<title>Java Smart Home Simulator</title>

<!-- Latest compiled and minified CSS -->
<style type="text/css">
.style1 {
text-align: center;
margin-left: 369px;
}
</style>
<script type="text/javascript">
</script>
<script type="text/javascript">
<!--
// -->
</script>
</head>
<body onload="FP_preloadImgs(/*url*/"button32.jpg",/*url*/"button33.jpg")">

<?php
require_once("funciones.php");

//Creamos un objeto de la clase funciones
$rand = new funciones();

//obtenemos toda la información de la tabla luz
$rawdata = $rand->getAllInfo();

//nos creamos un array para almacenar el valor numérico
$valoresArray;

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

//en un bucle for obtenemos en cada iteración el valor numérico y lo almacenamos en el array
for($i = 0 ;$i<count($rawdata);$i++){
    $valoresArray[$i]= $rawdata[$i][0];

    //OBTENEMOS EL TIMESTAMP

}
$timeArray;
$rawdato = $rand->getDia();
for($i = 0 ;$i<count($rawdato);$i++){

    $time= $rawdato[$i][0];
    $date = new DateTime($time);
    //ALMACENAMOS EL TIMESTAMP EN EL ARRAY
    $timeArray = $date->getTimestamp()*1000+(24 * 3600 * 1000);
}

?>
<script src="https://code.jquery.com/jquery.js"></script>
<!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
<script src="https://code.highcharts.com/stock/highstock.js"></script>
<script src="https://code.highcharts.com/modules/exporting.js"></script>

<div id="container">
</div>

<script type='text/javascript'>
$(function () {
    $('#container').highcharts({
        chart: {
            type: 'column'
        },
        title: {
            text: 'Número de veces que se ha conectado la luz por días'
        },
        xAxis: {
            type: 'datetime',
            dateTimeLabelFormats: {
                day: '%e of %b'
            }
        },
        yAxis: {
            min: 0,
            title: {
                text: 'Cantidad'
            }
        },
        legend: {
            enabled: false
        },
        series: [{
            name: 'luz',

            pointInterval: 24 * 3600 * 1000,

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

        pointStart: <?php echo $timeArray; ?>,
data: (function() {
    // obtenemos los valores de la base de datos
    var data = [];
    <?php
        for($i = 0 ;$i<count($rawdata);$i++){
        ?>
        data.push([<?php echo $valoresArray[$i];?>]);
    <?php } ?>
    return data;
})();

dataLabels: {
    enabled: true,
    rotation: -1,
    color: '#FFFFFF',
    align: 'center',
    format: '{point.y:.0f}', // one decimal
    y: 10, // 10 pixels down from the top
    style: {
        fontSize: '23px',
        fontFamily: 'Verdana, sans-serif'
    }
}
},1
});
//]]>
//introducimos un botón para regresar al menú principal
</script>
<p class="style1" style="width: 639px"><a href="../main/index.html">
</a></p>
</html>

```

```

funciones.php
<?php
class funciones{

    public $IDr = 0 ;
    //Función que crea y devuelve un objeto de conexión a la base de datos y chequea el estado de la misma.
    function conectarBD(){
        $server = "localhost";
        $usuario = "root";
        $pass = "";
        $BD = "sensores";
        //variable que guarda la conexión de la base de datos
        $conexion = mysqli_connect($server, $usuario, $pass, $BD);
        //Comprobamos si la conexión ha tenido éxito
        if(!$conexion){
            echo 'Ha sucedido un error inesperado en la conexión de la base de datos<br>';
        }
    }
}

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

//devolvemos el objeto de conexión para usarlo en las consultas
return $conexion;
}
/*Desconectar la conexión a la base de datos*/
function desconectarBD($conexion){
//Cierra la conexión y guarda el estado de la operación en una variable
$close = mysqli_close($conexion);
//Comprobamos si se ha cerrado la conexión correctamente
if(!$close){
echo 'Ha sucedido un error inesperado en la desconexion de la base de datos<br>';
}
//devuelve el estado del cierre de conexión
return $close;
}

//Devuelve un array multidimensional con el resultado de la consulta
function getArraySQL($sql){
//Creamos la conexión
$conexion = $this->conectarBD();
//generamos la consulta
if(!$result = mysqli_query($conexion, $sql)) die();

$rawdata = array();
//guardamos en un array multidimensional todos los datos de la consulta
$i=0;
while($row = mysqli_fetch_array($result))
{

//guardamos en rawdata todos los vectores/filas que nos devuelve la consulta
$rawdata[$i] = $row;

$i++;
}
//Cerramos la base de datos
$this->desconectarBD($conexion);
//devolvemos rawdata
return $rawdata;
}

//Consulta donde obtenemos la temperatura media por días
function getAllInfo(){
//Creamos la consulta
$sql = "SELECT AVG(VALOR) FROM TEMPERATURA GROUP BY DAY(tiempo);";
//obtenemos el array con toda la información
return $this->getArraySQL($sql);
}

//Obtenemos la fecha del primer valor generado.
function getDia(){
//Creamos la consulta
$sql = "select tiempo from temperatura order by tiempo asc limit 1;";
//obtenemos el array con toda la información
return $this->getArraySQL($sql);
}

//Obtenemos los valores máximos y mínimos recogidos por días.
function getminmax(){
//Creamos la consulta

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

    $sql = "select DATE_FORMAT(tiempo, '%e-%m-%Y'),MIN(valor),MAX(VALOR) from
temperatura GROUP BY DAY(tiempo);";
    //obtenemos el array con toda la información
    return $this->getArraySQL($sql);
}
}
?>

```

```

mediatemp.php
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
.style1 {
    text-align: center;
}
</style>
<script type="text/javascript">
<!--
// -->
</script>
</head>
<?php
//Importamos la clase que contiene nuestras funciones
require_once("funciones.php");

//Creamos un objeto de la clase funciones
$rand = new funciones();

//obtenemos toda la información de la tabla valores
$rawdata = $rand->getAllInfo();

//nos creamos un array para almacenar las medidas de temperatura capturadas por el sensor
$valoresArray;

//en un bucle for obtenemos en cada iteración el valor numérico

for($i = 0 ;$i<count($rawdata);$i++){
    $valoresArray[$i]= $rawdata[$i][0];

    //OBTENEMOS EL TIMESTAMP
}
$timeArray;
$rawdato = $rand->getDia();
for($i = 0 ;$i<count($rawdato);$i++){
    $time= $rawdato[$i][0];
    $date = new DateTime($time);
    //ALMACENAMOS EL TIMESTAMP EN EL ARRAY
    $timeArray = $date->getTimestamp()*1000;
}
//Generamos nuestra gráfica
?>
<script src="https://code.jquery.com/jquery.js"></script>
<!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

<script src="https://code.highcharts.com/stock/highstock.js"></script>
<script src="https://code.highcharts.com/modules/exporting.js"></script>

<body onload="FP_preloadImgs(/*url*/"button4F.jpg",/*url*/"button50.jpg")">

<div id="container">
</div>
<script type='text/javascript'>
$(function () {
  $('#container').highcharts({
    chart: {
      zoomType: 'x'
    },
    title: {
      text: 'Temperatura media por días'
    },
    subtitle: {
      text: document.ontouchstart === undefined ?
        'Hacer clic y arrastrar en el área de trazado para hacer zoom' :
        'Pulsar la tabla para hacer zoom'
    },
    xAxis: {
      type: 'datetime',
      minRange: 14 * 24 * 3600000 // en el eje x se muestran 4 días
    },
    yAxis: {
      title: {
        text: 'Grados Centígrados'
      }
    },
    legend: {
      enabled: false
    },
    plotOptions: {
      area: {
        fillColor: {
          linearGradient: { x1: 0, y1: 0, x2: 0, y2: 1 },
          stops: [
            [0, Highcharts.getOptions().colors[0]],
            [1, Highcharts.Color(Highcharts.getOptions().colors[0]).setOpacity(0).get('rgba')]
          ]
        },
        marker: {
          radius: 2
        },
        lineWidth: 1,
        states: {
          hover: {
            lineWidth: 1
          }
        },
        threshold: null
      }
    },
    series: [{
      type: 'area',
      name: 'Temperatura media',

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

pointInterval: 24 * 3600 * 1000,
pointStart: <?php echo $timeArray; ?>,
data: (function() {
    var data = [];
    <?php
        for($i = 0 ;$i<count($rawdata);$i++){
            ?>
            data.push([<?php echo $valoresArray[$i];?>]);
        <?php } ?>
    return data;
})();
    }
});
//]]>
//introducimos un botón para regresar al menú principal
</script>

<p>&nbsp;</p>
<p>&nbsp;</p>
<p class="style1" style="width: 1403px"><a href="../main/index.html">
</a></p>

</html>

```

---

```

vartemp.php
<!DOCTYPE html>
<html>
<?php
require_once("funciones.php");

//Creamos un objeto de la clase funciones
$rand = new funciones();

//obtenemos toda la información de la tabla valores
$rawdata = $rand->getminmax();

//nos creamos tres arrays para almacenar el tiempo y los valores máximos y mínimos
$timeArray;
$minArray;
$maxArray;
$cadena="";

//en un bucle for obtenemos en cada iteración los valores máximos y mínimos y el TIMESTAMP del
tiempo y lo almacenamos en los arrays
for($i = 0 ;$i<count($rawdata);$i++){
    $timeArray[$i]= $rawdata[$i][0];
    $minArray[$i]= $rawdata[$i][1];
    $maxArray[$i]= $rawdata[$i][2];

//Almacenamos los tiempos en una cadena
}

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

for($i = 0 ;$i<count($timeArray);$i++){
    $cadena= $cadena."".$timeArray[$i]."";
}

?>
<head>
<script type='text/javascript'
src='https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js'></script>
<script type='text/javascript'>
$(function () {

    $('#container').highcharts({

        chart: {
            type: 'columnrange',
            inverted: true
        },

        title: {
            text: 'VARIACIÓN DE LA TEMPERATURA POR DÍAS'
        },

        xAxis: {
            categories: [<?php echo $cadena;?>]
        },

        yAxis: {
            type: 'String',
            title: {
                text: 'Temperatura ( °C )'
            }
        },

        tooltip: {
            valueSuffix: '°C'
        },

        plotOptions: {
            columnrange: {
                dataLabels: {
                    enabled: true,
                    formatter: function () {
                        return this.y + '°C';
                    }
                }
            }
        },

        legend: {
            enabled: false
        },

        series: [{
            name: 'Temperatures',
            data: (function() {
                var data = [];
                <?php

```

|  |   |                                    |
|--|---|------------------------------------|
| <b>TFC</b><br><br><b>Integración redes telemáticas</b> | <b>Proyecto:</b> Estudio de las plataformas software existentes para la Internet de las cosas |                                    |
|  | <b>Tipo de documento:</b> Memoria.  |                                    |
|  | <b>Versión del documento:</b> V3.0  | <b>Fecha:</b> 13 de junio del 2015 |

```

        for($i = 0 ;$i<count($rawdata);$i++){
            ?>
            data.push([<?php echo $minArray[$i];?>,<?php echo $maxArray[$i];?>]);
            <?php } ?>
            return data;
        }()
    }
}

});
});

</script>
<style type="text/css">
.style1 {
    text-align: center;
    margin-left: 369px;
}
</style>
</head>
//introducimos un botón para regresar al menú principal

<body onload="FP_preloadImgs(/*url*/'button2E.jpg',/*url*/'button2F.jpg')">
<script src="http://code.highcharts.com/highcharts.js"></script>
<script src="http://code.highcharts.com/highcharts-more.js"></script>
<script src="http://code.highcharts.com/highcharts-3d.js"></script>
<div id="container" style="height: 522px"></div>
<p class="style1" style="width: 639px"><a href="../main/index.html">
</a></p>
</body>
</html>

```