UNIVERSIDAD DE EXTREMADURA

Escuela Politécnica

Grado en Ingeniería Informática en Ingeniería del Software

UNIVERSIDAD DE EXTREMADURA

Escuela Politécnica

Grado en Ingeniería Informática en Ingeniería del Software

Trabajo Fin de Grado

Propuesta de adquisición de densidad poblacional en tiempo real

Alumno: César Barro Carrasco

Tutor: Juan Carlos Preciado Rodríguez

Tabla de contenido

Índice de tablas

Índice de ilustraciones

Agradecimientos

A mi pareja, por ayudarme moralmente en los momentos más delicados, a mi familia por darme un apoyo incondicional y por supuesto a mi tutor Juan Carlos por sus consejos, implicación y alta disponibilidad.

Abstract

Currently, the desire to analyze the traffic of people in certain places grows in a notorious manner. This growth, according to Marta Fernández (2012), is due to the great interest about aspects such as the number of people accessing a local particular or the time remaining in this local. There are companies that require mechanisms to be able to control their capacity at any moment and implement marketing strategies accordingly. These mechanisms accounted for the number of people that exist in places where the population volume is high, and they help write reports to address statistical studies at the municipal level. This system not only is developed to control high population densities, It is also able to detect cases of intrusion, in which a detection is considered an exceptional case.

This work is entitled 'Proposed acquisition of population density in real time', and whose main objective is to develop a system that is able to collect information concerning the number of people through the use of receiving devices that you exist in places and at certain times. To achieve this goal, the system will perform a quantification of mobile devices by means of environment scans carried out by receiving antennas. According to Samuel Fernández (2016), in January 2016, 80 percent of the Spanish population was carrying a smartphone in your pocket. The idea of developing a system of population density, is born of this high percentage and their development begins after a meeting that takes place at the end of the year 2016 last July with the tutor of this project, which proposes the development of system.

For a correct quantification of people, the system assumes that a mobile device is equivalent to a person in a place, in a way that to obtain equivalent person - mobile, the data collected must ensure the identification of phones, something similar to what happens with the DNI in people or in the case of the automobile license plates. To ensure that a specific smartphone is exclusive in a place and it can be quantified, the data that is extracted from the smartphone is its Mac address, since it is characterized for being unique and identify a device around the world.

The results allow to make a comparison between the differents technologies used to carry out scans of the environment, as well as reflect the population activity that

occurs around the receiving devices, ensuring compliance the main objective described in this same overview.

Resumen

En la actualidad, el afán por analizar el tráfico de personas en lugares determinados crece de manera notoria. Este crecimiento, según Marta Fernández (2012), se debe al gran interés por conocer aspectos como la cantidad de personas que acceden a un determinado local o el tiempo que permanecen en el mismo. Empresas tales como restaurantes o supermercados requieren de mecanismos que sean capaces de controlar sus aforos en todo momento y poner en marcha estrategias de marketing en consecuencia. Asimismo, estos mecanismos permiten contabilizar el número de personas que hay en eventos tales como conciertos u otro tipo de festividades en los que el volumen demográfico es elevado, y ayudan a redactar informes para abordar estudios estadísticos a nivel municipal. Este sistema no solo está desarrollado para controlar densidades poblaciones elevadas, sino que también, es capaz de detectar casos de intrusismo, en los cuales una detección se considera un caso excepcional.

El presente trabajo constituye la documentación referente al proyecto de fin de carrera titulado 'Propuesta de adquisición de densidad poblacional en tiempo real', cuyo objetivo principal es *desarrollar un sistema que mediante el uso de dispositivos receptores sea capaz de recoger información referente al número de personas que existen en lugares y momentos determinados*. Para alcanzar dicho objetivo el sistema realizará una cuantificación de dispositivos móviles mediante escaneos de entorno llevados a cabo por antenas receptoras. Según Samuel Fernández (2016), en Enero de 2016, un 80 por ciento de la población española portaba un smartphone en su bolsillo. La idea de desarrollar un sistema de adquisición de densidad poblacional, nace de este elevado porcentaje y su desarrollo da comienzo tras una reunión que tiene lugar a finales del mes de Julio del pasado año 2016 con el tutor de este proyecto, en la cual propone la elaboración de dicho sistema.

Para una correcta cuantificación de personas, el sistema asume que un dispositivo móvil equivale a una persona en un lugar, de manera que, para obtener la equivalencia persona - móvil, los datos que se recogen deben asegurar la identificación de teléfonos, algo parecido a lo que sucede con el DNI en personas o con las matrículas para el caso de los automóviles. Para garantizar que un determinado smartphone es exclusivo en un lugar y poderlo cuantificar, el dato que se extrae de dicho smartphone

es su dirección Mac, ya que ésta se caracteriza por ser única e identificar a un solo dispositivo a nivel mundial.

Los resultados obtenidos permiten realizar una comparativa entre las diversas tecnologías empleadas para llevar a cabo los escaneos de entorno, así como reflejar la actividad poblacional que se produce alrededor de los dispositivos receptores, garantizando así, que se cumple el principal objetivo descrito en este mismo resumen.

1. Introducción.

En este primer capítulo, se explica con más detalle en qué consisten los objetivos de este proyecto, organizados en objetivo principal, objetivos secundarios y complementarios. Además el capítulo incluye la arquitectura del sistema en la cual se pueden detectar los módulos o partes fundamentales que componen el sistema. Finalmente, el capítulo incluye un listado de requisitos y motivaciones que facilitan el desarrollo.

1.1. El objetivo principal del sistema.

Como se ha comentado en el resumen de este proyecto, el objetivo principal del sistema es desarrollar un sistema que sea capaz de contabilizar el número de personas en lugares y momentos determinados. Además el sistema permite saber cuánto tiempo permanece una persona en el lugar. El uso de este sistema de detección puede llevarse a cabo para dos finalidades distintas que incluyen la contabilidad de personas:

- Contabilizar aforos de personas para estudios de poblaciones en lugares públicos como supermercados, restaurantes, espacios al aire libre donde se celebren conciertos o eventos en los cuales su volumen poblacional sea elevado. Una vez instalado este sistema pueden llevarse a cabo estrategias de marketing en consecuencia a la densidad poblacional captada.

- Detección de accesos indebidos: El uso de este sistema va más allá de detecciones de aforos para estudiar densidades de poblaciones. También puede usarse para detectar individuos no autorizados en lugares públicos durante horas de inactividad, por ejemplo en polígonos industriales, espacios ganaderos en el campo o en supermercados. Por ejemplo, si se produce una detección en un supermercado a las cuatro de la mañana, cuando éste lleva cerrado desde las diez, es muy probable que se trate de personal no autorizado.

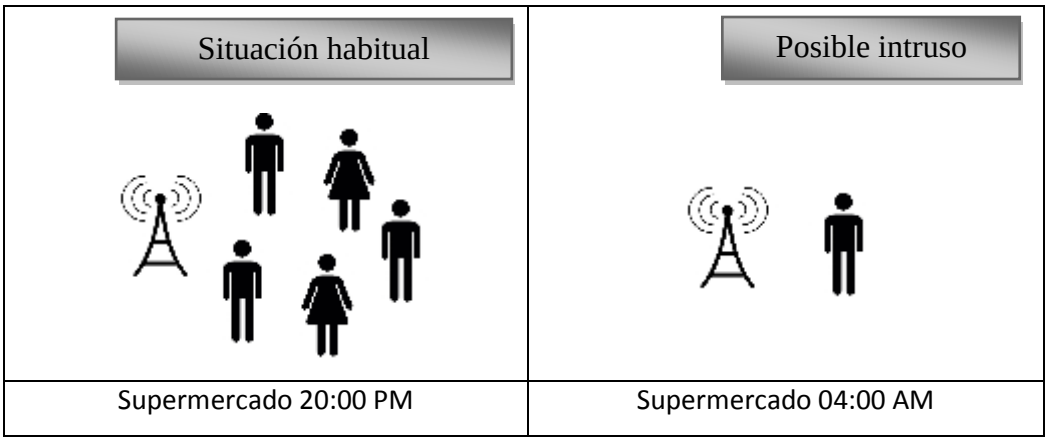


Ilustración 2. Tipos de detección. Fuene: Elaboración propia

1.2. Objetivos secundarios del sistema.

En este apartado se describen cuales son aquellos objetivos que constituyen el objetivo principal de este proyecto (contabilizar población). Son los siguientes:

1. Se hará uso de dispositivos receptores que se instalarán en lugares y serán los encargados de recoger la información que se va a almacenar en el sistema.

2. La información recogida por los receptores referente a la población existente en un lugar determinado, se almacena en una base de datos.

3. Se hace uso de un servidor web cuya función será recoger la información obtenida por los receptores y almacenarla en la base de datos.

4. Los datos obtenidos por las antenas se deben poder visualizar en informes, ofreciendo así, una manera más cómoda y atractiva de entender la información recogida.

1.3. Objetivos complementarios del sistema.

Los objetivos que se citan a continuación, constituyen a cada uno de los objetivos secundarios expuestos en el sub apartado anterior, de tal forma que juntos los tres tipos de objetivos (principal, secundarios y complementarios) forman parte del total de objetivos a tener en cuenta para el desarrollo del presente sistema.

El sistema debe ofrecer escalabilidad, es decir, debe permitir la conexión de un número variable de antenas a la vez.

1. Proximidad. Se debe poder medir la distancia que existe entre un dispositivo receptor y una detección.
2. La lista negra. Debe existir una lista que almacene aquellos dispositivos que no se consideren oportunos para ser almacenados en el sistema.
3. El número de detecciones recogidas en un determinado momento debe ajustarse al total de la población que existe en el lugar escaneado.
4. La base de datos escogida debe ser relacional y ofrecer escalabilidad horizontal, para dar soporte al sistema también escalable (primer objetivo complementario descrito).

5. El servidor debe ofrecer una interfaz gráfica que permita visualizar las antenas que esperan ser activadas y activar una o varias a la vez, asignándoles un tiempo de recarga (tiempo que transcurre desde que finaliza un escaneo y da comienzo el siguiente).
6. El servidor debe permitir localizar la ubicación en la que se encuentran instaladas.
7. El servidor debe ofrecer la funcionalidad de insertar un nombre de dispositivo en la lista negra en su interfaz gráfica.
8. La herramienta de visualización debe realizar constantes consultas a la base de datos para ofrecer la información actualizada al usuario.
9. Las antenas deben ser automáticas, es decir los programas de detección se ejecutan al inicio, de forma que el usuario debe limitarse a conectarlas a la red eléctrica sin interactividad alguna.
10. Los dispositivos receptores tienen un servicio web en el que ponen a disposición del servidor los datos obtenidos. Por lo tanto deben tener acceso a Internet y su configuración debe de permitir accesos desde el exterior.
11. Cada detección recogida por un receptor durante un escaneo debe ser única, es decir, no habrá dos detecciones iguales en una misma búsqueda.

OP Contabilizar personas	**OS_1-** Uso de dispositivos receptores encargados de recoger la información.	**OC_1-** El sistema debe ofrecer escalabilidad.
		OC_2- Medir la proximidad entre dispositivos detectados y receptor.
		OC_3- Debe existir una lista negra que almacene nombres de dispositivos no deseados.
		OC_4- El número de detecciones debe ajustarse al máximo a la realidad.
		OC_{10}- Las antenas deben iniciar los escaneos de forma automática.
		OC_{11}- Los dispositivos receptores ponen la información a disposición del servidor a través de un servidor web.
		OC_{12}- Detecciones únicas en cada escaneo.
	OS_2- La información se almacena en una base de datos.	**OC_5-** La base de datos debe ser relacional y debe permitir escalabilidad horizontal.
	OS_3- El servidor se debe de encargar se recoger la información de las antenas y la almacena en la base de datos.	**OC_6-** El servidor debe permitir activar una o varias antenas conectadas a la vez, asignándoles un tiempo de recarga.
		OC_7- El servidor debe poder localizar la ubicación de todas las antenas conectadas.
		OC_8- El servidor debe ofrecer la funcionalidad de añadir nombres de dispositivos no deseados a lista negra.
		OC_{11}- Los dispositivos receptores ponen la información a disposición del servidor a través de un servidor web.
	OS_4- La información se debe poder visualizar haciendo uso de informes.	**OC_9-** La herramienta de visualización debe de realizar constantes consultas a la base de datos para ofrecer la información actualizada.

Tabla 1. Objetivo principal (OP), objetivos secundarios (OSn) y objetivos complementarios (OCn). Fuente Elaboración propia.

1.4. Arquitectura del sistema.

El sistema se estructura en tres módulos interconectados. El primero incluye la tarea de detección de poblaciones, en la cual una antena receptora realiza escaneos (de forma periódica) de su entorno buscando indicios de actividad. La actividad, como ya se ha comentado, puede ser baja o nula, es decir para detectar un acceso indebido, o habitual en la que se contabiliza una población para llevar a cabo un estudio de la misma. El segundo módulo constituye todo lo referente al almacenado de la información recogida por los dispositivos receptores en lugares y momentos determinados, en donde se espera que la estructura de almacenamiento soporte un volumen de datos de carácter escalable, debido al alto volumen de datos que espera recogerse y a la propia escalabilidad que ofrece el sistema a desarrollar. Por último, el tercero de los módulos engloba todo lo relacionado con la visualización de datos recogidos por las antenas receptoras.

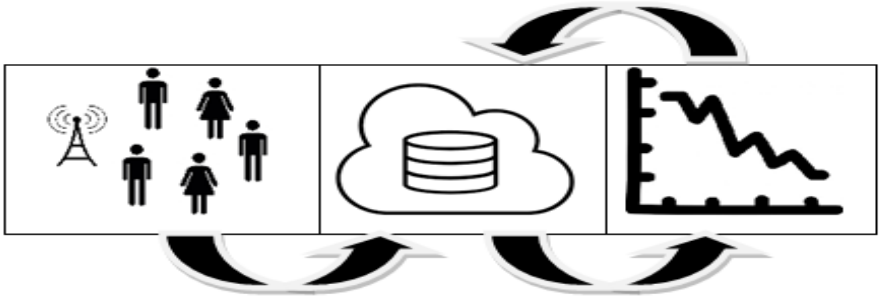


Ilustración 3. Arquitectura del sistema. Fuente Elaboración propia.

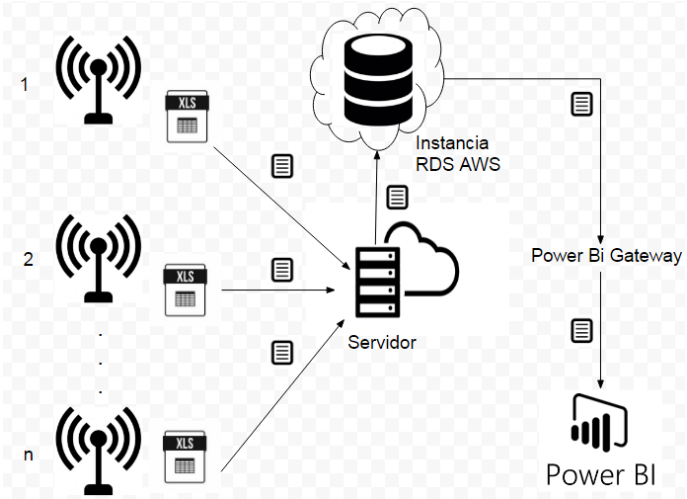


Ilustración 4. Diagrama de flujo de datos. Comunicación entre los agentes participantes en el sistema Fuente: Elaboración propia.

El módulo que se corresponde con la visualización de datos mediante informes, debe mantener una conexión activa con la base de datos, de tal forma que un usuario podrá ver cuando lo desee, la información del sistema actualizada.

1.5. Requisitos del sistema.

En este subapartado se mencionan aquellos requisitos que se consideran fundamentales para llevar a cabo la solución final del sistema, esto se debe a que son criterios que exigen una metodología de desarrollo del sistema más efectiva y organizada.

1.5.1. Requisitos funcionales.

- Los receptores deben disponer de un software de detección de dispositivos.
- Para que un receptor empiece a escanear su entorno debe ser activada por medio del servidor web.
- Un receptor solamente puede ser activado cuando se le asigna un tiempo de refresco para el escaneo.
- Un receptor debe conectarse previamente al sistema para poder ser activado por el servidor web.
- Se debe disponer de un sistema de gestión de bases de datos con modelado relacional, con escalabilidad horizontal para el almacenamiento permanente de datos.
- Para la visualización de datos, se requiere de una herramienta conectada al sistema con acceso a la información en tiempo real y permitir la interacción de usuarios con la misma.

1.5.2. Requisitos no funcionales.

- El software de detección se desarrolla en lenguaje de programación Java.
- El tiempo de consulta por parte del servidor web a los dispositivos receptores debe mantener un equilibrio entre carga de datos y número de accesos. Por ejemplo si el tiempo que transcurre entre una consulta al receptor y la siguiente es demasiado largo, la carga de datos será mayor, pero por el contrario si este tiempo es corto el número de accesos aumentará.
- La carga de trabajo sobre la gestión de dispositivos detectados recae, en su mayor parte en los dispositivos receptores, de manera que la tarea del servidor se limita a recoger los datos detectados por dichos receptores e insertarlos en la base de datos.
- Cada dispositivo receptor debe poner la información obtenida a disposición del servidor.
- Un receptor no vuelve a enviar una información que ya ha sido guardada previamente.
- Para el desarrollo del servidor web se hace uso de la herramienta Webratio.
- Los receptores utilizan servicios web tomcat para poner a disposición la información obtenida.

1.6. Idea del proyecto.

Teniendo en cuenta la estandarización y extensión actual de las tecnologías de conexión inalámbrica, y a su vez asumiendo el papel fundamental que juegan en la sociedad, como es el caso de la integración del Internet de las cosas (IOT), en la que el día de mañana cualquier objeto, desde un reloj hasta un inmueble, podrían estar interconectados inalámbricamente a la red, nace la idea de desarrollar el sistema descrito en el apartado en el que se detalla el objetivo principal del proyecto (llevar la contabilidad de personas en lugares y momentos determinados) , por lo tanto dicho objetivo se va a abordar haciendo uso de las tecnologías inalámbricas existentes.

La presente idea, surge considerando que un alto porcentaje de la sociedad actual utiliza dispositivos smartphones, según Justo (2017), en el año 2012 tan solo un

41 por ciento de la población española contaba con un smartphone, mientras que cinco años más tarde, en 2017 y según menciona el informe 'Google Consumer Barometer Report', el porcentaje se ha doblado. Los datos que se recogerán en el sistema a desarrollar se utilizarán para fines estadísticos y tendrán como objetivo informar sobre estados de aforos para aplicar estrategias de marketing en consecuencia.

1.7. Motivaciones.

A continuación se especifican las motivaciones que han hecho posible el desarrollo del sistema. Se clasifican en dos: personales y de desarrollo.

1.7.1. Motivaciones de desarrollo del proyecto

- Explotación de la expansión de la tecnología inalámbrica en la sociedad actual.
- Beneficio del uso cada vez más habitual de teléfonos inteligentes por dicha sociedad para el desarrollo del sistema.
- Necesidad de sistema de análisis de entorno poblacional en ciertos lugares para aplicar estrategias de marketing.
- Estudio y posterior comparativa de tecnologías de detección inalámbrica utilizadas en el sistema.

1.7.2. Motivaciones personales

- Interés por conocer los distintos mecanismos de detección inalámbrica de datos existentes en la actualidad y llevar a cabo la comparativa mencionada en el apartado anterior.
- Desarrollo de los programas que utilicen dichos mecanismos.
- Aprender a utilizar dispositivos que permitan llevar a cabo las detecciones.
- Manejar plataformas que permitan utilizar servicios como hosting de sites o de bases de datos en la nube.
- Desarrollo de comunicación sostenible entre los distintos agentes que intervienen en el sistema.

- Búsqueda de un equilibrio en el trabajo de dichos agentes de manera que no se produzcan sobrecargas en las transferencias de información.
- Conocer de cerca los computadores de placa reducida u ordenadores de bolsillo, más conocidos como Raspberry Pi y aprender a programarlos para que puedan cumplir su función en el sistema.
- Programación en lenguaje Java del software de detección de datos, poniendo en práctica lo aprendido durante el período de formación en la carrera.
- Utilización de la herramienta Webratio para desarrollar un servidor web.
- Aprender a utilizar los servicios proporcionados por la plataforma Amazon para hosting y alojamientos de bases de datos
- Entender el funcionamiento de herramientas de análisis de negocios para crear informes de visualización de los datos recogidos.

2. Dominio tecnológico

En este segundo capítulo se explican indicadores de métrica utilizados en la actualidad. Estas herramientas se utilizan para conocer información sobre el estado actual de una determinada empresa (ventas, número de clientes,...) y aplicar estrategias de marketing en consecuencia. Para obtener información referente a dichos indicadores, se van a aplicar técnicas de rastreo. Estas técnicas explotan, la cada vez más extensa conexión de dispositivos de forma inalámbrica. A su vez, se va a realizar un estudio para conocer más de cerca los mecanismos que ofrecen este tipo de conexiones.

2.1. Los Indicadores Clave de Desempeño (KPI)

Según la web "Comenzando desde cero" (2017), un KPI (Key Performance Indicators), es un indicador que ofrece información referente al rendimiento del progreso en función de ciertos objetivos a cumplir por una determinada empresa. Los datos que proporciona este tipo de métrica están relacionados con las ventas que se producen durante un período de tiempo, el número de personas que acceden a un local y terminan por no realizar compra e incluso pasan por el lugar pero no acceden. Estos datos son fundamentales para conocer el estado de la empresa y aplicar las técnicas de marketing que se crean convenientes. Una de las preguntas que uno debe hacerse a la hora de escoger un KPI es qué objetivo quiere perseguir e incluso si éste está ligado a los propios objetivos de la empresa. En el sistema a desarrollar se va a utilizar un indicador que va a proporcionar datos relacionados con la densidad poblacional en ciertos lugares de carácter público.

Ilustración 5. Ciclo de los KPI's. Fuente: Web Comenzando desde cero (2017).

2.2. La tecnología Bluetooth

Según el artículo 'Analizando Bluetooth' publicado en la web del Instituto Nacional de Ciberseguridad en España S.A (INCIBE) el 26 de Julio de 2016, la tecnología Bluetooth, utiliza una modulación de tipo espectro ensanchado por salto de frecuencia (FHSS) con 79 frecuencias, lo que la convierte en una de las tecnologías inalámbricas menos sensible a problemas derivados del ruido e interferencias. Por otro lado, esta tecnología se extiende cada vez más (desde teléfonos móviles hasta relojes inteligentes). Son éstos los motivos por los que se pretende recoger información de antenas receptoras Bluetooth, ya que dicha información tiene una alta fiabilidad y queda exenta de ruidos que afectan a la integridad de datos de carácter general, de dispositivos conectados mediante esta tecnología.

La tecnología Bluetooth es una tecnología inalámbrica que permite la conexión de dispositivos con otros dispositivos. La conexión puede llevarse a cabo de uno a uno o de uno a varios dispositivos. El alcance de conexión entre dispositivos puede llegar hasta los doscientos metros de distancia con la aparición de la nueva versión 5 de Bluetooth, el pasado 26 de Junio de 2016. Son muchos usuarios los que hoy en día utilizan esta tecnología, ya sea para conectar una impresora o para conectar los auriculares a un aparato de música.

2.2.1. Ventajas que surgen con la nueva versión de Bluetooth

Según el artículo " 5 ventajas de Bluetooth 5, la última versión de la popular tecnología inalámbrica " de la BBC Mundo publicado el 13 de Junio de 2016, cuando Bluetooth nació en 1994, su principal objetivo fue el intercambio archivos entre dispositivos sin coste alguno y sin conectarse por cable. Tras varias generaciones de Bluetooth posteriores, con la llegada de Bluetooth 5, esta tecnología ofrece ahora las siguientes ventajas:

- **Más veloz que la anterior versión 4.2**: Bluetooth 5 puede transferir hasta 100 Mbps.

- **Doble de alcance respecto a la anterior versión**: En la versión anterior esta tecnología no sobrepasaba los cien metros de alcance. Ahora puede llegar hasta los doscientos metros.

- **Geolocalización y navegación "offline"**: En la nueva versión Bluetooth según explica Mark Powell, Director ejecutivo de Bluetooth SIG, la nueva versión cuenta con nuevas funcionalidades de servicios sin conexión, como información relevante de ubicación y navegación.

- **Menos gasto energético.**

- **Mejores paquetes de aviso:** Estos paquetes se envían de un dispositivo A a uno B para avisar de que el dispositivo A se encuentra disponible para una posible conexión entre ambos.

2.2.2. Vulnerabilidades que presenta la tecnología Bluetooth

Asimismo, se citan algunas de las vulnerabilidades en la seguridad que se destacan en el mismo informe publicado por Incibe, que por desgracia, aún existen en la tecnología a la que se hace referencia:

- Aún se permiten elaborar códigos de pin cortos, de 1 a 16 bytes, para realizar un emparejamiento de dispositivos.
- No existe una manera robusta de crear y distribuir un código pin.
- No existe tampoco la autenticación de usuarios, tan solo se autentica por identificación de dispositivos.
- No existe bloqueo de autenticación tras varios intentos de conexión.
- El esquema de transmisión de claves es vulnerables a ataques MITM (Man In The Middle).
- Tampoco está demostrada que los números pseudoaleatorios para el código pin sean criptográficamente fiables, dando lugar a posibles repeticiones.
- La compatibilidad con versiones anteriores Bluetooth puede dar lugar a servicios sin seguridad o que utilizan protocolos entre niveles ya obsoletos.

2.2.3. Consejos para mejorar la seguridad mediante el uso de Bluetooth.

Tras los resultados obtenidos por las pruebas realizadas y reflejadas en el informe “Seguridad en Bluetooth: fortalezas y debilidades” publicado por Incibe el 29 de Septiembre de 2016, llegan a las siguientes conclusiones que proponen a los usuarios para tener en cuenta y mitigar así, las vulnerabilidades que se mencionan en el subapartado anterior. Estas, son las siguientes:

- Hacer uso del cifrado de comunicaciones cuando sea posible.
- No aceptar conexiones de dispositivos desconocidos.
- Revisar con periodicidad la lista de dispositivos conocidos, evitando así conexiones de dispositivos maliciosos.
- Asignar nombres a dispositivos conocidos que no revelan su marca o modelo, para restringir, de esta manera, conexiones a dispositivos no deseados.
- Permanecer en modo invisible para evitar ser visto por otros dispositivos.

2.2.4. Bluetooth para el futuro

Por un lado se hace uso de la tecnología Bluetooth. Esta tecnología, en constante actualización, se prevé que para un futuro, no muy lejano, forme parte de la vida cotidiana de muchas personas. Según SIG (Bluetooth Special Interest Group), se espera que para 2020 se vendan 1 de cada 3 dispositivos con conectividad Bluetooth integrada, para reforzar IoT (Internet of things).

No obstante aunque esta tecnología ha estado en constante actualización desde que se fundó en 1998, aún existen aspectos que se deben mejorar, como la compresión de audio. Así que con la aparición de la versión 5 se han potenciado y/o mejorado algunas características de versiones anteriores, como se comenta en el apartado "1.1. La tecnología Bluetooth", pero dicha mejora queda solo para el IoT (Internet of things), tal y como se menciona en el artículo Bluetooth 5 para el Internet de las cosas, publicado por Roberto Solé el 8 de Diciembre de 2016 en hardwaresfera.com. Por otro lado son

empresas como Apple que ya han eliminado el conector jack para sustituirlo por esta tecnología que avanza a pasos gigantescos.

Según un informe publicado en la web http://www.marketsandmarkets.com/ por marketsandmarket.com en Abril de 2016, la demanda de dispositivos inalámbricos así como dispositivos de audio o barras de sonido, tienen su evolución en constante aumento. Son países como Estados Unidos, China, Japón o India los que tienen mayor índice de demanda sobre esta tecnología. Se espera que para el año 2022 el mercado dedicado a la industria inalámbrica pueda alcanzar unos 54,07 dólares americanos.

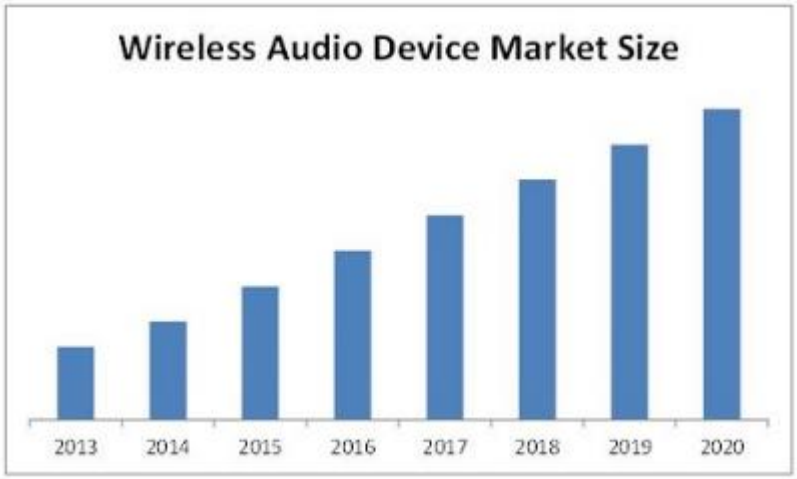


Ilustración 6. Use of cloud computing services (% of enterprises).
Fuente: Eurostat (2017).

Además de compañías relacionadas con dispositivos de audio inalámbrico, existen otras empresas que se ven beneficiadas por estos avances de la tecnología Bluetooth como las que se dedican a la domótica o a todo lo relacionado con las ciudades inteligentes. Otro aspecto a considerar, son las nuevas *mesh network* (red mallada), donde se pretende conseguir una funcionalidad parecida a las conexiones existentes hoy en día que se comunican por el protocolo Wi-Fi.

Ilustración 7. Bluetooth. Fuente: La web gcflearnfree (2017).

2.3. La tecnología WiFi

En este subapartado se explica en qué consiste la tecnología WiFi, cual es la seguridad que ofrece este tipo de acceso a Internet, cuales son los estándares que existen en la actualidad y que consejos deben llevarse a cabo para proteger un equipo siempre que se hace uso de este método de conexión a la red de redes.

2.3.1. Las conexiones Wifi

WiFi, cuyas siglas significan fidelidad inalámbrica (Wireless Fidelity) hace referencia a una tecnología de conexión inalámbrica a Internet, utilizada por dispositivos como ordenadores portátiles, teléfonos móviles, tabletas, etc… Además, no solo permite la conexión a Internet, sino que también proporciona conectividad entre dispositivos.

Según el artículo '¿Qué es WiFi? ¿Qué significa y para qué sirve?' publicado en la web 'valortop' por Sandra Fernández Moreno el 26 de Junio de 2015, La tecnología está limitada por el alcance, que puede variar entre los 5 y 150 metros, dependiendo de la señal emitida por el dispositivo emisor. Además de ser un tipo de tecnología, es una marca creada por Wi-Fi Alliance que diseñó este medio con el objetivo de que fuera compatible para la conexión de distintos dispositivos de forma inalámbrica. Así nació el estándar 802.11.

El funcionamiento de WiFi requiere de la participación de un enrutador conectado por cable a la red, cuyo trabajo consiste en transformar una señal digital en ondas de radio y transmitirlas por el aire. Estas ondas son captadas y decodificadas por los dispositivos que se encuentran dentro de un radio de alcance. Un dispositivo que capta una señal de radio, la transforma de nuevo en información digital para que el procesador del mismo sea capaz de procesarla. De esta forma se produce una conexión a una red de manera inalámbrica.

2.3.2. Seguridad en WiFi.

Debido a que en este mecanismo de conexión, las ondas se transmiten por el aire y son capaces de traspasar paredes, si no se aplican ciertas estrategias de seguridad, la información puede ponerse a disposición de usuarios con intenciones maliciosas. A continuación se citan algunos de los consejos a poner en práctica para reforzar la seguridad si se dispone de un enrutador que emite señal WiFi.

- Filtrado por dirección MAC (Media Access Control): Se realiza creando una lista blanca de dispositivos en el router. El router sólo permite conexiones de dispositivos cuyas direcciones MAC están contenidas en la lista, rechazando así, todas las demás conexiones.

- Cambiar la contraseña que viene por defecto en el router: Por norma general, todos los enrutadores disponen de una contraseña por defecto. Por desgracia existen programas que contienen diccionarios donde aparecen la mayor parte de este tipo de contraseñas. Un software de este tipo se encarga de aplicar ataques de fuerza bruta, utilizando todas las contraseñas del diccionario sobre los enrutadores, hasta que consigue averiguar la contraseña correcta. Son un ejemplo las contraseñas WEP. Se recomienda, por lo tanto, hacer uso de contraseñas con un sistema de encriptación WPA2, por ser el más actualizado.

- La contraseña debe contener el máximo posible de dígitos y alternar entre minúsculas, mayúsculas, dígitos y caracteres especiales.

- Apagar la opción WiFi del enrutador mientras no se haga uso de la misma.

2.3.3. Tipos de WiFi.

Existen distintos tipos de WiFi según el estándar al que pertenecen, la utilización de uno u otro, afecta a la velocidad de transmisión de la onda. A continuación se muestra un cuadro comparativo entre los distintos estándares existentes.

Nombre del estándar	Publicación	Velocidad máxima de transmisión	Frecuencia	Nº de canales
IEEE 802.11	1997	2 Mbps	2,4 Ghz	3
IEEE 802.11a	1999	54Mbps	5 Ghz	12
IEEE 802.11b	1999	11 Mbps	2,4 Ghz	3
IEEE 802.11g	2003	54 Mbps	2,4 Ghz	3
IEEE 802.11n	2007	600 Mbps (teóricos)	2,4 o 5 GHz	12
IEEE 802.11ac	2014	1.300 Mbps	5 Ghz	24

Tabla 2. Estándares de Wifi. Fuente: Web de Intel (2017).

2.3.4. WiFi para el futuro

Según el artículo titulado, 'Este es el futuro del Wi-Fi de los próximos 4 años según la Wi-Fi Alliance' escrito por Sergio De Luz el 9 de Enero de 2016, asegura que según la WiFi Alliance, compañía que creó este mecanismo de conexión inalámbrica, que a finales de 2015, los dispositivos que utilizaban WiFi ascendía a 12.000 millones, y para el año 2016 se esperaba que se sumasen otros 3.000 millones más, por lo que se deduce que el ciclo de vida de esta tecnología será largo y que sus capacidades de velocidad y alcance pueden crecer de una forma que aún no somos capaces de imaginar.

2.3.5. La tecnología de múltiple entrada: MU-MIMO

Hasta la fecha cuando nos hemos conectado a un router, no hemos situado en una cola de dispositivos a la espera de que el router nos envíe la señal para poder enviar o recibir un paquete. Por este motivo cuando estamos en una red donde existen un número elevado de conexiones a un router, podemos observar que la conexión se ralentiza. Esto se debe a la tecnología de conexión llamada SU-MIMO donde un enrutador solamente es capaz de conectarse a un único dispositivo simultáneamente.

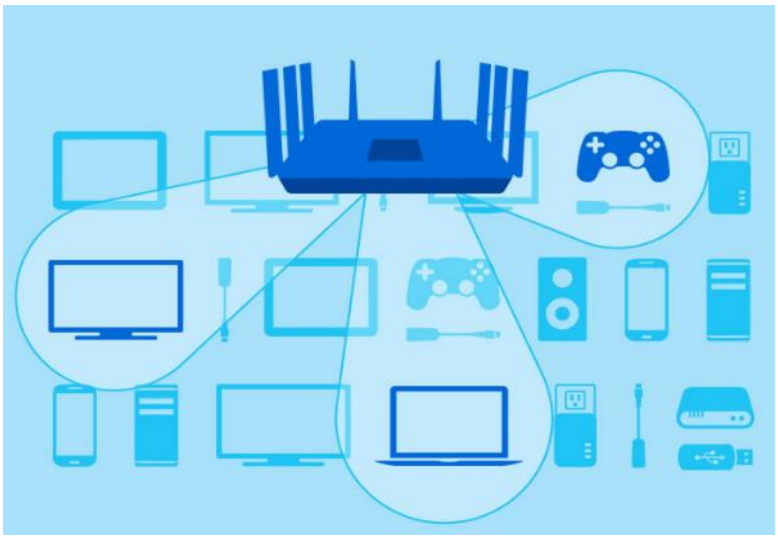


Ilustración 8. MU-MIMO. Fuente: Web Linksys (2017).

La tecnología de múltiple entrada que ofrece WiFi Alliance, donde existen múltiples salidas para múltiples usuarios, reduciendo de forma sustancial el tiempo de espera y por lo tanto aumentando la velocidad de conexión, permite conectarse de forma simultánea a varios dispositivos, mejorando la experiencia de varios usuarios conectados y compitiendo por tener el mayor ancho de banda en un conexión WiFi compartida. Son ejemplos de conexiones que requieren de esta tecnología, actividades como el streaming o los juegos online.

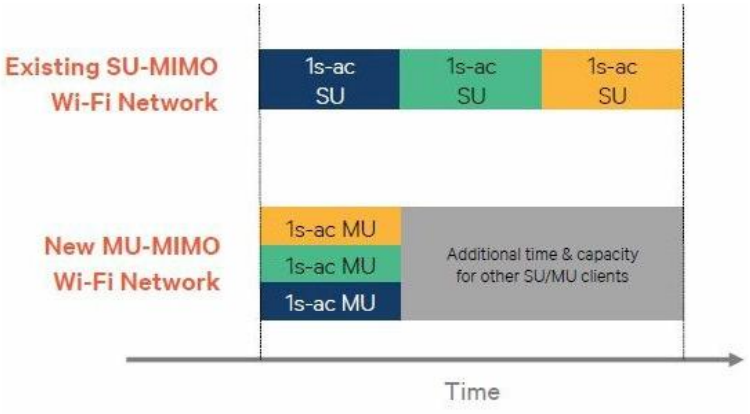


Ilustración 9. Comparativa SU-MIMO, MU-MIMO. Fuente: Web redeszone (2015)

Desafortunadamente en la actualidad no todos lo routers son compatibles con esta nueva tecnología. Los routers que trabajan con Wireless A, B, G y N no son compatibles con la tecnología MU-MIMO. A su vez, para obtener el máximo rendimiento posible, deben ser también compatibles con MU-MIMO. Se espera que para finales de 2017, se implante ya en dispositivos como tablets, teléfonos móviles, etc… La compañía Linksys ya hace uso de esta tecnología, y se espera que un futuro muy próximo, se implante de tal forma que sustituya al antiguo método de conexión, SU-MIMO.

2.4. Almacenamiento en la nube

Otro aspecto a tener en cuenta, ya que también ocupa su lugar en el sistema, es el almacenamiento en la nube. El sistema almacenará información en esta estructura de forma permanente, sin preocuparse por el tamaño necesario para el almacenamiento o si la información puede perderse.

El cloud computing o computación en la nube consiste en la posibilidad de ofrecer servicios a través de Internet. La computación en la nube es una tecnología nueva que busca tener todos nuestros archivos e información en Internet sin tener que poseer la capacidad suficiente para almacenamiento de la información.

La palabra nube, se utiliza para hacer referencia a Internet en representaciones como diagramas, haciendo referencia a la abstracción que supone de la infraestructura que representa. Según datos de un informe publicado en 2014 en Eurostat, Statistics Explained, por Konstantinos Giannakouris, María SMIHILY, en el mismo año 2014 solo un 19 por ciento de empresas europeas utilizaban la nube para gestionar correo electrónico y almacenamiento de archivos en formato electrónico. De este 19 por ciento tan solo un 46 por ciento utilizaba la nube para aplicaciones de carácter financiero o contable. Asimismo cuatro de cada diez empresas, es decir un 39 por ciento informaba de tener miedos o inquietudes relacionadas con riesgos de fallos de seguridad como primer limitante para hacer uso de este medio. Por consiguiente un 42 por ciento ponía como principal impedimento la falta de conocimiento de informática para utilizar el cloud computing.

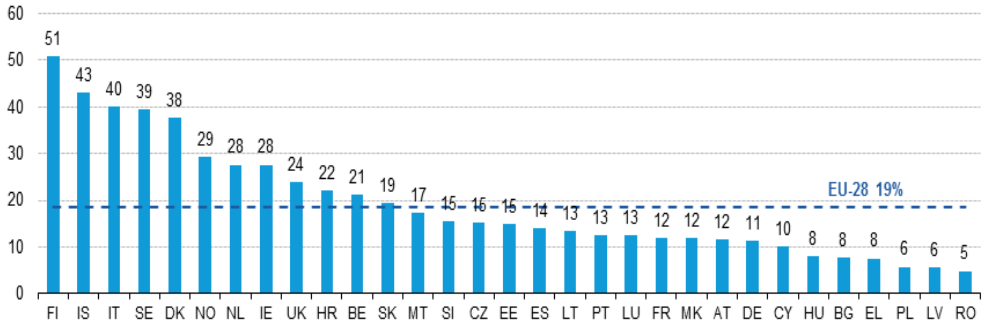


Ilustración 10. El uso de los servicios de computación en la nube (% de empresas por países europeos). Fuente: Dataprius (2014).

2.4.1. Ventajas del almacenamiento en la nube.

Son varias las ventajas que ofrece esta nueva forma de almacenamiento de información, todavía poco aceptada por la sociedad. A continuación se citan algunas de ellas:

- Bajo coste. Productos gratuitos o pagos mensuales fijos por utilización, sin costes adicionales, dado que no hay que invertir en gran infraestructura, ni en licencias
- Seguridad. Los datos siempre están seguros ya que la información viaja cifrada entre local y servidores.
- No hay necesidad de poseer una gran capacidad de almacenamiento.
- Mayor rapidez en el trabajo al estar basado en web.
- Acceso a toda la información.
- Acceso cuando quiera y donde quiera, sólo con conexión a Internet.

2.5. Privacidad

Actualmente, la sociedad tiene el derecho fundamental a la intimidad, es por ello que se deben tener en cuenta ciertos aspectos cuando se desarrolla un sistema de detección de dispositivos, en cuanto al almacenamiento de información referente a un usuario se refiere. Si pensamos en que un usuario, de manera consciente o no, permite ser visto por otros dispositivos, la información referente a la identificación de un dispositivo se puede almacenar sin ningún tipo de problema. Es un ejemplo cuando un usuario marca la opción modo visible en la configuración de Bluetooth, e incluso

cuando está compartiendo conexión a Internet con otros usuarios, es decir, que asume el rol de punto de acceso, debe mostrar el essid para que los demás puedan conectarse.

A su vez, existen herramientas que permiten realizar tareas de Sniffing, que consisten en monitorizar el tráfico de red existente, que se sitúan dentro de los límites legales siempre y cuando estas tareas se realicen en un ámbito didáctico. De tal forma, es importante saber que algunas herramientas explicadas en la documentación de este proyecto pueden caer en manos de personas con intenciones fraudulentas.

3. Análisis

En este apartado se detallan algunas ideas con el fin de planificar el desarrollo del proyecto. Durante la fase de análisis es posible que se tomen algunas decisiones o estrategias que en fases posteriores puedan ser modificadas o eliminadas respecto de lo que podríamos denominar una posible solución final. Asimismo, se realiza un estudio de las ventajas e inconvenientes de las metodologías más utilizadas para cada una de las tecnologías inalámbricas actuales, de forma que dicho estudio facilite la elección de aquellas que se ajusten a los objetivos del proyecto. A su vez, se elegirán las que mejor se adapten al sistema de leyes en España cuyo coste económico no superen ciertas cuantías.

3.1. Mecanismos de detección Bluetooth

En este subapartado, se realiza un estudio que trata la temática sobre los mecanismos o herramientas que existen en la actualidad para abordar detecciones de dispositivos conectados vía Bluetooth. Finalmente y tras el estudio de dichos mecanismos, se lleva cabo una comparativa, en la que se eligen aquellas que mejor se adaptan al desarrollo del proyecto.

3.1.1. Escaneo de dispositivos Bluetooth. La librería Bluecove.

Una de las herramientas utilizadas para el escaneo de dispositivos conectados vía Bluetooth, es la librería Bluecove. Esta librería java implementa la API JSR-82, que permite a los desarrolladores realizar aplicaciones que conectan con los dispositivos integrados Bluetooth en los equipos donde se van a llevar a cabo los escaneos. Bluecove es capaz de soportar los siguientes roles:

- SDAP: Descubrimiento de servicios.
- RFCOMM: Emulación del puerto serie. Basado en el protocolo L2CAP. Permite hasta 60 conexiones simultáneas entre dos dispositivos distintos.
- L2CAP: Enlace lógico y protocolo de adaptación. ESte protocolo permite transferencia de paquetes de hasta 64 KB.

- OBEX: Protocolo de intercambio de archivos.
- Para la detección de dispositivos, en este proyecto se va a hacer uso de la interfaz DiscoveryListener, incluida en la propia librería. Esta interfaz contiene una serie de métodos que se citan a continuación:

- deviceDiscovered: Este método se invoca cada vez que el dispositivos receptor recibe la señal de un dispositivo en las inmediaciones de su entorno.
- inquiryCompleted: El método inquiryCompleted indica con su ejecución, que la búsqueda de dispositivos ha sido finalizada.
- serviceDiscovered: Cada vez que el receptor encuentra un servicio Bluetooth disponible, se ejecuta este método.
- serviceSearchCompleted: Este método se invoca cada vez que finaliza el escaneo de servicios Bluetooth.

A efectos del sistema a desarrollar, parece que tan solo tienen nexo con los objetivos del mismo, los dos primeros métodos: deviceDiscovered e inquiryCompleted. El primer método, incluye las acciones necesarias que se deben realizar en el momento en el que se detecta un dispositivo, mientras que el segundo método se utiliza para informar de que una determinada búsqueda de dispositivos ha finalizado.

Para la obtención de información referente a un dispositivo detectado, la API incluye las dos siguientes funciones:

- getBluetoothAddress: Obtiene la dirección física del dispositivo Bluetooth remoto.
- getFriendlyName: Gracias a este método se obtiene el essid que identifica al dispositivo. El essid del dispositivo ha podido ser modificado por el usuario o por el contrario puede conservar el nombre de fábrica.

Para el montaje de una infraestructura de antenas receptoras, que utilizan la librería Bluecove como herramienta para escanear el entorno en búsqueda de dispositivos conectados mediante la tecnología Bluetooth se especifica en la siguiente tabla:

Dispositivo	Precio
Raspberry Pi 3	29,95 €
Dongle Bluetooth	5 €
Librería Bluecove	Gratuita
Total del montaje	**34,95 €**

Tabla 3. Presupuesto de montaje de sistema de detección Bluetooth. Fuente: Web RaspberryShop

3.1.2. El comando hcitool.

Este comando, perteneciente a la família de comando de Linux, realiza escanea de entorno buscando conexiones bluetooth. Obviamente, para asegurarse de que el comando funciona de manera correcta se debe asegurar de que existe un dispositivo Bluetooth en el equipo desde donde se va ejecutar el comando. Hcitool ofrece, mediante sus búsquedas, información referente a la dirección física y al nombre del dispositivo detectado.

Cabe decir, que un nombre de un determinado dispositivo detectado puede hacer referencia a la persona (debido a que ésta ha cambiado de forma voluntaria el nombre del aparato Bluetooth que lleva integrado). Por otro lado, el nombre detectado podría ser la marca o el modelo del dispositivo asignado en fábrica. Al ejecutar en un terminal Linux el comando "hcitool -help" la herramienta ofrece una serie de funciones atribuidas a dicho comando donde puede apreciarse que existe una función llamada "scan", de tal forma que parece tratarse de la función que lleva a cabo las búsquedas de conexiones Bluetooth.

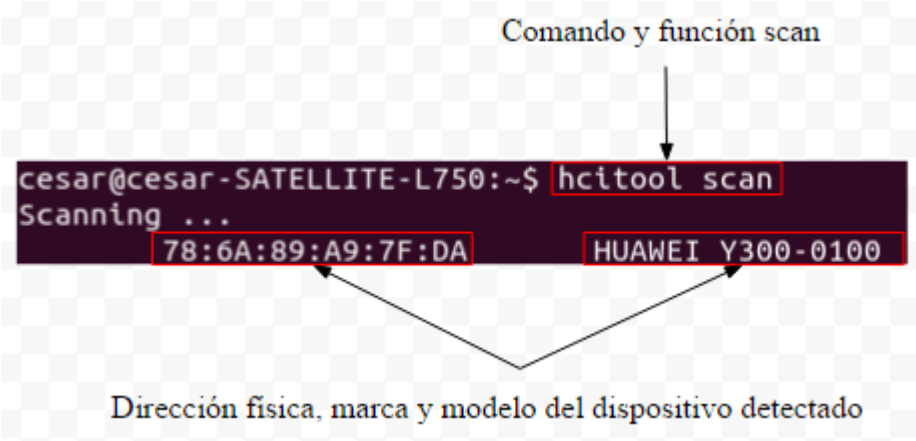


Ilustración 11. Resultado de comando hcitool. Fuente: Elaboración propia.

El coste para el montaje de una infraestructura que utiliza el comando hcitool de Linux como herramienta de detección de dispositivos, es el mismo que se tiene en cuenta en la detección de dispositivos mediante la librería Bluecove. Esto se debe a que se requieren los mismos dispositivos para el montaje.

3.1.3. Comparativa de mecanismos de detección Bluetooth

La función de ambos mecanismos descritos responden bastante bien a los objetivos del sistema, ya que al devolver un identificador único (dirección física) de un determinado dispositivo, puede llevarse a cabo un conteo de usuarios en lugares y tiempos determinados.

Mecanismo	Precio de dispositivo receptor	Alcance (metros)	Ratio de transmisión (Mbps)	Plataforma
Librería Bluecove	**34,95 €**	10 - 100	24 Mbps	Indiferente
Función scan de Hcitool	**34,95 €**	10 - 100	24 Mbps	Linux

Tabla 4. Comparativa entre mecanismos de detección Bluetooth. Fuente: Elaboración propia.

Como puede observarse en la tabla apenas se aprecian diferencias existente entre los dos mecanismos, es decir la potencia de detección y de transmisión no depende del mecanismo, si no de los dispositivos integrados e instalados que realizan las tareas de escaneo de entorno, no obstante, el mecanismo elegido para realizar búsquedas de conexiones Bluetooth es la librería Bluecove por las siguientes razones:

- La librería Bluecove, para poder ejecutarse, no depende del sistema operativo implantado en una máquina, mientras que por otro lado, la función scan del comando hcitool solo se ejecuta en computadoras con sistema operativo Linux.
- La facilidad de uso de Bluecove ayuda a su elección, debido a que se trata de una librería a la que tan solo hay que implementar sus métodos. Además ya dispone de funciones que devuelven la información que se precisa en el sistema.

3.2. Mecanismos de detección de conexiones a Internet.

En este subapartado, se realiza un estudio que trata los mecanismos o herramientas que existen en la actualidad, que en este caso abordan detecciones de dispositivos conectados a Internet. Finalmente y tras el estudio de dichos mecanismos, se lleva cabo una comparativa, en la que se eligen aquellas que mejor se adaptan al desarrollo del proyecto.

3.2.1. El dispositivo StingRay.

El dispositivo StingRay, según De Frutos (2017) explica en la web Androidsis, se utiliza para detectar dispositivos cercanos a las inmediaciones del mismo, de donde puede extraerse información referente al usuario del teléfono móvil detectado, como por ejemplo, el propio número de teléfono o la localización.

El modus operandi de Sting Ray, consiste en situarse entre el dispositivo móvil y el punto de acceso al cual el teléfono está conectado (man-in-the-middle). De tal forma que la información de cualquier dispositivo conectado a la red, queda registrada en un software asociado a StingRay.

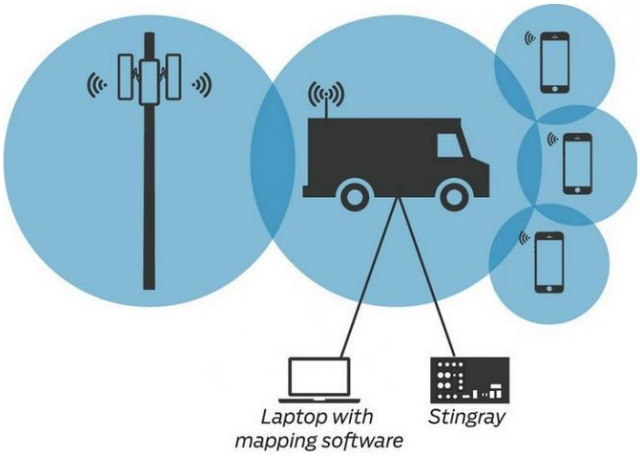


Ilustración 12. Esquema de funcionamiento de dispositivo StingRay. Fuente: Web Androidsis (2017).

Al parecer se trata de un sistema, que según el artículo citado, tiene una precisión bastante exacta en cuanto a la detección de dispositivos se refiere, y por lo tanto, a primera vista, se ajusta a los objetivos descritos más abajo. No obstante, existen ciertas desventajas a tener cuenta si se pretende hacer uso de esa tecnología, a continuación se citan las siguientes:

- Tienen un coste elevado. Un Sting Ray tiene un coste de 75.000 euros cada uno de ellos.
- Para llevar a cabo la localización exacta de un teléfono móvil se precisa de varios de estos dispositivos. Esto se debe a que utiliza el mismo principio de triangulación que emplean los dispositivos GPS para determinar la posición de un punto en el plano.
- Este dispositivo, es utilizado de forma general por el FBI, NSA (Agencia de Seguridad Nacional) y otras divisiones gubernamentales que pueden desempeñar actividades relacionadas con el espionaje.

Ilustración 13. Dispositivo StingRay. Fuente: Web Androidsis (2017).

El uso de este sistema para detectar información de dispositivos requiere de un StingRay y el software necesario para para interceptar dicha información. En la tabla siguiente se indican los coste de aplicar este sistema de detección:

Dispositivo	**Precio**
StingRay	7500 €
Software de detección	No especificado
Total del montaje	**+7500 €**

Tabla 5. Presupuesto para un montaje de sistema de detección haciendo uso de StingRay. Fuente: Web Androidsis (2017)

3.2.2. Software que utiliza StingRay.

Como ya se ha visto, el dispositivo StingRay es capaz de interceptar información debido a que éste se posiciona dentro del área de cobertura que encapsula a un determinado usuario y el dispositivo StingRay. Para la obtención de datos, necesita de un software que permita llevar a cabo tales actividades, de modo que según Gerardo (2013) publica en la web bitmovil.com, algunos de los programas que utiliza StingRay son los siguientes:

- FishHawk: Utiliza la configuración original del dispositivo para interceptar conversaciones telefónicas.

- Porpoise: Amplía las capacidades del dispositivo para añadir una funcionalidad extra: Obtención de mensajes de texto. Esta nueva capacidad requiere el uso de un pendrive para almacenar la información referente a los mensajes obtenidos.

3.2.3. Open BTS. Creación de infraestructuras propias GSM.

Open BTS, es un proyecto open source, es decir, de código abierto, utilizado para el desarrollo de un sistema que reemplaza a las tradicionales redes de compañías telefónicas, creando una BTS (Base Transceiver Station) privada, es decir el sistema es capaz de crear su propia red privada. Este sistema, permite la conexión a teléfonos móviles y poder realizar comunicaciones 2G mediante GSM (Sistema Global para las comunicaciones Móviles), e incluso 3g mediante UMTS para una mayor velocidad en la transmisión de datos. Este sistema, ha sido probado, según el artículo "OpenBTS: Cómo crear tu propia infraestructura GSM (2017)", en lugares remotos dando como resultados conexiones que proporcionaban a los usuarios cobertura de red privada permitiendo así, transferencias de datos entre dispositivos enrutando correctamente las llamadas de los usuarios dentro de la cobertura de la red. El nexo de esta metodología con los objetivos del sistema a desarrollar, se focaliza en la obtención de información de los dispositivos que se conectan a este tipo de redes.

En el artículo que publica Margaritelli (2016), es posible desarrollar una BTS casera y de bajo coste. En dicho artículo, explica de forma detallada, los pasos a seguir para la obtención de la misma. A su vez, para el montaje precisa de los dispositivos que se citan en la siguiente tabla, acompañados de su precio en el mercado.

Dispositivo	**Precio**
Módulo Blade RF x40	400 €
Raspberry Pi versión 3	50 €
Batería USB	35 €
2 antenas Quad-Band SMA	2 x 8 €
Micro SD mayor o igual a 8 GB	Desde 7 €
Total del montaje	**508 €**

Tabla 6. Presupuesto para un montaje casero de una infraestructura GSM. Fuente: Web Evilsocket (2016)

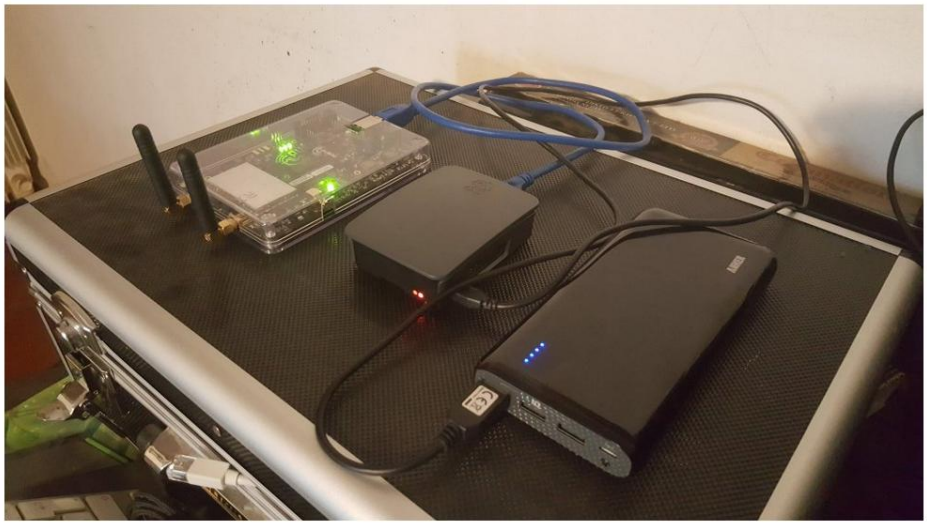


Ilustración 14. Montaje de una infraestructura GSM. Fuente: Web Evilsocket (2016)

3.2.3.1. El servidor Asterisk

Este servidor fue creado por Mark Spencer de Digium, empresa situada en el estado de Alabama (EEUU), que se dedica al desarrollo de software y hardware para la comunicación y telefonía, como es el caso de Asterisk. Esta herramienta tiene funcionalidad de central telefónica, es decir, recibe llamadas entrantes y las conecta con dispositivos destino dentro de la misma red privada o con dispositivos situados en otras redes exteriores.

Ilustración 15. Herramienta Asterisk. Fuente: Wikipedia (2017).

3.2.3.2. Inconvenientes de desarrollar una infraestructura propia de conexión GSM.

Aunque la idea de crear una estación propia de servicio de comunicación vía GSM parece atractiva, deben tenerse en cuenta algunos aspectos que se citan a continuación:

- En España es ilegal: Se precisa de licencias concedidas por compañías telefónicas, tras pagar un coste económico elevado.
- Su implantación debe llevarse a cabo en lugares privados y cubiertos, esto se debe a que el lugar no debe tener ningún tipo de cobertura telefónica de ninguna compañía.

3.2.4. La herramienta Wireshark

Esta herramienta, según Relancio (2013) explica en el artículo "Wireshark, un gran analizador de protocolos" de la web www.seas.es es capaz de monitorizar todos y cada uno de los paquetes que detecta nuestra tarjeta de red, es decir, realiza el trabajo de un sniffer con un interfaz atractiva e intuitiva a la cual se pueden aplicar una serie de filtros de una manera sencilla. Este programa se utiliza, de forma habitual, para observar la situación en la que se encuentra la situación de la red a la que nuestro dispositivo se encuentra conectado, o bien las redes del entorno. Por ejemplo, es posible ver qué dispositivos se encuentran conectados a una red en un momento determinado. Esto permite detectar intrusos.

En la actualidad, es habitual utilizar Wireshark para fines didácticos, aunque sin embargo esta herramienta, en muchas ocasiones puede emplearse para llevar a cabo acciones fraudulentas, como obtener información privada de un individuo u obtener la clave de acceso de tipo WEP de un determinado router, para acceder a redes privadas de manera gratuita.

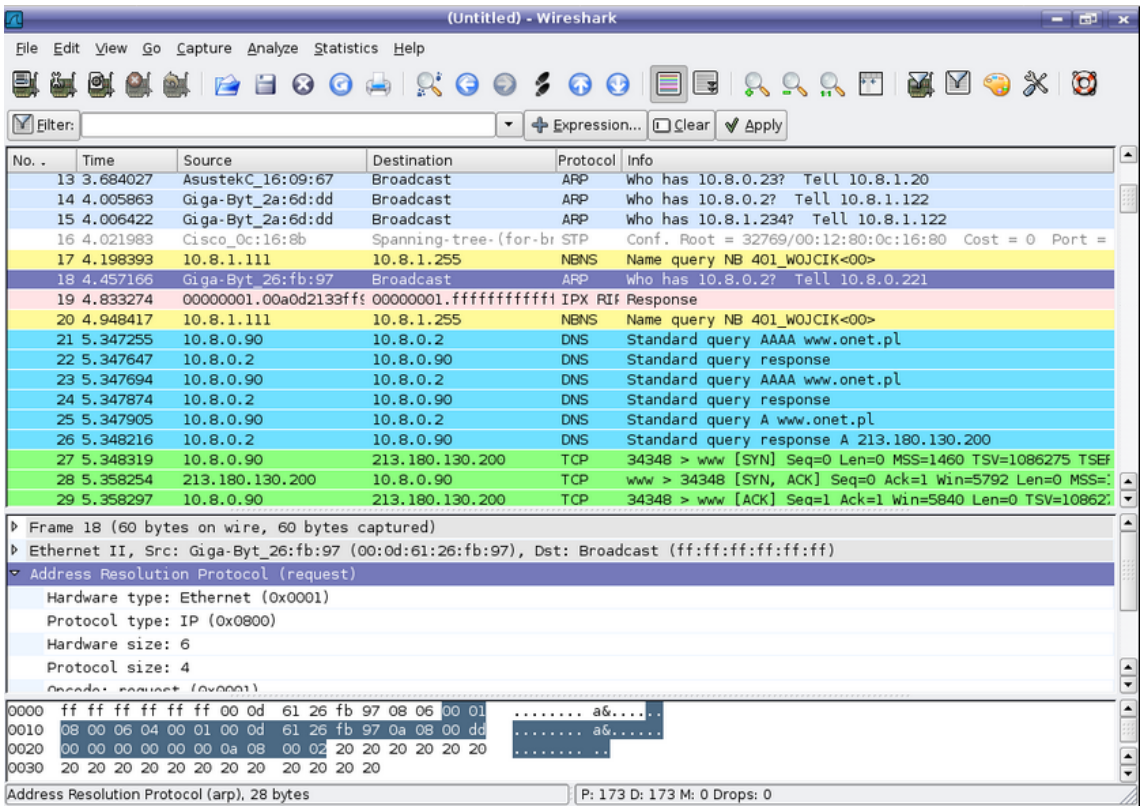


Ilustración 16. Interfaz gráfica de Wireshark. Fuente: Wikipedia (2017).

La herramienta ofrece además, una forma de recoger información sin utilizar interfaz gráfica. Se trata del comando tshark. Este comando utilizado en terminales, realiza las mismas tareas y permite filtrar la misma información que su versión gráfica, mediante una serie de parámetros que se añaden al propio comando.

Wireshark podría ser una buena candidata para utilizarse en el desarrollo del proyecto ya que, como se explica en el artículo citado, se utiliza para usos didácticos, aunque detecte información que identifica a un determinado dispositivo. Además, los datos que pretenden almacenarse en el sistema, son aquellas que hacen referencia, tan solo al distintivo del dispositivo. Se descarta, sin lugar a dudas, la gestión de cualquier otro tipo de información que pueda someter al sistema a permanecer fuera de los límites legales en este país.

Los costes del montaje de un sistema de detección de dispositivos conectados a Internet mediante la herramienta Wireshark, se reflejan en la siguiente tabla:

Dispositivo	Precio
Raspberry Pi 3	29,95 €
SunFounder RT5370 USB Wireless Network	10,99 €
Herramienta Wireshark	Gratuita
Total del montaje	**40,94 €**

Tabla 7. Presupuesto para un montaje de sistema de detección haciendo uso de la herramienta Wireshark. Fuentes: Webs Cosas de Móvil y RaspberryShop.

3.2.5. Detección de puntos de acceso.

Esta es una de las metodologías que no corren ningún peligro de cruzar la frontera que separa lo legal de lo ilegal. Esto se debe a que un determinado usuario de un dispositivo tiene la elección de permitir que su aparato sea visto o no por otros. Es un ejemplo cuando una persona desea compartir su conexión a Internet con otros usuarios, ya que éste permite que essid o nombre asignado al dispositivo sea visto por otros para poder conectarse. Aunque si bien es cierto, que a esta forma de acceso se le asigne una contraseña, no limita el poder visualizar dicho essid mediante algunas herramientas de escaneo de puntos de accesos como por ejemplo el comando “iwlist”, el cual su funcionamiento se explica más adelante, en la documentación de este mismo proyecto.

Para el montaje de una infraestructura que utiliza un sistema de detección de puntos de acceso, se requiere de los mismo dispositivos utilizados en para montar un sistema de detección monitorizando tramas, por tanto tiene el mismo coste, ya que el software a desarrollar implementa el comando "iwlist" y por lo tanto no suma al coste ningún importe adicional.

3.2.6. Comparativa de mecanismos de detección de dispositivos conectados a Internet

En este subapartado se pretende realizar un resumen de las ventajas e inconvenientes que presentan cada una de las metodologías comentadas a lo largo del capítulo, dando lugar así, a criterios que van a ayudar en la elección de una o varias de dichas tecnologías para el desarrollo del proyecto. La comparativa se lleva a cabo mediante el siguiente cuadro:

Tecnología	Precio de dispositivo receptor	Alcance	Ratio de transmisión	Legalidad en España
Detección de puntos de acceso	**40,94 €**	Depende del dispositivo de detección. Uno de bajo coste puede lograr un alcance de hasta 100 metros.	Hasta 1300 Mbps	Sí
Wireshark	**40,94 €**	Depende del dispositivo de detección. Uno de bajo coste puede lograr un alcance de hasta 100 metros.	Hasta 1300 Mbps	Sí
StingRay	**+ 7500 €**	Desde 500 metros	No especificado	No, tan solo en autoridades policiales en EEUU.
OpenBTS	**~ 500 €**	Depende del dispositivo receptor de conexiones empleado en la red.	No especificado	No. Existe una manera remota mediante pagos de cuantías elevadas a las compañías telefónicas del país.

Tabla 8. Comparativa de mecanismos de detección de conexiones a Internet. Fuentes: Wikipedia (2017), Evilsocket (2016), Androidsis (2017).

Aunque se podría pensar que ciertos mecanismos mencionados en la tabla podrían utilizarse, pese a que carezcan de legalidad en este país, ya que podría llevarse a cabo un experimento con fines didácticos, pueden encontrarse beneficios en los dos primeros mecanismos citados (detección de puntos de acceso y Wireshark). Ambas herramientas permiten una implantación de bajo coste, ya que lo único que se necesitaría es un dongle que sea capaz de detectar conexiones. Además las dos permiten obtener información que identifica de manera única a un dispositivo de detectado en tiempo de escaneo. En lo que respecta al ratio de transmisión, al igual que ocurre con el alcance, también depende del dispositivo receptor que el desarrollador desee implantar en su sistema, flexibilizando así, el ratio si se obtienen dispositivos receptores de mayor capacidad sobre esta característica.

3.3. Conclusiones tras el estudio de mecanismos.

Finalmente y una vez mencionadas las herramientas de detección de estas dos tecnologías y teniendo en cuenta, como criterios algunos factores que se reflejan en las tablas comparativas como compatibilidades con el sistema legal de este país, el coste de llevarlas a cabo o incluso las dependencias sobre plataformas que tienen, se decide finalmente que este proyecto utilizará los siguientes mecanismos de detección:

Bluetooth	**WiFi**
Librería Bluecove	Monitorización de tramas
	Detección de puntos de acceso

Tabla 9. Mecanismos de detección seleccionados por tecnología. Fuente: Elaboración propia.

4. Diseño del sistema.

En este capítulo se plantean algunas estrategias que pueden facilitar el desarrollo del programa. Estas ideas, describen con más detalle, los roles que participan en el sistema y las funciones que desempeñan cada uno de ellos. Además se dan a conocer posibles soluciones, que deben tenerse en cuenta para solucionar problemas que pueden surgir durante el desarrollo, y que afectan al rendimiento y a la comunicación entre dichos agentes.

Asimismo, se exponen ejemplos de tipo de información, considerada relevante para ser almacenada, e incluso se plantea una herramienta para poder visualizar e interpretar los datos almacenados. Es posible que en un futuro, algunas de las estrategias aquí descritas se modifiquen.

4.1. Estrategias para el desarrollo del sistema.

- **Agentes que participan en el sistema:** El sistema se organiza en un conjunto de antenas receptoras que escanean el entorno y un servidor que se encarga de poner en marcha dichas antenas y enviar la información recopilada por éstas a una base de datos.
- **El trabajo recae en mayor medida en el dispositivo receptor:** El mayor trabajo lo realizan las antenas receptoras. El servidor accede cada cierto tiempo al receptor para recoger la información obtenida. El tiempo de acceso del servidor a los receptores debe ser una variable que por un lado no tenga un valor muy elevado, ya que **a mayor tiempo de acceso, mayor carga de datos** para transferir.
- **La comunicación entre los agentes participantes:** Debe existir un medio de comunicación entre las antenas receptoras y el servidor, para ello se va a hacer uso de una base de datos
- **Tipo de información:** La información obtenida debe de poder identificar de manera exclusiva a un determinado dispositivo captado por el receptor. Además debe registrarse información de la propia antena como pueden ser las coordenadas de localización. A su vez, debe existir una lista que contenga aquellos dispositivos que no se desean almacenar en la base de datos. Por

ejemplo, si en un espacio cerrado una antena receptora recibe la señal de un dispositivo que siempre permanece en el lugar, como es el caso de una impresora o una televisión, no resulta relevante almacenar información sobre el dispositivo. Por este motivo, el identificador del dispositivo estará contenido en dicha lista.

- **Visualización:** Una vez almacenados los datos se debe de hacer uso de una herramienta que permita analizar e interactuar con una cantidad masiva de datos, recabando información mediante peticiones a la base de datos.
- **El sistema de gestión de bases de datos:** Para el almacenamiento de información debe hacerse uso de un sistema relacional. Esto se debe a la necesidad de tener varias tablas conectadas. Por ejemplo, una primera tabla debe mostrar información de todas las antenas conectadas al sistema, y otra tabla puede mostrar información sobre los dispositivos captados por las antenas.

4.2. Soluciones propuestas

En este subapartado, se van a proponer las soluciones que van a facilitar el alcance de los objetivos que contribuyen al correcto desarrollo del sistema. El subapartado "Soluciones propuestas" se va descomponer en los tres módulos en los que se divide el sistema (detección de dispositivos, almacenamiento de información y visualización o control de la misma), aportando una serie de soluciones a cada uno de los módulos citados.

4.2.1. Detección de dispositivos

El sistema debe tener un conjunto de antenas receptoras y un servidor web que consulte la información obtenida por las antenas. Para llevar a cabo la función de antenas receptoras se hará uso de ordenadores de placa reducida, es decir las Raspberry Pi y se le instalarán los medios necesarios para que sean capaces de recibir señal Wifi y Bluetooth.

4.2.1.1. Dispositivos receptores: Raspberry Pi

Las Raspberry Pi son ordenadores de bolsillo, también llamados ordenadores de placa reducida, que son capaces de realizar las mismas actividades que llevaría a cabo

un ordenador corriente. Aunque las capacidades de procesamiento o almacenado, son todavía, inferiores a los ordenadores comunes, estos computadores son utilizados en una gran variedad de proyectos. Algunas de las ventajas que presentan son las siguientes:

- Tamaño reducido: Son ordenadores pequeños que pueden ser integrados en cualquier sistema.
- Disponen de puertos USB, HDMI, mini jacks de audio y RJ45 para conexión por cable de red vía Ethernet.
- Además incluyen (a tamaño reducido), el mismo Hardware que un ordenador habitual, como memoria RAM, un procesador, una tarjeta gráfica, una tarjeta Bluetooth integrada y en su última versión además tienen una tarjeta Wireless para conexiones Wi-Fi.
- Consumen muy poco. Un Raspberry puede consumir entre 4 y 5 vatios mientras trabajan. En su lugar un ordenador corriente puede consumir del orden de 30 vatios. Asimismo un ordenador corriente, en su estado de reposo puede llegar a consumir hasta 5 vatios mientras que una Raspberry PI tiene su consumo, en estado de reposo, en 0,5 Vatios.

Modelo Raspberry Pi	Frecuencia de procesador	Capacidad de memoria Ram
Raspberry Pi 2 Model B	900 Mhz (overclock 1 Ghz)	LPDDR2 de 1 GB
Raspberry Pi 3	1,2 Ghz	LPDDR2 de 1 GB

Tabla 10. Comparativa de las dos Raspberries utilizadas en el desarrollo del sistema. Fuente: Web RaspberryShop (2017).

Ilustración 17. Raspberry Pi 3. Fuente: Web RaspberryShop (2017).

La configuración de funciones que la Raspberry debe llevar a cabo para lograr el alcance del principal objetivo, es decir, contabilizar personas, se va a desarrollar sobre un sistema operativo totalmente adaptado para este tipo de ordenadores de tamaño reducido.

4.2.1.1.1. Sistema Operativo Raspbian

Raspbian es una distribución de GNU/Linux y por tanto se trata de un sistema operativo libre basado en Debian, que fue diseñado para instalarse en las Raspberry. Para este proyecto se utiliza la versión Raspbian Jessie Lite por ser una versión ligera. Este aspecto es una ventaja, ya que el sistema operativo puede instalarse en memorias de SD de tan solo 8 gigabytes.

Una de las herramientas que integra este sistema operativo es "raspi-config", que permite modificar configuraciones en el sistema operativo sin necesidad de editar manualmente ningún fichero de configuración, así como la hora y fechas o idiomas para el teclado.

4.2.1.1.2. Programas para instalar en las antenas.

En el sistema, cada Raspberry dispondrá de dos programas (WiFi y Bluetooth) de rastreo desarrollados en lenguaje de programación Java, los cuales se van a ejecutar en paralelo, sobre un sistema operativo Raspbian. Estos programas, cuya función será rastrear sus entornos buscando dispositivos conectados de manera inalámbrica, se

encargarán de almacenar en un fichero, en formato hoja de cálculo, toda la información obtenida.

En este proyecto se va a realizar una comparativa de las dos tecnologías de detección WiFi escogidas en el capítulo anterior (Monitorización de tramas y escaneo de puntos de acceso). Por lo tanto, una antena que se sitúe en un determinado lugar tendrá instalado un programa que detecta puntos de acceso y otra antena va a tener instalado el programa de monitorización de tramas, además del programa de detección Bluetooth en ambas.

Antena en ubicación A		**Antena en ubicación B**	
	Detección Bluetooth **Detección de puntos de acceso**		Detección Bluetooth **Monitorización de tramas**

Tabla 11. Programas a instalar en antenas receptoras. Fuente: Elaboración propia.

4.2.1.1.3. La lista negra.

La lista negra es una tabla que contiene aquellos nombres de dispositivos o máscaras, que deben ser comprobados en el momento de una detección, para evitar almacenar detecciones estáticas. Una detección estática se considera aquella que detrás del dispositivo detectado no se encuentra una persona. Por ejemplo en el caso de un televisor que siempre permanece en el lugar escaneado, no merece la pena ser almacenado ya que tras él no se encuentra ninguna persona. De igual forma sucede con una impresora. De esta manera la información que se almacena pretende ajustarse al máximo a la realidad, asegurándose así, de que un dispositivo equivale a una persona (teléfono móvil o reloj Bluetooth) con casi toda certeza. Por lo tanto, si el administrador de antenas, en un momento de visualización de información, se da cuenta de que existen nuevos dispositivos que no merecen la pena ser recopilados, tan solo debe escribir en la tabla de la base de datos la nueva máscara (TV o Laser) o el nombre completo de un dispositivo a evitar (TV LG [3600] o LaserJet 1200). La inserción de máscaras en la lista negra puede resultar útil si en el lugar existen más de un televisor y se ha detectado que sus nombres comienzan por "TV", de esta manera basta con insertar

"TV" para que se desprecien todos los dispositivos detectados cuyos nombres comiencen por estas dos letras.

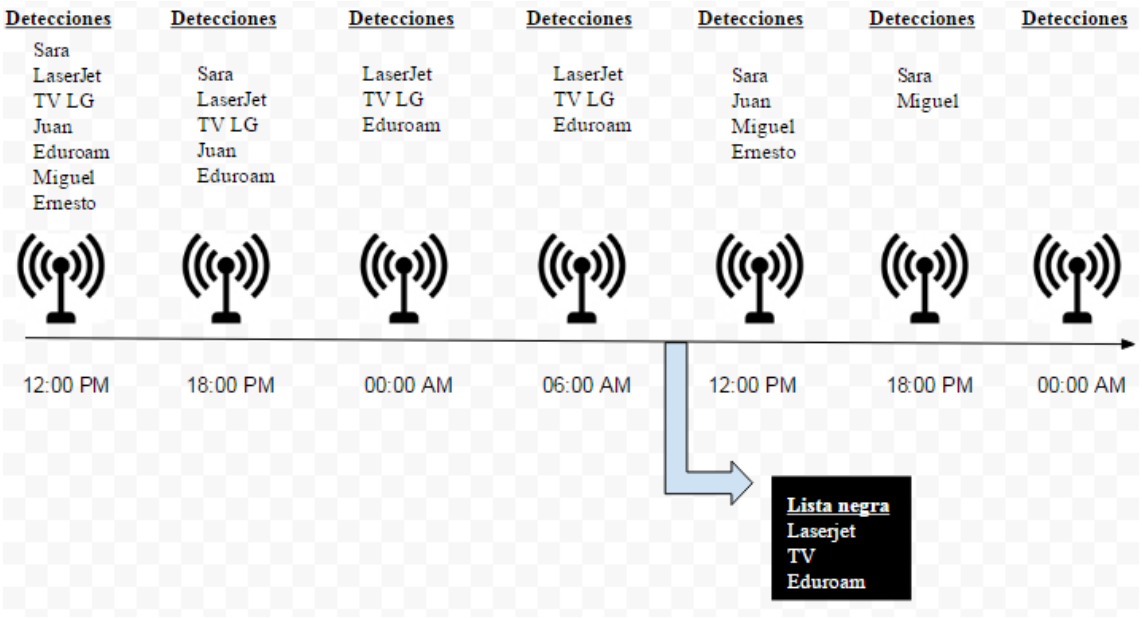


Ilustración 18. Instalación de antena. Adición de máscaras a lista negra. Fuente Elaboración propia.

Una vez instalada en un determinado lugar, se aconseja detectar qué dispositivos deben descartarse ya que no forman parte de la población. Una forma práctica de conseguir detectar estos dispositivos, es añadir a lista negra aquellos nombres que se detectan en horas de baja actividad o inactividad en el lugar, es decir que en condiciones normales no debería existir ningún tipo de población. Estos dispositivos que se detectan, durante horas de actividad pueden pasar desapercibidos. Esta tarea, se aconseja llevar a cabo lo antes posible tras la instalación de una antena, para la obtención de medidas netas.

Una de las características que se propone, es la capacidad de insertar nombres de dispositivos en lista negra de forma automática. De manera que cuando una antena receptora se implanta en un determinado lugar la primera tarea aconsejable realizar, es la inserción de dispositivos de carácter estático. Esta tarea, se puede llevar a cabo de manera manual o automática. Para el desarrollo de la segunda opción, la antena debe permanecer en un lugar durante horas de baja actividad o inactividad. Esto se debe a que la inserción se realiza tras consultar a las tablas en base de datos que almacenan

nombres de dispositivos, que han sido detectados entre estas horas de baja o nula población.

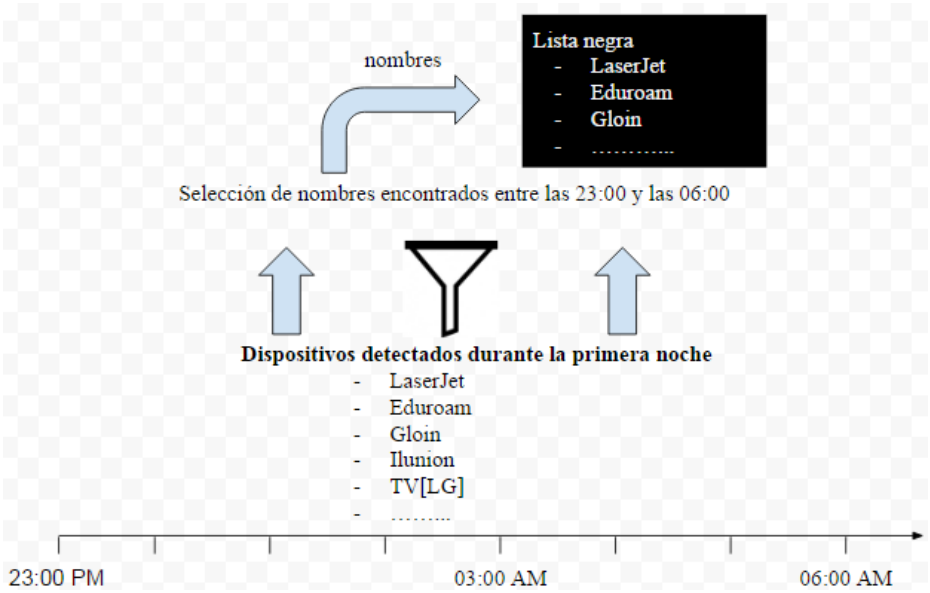


Ilustración 19. Inserción automática en lista negra. Ejemplo de un supermercado. Fuente: Elaboración propia.

4.2.1.1.4. La información a disposición del servidor web.

Cuando una antena detecta un dispositivo en una de sus búsquedas, si cuyo nombre no se encuentra en la lista negra, la antena almacena la información referente al dispositivo detectado, en un fichero en formato XLS. La información de este fichero debe ser consultada por el servidor web, de modo que, éste debe de poder tener al acceso al fichero. Para solventar este problema y alcanzar el objetivo "**Los dispositivos receptores ponen la información a disposición del servidor a través de un servidor web** ", la antena integra un servicio web Tomcat en el que va a alojar el fichero, permitiendo de esta manera el acceso al fichero y poder realizar las consultas.

4.2.1.1.5. Integraciones a la Raspberry Pi para recepción de dispositivos.

En este sistema, se van a utilizar dos Raspberry, las cuales integran tarjetas Bluetooth para poder ser interconectados con más dispositivos. Además a partir de la versión 3, integran también una tarjeta Wireless que viene muy bien para llevar a cabo los escaneos de dispositivos conectados vía WiFi. Sin embargo, se dispone de una

Raspberry en la versión dos y una Raspberry en la versión tres, de modo que para la primera, se le debe añadir una antena repetidora, para que pueda ejecutar escaneos y detecciones de puntos de acceso WiFi. La antena que se ha utilizado dispone de conexión USB de manera que basta con enchufarla y esperar a que el ordenador la reconozca (Plug and Play).

Ilustración 20. Raspberry con antena conectada vía USB. Fuente: Web Cosas de Móvil (2017).

En cuanto a experiencias vividas con las tarjetas Bluetooth integradas se aprecia que el alcance no supera los diez metros de distancia, de modo que se opta por incluir en ambas Raspberry, dos dongle o dispositivos de recepción de dispositivos remotos, para la tecnología Bluetooth. Estos dispositivos son muy comunes en la sociedad y son muy fáciles encontrarlos. También son dispositivos USB, así que basta con enchufarlos, al igual que la antena receptora de dispositivos WiFi en la Raspberry de versión dos, y esperar a que la Raspberry reconozca el dispositivo.

Un factor que hay que tener en cuenta, es la amplitud de los lugares de los cuales se pretende obtener información, así que la capacidad de los dongle de recepción Bluetooth debe ser proporcional a dicha amplitud de los lugares. Dado que por el momento solamente se tienen funcionamiento dos Raspberry, una ubicada en cafetería, y la otra en unas oficinas de trabajo, cuya amplitud superan los 100 metros cuadrados, los dongles deben tener una capacidad de alcance al menos superior a los 50 metros, para tomar muestras de datos significativas.

Ilustración 21. Dongle Bluetooth. Fuente: Web Ioffer (2017).

Ocurre lo mismo en el caso del WiFi, dado su poca capacidad de alcance, la tarjeta Wireless integrada en la Raspberry en la versión tres, no es suficiente para obtener una muestra que resulte relevante para la experiencia. Por lo que también se le añade una antena receptora de puntos de acceso WiFI a la Raspberry que se dispone en la versión tres, y aumentar así, su capacidad de alcance.

4.2.1.1.6. La conexión a Internet de la Raspberry.

A su vez, las antenas receptoras, deben tener conexión a Internet, esto es así ya que cada vez que se conecten al sistema, deben acceder a la instancia RDS de Amazon Web Services, que se encuentra en la nube. De modo que para ofrecer portabilidad a la antena, se toma la decisión de conectarlas a Internet mediante conectividad 3g. Para ello se han formalizado dos contratos con una compañía telefónica y se han obtenido dos tarjetas SIM con conexión de datos cada una de ellas.

Un aspecto importante a tener en cuenta, es la elección de la compañía. Existen compañías telefónicas que asignan falsas IPs públicas a las conexiones de sus clientes, es decir se trata de IPs que pertenecen a redes privadas y por lo tanto, si se pretende alojar una web en un equipo cliente de una compañía telefónica de este tipo, la web creada no puede ser accedida desde un equipo que pertenece a otra red diferente. La compañía elegida para el desarrollo del proyecto asigna IPs públicas reales. Estas IPs son dinámicas y cambian cada dos días aproximadamente. Por lo tanto, gracias a la correcta elección de compañía proveedora de servicios de Internet, se dispondrá de servicios web alojados en las antenas receptoras perfectamente accesibles desde redes ajenas.

En cuanto a la conexión de las tarjetas en las Raspberry, se ha hecho uso de los comunes módems usb, en la actualidad casi desfasados, pero sin embargo, son de gran ayuda para alcanzar el objetivo "Las antenas deben tener acceso a Internet".

Ilustración 22. Dispositivo módem USB vodafone. Fuente: Web Gizmos (2017).

Estos módems contienen una cavidad por donde se inserta la tarjeta SIM, parecida a las de los teléfonos móviles. Una vez insertada la tarjeta, tan solo basta con ser conectada a la Raspberry mediante USB (Plug And Play), donde el dispositivo será reconocido en unos instantes, sin necesidad de reiniciar la Raspberry.

4.2.1.2. El servidor web

El desarrollo del servidor web se lleva a cabo con la herramienta Webratio. Esta plataforma permite el desarrollo de aplicaciones web sin escribir una sola línea de código. El usuario tan solo diseña el modelo conceptual de la aplicación y la plataforma se ocupa de escribir el código fuente necesario. La conexión a la base de datos será de tipo MySQL. El servidor debe tener asignadas una serie de rutinas que le permiten acceder a las antenas receptoras cada cierto tiempo programado. Para poder localizar la ubicación de las antenas y obtener la información referente a detecciones, el servidor debe extraer la URL de la tabla de la base de datos. Asimismo, en el servidor se debe añadir formularios donde se van a poder activar las antenas conectadas al sistema, mediante la asignación de un tiempo de recarga. Además, debe contar con otro formulario para la inserción de nombres de dispositivos a lista negra. En un futuro, tras ser insertado un nombre de dispositivo en lista, si una antena detecta un dispositivo con dicho nombre, el dispositivo se ignora y no se almacena.

El servidor web, se aloja en una instancia proporcionada por el servicio Elastic Beanstalk de Amazon sobre una plataforma Tomcat. El propio servicio, una vez despliega una versión del servidor web proporciona un nombre de dominio a la zona donde se encuentra alojada dicha versión del servidor.

4.2.1.3. Amazon Web Services. Elastic Beanstalk.

Esta herramienta o servicio, también proporcionado por Amazon Web Services, ofrece hosting o almacenamiento de páginas web. El hosting que pertenece a la capa gratuita de Amazon no debe superar un límite de tiempo de un año. Este servicio contiene plataformas como Apache Tomcat, entre otras que hacen que desplegar una aplicación web en la nube sea una tarea fácil de desarrollar. Basta con subir un fichero war o archivo de aplicación web y el propio servidor web de Elastic Beanstalk se encarga de extraer los archivos contenidos en el war y desplegar la aplicación de forma automática proporcionando un nombre de dominio para el acceso.

El blog cuenta con un vídeo tutorial que explica cómo generar un plan de despliegue el cual crea un fichero en formato war, necesario para realizar la subida de aplicación a Beanstalk y cómo este servicio de Amazon junto con Tomcat (servicio web utilizado en el sistema a desarrollar), es capaz de desplegar la aplicación y posteriormente generar un nombre de dominio accesible al hosting en el que se encuentra alojada la web.

4.2.1.4. El tiempo de recarga.

Tanto para el caso del programa de detecciones Bluetooth como el de detecciones vía WiFi, los dispositivos receptores tienen un tiempo, el cual se programa desde el servidor web, que indica el tiempo que debe existir entre que finaliza un escaneo de entorno y da comienzo el siguiente. Este tiempo es una variable a tener en cuenta ya que afecta directamente a dos factores fundamentales para el almacenado de datos.

Valor de variable temporal	Factores afectados
Tiempo muy elevado	La integridad de la información recibida. Si el tiempo tiende a ser extremadamente largo, puede perderse información relevante, ya que puede ocurrir que un determinado dispositivo visite el lugar y el dispositivo receptor no lo detecto debido a su tiempo de descanso.
Tiempo muy bajo	Datos altamente redundantes. Puede ocurrir que un usuario visite el lugar y permanezca allí durante unas horas. Si el tiempo de refresco es extremadamente bajo, se almacenará información de carácter redundante. Por ejemplo si un alumno permanece una hora en cafetería y el tiempo de recarga es de sesenta segundos, es decir un minuto, se producirán aproximadamente sesenta inserciones, sin embargo si el tiempo de recarga está en dos minutos, las inserciones en el fichero, aproximadamente se verán reducidas a la mitad.

Tabla 12. Contraste de valores para el tiempo de recarga. Fuente: Elaboración propia.

La web cuenta con la página principal, en la cual el usuario puede visualizar la ubicación de la antena y configurar el tiempo de recarga de la misma. Este tiempo, el usuario, puede configurarlo de forma individual o masiva, es decir, puede configurar un tiempo para todas las antenas conectadas en el momento o para una en concreto, indicando el identificador de antena para la cual se le va asignar el valor del tiempo.

4.2.2. La base de datos.

La base de datos va a ser gestionada por dos agentes participantes en el sistema, una puerta de enlace o gateway que conecta la herramienta la base de datos con la herramienta de visualización (descrito más adelante en este mismo capítulo), y el servidor web que va a recoger los datos obtenidos por las antenas y los va almacenar en la base de datos. El tipo de información que puede considerarse de mayor relevancia, y por lo tanto debe almacenarse, es **el identificador** de la antena receptora, **las coordenadas** del lugar donde se encuentra, **el nombre del dispositivo captado** en un escaneo, el momento en el que ha sido recogida la información del dispositivo y para

evitar que en un mismo escaneo se detecten dos dispositivos con el mismo nombre se incluye la **dirección física** del dispositivo.

Como se ha comentado con anterioridad, la información obtenida se pretende almacenar en una base de datos que utiliza un **sistema gestor de base de datos de tipo relacional**. Esto se debe a la dependencia existente entre tablas del sistema. La base de datos va a estar alojada en la nube y ofrecerá escalabilidad de carácter horizontal, lo que va a permitir despreocuparse por el espacio que van a ocupar los datos recogidos por las antenas. Para cumplir con los requisitos relacional y escalable se va a hacer uso del servicio RDS que ofrece Amazon Web Services.

4.2.2.1. Amazon Web Services: Instancias RDS

El servicio RDS de Amazon puede ser una buena idea para alojar una base de datos, ya que el servicio ofrece un sistema que facilita las tareas de configuración, utilización y escalado de una base de datos relacional en la nube. Además ofrece escalabilidad de carácter horizontal, donde dicho escalado es totalmente transparente al usuario. Esto significa que el sistema está preparado para el almacenamiento masivo de información bajo demanda, de manera que no debemos de preocuparnos por el crecimiento de datos en nuestras tablas.

En los anexos de la documentación de este proyecto se incluye un enlace a un blog donde se encuentra un vídeo tutorial en el que se detallan los pasos a seguir para la creación de una instancia RDS de Amazon Web Services.

Una vez se tiene un espacio donde poder almacenar la base de datos, se procede a la creación de tablas y atributos, para ello se hace uso de la herramienta MySQL Workbench.

4.2.2.2. MySQL Workbench

MySQL Workbench es una herramienta para la gestión de bases de datos MySQL (creación, edición y borrado de tablas). Basta con añadir la dirección del servidor MySQL junto con las credenciales para tener acceso a las bases de datos.

Esta aplicación debe conectar conectar con el servidor MySQL de la instancia RDS de Amazon, ya que es allí donde se encuentra nuestra base de datos para poder crear las tablas y las relaciones existentes entre ellas. El programa al iniciar pide la URL donde se encuentra el servidor MySQL junto con las credenciales de acceso a la base de datos. A partir de aquí pueden realizarse toda clase de gestiones sobre la misma como consultas, inserciones, modificaciones, borrados y toda clase de modificaciones sobre atributos de tablas.

4.2.2.3. El modelo de datos del sistema.

En este apartado se detalla la estructura de tablas del sistema y las relaciones entre ellas. Hay que considerar que debe almacenarse la información de un receptor, los datos obtenidos de los escaneos y una lista negra que contiene aquellos identificadores de dispositivos que no se consideran relevantes por su carácter estático. Esta lista negra puede ser común a las dos tipos de escaneo (Bluetooth y WiFi). Por lo tanto la especificación de tablas queda de la siguiente forma:

Nombre de la tabla	**Descripción**
Información_Antena	Almacena información de dispositivos receptores
Nombre de atributos	**Descripciones**
idAntena 🔑	Identificador de antena. Tipo de atributo: entero
nombreUbicacion	Ubicación de la antena. Tipo de atributo: texto
dirFileBluetooth	Dirección URL del fichero de datos Bluetooth. Tipo de atributo: texto.
Coordenadas	Coordenadas de localización del receptor. Tipo de atributo: texto.
Refresco	Valor de refresco de escaneo de la antena. Tipo de atributo: entero.
Latitud	Latitud de las coordenadas de localización de la antena. Tipo de

	atributo: doble.
Longitud	Longitud de las coordenadas de localización de la antena. Tipo de atributo: doble.
Lectura del fichero	Atributo de carácter bandera que cambia de valor cuando el servidor lee el fichero de datos de la antena. Tipo de atributo: entero.
dirFileWiFi	Dirección URL del fichero de datos WiFi. Tipo de valor: texto.

Tabla 13. Tabla de información de antena en la base de datos. Fuente: Elaboración propia.

Nombre de la tabla	Descripción
Datos_Wifi	Almacena información de los dispositivos captados por las antenas receptoras
Nombre de atributos	**Descripciones**
idAntena (Foreign key to Información_Antena)	Identificador de la antena que ha captado el dispositivo. Tipo de atributo: entero.
essidDevice	Distintivo del dispositivo. Este puede ser de fábrica, donde normalmente se indica la marca y el modelo del dispositivo captado, o en algunos casos, el usuario sustituye este identificador por otro alias. Tipo de atributo: texto
tiempoAcceso	Tiempo en el cual, un dispositivo es captado una antena. Tipo de atributo: temporal.
macDevice	Dirección física del dispositivo. Tipo de atributo: texto.

Tabla 14. Datos de conexiones vía Wifi en base de datos. Fuente: Elaboración propia.

Nombre de la tabla	Descripción
Datos_Bluetooth	Almacena información de los dispositivos captados por las antenas receptoras
Nombre de atributos	**Descripciones**
idAntena (Foreign key to Información_Antena)	Identificador de la antena que ha captado el dispositivo. Tipo de atributo: entero.
direccionLocalDisp	Dirección física del dispositivo. Tipo de atributo: texto.

tiempoAccesoDisp	Tiempo en el cual un dispositivo es captado una antena. Tipo de atributo: temporal.
Nombre dispositivo	Distintivo del dispositivo. Este puede ser de fábrica, donde normalmente se indica la marca y el modelo del dispositivo captado, o en algunos casos, el usuario sustituye este identificador por otro alias. Tipo de atributo: texto.

Tabla 15. Datos de detecciones Bluetooth en base de datos.
Fuente: Elaboración propia.

Nombre de la tabla	**Descripción de los datos almacenados**
ListaNegra	Esta tabla almacena nombres o máscaras de palabras, de tal forma que si una antena receptora detecta un dispositivo cuyo nombre se corresponde en su totalidad o parcialidad con algunos de los que aparecen en esta tabla, la información de este dispositivo no se almacena. Es un ejemplo todos los dispositivos cuyo nombre contiene la palabra Láser. Debido a que probablemente se trata de una impresora, este dispositivo no se almacena por considerarse de carácter estático.
Nombre de atributos	**Descripciones**
NombreDispositivo	Nombre o patrón de un dispositivo. Tipo de atributo: texto.

Tabla 16. Tabla de de lista negra. Fuente: Elaboración propia.

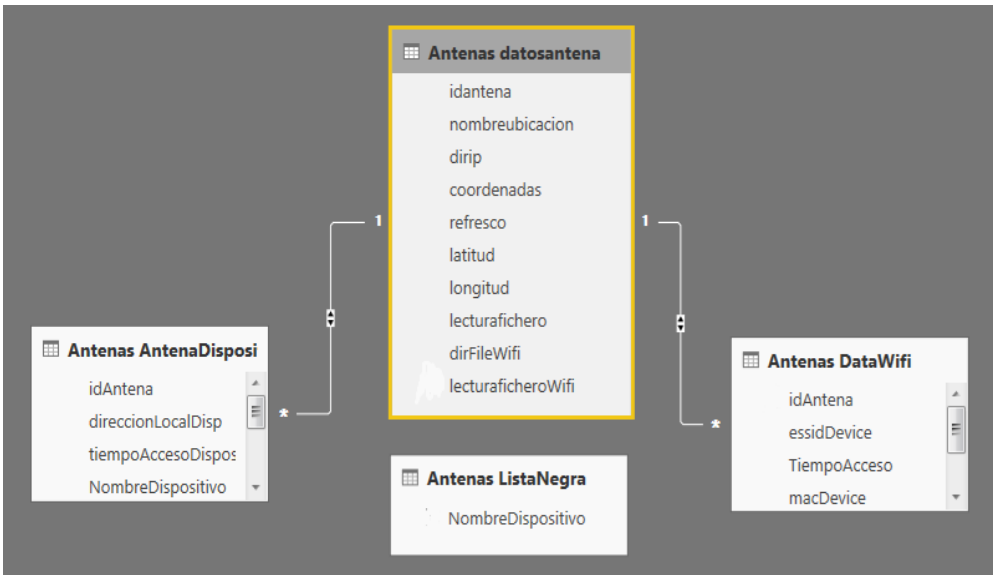


Ilustración 23. Modelo relacional del sistema. Fuente: Elaboración propia.

4.2.2.4. La base de datos como sistema de comunicación entre servidor y antenas.

La base de datos del sistema juega un papel fundamental, ya que aparte de almacenar información y comunicar con la herramienta de visualización para mostrar los datos recogidos en informes, se utiliza como herramienta de comunicación entre servidor web y antenas. Cuando la antena se conecta al sistema, indica el lugar donde se encuentran los ficheros de datos que el servidor debe leer y recoger la información, para a continuación almacenarla en la base de datos. La tabla que contiene información de la antena, tiene dos atributos de carácter bandera, uno para información WiFi y otro para Bluetooth, donde el servidor cambia el valor de 0 a 1 cada vez que accede a los datos y la antena retorna a 0 de nuevo el valor del atributo en el momento que sabe que el servidor ha accedido a la información y por consiguiente interpreta, a su vez, que la información ha sido almacenada de forma permanente en la base de datos del sistema. En este momento la antena borra el fichero de datos para almacenar nuevos datos.

4.2.3. Herramienta de visualización Power Bi.

Para visualizar la información obtenida por las antenas, se va a hacer uso Power Bi. Se trata una herramienta de Business Intelligence que tiene la capacidad de conectarse a una base de datos y mostrar la información mediante el uso de informes con gráficos interactivos, es decir, el usuario puede interactuar con estos informes para visualizar la información de una manera u otra. La aplicación cuenta con conexiones a bases de datos de varios tipos como MySQL, SQL Server, entre otras. Pero no solo es capaz de recopilar datos de una base de datos, si no que también reconoce como fuente de datos a ficheros csv, excel o ficheros propios de Power Bi con extensión .pbix, entre otros.

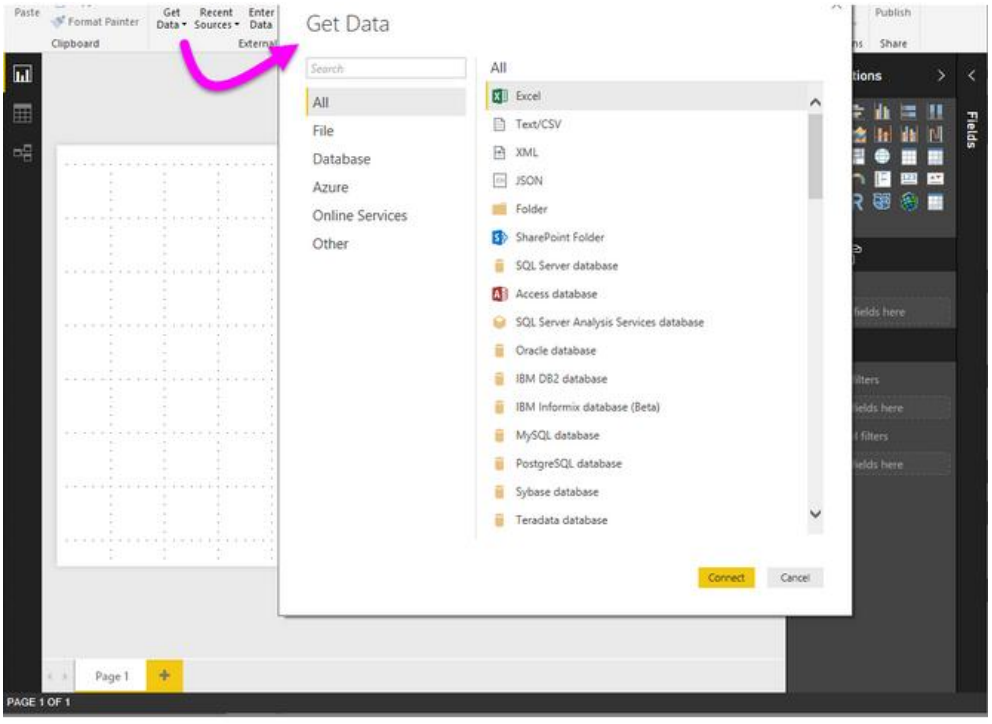


Ilustración 24. Power Bi. Fuente: Elaboración propia.

La herramienta cuenta con tres vistas: Una para visualizar los datos como filas en la tabla, otra para poder ver el modelo relacional de una base de datos y otra para poder crear informes con una amplia gama de tipos de gráficos en una paleta. La paleta puede ser ampliada con más gráficos descargables de la página oficial, permitiendo así,

crear gráficos personalizados de todo tipo. Cada gráfico en el informe cuenta con varios campos, adaptados a cada gráfico, para rellenar con los atributos de las tablas. A su vez cada informe tiene una serie de filtros para poder mostrar los datos que el desarrollador cree conveniente descartando otros. Por último Power Bi tiene la característica de crear consultas personalizadas, para ello se hace uso de la vista Query donde se puede editar todo tipos de consultas a las fuente de datos.

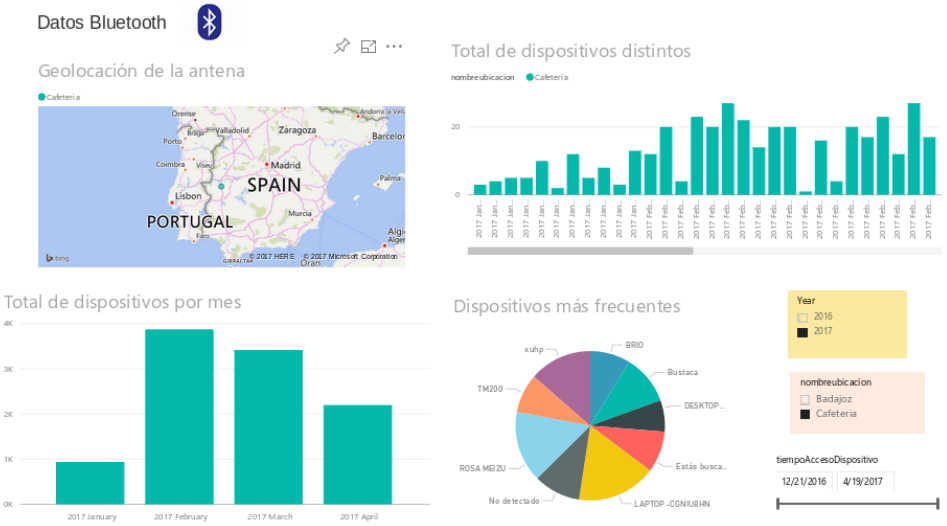


Ilustración 25. Datos Bluetooth y gráficos de los datos captados. Fuente: Elaboración propia.

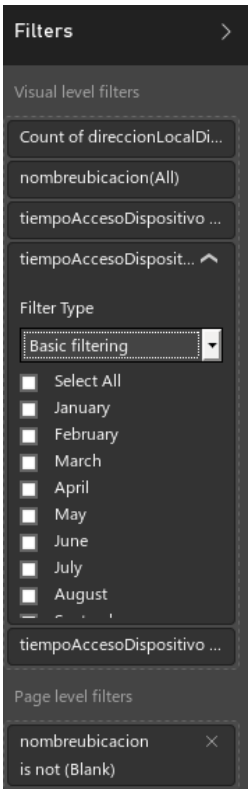


Ilustración 26. Filtrado básico para atributo temporal en Power Bi.
Fuente: Elaboración propia.

4.2.3.1. Comunicación entre Power Bi y la Base de datos.

Power Bi, es muy recomendable para proyectos en los que se pretende recoger datos y visualizarlos de una forma más amigable y poder interactuar con la información. La herramienta, en su versión web (la cual se puede descargar como una aplicación en modo local: Power Bi Desktop) tiene una limitación. Ésta es incapaz, aún en su versión más reciente, de conectar con fuentes de datos que no sean de la marca Microsoft, como por ejemplo Azure. Esto presenta un problema para el proyecto que se está desarrollando ya que la base de datos que se utiliza está alojada en una instancia RDS de Amazon, así que se requiere de alguna aplicación que realice consultas a la base de datos cada cierto tiempo y envíe la información a Power Bi web. Para ello, se requiere, por un lado tener una cuenta pro de Power Bi, y descargar la herramienta Power Bi Gateway. Esta herramienta se puede descargar, de manera gratuita del propio menú de configuración de la web de Power Bi.

4.2.3.2. Power Bi Gateway

Power Bi Gateway es una herramienta cuya función es hacer las veces de intermediaria entre la fuente de datos y la aplicación. Una vez instalada y configurada la aplicación, podemos indicar una programación de actualización de datos en Power Bi para nuestros informes. De tal forma que es posible programar una serie de horas para que nuestra Gateway realice las consultas necesarias para tener actualizados los informes.

Ilustración 27. Programación de actualizaciones en Power Bi. Fuente: Elaboración propia.

Para que las consultas y actualizaciones de datos en nuestro sitio web de Power Bi puedan llevarse a cabo, el ordenador que tiene instalada la puerta de enlace debe estar encendido y con la aplicación iniciada. Normalmente y por defecto, la aplicación se inicia al arrancar, pero si esto no sucede, es recomendable configurar el arranque de Windows para que inicie el servicio de la puerta de enlace al iniciar el sistema operativo. Cabe decir que como desventaja, Gateway, es solo instalable en sistema operativo Windows.

4.3. Arquitectura de la solución del proyecto

En este último subapartado del capítulo "Diseño del sistema", se recupera la tabla inicial de la arquitectura del sistema, en la cual se reflejan los tres módulos en los que se descompone el proyecto y para cada uno de ellos se va a indicar que tecnologías o herramientas van a constituir su realización:

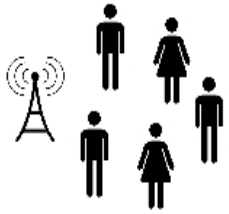
Arquitectura de la solución del sistema. Tecnologías y herramientas aplicadas		
		
Detecciones	**Almacenamiento**	**Visualizaciones de datos**
- Raspberry Pi con sistema operativo Raspbian y dos programas instalados: 1) Detecciones Bluetooth. 2.1) Puntos de acceso . 2.2) Monitorización de tramas. - Servicio web Tomcat instalado en antena para poner fichero de datos a disposición del servidor. - Servidor web para recogida de datos e inserción en base de datos.	- Instancia RDS de Amazon Web Services. - Base de datos con tablas para almacenar información referente a datos de una antena, datos de dispositivos detectados y lista negra. - MySQL Workbench para la creación de tablas , atributos y relaciones entre tablas.	- Herramienta Power Bi para visualización de informes. - Gateway Power Bi para comunicar con la base de datos y realizar consultar para mostrar datos en Power Bi.

Tabla 17. Arquitectura de la solución del proyecto. Fuente: Elaboración propia

5. Desarrollo.

En el capítulo de desarrollo se pretenden mostrar todos los procedimientos que se siguen durante el desarrollo del sistema, además de aplicar todas y cada una de las estrategias y soluciones propuestas comentadas en el capítulo anterior. A su vez se explican los algoritmos utilizados para el escaneo de dispositivos. Se detallan tres algoritmos utilizados para el desarrollo de programas de detección: Uno para la recepción de dispositivos Bluetooth y otros dos (puntos de acceso y monitorización de tramas) que van a llevar a cabo las detecciones a través de conexiones WiFi. Este capítulo cuenta, además, el procedimiento para la realización del servidor web y elaboración de métodos que se ocupan de asegurase que un nombre de un dispositivo no aparece en la lista negra antes de proceder a su almacenamiento. A su vez incluye detalles sobre la creación de informes, para los datos obtenidos utilizando la herramienta Power Bi, así como las configuraciones necesarias a llevar a cabo en la Raspberry Pi para su correcto funcionamiento en el sistema.

5.1. Desarrollo de los métodos de la clase DiscoveryListener de la librería Bluecove en Java.

En este subapartado, se explica la manera en la que se desarrolla la detección de dispositivos Bluetooth desde el dispositivo Raspberry. Como ya se ha dicho en capítulos anteriores, para evitar trabajo al servidor, la propia Raspberry almacena la información de las detecciones de dispositivos Bluetooth, en su propia memoria SD, de manera que la implementación sigue los pasos para enviar los datos a un fichero. En el caso del sistema se ha optado por una hoja de cálculo de extensión xls. Inicialmente tenemos un bucle en el programa principal que realiza llamadas a la interfaz Bluecove cada cierto tiempo. Cada vez que se invoca Bluecove se produce un escaneo del entorno. Este tiempo se configura desde el servidor web.

```
boolean started = LocalDevice.getLocalDevice().getDiscoveryAgent().startInquiry(DiscoveryAgent.GIAC,
        listener);
```

Ilustración 28.Inicialización de Bluecove. Fuente: Elaboración propia.

En la sentencia para invocar a la interfaz Bluecove, lo que se hace es llamar al agente descubridor del dispositivo local para que comience la búsqueda de dispositivos. Una vez ha dado comienzo el escaneo de entorno, todos los hilos activos en ese momento esperan a ser notificados para continuar con el programa.

Para el escaneo de dispositivos se crea una lista que va a almacenar objetos de tipo RemoteDevice. En esta lista se almacenarán todos los dispositivos detectados en un escaneo de entorno.

5.1.1. Método DeviceDiscovered.

Este método realiza una serie de acciones cada vez que descubre a un nuevo dispositivo, la primera acción a desempeñar es almacenar el dispositivo en la lista de dispositivos encontrados. Una vez hecho esto, se dispone a quedarse con la información que le ofrece el escaneo. Esta información se compone del nombre y la dirección física del dispositivo. Además y como ya se ha comentado en capítulos anteriores, conviene almacenar también el identificador de la antena y una marca de tiempo que hace referencia al momento preciso en el que el dispositivo ha sido detectado.

```
151    static DiscoveryListener listener = new DiscoveryListener() {
152
153        @SuppressWarnings("deprecation")
154
155        @Override
156        public void deviceDiscovered(RemoteDevice btDevice, DeviceClass cod) {
157
158            Calendar d = Calendar.getInstance();
159
160            dispEncontrados.addElement(btDevice);
161            String nombreDisp;
162            try {
163                nombreDisp = btDevice.getFriendlyName(false);
164                System.out.println("- Nombre: " + nombreDisp);
165
166            } catch (IOException canGetDeviceName) {
167                System.out.println("- Nombre no detectado");
168                nombreDisp = "No detectado";
169            }
170
171            String dirLocal = btDevice.getBluetoothAddress();
172            System.out.println("- Direccion local: " + dirLocal);
173
174            java.text.SimpleDateFormat sdf = new java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
175
176            String TimeStamp = sdf.format(d.getTime());
177
178            if (nombreDisp == null)
179                nombreDisp = "No detectado";
```

Ilustración 29. Método DeviceDiscovered. Fuente: Elaboración propia.

En el método se utilizan dos funciones para obtener el nombre y la dirección física del dispositivo. Estos métodos son getFriendlyName y getBluetoothAddress. Estos métodos devuelven valores de tipo String (líneas 161 - 171). Es muy probable que que el usuario del dispositivo detectado no tenga configurado un nombre o que el propio método getFriendlyName haya fallado en la detección del nombre. Es entonces cuando se maneja el error con un try catch (líneas 162 - 169), de dónde se asigna a la variable nombreDisp (nombre del dispositivo detectado), el valor "No detectado". Estos errores se optan por corregirse de esta manera, ya que el programa cae devolviendo una excepción de tipo NullPointerException y es necesario manejarla.

A su vez, se recoge la marca de tiempo asignando un formato personalizado. En este caso, el formato de marca que se asigna es de tipo Americano.

```
181             if (!comprobarPatronListaNegra(nombreDisp)) {
182
183                 HSSFRow fila = hoja.createRow(numFila);
184                 numFila++;
185
186                 HSSFCell celdaIdAntena = fila.createCell((short) 0);
187                 HSSFCell celdadireccionLocal = fila.createCell((short) 1);
188                 HSSFCell celdaTiempo = fila.createCell((short) 2);
189                 HSSFCell celdaNombreDisp = fila.createCell((short) 3);
190
191                 String idenAntena = String.valueOf(idAntena);
192                 HSSFRichTextString idAntena = new HSSFRichTextString(idenAntena);
193                 celdaIdAntena.setCellValue(idAntena);
194
195                 HSSFRichTextString dirLocalDisp = new HSSFRichTextString(dirLocal);
196                 celdadireccionLocal.setCellValue(dirLocalDisp);
197
198                 HSSFRichTextString tiempoAcc = new HSSFRichTextString(TimeStamp);
199                 celdaTiempo.setCellValue(tiempoAcc);
200
201                 HSSFRichTextString nDisp = new HSSFRichTextString(nombreDisp);
202                 celdaNombreDisp.setCellValue(nDisp);
203
204                 try {
205
206                     FileOutputStream fos = new FileOutputStream(excelDatos);
207                     libro.write(fos);
208                     fos.close();
209
210                 } catch (Exception ex) {
211                     System.out.println("Fichero no encontrado");
212                 }
213             }
```

Ilustración 30. Método DeviceDiscovered. Comprobación en lista negra. Fuente: Elaboración propia.

La segunda parte de este método, se basa en el almacenamiento de la información obtenida en la búsqueda de un dispositivo. Cuando se ha detectado el

dispositivo, antes de proceder a su almacenamiento, debe comprobarse que su nombre no está contenido en la lista negra (línea 181). Esta lista contiene todas las máscaras de nombres que no deben ser almacenados. Por ejemplo, una de las máscaras que contiene es la palabra TV. De manera que cada vez que una antena detecta un dispositivo cuyo nombre contiene la palabra TV. El dispositivo se descarta y no se guarda, evitando así el almacenamiento de dispositivos estáticos que siempre permanecen en el lugar, como es el caso de una televisión.

Para el almacenado de datos en un fichero con formato hoja de cálculo, se hace uso de la librería POI. Esta librería permite leer y escribir en un documento de dicho formato realizando la gestión por hoja, fila y celda. En nuestro caso, la información se va guardar por fila (líneas 183 - 202), de donde, cada fila va a contener cuatro celdas y cada una de las celdas va a almacenar la información, de la siguiente forma:

Identificador de la antena receptora	Nombre del dispositivo detectado	Dirección física de dispositivo detectado	Marca de tiempo de la detección del dispositivo

Tabla 18. Estructura de almacenamiento de detecciones Bluetooth en el fichero. Fuente: Elaboración propia.

Finalmente, se crea un flujo de salida que apunta a nuestro fichero de datos y se crea una operación de escritura donde se escriben las celdas que ya contienen los datos que se pretenden almacenar (líneas 204 - 212), de donde cabe destacar el manejo de la excepción con un try catch en caso de no encontrar el fichero o que éste estuviese corrupto.

5.1.1.1. Desarrollo de método de comprobación de nombres en lista negra.

Cuando inicia el programa de detección de dispositivos de una antena receptora, da comienzo a la descarga de ítems de la tabla lista negra en una lista de java. En ella se almacenan todas y cada una de las máscaras o nombres completos de dispositivos que existen en la tabla. Cada vez que se comprueba si un determinado dispositivo existe en la lista, se comprueba mediante el método “comprobarPatronListaNegra”.

```
240    public static boolean comprobarPatronListaNegra(String nombreDisp) {
241        boolean enc = false;
242        for (int i = 0; i < ListaNegra.size(); i++) {
243            Pattern p = Pattern.compile(ListaNegra.get(i));
244            Matcher m = p.matcher(nombreDisp);
245            if (m.find())
246                enc = true;
247        }
248        return enc;
249    }
```

Ilustración 31. Comprobación mediante el método "comprobarPatronListaNegra". Fuente: Elaboración propia.

El método recibe por parámetro un nombre. A continuación recorre la lista y haciendo uso de las clases Pattern y Matcher, se comprueba si la etiqueta del ítem recogido de la lista coincide con el nombre del dispositivo recibido por parámetro (identificador del dispositivo detectado). Si el resultado obtenido por las comparaciones realizadas con el método find() (línea 245) del objeto de la clase Matcher devuelve true, significa que el patrón coincide con el nombre del dispositivo. Por lo tanto, se desprecia, así que en este caso no se va a almacenar.

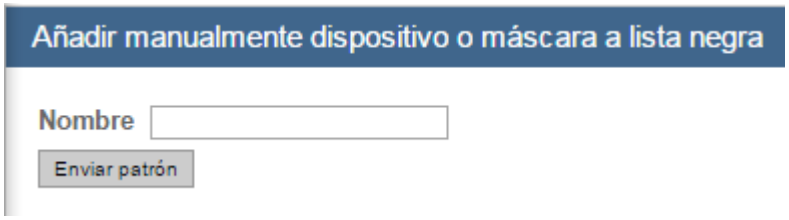


Ilustración 32. Formulario para añadir manualmente máscaras a lista negra. Fuente: Elaboración propia.

5.1.1.1.1. Desarrollo de un mecanismo de inserción automática de nombres en lista negra.

Para el desarrollo de un mecanismo de inserción automático de nombres en lista negra, debe realizarse una consulta que devuelva aquellos nombres que pertenecen a dispositivos detectados entre horas de baja actividad. A continuación se muestra un ejemplo de inserción en lista negra mediante sinxtaxis SQL:

insert into ListaNegra ***(SELECT distinct(essiddevice) from DataDevices***

where HOUR(TiempoAcceso) between '22:30' and '7:00'

having essidDevice not like ' ');

La sentencia realiza una consulta sobre todos los nombres de dispositivos distintos de la tabla de detecciones de la antena que han sido descubiertos en horas que comprenden entre las diez y media de la noche y las siete de la mañana. Este puede ser un ejemplo de inserción automática en lista negra si una antena receptora fuese instalada en un supermercado, entendiendo que a partir de las siete de la mañana pueden existir detecciones de personas como personal de limpieza u otro tipo de empleados, mientras que a partir de las diez y media no debería haber ningún tipo de detección. De lo contrario podría tratarse de un caso de acceso inadecuado.

5.1.2. Método InquiryCompleted

Una vez concluida la búsqueda de dispositivos debemos indicar al programa que debe continuar. Para ello se hace uso del método notifyAll. Este método se va a encargar de mandar un aviso a los hilos que están esperando mediante el método wait. Para llevar a cabo esta comunicación, se crea un objeto de la clase Object (HilosEventoCompletado). Este objeto espera en el programa principal (main) a que se le avise para que pueda continuar.

```
305 boolean started = LocalDevice.getLocalDevice().getDiscoveryAgent().startInquiry(DiscoveryAgent.GIAC,
306         listener);
307
308 System.out.println("************************************************************");
309 System.out.println("Búsqueda número " + i++ + ": ");
310
311 System.out.println("Buscando dispositivos, espere...");
312
313 HilosEventoCompletado.wait();
```

Ilustración 33. Comprobación mediante el método Método InquiryCompleted. Fuente: Elaboración propia.

En la imagen puede apreciarse la llamada al agente descubridor de Bluecove (líneas 305 - 306), mientras el objeto espera a que se le notifique que el proceso de escaneo ha finalizado (línea 313).

```
164        @Override
165        public void inquiryCompleted(int discType) {
166
167            System.out.println("\nInformación: Detección de dispositivos completada");
168
169            synchronized (HilosEventoCompletado) {
170                HilosEventoCompletado.notifyAll();
171            }
172        }
```

Ilustración 34. Agente descubridor de Bluecove. Fuente: Elaboración propia.

Cuando el proceso finaliza muestra un mensaje un por pantalla indicando que proceso ha finalizado y sincroniza con el objeto (HilosEventoCompletado), notificando que se ha dado por concluida la búsqueda de dispositivos (líneas 169 - 170). El programa principal continúa con las siguientes acciones a realizar.

5.2. Escaneo de puntos de acceso. Comando iwlist.

Otra manera de llevar a cabo un control de densidad poblacional, como se ha visto en el capítulo “Análisis”, es realizar búsquedas de puntos de accesos WiFi. Son muchos los dispositivos, que hoy en día llevan activadas configuraciones de puntos de accesos en sus dispositivos móviles para compartir su red con otros usuarios, de manera que, parece buena idea ver los resultados que ofrece este mecanismo de escaneo.

Asimismo, para poder realizar detecciones de entorno de dispositivos conectados vía Wifi, se hace uso del comando iwlist. Este comando obtiene información detallada sobre las posibles redes Wifi dentro de un perímetro determinado, que depende como siempre, de las características de la tarjeta Wireless del dispositivo receptor en cuanto a capacidad de alcance se refiere.

Para ejecutar un escaneo de puntos de acceso WiFi, necesitamos ejecutar la función scan del comando iwlist. Para ello debemos indicarle la interfaz del dispositivo receptor que queremos utilizar para escanear el entorno. La estructura del comando es la siguiente:

Comando	Interfaz del receptor	Función
Iwlist	wlanN (N=0,1,2,...)	Scan

Tabla 19. Estructura del comando "iwlist". Fuente: Elaboración propia.

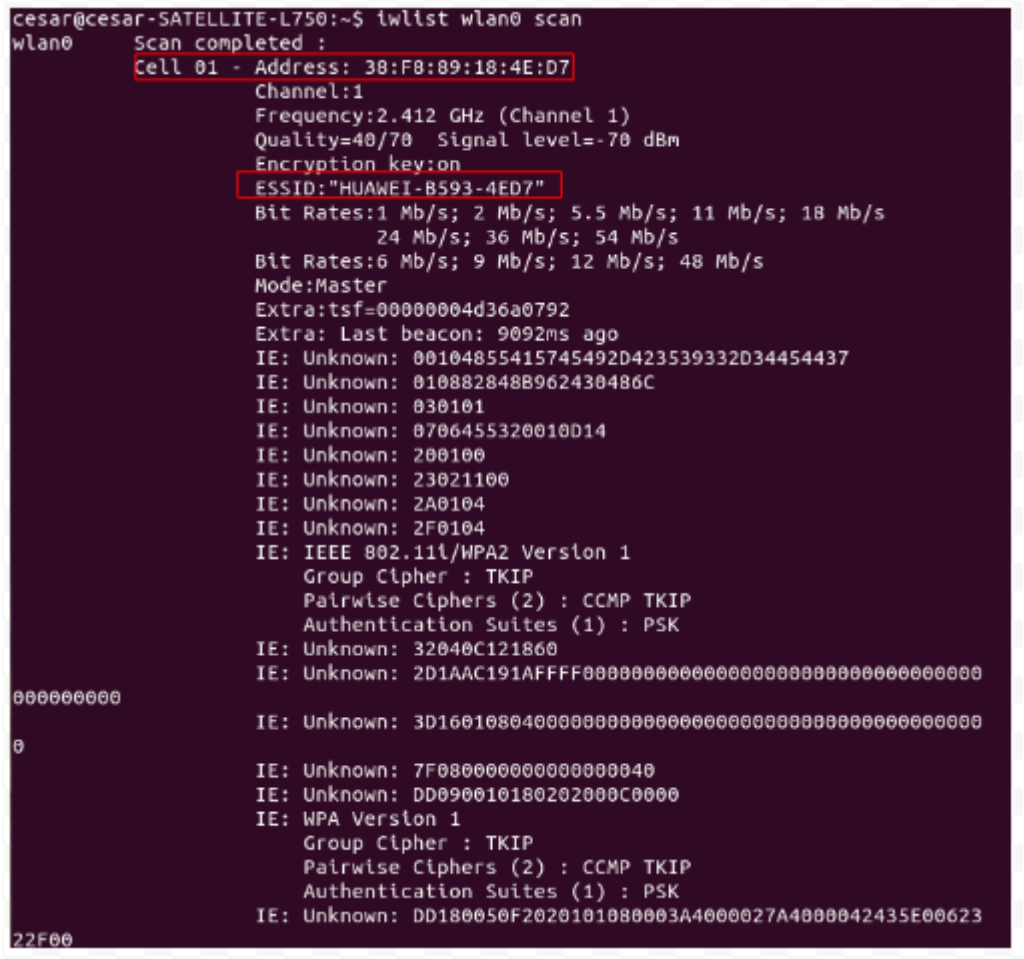


Ilustración 35. Resultado de aplicar el comando iwlist. Fuente: Elaboración propia.

De donde se recopila solamente la información que se considera relevante para su almacenamiento. Como ya se ha dicho en capítulos anteriores, el sistema almacena información de un dispositivo detectado referente a su nombre y a su dirección física.

De tal forma que el programa desarrollado en Java para la detección de dispositivos conectados mediante vía WiFi, parte de este comando para el escaneo y mapeo de información de detecciones.

Asimismo, si se ejecuta en una terminal el comando tal y como se ha explicado en el apartado anterior, el sistema nos ofrece toda la información que se ha mostrado en la imagen. Para poder filtrar solamente aquellos datos que se necesitan, se puede utilizar la función awk. Esta función se ocupa de filtrar todas aquellas líneas de la salida por pantalla que contengan los máscaras que se le indican. De tal forma que si se pretende recoger datos referentes al essid o nombre del dispositivo y de la dirección física, tan solo interesa quedarse con aquellas líneas que comienzan por la palabra "Cell" y "ESSID" ya que las líneas que contienen estas palabras contienen, a su vez, la información que se pretende almacenar.

```
cesar@cesar-SATELLITE-L750:~$ iwlist wlan0 scan | awk '/ESSID/ || /Cell/'
          Cell 01 - Address: 38:F8:89:18:4E:D7
                    ESSID:"HUAWEI-B593-4ED7"
```

Ilustración 36. Adición de filtros al comando iwlist. Fuente: Elaboración propia.

5.2.1. El programa de escaneos de puntos de acceso en Java.

Para realizar un escaneo de entorno, se necesita ejecutar el comando iwlist, tal como se ha explicado. Dado que se trata de un comando propio de terminal Linux, se requiere de alguna herramienta con la que interactuar con comandos de terminal desde un programa escrito en lenguaje Java, para posteriormente recoger la información que interesa almacenar, es entonces cuando entra en juego la clase Runtime. Esta clase permite mediante el método exec(), ejecutar un determinado comando dentro de un programa escrito en Java.

```
String[] cmd = { "bash", "-c", "iwlist wlan0 scan | awk '/ESSID/ || /Cell/'" };
Runtime rt = Runtime.getRuntime();
Process proc = rt.exec(cmd);
```

Ilustración 37. Ejecución de iwlist desde Java. Fuente: Elaboración propia.

Una manera de recoger la información resultante de ejecutar el comando, es leer en un buffer dicha información. A continuación se recoge en variables de tipo cadena cada una de las líneas devueltas. Es posible que tenga que ajustarse haciendo uso del método substring para descartar información no deseada de una línea, por ejemplo, en el

caso de la línea “Cell 01 - Address: 38:F8:89:18:4E:D7”, tan solo se va a almacenar la dirección descartando la cadena “Cell 01 - Address:”.

```
144         int i = 0;
145         while (true) {
146             String[] cmd = { "bash", "-c", "iwlist wlan0 scan | awk '/ESSID/ || /Cell/'" };
147             Runtime rt = Runtime.getRuntime();
148             List<String> datosDisp = new LinkedList<String>();
149 
150             System.out.println("\n ( Búsqueda número: " + i + " )\n");
151             Process proc = rt.exec(cmd);
152 
153             BufferedReader is = new BufferedReader(new InputStreamReader(proc.getInputStream()));
154 
155             while ((dispositivo = is.readLine()) != null) {
156                 Calendar date = Calendar.getInstance();
157                 java.text.SimpleDateFormat sdf = new java.text.SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
158 
159                 datosDisp.add(dispositivo.substring(27, dispositivo.length()));
160                 if (datosDisp.size() == 2) {
161 
162                     String nombreDispositivo = datosDisp.get(1).substring(0, datosDisp.get(1).length() - 1);
163                     String macDispositivo = datosDisp.get(0);
164 
165                     System.out.println("- Nombre de dispositivo: " + nombreDispositivo);
166                     System.out.println("- Mac de dispositivo: " + macDispositivo);
167 
168                     if (!comprobarPatronListaNegra(nombreDispositivo)) {
169                         try {
170                             Thread.sleep(1000);
171                             ntwk.addDeviceToFile(nombreDispositivo, macDispositivo, sdf.format(date.getTime()));
172                         } catch (InterruptedException e) {
173                             // TODO Auto-generated catch block
174                             e.printStackTrace();
175                         }
176                     }
177                     datosDisp.clear();
178                     System.out.print("\n");
179                 }
180 
181             }
```

Ilustración 38. Programa en Java para el escaneo WiFi. Fuente: Elaboración propia.

En la imagen se muestra la manera en la que se lleva a cabo la recolección de los datos necesarios para identificar un dispositivo detectado. El método lee del buffer de lectura la información que devuelve el comando ejecutado mediante el método exec de la clase Java Runtime. A continuación, almacena en primer lugar la dirección física del dispositivo y finalmente el essid o nombre del dispositivo. Del mismo modo que se hacía en el programa para escanear dispositivos conectados vía Bluetooth, debe comprobarse en una lista negra para evitar dispositivos de carácter estático, en el caso de Wifi, es muy común encontrarse con detecciones cuyo essid contienen la máscara o patrón Láser. Es el caso de la posible detección una impresora. De tal forma, que podría incluirse la etiqueta Láser en la lista negra. Asimismo, la lista negra es común a las dos tecnologías (Bluetooth y WiFi).

```
103     public static boolean comprobarPatronListaNegra(String nombreDisp) {
104         boolean enc = false;
105         for (int i = 0; i < ListaNegra.size(); i++) {
106             Pattern p = Pattern.compile(ListaNegra.get(i));
107             Matcher m = p.matcher(nombreDisp);
108             if (m.find())
109                 enc = true;
110         }
111         return enc;
112     }
```

Ilustración 39. Fragmento de programa para comprobar patrón en lista negra. Fuente: Elaboración propia.

Al igual que en las comprobaciones en lista negra en busca de máscaras para no almacenar dispositivos estáticos, aquí se utilizan también las clases Pattern y Matcher para obtener respuesta de similitudes entre cadenas o comprobaciones sobre si una cadena está contenida en otra. Cuando inicia el programa se descarga de la base de datos la información contenida en la tabla "Lista Negra" y se vuelca en una colección de datos de tipos String en Java, también llamada "ListaNegra". Es ahí donde el método comprobarPatronListaNegra() realiza las comprobaciones consultando en la colección de datos.

Una vez comprobado que no existe un determinado dispositivo en colección de nombres en el momento de la detección del dispositivo, de la misma forma que ocurre en el programa de detecciones de dispositivos vía Bluetooth, se almacena en un fichero en formato hoja de cálculo. El fichero, como ya se ha comentado, está en todo momento, a disposición del servidor a través de un servidor Tomcat instalado e iniciado en la propia antena o dispositivo receptor. A continuación se muestra el método que se ocupa de insertar cada dato informativo en la celda correspondiente dentro de la hoja de cálculo.

```
50      public void addDeviceToFile(String device, String mac, String TimeStamp) {
51          HSSFRow fila = hoja.createRow(numFila);
52          numFila++;
53
54          HSSFCell celdaIdAntena = fila.createCell((short) 0);
55          HSSFCell celdaDevice = fila.createCell((short) 1);
56          HSSFCell celdaTiempo = fila.createCell((short) 2);
57          HSSFCell celdaMac = fila.createCell((short) 3);
58
59          String idenAntena = String.valueOf(idAntena);
60          HSSFRichTextString idAntena = new HSSFRichTextString(idenAntena);
61          celdaIdAntena.setCellValue(idAntena);
62
63          HSSFRichTextString dirDisp = new HSSFRichTextString(device);
64          celdaDevice.setCellValue(dirDisp);
65
66          HSSFRichTextString tiempoAcc = new HSSFRichTextString(TimeStamp);
67          celdaTiempo.setCellValue(tiempoAcc);
68
69          HSSFRichTextString dirMacDisp = new HSSFRichTextString(mac);
70          celdaMac.setCellValue(dirMacDisp);
71
72          try {
73              FileOutputStream fos = new FileOutputStream(excelDatos);
74              libro.write(fos);
75              fos.close();
76          } catch (Exception ex) {
77              System.out.println("Fichero no encontrado");
78          }
79      }
```

Ilustración 40. Inserción de datos en fichero en formato XLS. Fuente: Elaboración propia.

Como en el caso de detecciones Bluetooth, el método empleado para inserciones en el fichero en formato hoja de cálculo utiliza la librería POI, en la cual define un serie de celdas por fila, tantas como atributos necesitamos almacenar en la base de datos, y a continuación realiza las inserciones de valores en sus respectivas celdas, identificador de antena, essid del dispositivo remoto, dirección física y tiempo de detección del mismo.

5.3. Escaneo de paquetes de red. Comando tshark de Wireshark.

El comando tshark de Wireshark realiza las mismas funciones de escaneo de la misma forma que que Wireshark lo hace. Además es capaz de filtrar la información, mediante una serie de parámetros que acompañan al comando, de manera que permite personalizar las búsquedas obteniendo los datos que se desean obtener.

En este capítulo se explica cómo se utiliza tshark en el sistema, que filtros se utilizan para obtener la información que se pretende almacenar (dirección física de dispositivo y essid). para poder utilizar el comando lo primero que hay que hay que hacer es cerciorarse de que el receptor que disponemos para la recepción de paquetes, es capaz de habilitar el modo monitor. Esto se consigue mediante el comando *sudo iwconfig interfaz mode monitor*, de donde la interfaz se refiere a dicho receptor de paquetes. Una vez asegurada la herramienta de recepción, se realiza la instalación de la herramienta Wireshark (tshark para interfaz de comando). La instalación como cualquier otra en cualquier sistema operativo Linux se realiza mediante el comando *sudo apt-get install tshark.* Una vez instalados todos los mecanismos de recepción, se implementa el comando personalizado para la recepción de los dos datos mencionados que identifican a los dispositivos que se detectan. Lo primero es averiguar sobre el comando, es decir conocer cuales son los parámetros que se necesitan para filtrar la información. Tras un estudio del comando, se ha aprendido que para filtrar dicha información se utiliza tshark de la siguiente manera:

sudo tshark -i wlan0 -a duration:60 -T fields -e wlan.sa_resolved -e wlan.sa -E separator=,

El comando se podría dividir en tres partes, se muestran en la siguiente tabla:

Parámetros	Detalles
Opción -i Wlan0	Interfaz a utilizar. Debe tener activado el modo monitor.
duration:60	Duración en segundos de la búsqueda de paquetes.
-e wlan.sa_resolved (essid del dispositivo) -e wlan.sa (mac del dispositivo) -E separator=, (split o separador)	Información a escoger y separador a utilizar

Tabla 20. Partes del comando tshark, incluidos parámetros y filtrado. Fuente: Elaboración propia.

A partir de aquí, ya se puede empezar el desarrollo de una aplicación que utilice este comando y mapee los datos de forma que se permita posteriormente su almacenado en una estructura. En el siguiente subapartado se detallan los pasos a seguir para la elaboración de un software que permita retener la información indicada.

5.3.1. Tshark en Java. Elaboración del software de detección.

Lo primero que debe llevar a cabo este programa, es una descarga de nombres de lista negra para posteriormente consultarlos y almacenar la información considerando el no figurar en la información descargada. Es decir, al igual que sucede con las otras dos tecnologías, la antena comprueba previamente en la lista negra si el dispositivo que acaba de detectar figura en dicha lista. Para ello se utiliza el mismo mecanismo que en el programa de detección de puntos de acceso. A continuación y si dicho dispositivo detectado no se encuentra en la lista, los datos referentes a éste, se almacenan en un registro o fila de un fichero en formato xls. Cada una de las cuatro celdas que componen cada fila o registro del fichero xls contiene la siguiente información:

Identificador de la antena receptora	Essid del dispositivo detectado	Mac del dispositivo detectado	Marca de tiempo de la detección del dispositivo

Tabla 21. Estructura de celdas en fichero XLS para almacenado de información Wifi con comando tshark. Fuente: Elaboración propia.

Para poder ejecutar el comando tshark en java, se hace uso de la clase Runtime y se obtiene la salida o el resultado del comando en un buffer.

```
Process proc = rt.exec("sudo tshark " +
        "-i rename4 " +
        "-a duration:60 " +
        "-T fields -e wlan.sa_resolved -e wlan.sa " +
        "-E separator=,");
BufferedReader is = new BufferedReader(new InputStreamReader(proc.getInputStream()));
```

Ilustración 41. Ejecución del comando tshark en Java para escaneo de paquetes.

Una vez almacenada la información, la detección de dispositivos continúa repitiendo el proceso de almacenaje en del fichero hasta que finaliza el tiempo de escaneo, especificado en el parámetro *-a duration:(segundos) de comando.* Durante escaneo y escaneo el programa realiza una consulta a la base de datos para averiguar, mediante un atributo de la tabla 'DatosAntena', si el servidor ha realizado la consulta de los datos en el fichero xls (inserta un 1), para poder almacenarlos de manera permanente en la base de datos de la instancia RDS de los servicios de Amazon. Si es el caso, el programa reinicia el fichero, es decir pone a cero o borra la información de dicho fichero para posteriores inserciones e indica al servidor (también haciendo uso del mismo atributo, inserta un 0) que ha sido consciente de la lectura del servidor. Para el almacenado de datos en el fichero y de igual forma que en el programa de detección de puntos de acceso, se utiliza la librería POI, la cual se encarga de definir un libro xls, una página y las filas y celdas necesarias para almacenar la información.

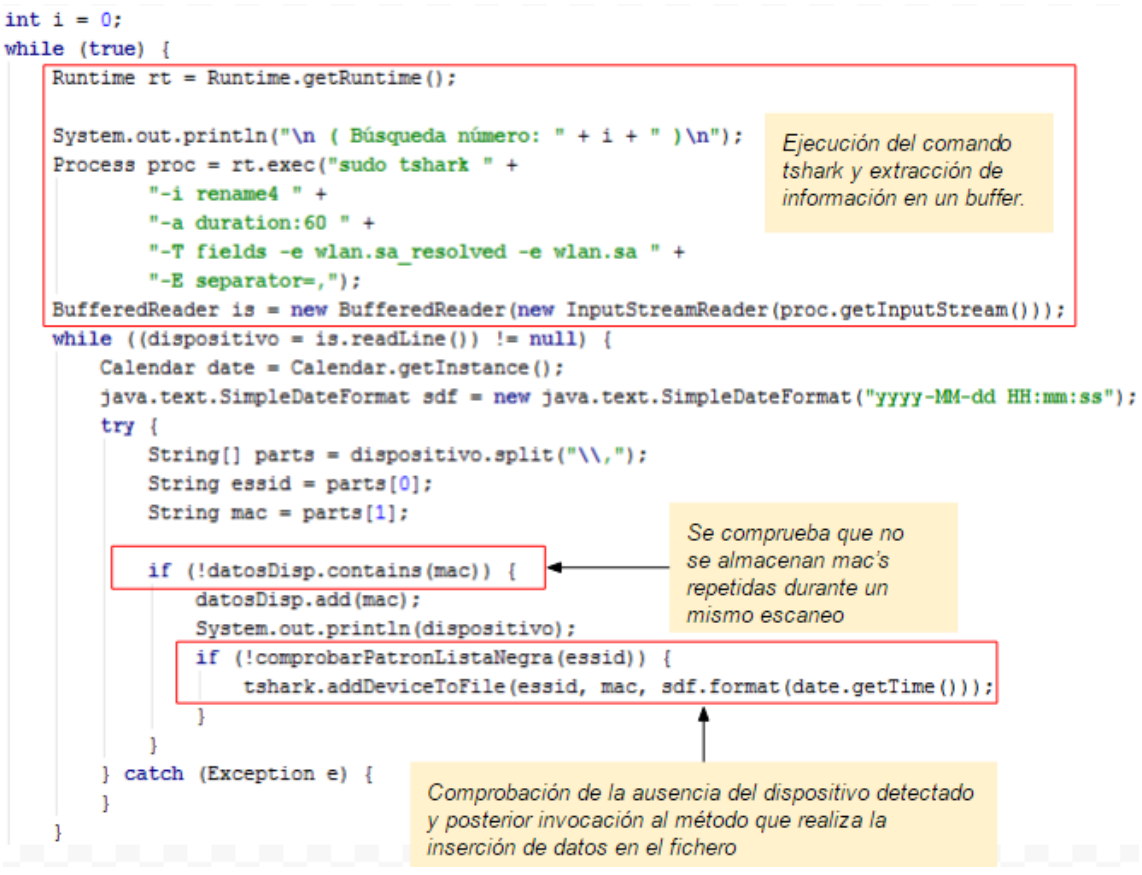


Ilustración 42. Estructura completa del almacenado de dispositivos mediante escaneos de paquetes utilizando tshark. Fuente: Elaboración propia.

5.4. Configuración de la Raspberry Pi.

En este apartado se van a detallar los pasos a seguir para configurar la Raspberry de manera que pueda asumir su rol de antena receptora dentro del sistema. En el capítulo de configuración de la Raspberry, se pretende realizar un repaso de todas y cada una de las herramientas necesarias para poder ejecutar el programa de escaneos de entorno en tecnologías Bluetooth y WiFi y poner los datos a disposición del servidor.

5.4.1. El script Sakis3g para conexiones 3g a Internet.

Una vez se dispone de los medios necesarios para llevar a cabo la conexión 3g (módem USB con ranura para SIM y tarjeta asociada a compañía para proveer el servicio y asimismo el acceso a Internet), se requiere de un software que permita llevar a cabo la conexión a la red pública.

Sakis3g es un script muy utilizado en conexiones 3g en Raspberry Pi, por su facilidad de uso. Cabe destacar, que sakis3g funciona con conexiones punto a punto utilizando el protocolo PPP. Este protocolo establece conexión entre un par de dispositivos autenticados. Este modelo de conexión se lleva a cabo en redes de telefonía móvil. Para instalar el servicio PPPd, es necesario ejecutar el siguiente comando:

- *sudo apt-get install pppd*

Sakis3g es un programa o script que se encarga de hacer el trabajo sucio por nosotros, es decir sus acciones son reconocer el módem, reconocer la compañía telefónica y pedir al usuario una serie de datos en cuanto a la configuración de la tarjeta SIM insertada en el módem USB se refiere. Para ello lo primero que se debe tener actualizado es el sistema operativo, así que se ejecutan en primer lugar los comandos:

- *sudo apt-get update*
- *sudo apt-get upgrade*

Una vez se tiene actualizado el sistema operativo se descarga el script con el comando:

– sudo wget “nombre del recurso a descargar”

El recurso descargado es un paquete comprimido en formato .tar.gz, de modo que se descomprime el paquete que contiene sakis3g y se le proporciona permisos de ejecución mediante los comandos:

– tar -xvf sakis3g.tar
– sudo chmod +x sakis3g

En este momento ya se tiene el script listo para ser ejecutado. Cuando el script se abre, ofrece un asistente en la que se deben seleccionar las opciones necesarias que pertenecen a la configuración del módem y a la compañía proveedora del servicio para la conexión de datos. El asistente configura la siguiente información:

– Método de conexión: 3g.
– Categoría del módem: Dispositivo USB, módem Bluetooth…
– Marca y modelo del módem USB.
– Interfaz en la que se encuentra el dispositivo dentro de la Raspberry.
– Selección de información de credenciales en APN. Información de fácil alcance en la web de la compañía: Nombre del punto de acceso (APN), usuario y contraseña por defecto.

Una vez terminado el asistente, el dispositivo se conecta automáticamente a Internet. Ahora al ejecutar *ifconfig* debe visualizarse un IP, que como se ha dicho debe ser pública real. Esto se sabe si la máscara que se visualiza es *255.255.255.255*.

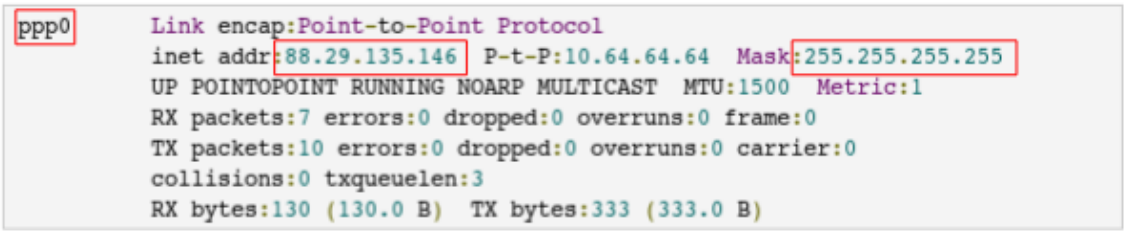


Ilustración 43. Resultado de ifconfig. Asignación de ip única a interfaz ppp. Fuente: Elaboración propia.

5.4.1.1. Automatización del proceso de conexión a Internet.

Uno de los objetivos de este proyecto, es automatizar todos y cada uno de los agentes que intervienen en él, de manera que un tipo de automatización que se lleva a cabo en el sistema, es que la antena una vez se conecte a la red eléctrica, debe ser capaz de conectarse automáticamente a Internet y a su vez comunicarse con el servidor mediante la base de datos. Así que en este subapartado se explica cómo se lleva a cabo dicha automatización, de manera que el usuario no tenga que configurar manualmente la conexión 3g cada vez que pretenda conectar la antena a Internet.

Según Vázquez, Saúl, en una entrada publicada en su blog el 20 de Octubre de 2013, para que el script funcione, éste debe ser movido desde la carpeta donde se encuentra inicialmente, hacia el directorio */opt.* A continuación, se le asigna el propietario root al script y se crea un enlace simbólico del mismo en la ruta /usr/bin, directorio donde se encuentran muchos de los programas ejecutables del sistema operativo Raspbian. Todo esto se consigue ejecutando los siguientes comandos:

- *sudo mkdir /opt/sakis3g*
- *sudo mv carpeta_origen_sakis3g /opt/sakis3g*
- *sudo chown root:root /opt/sakis3g/sakis3g*
- *sudo ln -s /opt/sakis3g/sakis3g /usr/bin*

A continuación, se accede al fichero /etc/sudoers y se configuran las reglas que tiene el usuario root, cada vez que se ejecuta un programa mediante el comando sudo, es decir como administrador del sistema. Para acceder a sudoers se accede mediante el comando *sudo visudo*. En este fichero se va a indicar que se al ejecutar el programa sakis3g mediante los permisos de administrador el sistema no precise de los credenciales del mismo. Para ello se añade al final de la línea la siguiente instrucción, de donde PI es el usuario del sistema operativo Raspbian en la Raspberry Pi:

- *pi ALL=NOPASSWD: /opt/sakis3g/sakis3g*

Por último solo queda configurar la información referente a los credenciales de la APN pertenecientes a la compañía proveedora de los servicios de conexión a Internet, el identificador que el sistema asigna al modem cuando es detectado tras su conexión

Plug and Play, y por último, de la interfaz USB por la cual el modem se conecta a la Raspberry. Para añadir toda esta informacion se crea un fichero de configuración en el directorio /etc. Ahora, solo basta con añadir las siguientes líneas al fichero creado. Estas líneas contienen la información indicada más arriba.

- USBDRIVER="option"
- USBINTERFACE="0"
- APN="airtelnet.es"
- APN_USER="vodafone"
- APN_PASS="vodafone"
- MODEM="19d2:0117"

Una vez realizada toda la configuración, para poder ejecutar el script se utiliza el siguiente comando:

- */opt/sakis3g/sakis3g --sudo "connect"*

Finalmente, para poder ejecutar este comando al inicio y permitir de esta forma que la antena sea capaz de conectarse a Internet al arrancar el sistema operativo, se ha pensado en el fichero /etc/rc.local. Cualquier sistema Unix, en este caso Raspbian, ejecuta todas las instrucciones que se encuentra dentro del programa contenido en este fichero en tiempo de arranque, es por ello se ha pensado en incluir el anterior comando en el programa de este fichero. De manera que se añade la instrucción en el mismo justo antes de que el programa de arranque finalice, es decir justo antes de linea que contiene la instrucción *exit 0*.

Ilustración 44. Comando para conexión desde Raspberry a Internet mediante módem USB a través de herramienta Sakis3g. Fuente: Elaboración propia.

5.4.2. Preparación del entorno web. Instalación de Tomcat.

Como ya se ha dicho, la antena receptora debe disponer de un servicio web que pone a disposición del servidor para ofrecer la información obtenida en un fichero de datos en formato xls. Para ello se hace uso del servicio web Tomcat. Se instala de una manera muy sencilla. Basta con ejecutar el comando siguiente:

- sudo apt-get install tomcat

Una vez instalada el servicio se va a alojar un fichero en formato xls en una carpeta. La carpeta se debe alojar dentro del directorio webapps, situado en la carpeta de instalación de tomcat. Por defecto tomcat se instala en el path /var/lib/tomcat/webapps. En este directorio se crea una carpeta llamada Antena Raspberry, dentro de la cual se incluye el fichero xls. Cada fila de este fichero hará referencia a un registro en el que se almacenará la información de un nuevo dispositivo detectado. No es necesario asignar permisos al directorio que contiene el fichero xls, esto se debe a que se encuentra dentro de la carpeta webapps, y a su vez esta carpeta es la que comparte por defecto tomcat en la red, de manera que tiene ya permisos de lectura asignados. Para poder realizar

pruebas de que realmente el fichero está compartido mediante protocolo http, puede realizarse una prueba. Gracias a que ya se tiene acceso a Internet debido a la conexión del modem con la tarjeta SIM integrada en la Raspberry Pi y configurada, si se reinicia el dispositivo, automáticamente, obtiene una IP de carácter pública. De igual manera si escribimos desde el navegador de cualquier equipo la URL donde se encuentra el fichero se puede descargar y ver la información que este contiene. La URL se compone de la ip asignada por la compañía proveedora de servicios de acceso a Internet seguido del puerto 8080. Este puerto es el que utiliza tomcat para conexiones mediante protocolo http al servicio web y a su vez al recurso que ofrece. A continuación, basta con indicar la ruta donde se ubica dicho recurso, en este caso el fichero. La ruta del archivo a compartir parte, por lo tanto, de la carpeta webapps, es decir la carpeta raíz que debe indicarse surge a partir del directorio webapps. La URL queda de la siguiente forma:

IP asignada por proveedor	Puerto utilizado por Tomcat	Ruta del archivo a compartir
47.59.202.244	:8080	/Antena_Raspberry/fichero.xls

Tabla 22. URL generada por antena receptora para permitir acceso a servidor a fichero de datos recogidos. Fuente: Elaboración propia.

Esta dirección URL, la antena, la inserta cada vez que se conecta al sistema en la base de datos ubicada en la instancia RDS de Amazon, como ya se sabe esta base datos se comparte entre todas las antenas conectadas y el servidor. De tal forma que cuando el servidor accede a dicha base de datos, tiene a su disposición la URL donde se encuentra el fichero de datos y es entonces cuando obtiene la información referente a los dispositivos detectados por dichas antenas receptoras durante un periodo de tiempo establecido.

5.5. Desarrollo del servidor web.

En este subapartado se cuentan los detalles para llevar a cabo el desarrollo de las funciones que desempeña el servidor, entre las cuales se encuentran: la consulta de los

datos referentes a dispositivos detectados por las antenas receptoras, los cuales se encuentran en ficheros alojados en las propias antenas, la activación de éstas o las inserciones de nombres en la lista negra.

5.5.1. Conexión con la base de datos.

El servidor web conecta con cada una de las antenas receptoras a través de los servicios web Tomcat instalados en las mismas, gracias a la conexión del proyecto web con la base de datos alojado en la instancia RDS de Amazon Web Services.

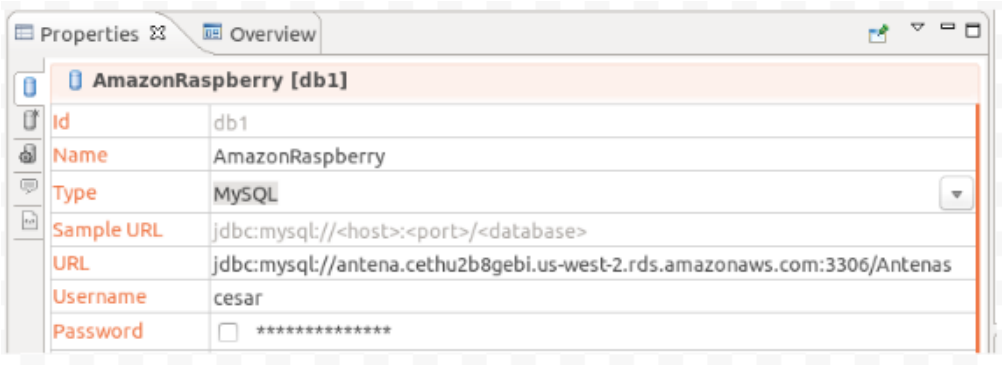


Ilustración 45. Conexión con instancia RDS desde herramienta Webratio. Fuente: Elaboración propia.

El modelo relacional del proyecto web tiene una conexión MySQL con el servidor de bases de datos en la instancia RDS. Gracias a la conexión con la instancia, el servidor conoce todas y cada una de las URLs de las antenas conectadas al sistema.

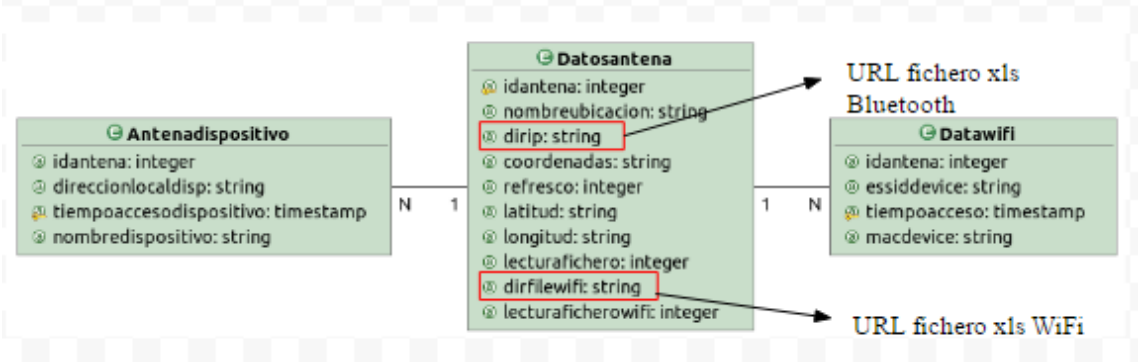


Ilustración 46. Almacenado en tabla "DatosAntena" de URLs (Datos Bluetooth y Wifi). Fuente: Elaboración propia.

5.5.2. Desarrollo del mecanismo de acceso a las antenas.

El servidor debe acceder a la información obtenida por las antenas. Para que esto sea posible, se han definido una serie de rutinas en el propio servidor. Las rutinas son desarrolladas como triggers en Webratio, y acceden a la información almacenada en ficheros xls por las antenas, en tiempos establecidos en el momento en el que se desarrolla la aplicación web. Este tiempo inicialmente es estático, es decir, el usuario no puede decidir el tiempo en el que se produce una consulta desde el servidor a una antena receptora. En este caso el valor del tiempo, afecta directamente a otros dos factores, también fundamentales para el buen funcionamiento del sistema.

Valor de variable temporal	Factores afectados
Tiempo muy elevado	Disminución de la frecuencia de accesos por parte del servidor, sin embargo aumenta el tamaño de la información en el momento en el que los datos viajan desde un receptor hacia el servidor y desde éste hacia la base de datos. Se ve afectada la rapidez del sistema.
Tiempo muy bajo	Aumento de la frecuencia de accesos por parte del servidor con tamaños de datos más ligeros. El trabajo del servidor crece en cuanto a accesos.

Tabla 23. Contraste de valores para el tiempo de acceso a los datos recogidos por la antena por parte del servidor. Fuente: Elaboración propia.

A continuación se muestra el esquema, correspondiente al trigger que se encarga de la recogida de datos en cada antena, donde se explican cada una de las acciones que se llevan a cabo en la transferencia de datos desde el dispositivo receptor hacia la base datos. El bucle exterior (Loop3), recoge en atributos de tipo String las urls de cada uno de los ficheros en formato hoja de datos (xls), de todas las antenas conectadas y se asignan a una Excel Unit, la cual recoge todos los registros de cada fichero. Cada registro se corresponde con información de la antena receptora (identificador) y de los dispositivos detectados (essid o nombre, dirección física y tiempo de detección). Estos registros se iteran en un segundo bucle (Bucle), donde finalmente, son insertados en la

base de datos. Esta manera de desarrollar la transferencia desde la antena hacia la base de datos, se realizan de la misma forma tanto para almacenamiento de dispositivos detectados vía WiFi como para Bluetooth.

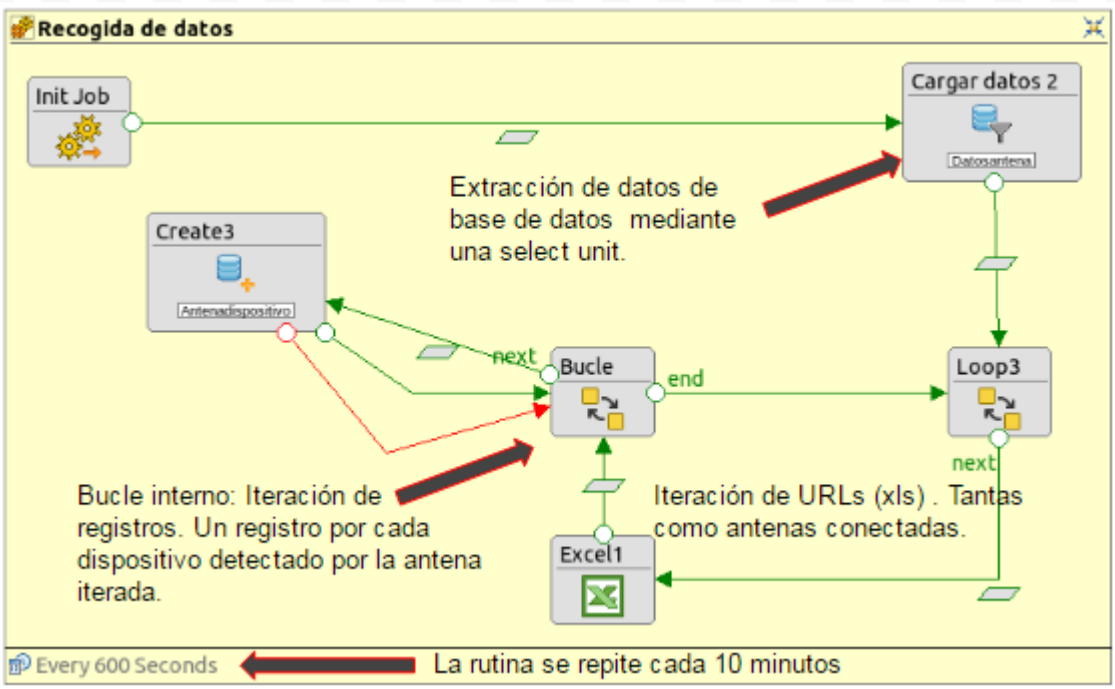


Ilustración 47. Recogida de datos por parte del servidor. Fuente: Elaboración propia.

La antena, a su vez, debe tener constancia de cuando se ha producido una lectura del fichero de datos. Cada vez que el servidor accede al dispositivo, éste pone a cero el fichero de datos e inicia una nueva recopilación de datos. De esta forma no se ve aceptada la capacidad de almacenamiento de la antena. Esto se hace mediante un atributo en la tabla de la propia antena, el cual, el proyecto cambia su valor a uno cada vez que accede al fichero de la antena, y en cuanto ésta es consciente de la lectura y en respuesta, vuelve a iniciar el valor del atributo a cero. Es entonces cuando reinicia el fichero de datos xls. Esta comunicación antena-servidor se lleva a cabo tanto en el almacenamiento de dispositivos detectados vía Bluetooth como WiFi.

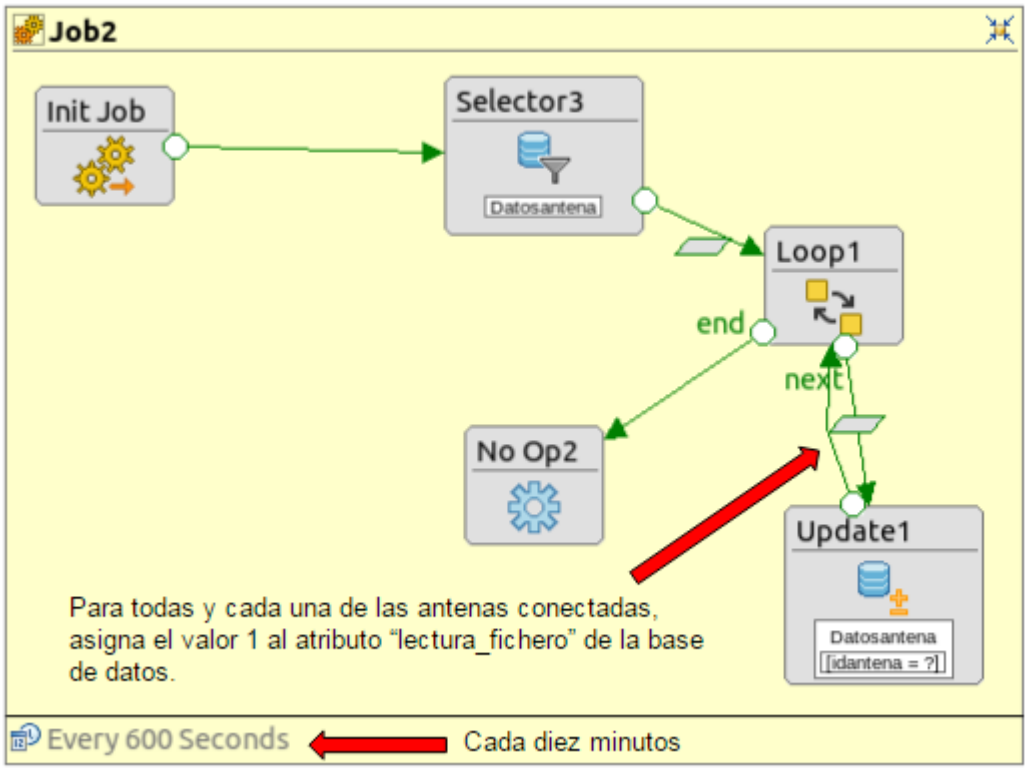


Ilustración 48. Recogida de datos. Fuente: Elaboración propia.

En el esquema siguiente se indica la trayectoria del flujo de información desde su origen, en todas y cada una de las antenas receptoras, hacia la herramienta de visualización. En este caso se trata de Power Bi.

5.5.3. Activación de antenas desde el servidor.

Para la activación de las antenas se crea un formulario que envía información a una Create Unit. Esta unidad inserta en la base de datos la información que se envía desde el formulario. El dato que envía es el valor de recarga, que hace referencia al tiempo que trascurre desde que finaliza un escaneo de dispositivos y da comienzo el siguiente. Cuando la antena observa que el valor en la base de datos es distinto a cero, comienza a escanear su entorno en búsqueda de dispositivos.

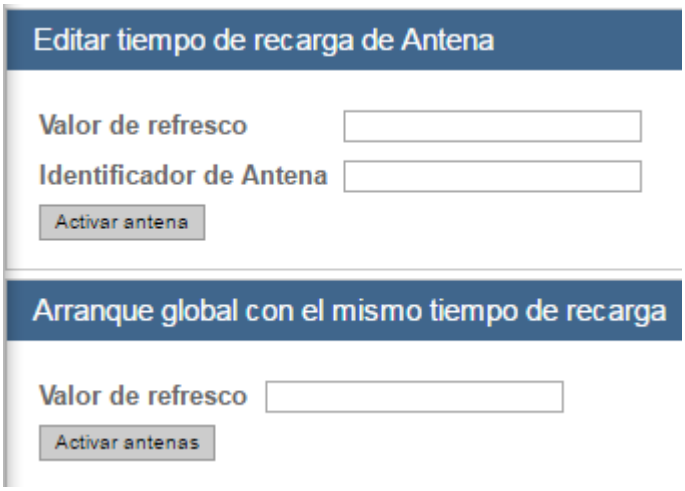


Ilustración 49. Formulario en servidor web para configurar el tiempo de recarga de las antenas receptoras. Fuente: Elaboración propia.

5.5.4. Desarrollo de mecanismos de inserción de nombres en lista negra.

Para el desarrollo de un mecanismo que permite insertar nombres en la lista negra, se ha elaborado un formulario que, de la misma manera que ocurre para el caso de asignaciones de tiempos de recargas, envía un valor de tipo cadena a una Create Unit. A continuación, la unidad, inserta el valor que recibe en la tabla de la base de datos que almacena los nombres de los dispositivos que no se van a almacenar en el caso de producirse sus detecciones.

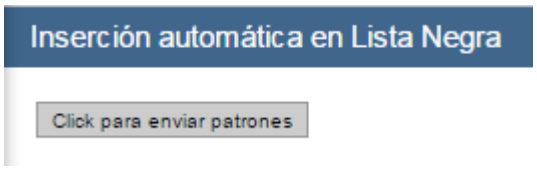


Ilustración 50. Formulario para ajuste de entorno automático. Fuente Elaboración propia.

5.6. Herramienta de visualización de datos. Power Bi.

La información que se pretende visualizar en los informes creados en Power Bi, debe estar lo más actualizada posible gracias a Power Bi Gateway, ya comentada con anterioridad. Esta herramienta, como ya se ha dicho realizará las consultas a la base

de datos, o bien de manera programada o mediante una orden al hacer clic en el botón refrescar de la aplicación web Power Bi. Si bien es cierto, los horarios para programar las actualizaciones de datos (consultas automáticas sobre la base de datos), se deben escoger con una cierta lógica, es decir deben aprovecharse al máximo, ya que desafortunadamente, una de las limitaciones de la herramienta, es que tan solo pueden programarse hasta ocho horarios distintos. Además el mínimo tiempo, que permite la herramienta, transcurrir entre un horario y el siguiente es de 30 minutos. De tal forma que si se programa una actualización para las 8:30 A.M., la siguiente actualización no podrá ser hasta, como mínimo, las 9:00 A.M.

Una vez conocida esta limitación de la herramienta, se toma la decisión de que las actualizaciones se lleven a cabo durante el horario matutino. Parece lógico que tanto para una antena situada en una zona común universitaria (cafetería) como para otra ubicada en una oficina, las horas de máxima detección de dispositivos pueden darse en horarios de mañana. De forma que las horas elegidas para las actualizaciones automáticas quedan de la siguiente forma: 9:00, 12:00, 13:00, 13:30, 14:00, 14:30, 16:00, 20:00, sumando un total de 8 horarios programados. Nótese que entre la una y las dos y media del mediodía, se producen las actualizaciones con la máxima frecuencia dentro de la jornada. Esto se debe a que la máxima detección de dispositivos en la antena que se encuentra en cafetería, se haya durante ese período de tiempo (El alumnado o profesorado que come en cafetería).

6. Análisis de resultados.

Una vez llegados a este punto del proyecto, se procede a detallar qué consultas se van a realizar sobre la base de datos para mostrarlas en la herramienta de visualización comentada en el subapartado anterior. Actualmente, existen dos antenas en dos ubicaciones distintas, de donde ambas han estado recopilando información durante varios meses. Estos datos recogidos se comentan mediante informes que muestran datos recogidos en lugares y tiempos establecidos, donde se observa la cantidad de personas que visitan el lugar.

6.1. Análisis de la población mediante la visualización de informes.

Los gráficos que se incluyen en este subapartado, muestran información que hace referencia a las consultas que se indican a continuación:

- **Localización de receptores** de dispositivos.
- Contador de **población distinta** en el **lugar** por **años, meses, días y horas**.
- Ranking de **dispositivos con un número mayor de frecuencias** en un lugar.
- Tiempo de **estancia de un usuario** en un lugar determinado.
- **Búsqueda personalizada de un determinado dispositivo en un lugar** por horas, de manera que permite observar el **tiempo de estancia de un individuo** en dicho lugar.

Ilustración 51. Mapa con la ubicación en cafetería de la Escuela Politécnica de Cáceres a nivel nacional. Fuente: Elaboración propia.

Ilustración 52. Mapa con la ubicación de cafetería a nivel municipal. Fuente: Elaboración propia.

En el mapa la herramienta de visualización de datos, recoge la latitud y la longitud de la base de datos en el registro de una antena y la muestra en el mapa. Además se le añade una leyenda cuyo atributo asignado es el nombre de la ubicación de la antena para poder saber qué punto en el mapa se corresponde con cada ubicación. Esto se recomienda cuando existen más de una antena conectadas al sistema.

Las siguientes tres imágenes muestran los datos obtenidos durante los períodos de tiempo anuales, mensuales y diarios.

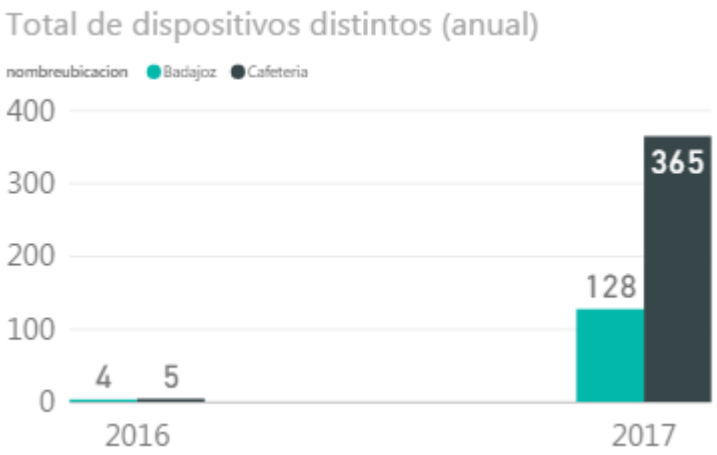


Ilustración 53. Total de dispositivos distintos recogidos por dos antenas receptoras en distintas ubicaciones a nivel anual. Fuente: Elaboración propia

En la figura anterior, se observa una diferencia notable entre la detección de dispositivos en el año 2016 con el actual 2017. Esto se debe a que la antena se instaló en

el lugar a finales del año 2016, a mediados del mes de Diciembre, es por esta razón que a día de hoy en Mayo, exista esta diferencia de detecciones, ya que en 2017 se ha estado recopilando información durante cinco meses. Por lo tanto esta comparativa no resulta (todavía) significativa.

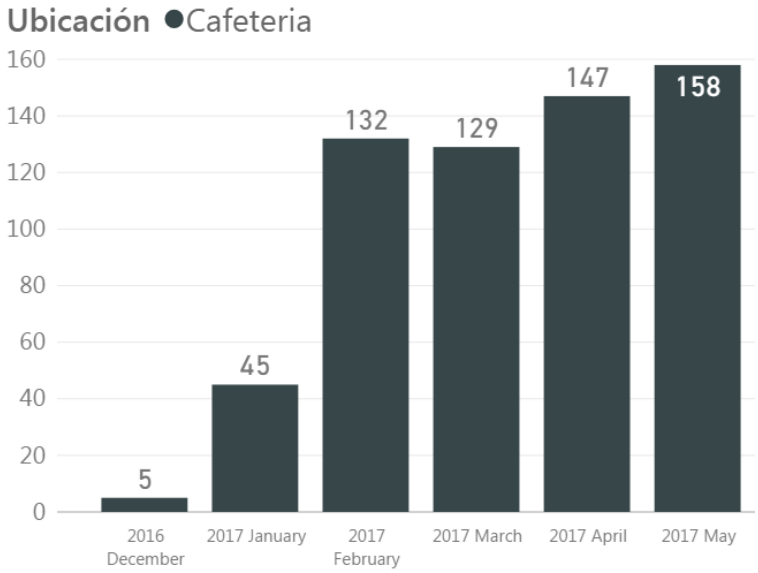


Ilustración 54. Detección distintiva a nivel mensual de dispositivos Bluetooth. Fuente: Elaboración propia.

A diferencia de la comparativa anual, aquí ya se pueden resaltar algunos datos que explican el comportamiento de personas en este lugar. Esta antena en cuestión ha sido instalada en la cafetería de la Escuela Politécnica de Cáceres, de manera, que la gran mayoría de usuarios en este lugar son alumnos y profesores, pero en su gran mayoría alumnos. En la imagen de la comparativa mensual se aprecian cambios entre la densidad poblacional durante los meses Diciembre, Enero y Febrero. En cuanto al mes de Diciembre, puede deberse al período vacacional existente durante esta época del año (Festividades navideñas) y a que la antena, como se ha comentado, fue instalada a mediados de este mismo mes. Por otro lado y como ya se sabe, al finalizar estas fiestas, da comienzo el período de exámenes, es por esta razón que en Enero aún no se considera época de mayor actividad en la facultad, ya que muchos alumnos prefieren estudiar en bibliotecas ajenas a la sala de estudios que ofrece la facultad y a la todavía carencia de clases de asignaturas. Mientras tanto, durante el mes de Febrero, es notable el crecimiento poblacional debido al inicio de clases de asignaturas del segundo cuatrimestre, y por lo tanto un aumento de población detectada en este lugar. Además

pueden observarse de forma clara que los cuatro meses (Febrero, Marzo, Abril y Mayo), son meses en los que existe mayor frecuencia estudiantil en el lugar.

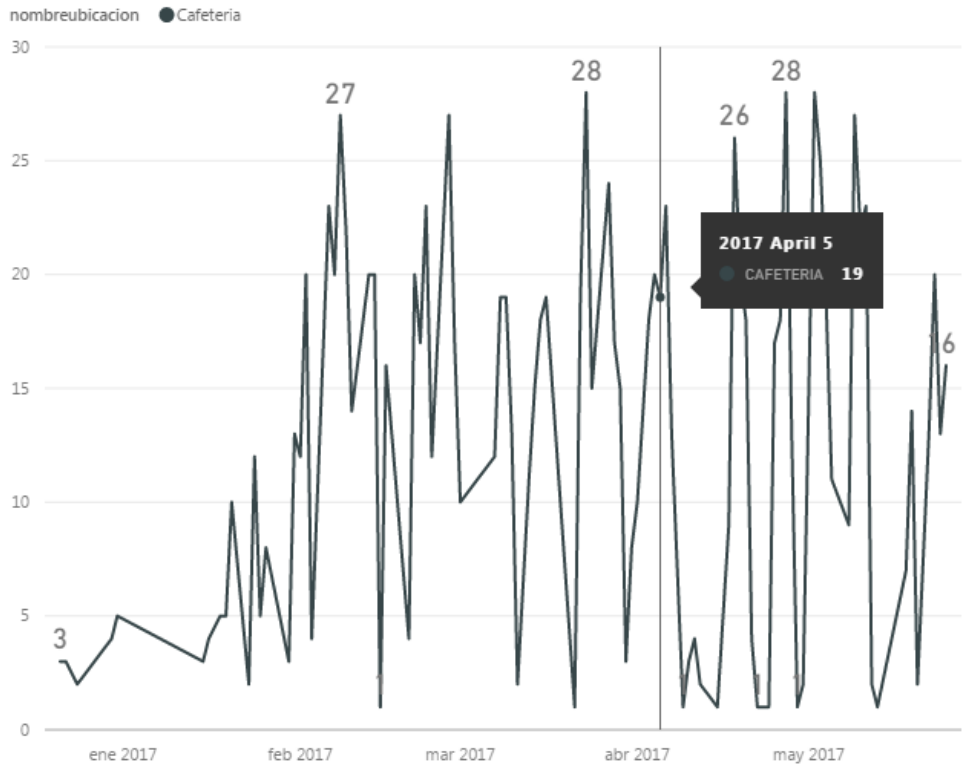


Ilustración 55. Detecciones diarias Bluetooth. Fuente: Elaboración propia.

En este caso, la mediciones se llevan a cabo por días, donde como se puede apreciar, el visual nos ofrece información sobre cuántos dispositivos distintos ha sido detectados en un día mediante conexiones Bluetooth en la misma cafetería de la Escuela Politécnica. Si se observa en la figura, el día 5 de Abril de 2017, se detectaron un máximo de 19 dispositivos distintos. Como ya se ha dicho, aunque la tecnología Bluetooth se implantó a mediados de los años 90, no es hasta la fecha de hoy que comienza su expansión con el internet de las cosas, no obstante, en la actualidad, todavía no ha llegado a la cumbre en cuanto a integración en la sociedad. Es por esta razón que son todavía pocas detecciones. Sin embargo, en la figura anterior, la cual muestra mediciones a nivel diario, se puede diferenciar también de forma clara, el período lectivo del no lectivo o festivo.

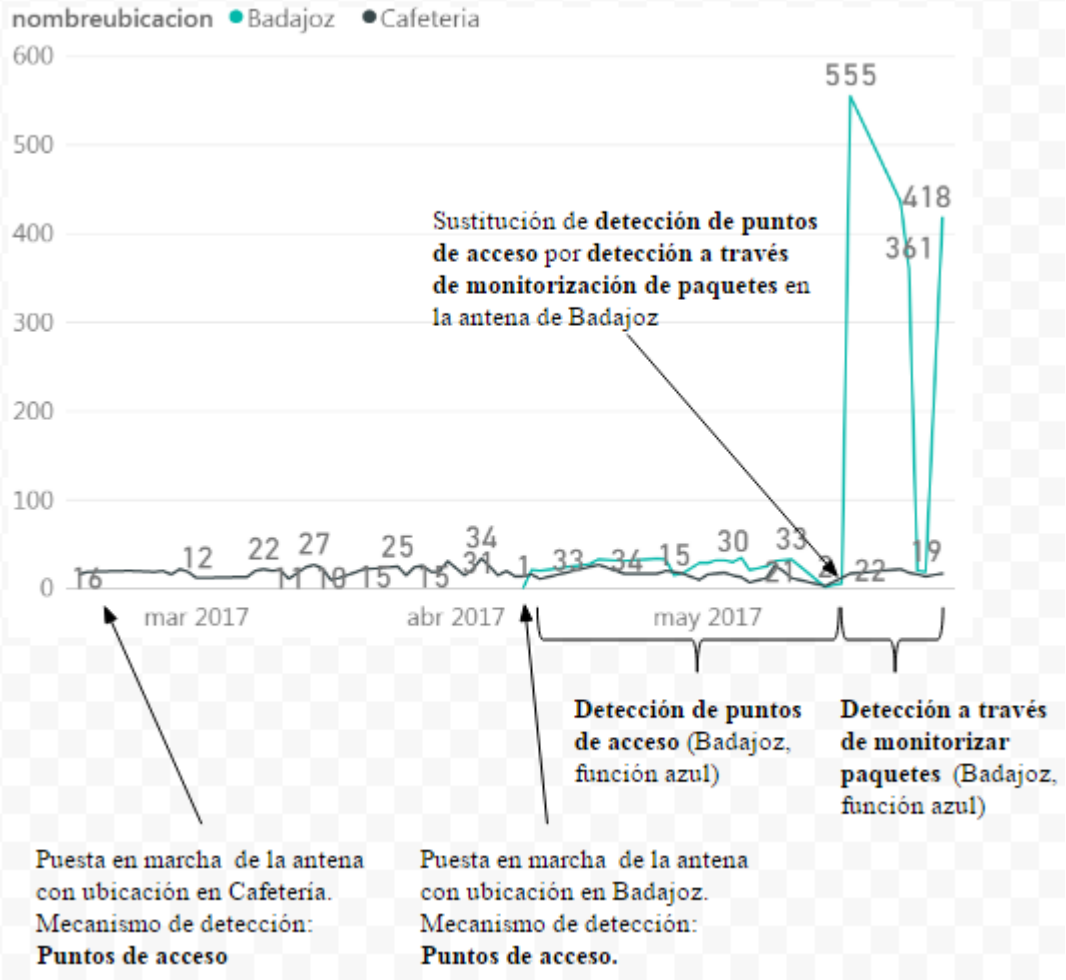


Ilustración 56. Comparativa de tecnologías de detección de dispositivos conectados a Internet. Fuente: Elaboración propia.

En la figura anterior, se comparan las dos tecnologías que detectan dispositivos conectados a la red de redes. Hasta mediados del mes de Febrero, las dos antenas utilizadas en este proyecto, tan solo han estado midiendo datos de conexiones Bluetooth. No es hasta dicha fecha que se implantan, inicialmente en la antena situada en Cafetería y dos meses más tarde en la antena que se ubica en Badajoz, el mecanismo de recopilación de datos referentes a puntos de acceso WiFi. Por lo tanto el contraste de ambas antenas no da comienzo hasta principios del mes de Abril. No obstante, cabe decir que las mediciones tienen una tendencia similar en cuanto a totales de mediciones por día. Es cierto que la antena situada en cafetería obtiene más detecciones de dispositivos distintos que la de Badajoz en un mismo día, pero, aún así se consideran pocas. De manera, que tras observar que utilizando el mecanismo de detección de puntos de acceso, las detecciones apenas sobrepasan las 30 al día, se decide hacer uso

del método de detección de dispositivos mediante la recolección de paquetes “tshark” a mediados del mes de Mayo. El éxito de esta nueva metodología fue inmediato. Cabe destacar el crecimiento notable que existe en cuanto a mediciones diarias, desde que se instala el nuevo mecanismo en la antena de Badajoz. A su vez, se refleja bastante bien la caída de mediciones (de 555 a 418), debido a la inserción de dispositivos estáticos en la lista negra.

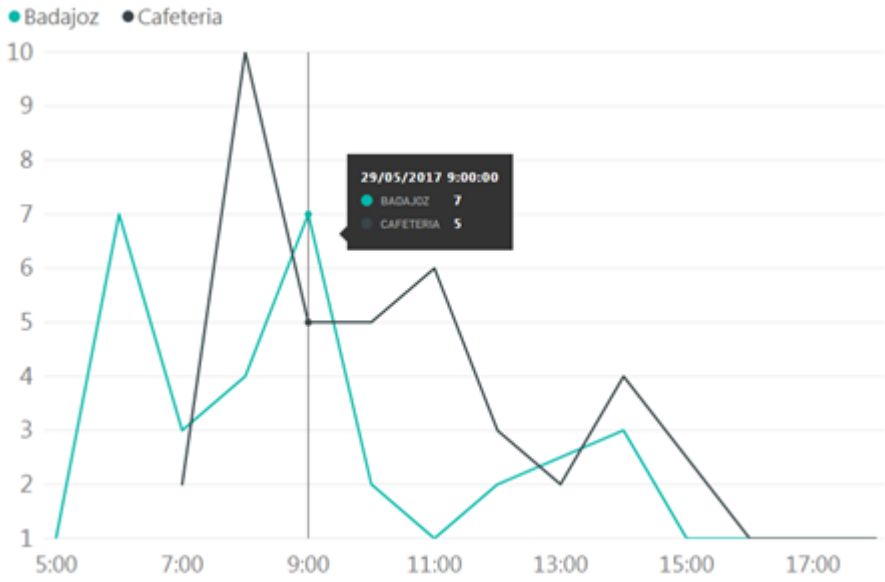


Ilustración 57. Visual de detecciones Bluetooth por hora. Fuente: Elaboración propia.

Una manera de llevar un control diario de detecciones de dispositivos, es mediante la visualización organizada por horas, la cual muestra el total de dispositivos distintos recogidos que tienen lugar entre determinadas horas del día. La figura muestra una comparativa de detecciones de dispositivos conectados vía Bluetooth recogidas el día 29 de Mayo de 2017. De igual forma, el mismo visual se puede configurar para ofrecer una vista semanal de detecciones. Se recuerda que la herramienta Power Bi es interactiva, con lo cual permite al usuario interactuar con los datos recogidos.

Ilustración 58. Visual de detecciones Bluetooth a nivel semanal en cafetería. Fuente: Elaboración propia.

La figura muestra las detecciones recogidas en la semana número 21 del año 2017, es decir la semana que va desde los días 22 a 28 de Mayo. Como es evidente, en los días 27 y 28 de la semana no se recogen datos por ser Sábado y Domingo respectivamente.

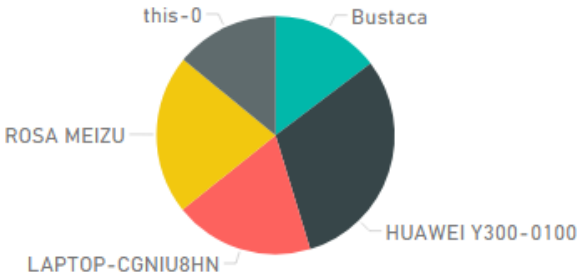
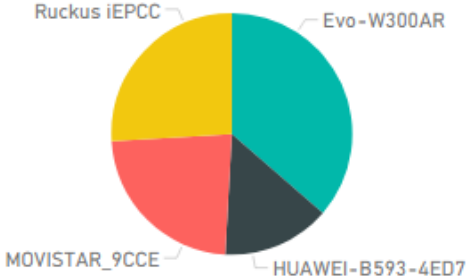
Dispositivos más frecuentes en cafetería	
Bluetooth	**Conexión a Internet**
	

Ilustración 59. Comparativa de dispositivos detectados con más frecuencia. Fuente: Elaboración propia.

La figura anterior recoge la información referente a aquellos dispositivos que superan un número de detecciones durante los escaneos emitidos por la antena receptora desde el momento en que se instaló en el lugar.

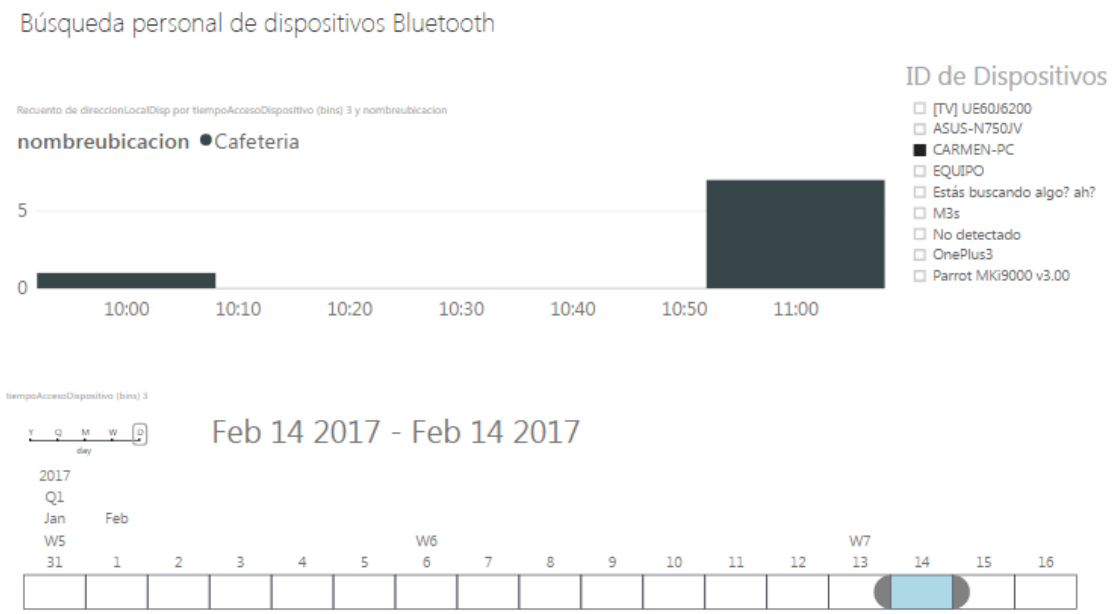


Ilustración 60. Búsqueda personalizada de dispositivos. Fuente: Elaboración propia.

La imagen muestra un visual de un informe en el cual se aprecian algunos datos recogidos, el día catorce de Febrero entre las diez y las once de la mañana, por la antena receptora situada en la cafetería de la Escuela Politécnica de Cáceres. El sistema recoge señales que provienen del dispositivo de una usuaria que según parece su nombre es Carmen y a juzgar por el identificador del dispositivo "CARMEN-PC", es muy probable que utilizara un ordenador portátil. Gracias a estos visuales, se puede medir el tiempo que una persona permanece en un determinado lugar. En algunas ocasiones puede ocurrir que la antena no reciba la señal de un determinado dispositivo entre dos momentos casi continuos puntuales. Por ejemplo, en la imagen, se ve como el dispositivo se detecta entre las diez y las diez y diez de la mañana y ya no vuelve a detectarse hasta el período de tiempo que va desde las once menos diez hasta las once. En estos casos se deduce que la usuaria del dispositivo ha permanecido allí durante todo el tiempo, sin embargo es probable que haya suspendido el portátil o que la propia antena no haya detectado el dispositivo.

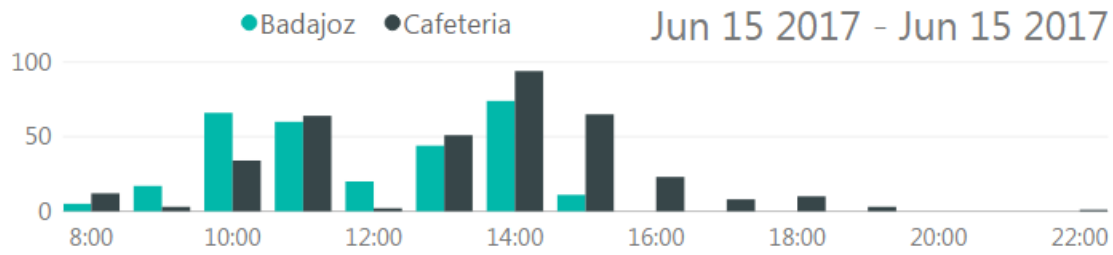


Ilustración 61. Horas de actividad. Posibilidad de acceso indebido. Fuente Elaboración propia.

La gráfica, que se observa en la ilustración número 60, informa sobre los horarios de máxima actividad poblacional. Para el caso de cafetería las detecciones se producen entre las ocho y las diez de la noche. Pero el momento en el cual se detectan más dispositivos, se produce a las dos del mediodía. Esto se debe a que profesores y estudiantes comen allí. Además informa que a partir de esta hora punta, las detecciones comienzan a menguar hasta las siete de la tarde. A partir de las ocho las detecciones son prácticamente nulas. A las diez se produce una detección. Este podría tratarse de un caso de acceso inadecuado, aunque es posible que aún se detecte algún dispositivo perteneciente a personal de limpieza.

Ilustración 62. Segundo posible caso de acceso indebido. Fuente: Elaboración propia.

En este segundo caso, también en cafetería, se produce una detección a las tres de la mañana, a diferencia del caso anterior que podría tratarse de algún trabajador que aún estuviera en el lugar ya que la detección se producía a las diez de la noche, en este caso la detección se produce en horas de actividad poblacional nula para este lugar, teniendo en cuenta que se trata una cafetería que se encuentra en la Escuela Politécnica y que a estas horas permanece cerrada. Por lo tanto podría tratarse de un posible caso de intrusismo. Desafortunadamente su nombre no ha sido detectado por la antena.

6.2. Estudio de casos de proximidad.

En este subapartado se lleva a cabo un estudio de los distintos casos de proximidad que se observan tras las detecciones. Se van a estudiar tres casos en cuyos gráficos se observa un crecimiento del número de detecciones sobre un determinado dispositivos en momentos consecutivos. El estudio propone determinar si el crecimiento citado puede deberse a la reducción de la distancia que se sitúa entre un dispositivo detectado y su receptor.

6.2.1. Primer caso de proximidad.

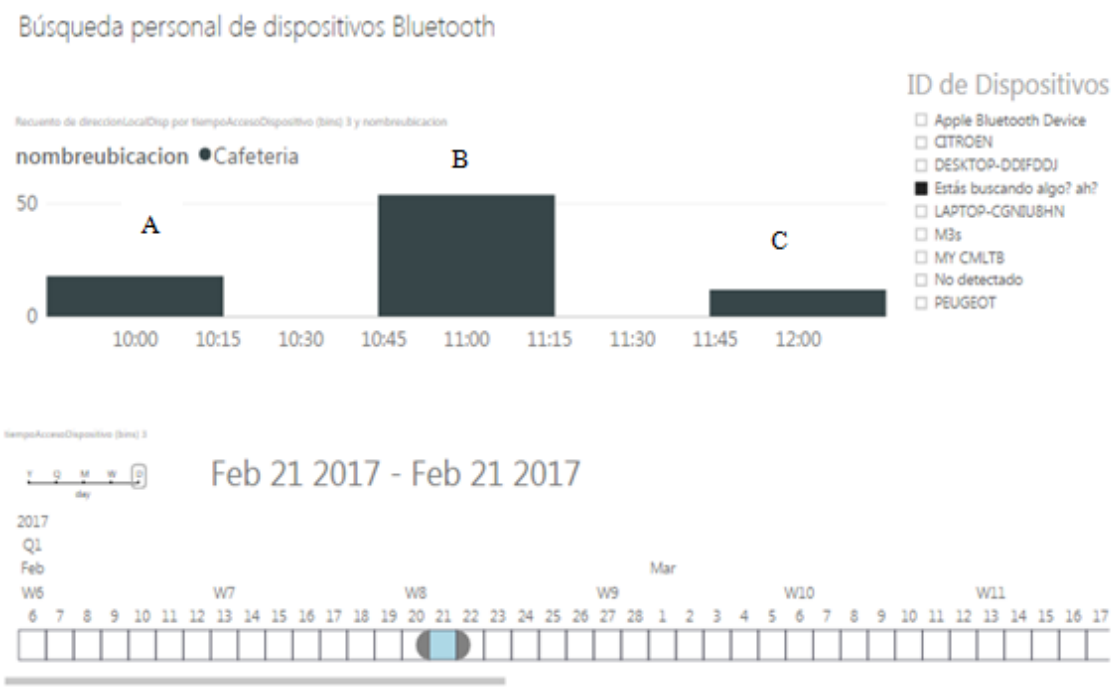


Ilustración 63. Primer posible caso de proximidad. Fuente: Elaboración propia.

La gráfica muestra datos recogidos, entre las diez y las doce de la mañana del 21 de Febrero de este mismo año 2017, de un dispositivo cuyo identificador es "¿Estás buscando algo ah?". De igual forma que en el caso anterior, se puede deducir, que aunque no se detecten dispositivos en los períodos de tiempo que comprenden entre las 10:15 - 10:45 y 11:15 - 11:45, es muy probable que el usuario permaneciera en el lugar durante las dos horas. En este caso entra en juego una nueva variable: La proximidad. El

segundo conjunto de detecciones (conjunto B), suma más de cincuenta si se compara con los otros dos (A y C), la diferencia es bastante notable. Esto puede significar que el usuario entre las 10:45 y las 11:15 se ha podido desplazar hacia un lugar más próximo a la antena, es decir ha reducido la distancia que se sitúa entre el propio usuario y el receptor de señal.

6.2.2. Segundo caso de proximidad.

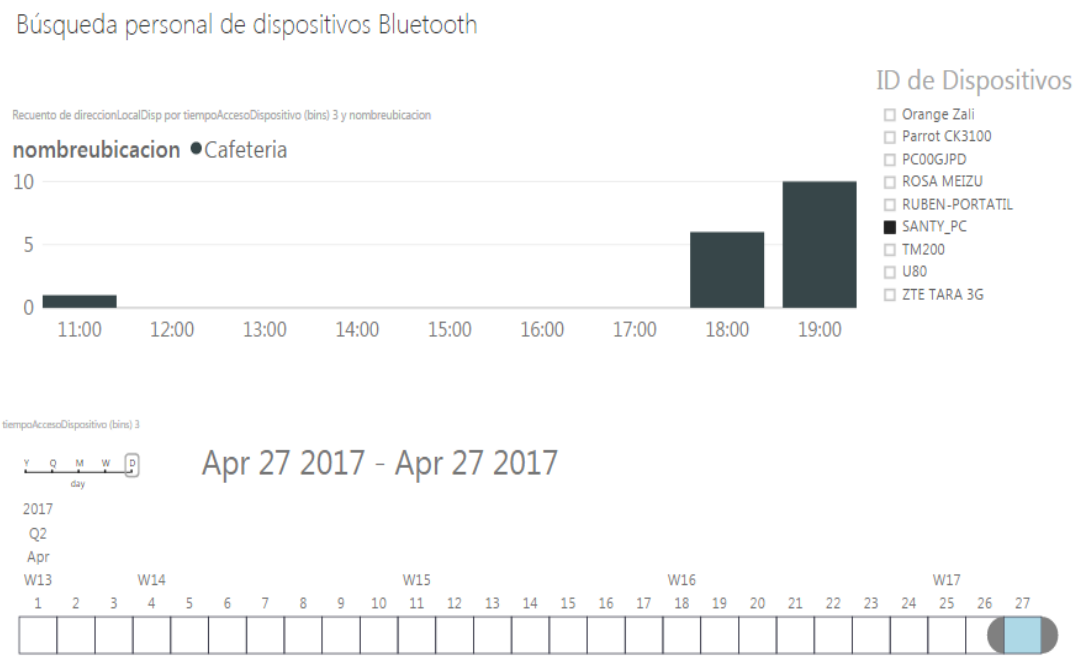


Ilustración 64. Segundo posible caso de proximidad. Fuente: Elaboración propia.

En este caso, el dispositivo cuyo nombre o identificador es SANTY-PC, accedió a cafetería en un momento que tiene lugar entre las once y las doce de la mañana y por la tarde volvió al lugar y permaneció durante aproximadamente dos horas. Es probable que durante las horas de la tarde se situase en un lugar más cercano a la antena receptora, ya que su número de detecciones es mayor por la tarde. Cabe señalar que los datos recogidos pueden haber sido resultado de otros factores como por ejemplo que durante la mañana el dispositivo Bluetooth del dispositivo estuviese encendido durante un período más corto que durante la tarde e incluso que el escaneo de entorno no haya detectado el dispositivo.

6.2.3. Tercer caso de proximidad.

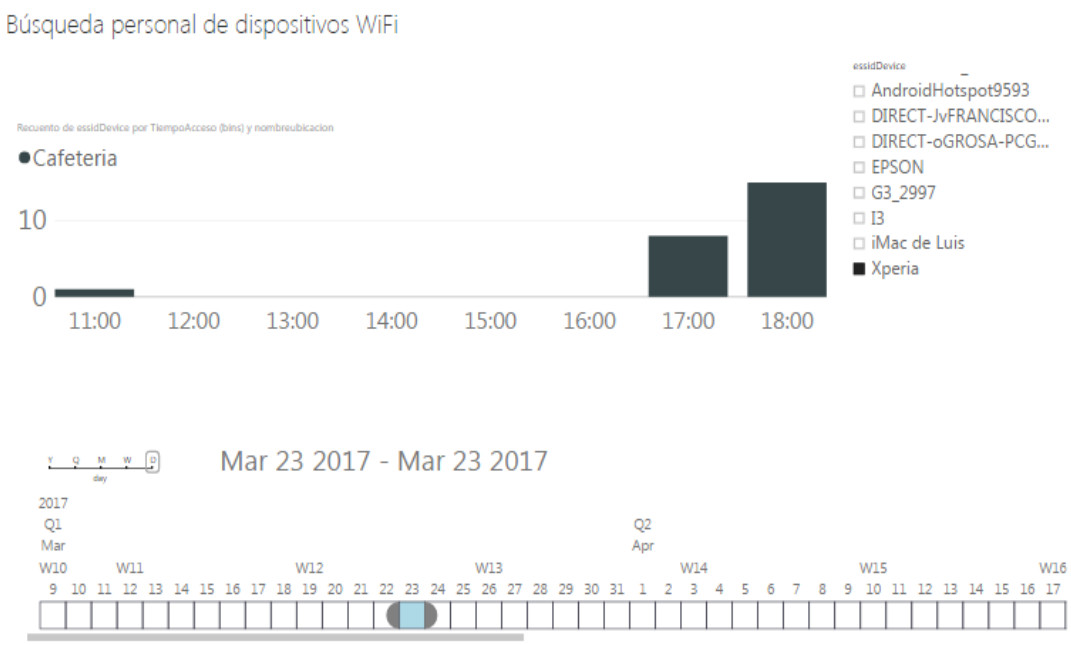


Ilustración 65. Tercer posible caso de proximidad. Fuente: Elaboración propia.

En este visual se reflejan datos recogidos por la antena receptora. En este caso los datos recogidos son de conexiones WiFi. El dispositivo con identificador Xperia ha sido detectado durante las once de la mañana y por la tarde entre las cinco y las siete. Como se puede observar el número de detecciones sobre el dispositivo es mayor durante la tarde. Es por esta razón que se cree que estamos ante un tercer caso de proximidad. Es posible que el dispositivo haya estado más cerca del receptor en las dos horas que ha permanecido en cafetería durante la tarde del día 23 de Marzo del año 2017.

6.2.4. Conclusiones sobre casos de proximidad.

Tras el estudio de los casos anteriores referentes a la proximidad entre un dispositivo detectado y su dispositivo receptor, a continuación se citan factores que podrían verificar la proximidad a la que se hace referencia y por el contrario, se citan factores que la hacen discutible y nos alejan, a su vez, del alcance del objetivo (Medir la proximidad entre dispositivos detectados y receptor).

Factor a favor	Factores en contra
- Contrastes notables del número de detecciones para un mismo dispositivo en momentos casi inmediatos.	- Posibilidad de que el receptor se encuentre en un período de descanso cuando el dispositivo se encuentra en las inmediaciones de la antena.
	- Fallo en la detección de señal.
	- El usuario decide si tiene el dispositivo en modo visible o apagado para los casos de detecciones Bluetooth y puntos de acceso.
	- El usuario puede tener o no el dispositivo encendido.

Tabla 24. Estudio de proximidad. Fuente: Elaboración propia.

Tras comprobar que el número de factores que hacen que la proximidad de un dispositivo detectado aumente con respecto a su dispositivo receptor es cuatro veces mayor en los factores en contra, se determina que la teoría de la proximidad queda pendiente para una mejora futura de este proyecto y por el momento asume un carácter empírico.

6.3. Inserciones en lista negra.

El siguiente gráfico muestra los resultados de un ejemplo de inserción de nombres en la lista negra. Los datos pertenecen a dispositivos detectados por dos antenas. Una antena situada en la cafetería de la Escuela Politécnica de Cáceres y otra en Badajoz. A la antena cuya ubicación es Badajoz y la cual escanea el entorno de unas oficinas, se le instala el programa de detección de dispositivos mediante monitorizaciones de tramas a finales del mes de Mayo. Nótese que se produce una caída de detecciones tras el primer día de instalación, que baja desde las 580 hasta las 420 aproximadamente hacia finales de la primera semana, en que la antena ha estado detectando dispositivos mediante la técnica de monitorización. Esto a la inserción de dispositivos detectados durante horas de baja actividad poblacional en el lugar. Las mediciones pasan de ser en bruto a netas y se ajustan al máximo a la población existente durante las siguientes jornadas. Al finalizar la semana laboral las mediciones son aproximadamente nulas

Ilustración 66. Tipos de medidas. Fuente: Elaboración propia.

6.4. Conclusiones sobre la comparativa de mecanismos detección

En este subapartado se observan mediante dos gráficas, los resultados obtenidos en mediciones. Estas gráficas muestran las detecciones de dispositivos distintos detectados durante el año 2017 y el mes de Junio del mismo año.

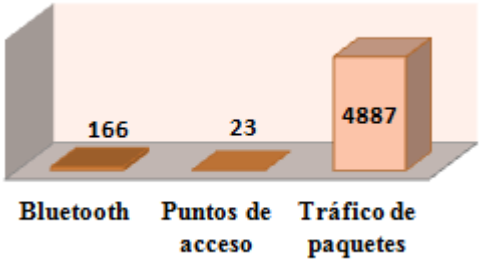


Ilustración 67. Mediciones por tecnologías para el mes de Junio de 2017.
Fuente: Elaboración propia.

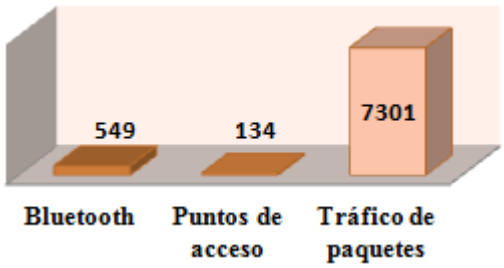


Ilustración 68. Mediciones por tecnologías a nivel anual hasta mediados del año 2017.
Fuente: Elaboración propia.

En las gráficas mostradas se aprecia como la detección mediante la monitorización en busca de tramas de paquetes sobresale de manera notoria frente a los otros dos mecanismos, tanto a nivel mensual como anual. En ellas se refleja la actitud que tiene un usuario hacia la posibilidad de ser detectado y en consecuencia desactiva o desactiva la visibilidad del mismo. Este es el caso que se retracta para los casos de detecciones Bluetooth y puntos de acceso, en el que en la mayoría de los casos los usuarios deciden desactivar esta visibilidad, probablemente por miedo o desconfianza a ser vistos por otros dispositivos. Por el contrario, para el caso de detecciones mediante monitorización de tramas, estos usuarios no saben que están siendo vistos por otros dispositivos, por lo tanto ignoran la posibilidad de que un receptor puede extraer información referente a la identificación de los dispositivos detectados.

6.5. Valoración de objetivos del proyecto.

La autovaloración de la consecución de objetivos mostrada en la siguiente tabla, se lleva a cabo por niveles, de donde cada objetivo cuya casillas coloreadas con colores (verde, amarillo y rojo), indican un porcentaje de alcance, los cuales aquí se reflejan:

Porcentaje de alcance de objetivos por color	
Verde	> 80%
Amarillo	> 50 % < 80 %
Rojo	0%

Tabla 25. Porcentajes de alcance de objetivos por colores. Fuente Elaboración propia.

Tabla de autovaloración objetivos del sistema		
OP Contabilizar personas	**OS_1- Uso de dispositivos receptores encargados de recoger la información.**	**OC_1- El sistema debe ofrecer escalabilidad.**
		OC_2- Medir la proximidad entre dispositivos detectados y receptor.
		OC_3- Debe existir una lista negra que almacene nombres de dispositivos no deseados.
		OC_4- El número de detecciones debe ajustarse al máximo a la realidad.
		OC_{10}- Las antenas deben iniciar los escaneos de forma automática.
		OC_{11}- Los dispositivos receptores ponen la información a disposición del servidor a través de un servidor web.
		OC_{12}- Detecciones únicas en cada escaneo.
	OS_2- La información se almacena en una base de datos.	**OC_5- La base de datos debe ser relacional y debe permitir escalabilidad horizontal.**
	OS_3- El servidor se debe de encargar se recoger la información de las antenas y la almacena en la base de datos.	**OC_6- El servidor debe permitir activar una o varias antenas conectadas a la vez, asignándoles un tiempo de recarga.**
		OC_7- El servidor debe poder localizar la ubicación de todas las antenas conectadas.
		OC_8- El servidor debe ofrecer la funcionalidad de añadir nombres de dispositivos no deseados a lista negra.
		OC_{11}- Los dispositivos receptores ponen la información a disposición del servidor a través de un servidor web.
	OS_4- La información se debe poder visualizar haciendo uso de informes.	**OC_9- La herramienta de visualización debe de realizar constantes consultas a la base de datos para ofrecer la información actualizada.**

Tabla 26. Autovaloración de objetivos. Fuente: Elaboración propia.

La tabla muestra dos objetivos de carácter complementario que no ha sobrepasado el 80 por ciento de su alcance. A continuación se hace una reflexión sobre sus porcentajes obtenidos:

- Medir la proximidad entre dispositivos detectados y receptor. La proximidad, como ya se ha explicado en el capítulo de análisis de resultados, este objetivo asume un carácter empírico y queda por lo tanto pendiente de conseguir su alcance en mejoras futuras de este proyecto. Su alcance tan solo podrá conseguirse si se termina demostrando factores que verifiquen que un dispositivo detectado se ha encontrado en un determinado momento más cerca o más lejos de una antena receptora.

- Las mediciones obtenidas deben ajustarse a la realidad. El problema aquí lo ocasionan los mecanismos de detección Bluetooth y de puntos de acceso, debido a que como ya se ha explicado, un usuario decide, en la mayoría de casos si es visto o no activando o desactivando su opción de visibilidad. Por lo tanto, tan solo se ajustan a la realidad las mediciones obtenidas mediante monitorización de tramas de paquetes, que añadiendo los determinados dispositivos a lista negra inicialmente, ofrecen unos resultados mucho más reales que en los otros dos casos anteriores.

7. Conclusiones

En este capítulo, se realiza un desglose de objetivos para valorar en qué medida se han llevado a cabo. Para las valoraciones se va a hacer uso de tres tablas: Objetivos generales y específicos de desarrollo y objetivos personales. La información que se indica en las tablas hace referencia al objetivo a cumplir, la valoración autocrítica sobre su consecución, los detalles llevados a cabo para obtener dichas consecuciones y en el caso de los objetivos personales, además se añade un grado de dificultad. Finalmente en el capítulo se lleva a cabo una reflexión a modo de conclusión, donde se añaden ciertas mejoras futuras al presente proyecto de fin de carrera.

7.1. La conclusión final.

El presente proyecto ha conseguido en su mayor medida los objetivos descritos, sin embargo existen ciertos aspectos que se detallan a continuación. Cada aspecto aquí detallado, hace referencia a cada uno de los mecanismos empleados para la detección de datos. Así mismo se comenta una mejora futura para el sistema que se va a llevar a cabo próximamente.

7.1.1. Detección Bluetooth.

Detección de pocos dispositivos en el escenario de escaneos. Esto se debe a la todavía creciente expansión de dispositivos conectados a través de este mecanismo. En un futuro cuando se establezca IoT (Internet de las cosas), este método de detección obtendrá mejores resultados. Asimismo, la decisión en el lado del usuario de ser visto o no a través de esta tecnología ha reducido el número de detecciones. Esto se debe a que en un teléfono móvil, un usuario, en la gran mayoría de casos, elige si desea ser visto o no activando la visibilidad. de su dispositivo.

7.1.2. Detección puntos de acceso.

Sucede algo parecido al caso Bluetooth. Son pocos dispositivos detectados. El usuario del dispositivo decide también ser visto o no, activando o desactivando la opción de visibilidad para utilizar su teléfono como punto de acceso a Internet.

7.1.3. Detección mediante Wireshark.

Éxito rotundo. Son las dos palabras que mejor definen la experiencia tras el uso de este mecanismo de detección. A diferencia de las dos anteriores, el usuario no sabe que está siendo monitorizado. Es la razón por la cual el número de detecciones se ajusta, en su mayor medida, a la realidad referente a la población que existe en los lugares escogidos para escanear. Cabe señalar, que el sistema se mantiene dentro de los dominios legales ya que utiliza una herramienta empleada para fines didácticos en ciclos formativos y carreras universitarias y no comprometen datos de carácter personal que invadan la intimidad de personas detectadas.

7.2. Mejoras futuras aplicadas al sistema.

Este proyecto se flexibiliza a llevar a cabo integraciones futuras en el mismo. A continuación se citan algunas de las mejoras que en un futuro pueden optimizar los servicios que el sistema desarrollado proporciona:

- **Servicio de almacenamiento de datos FireBase:** Una mejora que proporciona una visualización de datos en tiempo real son las bases de datos FireBase, de tal forma que los cambios en la visualización de datos, se producirían en el momento en que son modificados en la base de datos.

- **Matching Learning:** Una buena mejora es sin duda, el aprendizaje del flujo de la información, de tal forma que, como resultado, se podrían obtener predicciones, que permitan tener una previsión del comportamiento de la población en lugares y momentos determinados.

- **Proximidad:** En este proyecto no ha podido demostrarse que en un momento determinado podríamos estar ante un caso de proximidad, por eso se propone un estudio que sea capaz de demostrarlo.

- **Sistema de notificaciones** que envíe un aviso a un usuario del sistema, si se detecta un intruso en horas de inactividad en el lugar.

8. Manual de usuario.

Una vez finalizadas las conclusiones y llegando a la recta final de la documentación para el presente proyecto de fin de carrera, y con el objetivo de facilitar el uso del sistema a usuarios, en este capítulo se van a detallar los pasos a tener en cuenta para la instalación y puesta marcha del mismo.

a) El primer paso a realizar es escoger un lugar donde colocar el dispositivo receptor.
b) A continuación, buscar en el mismo una ubicación donde conectar a la red eléctrica el dispositivo. Una vez conectado el dispositivo, éste espera a ser activado.
c) Para activar el dispositivo, acceder al enlace del servidor. Una vez dentro, se pueden observar las antenas conectadas sin activar, ya que cuyos valores de refresco están todavía a cero. Para activarlas y que éstas empiecen a escanear sus entornos se les debe asignar un valor. Además, mediante el enlace "Localizar", se pueden observar en un mapa de Google sus ubicaciones a nivel global.

Ilustración 69. Antenas conectadas sin activar con refresco a cero. Fuente: Elaboración propia

d) Una vez instalado el receptor se aconseja dejar encendido el dispositivo durante horas de inactividad para detectar dispositivos de carácter estático. Estos dispositivos se deben descartar para obtener medidas de entorno que se ajusten al máximo a la realidad. Una vez pasada la primera noche se vuelve acceder al enlace del servidor y se recogen dos opciones:
   - Inserción manual de lista negra. Esta es una buena idea para insertar máscaras, por ejemplo, si se sabe que en el escenario pueden existir

dispositivos como impresoras, televisiones o monitores que pueden ser detectados por la antena receptora, pueden añadirse máscaras de tipo “Laser” o “TV” a lista negra. Esto se debe a que a dichos dispositivos se les ha podido asignar estos essid’s en fábrica. De esta manera la antena cuando detecta dispositivos con estos nombres, al estar insertados en lista negra, los desprecia y no los almacena.

- La segunda opción es la inserción automática. Se trata de la opción más cómoda. Para poder llevar a cabo esta opción se hace click en el botón del formulario “Inserción automática en Lista Negra” del mismo enlace de la web del servidor.

e) Una vez la antena recibe señal de dispositivos de carácter no estático, es decir dispositivos que provienen de personas que acceden al lugar escaneado, el usuario se debe crear una cuenta en la herramienta Power Bi. Este programa conectará con la base de datos del sistema. Para la configuración de conexión de Power Bi con la base de datos y generación de informes, el usuario puede hacer uso del video tutorial “Introducción a Power Bi. Conexión con datos”.

f) Finalmente, con todas estas configuraciones llevadas a cabo es suficiente, para disponer de un sistema de telemetría activado.

9. Referencias bibliográficas

Marta Fernández Melgarejo (26 de Marzo de 2012). *Indicadores clave para el éxito en Retail*. Recuperado de http://www.puromarketing.com/13/12554/claves-para-exito-retail.html (10)

Web Comenzando desde cero (15 de Mayo de 2017). *Qué es un KPI. Preguntas que debes hacerte antes de elegirlos.* Recuperado de: http://comenzandodecero.com/que-es-un-kpi/. (11)

Instituto Nacional de Ciberseguridad en España (26 de Julio de 2016). *Analizando Bluetooth.* Recuperado de https://www.certsi.es/blog/analizando-bluetooth. (12)

BBC Mundo (13 de Junio de 2016). *5 ventajas de Bluetooth 5, la última versión de la popular tecnología inalámbrica.* Recuperado de: http://www.bbc.com/mundo/noticias-36517394. (13).

Sandra Fernández Moreno (26 de Junio de 2015). *¿Qué es WiFi? ¿Qué significa y para qué sirve?*. Recuperado de http://www.valortop.com/blog/que-es-wifi-que-significa-y-para-que-sirve. (15).

David Justo (28 de Febrero de 2017). *El uso de 'smartphones' en España se duplica en los últimos cinco años.* Recuperado de: http://cadenaser.com/ser/2017/02/28/ciencia/1488281552_888684.html. 17

Bluetooth Special Interesting Group (25 de Febrero de 2017). *Bluetooth Smart & Smart Ready Market worth $5.57 billion by 2020*. Rcuperado de: http://www.marketsandmarkets.com/PressReleases/bluetooth-smart-ready.asp. 21.

Sergio De Luz (9 de Enero de 2016). *Este es el futuro del Wi-Fi de los próximos 4 años según la Wi-Fi Alliance.* Recuperado de https://www.redeszone.net/2016/01/09/este-es-el-futuro-del-wi-fi-de-los-proximos-4-anos-segun-la-wi-fi-alliance/.

Eurostat (28 de Enero de 2017). *Cloud computing statistics on the use by enterprises.* Recuperado de: http://ec.europa.eu/eurostat/statistics-explained/index.php/Cloud_computing_-_statistics_on_the_use_by_enterprises

Alfonso de Frutos (16 de Enero de 2017). *¿Se pueden detectar números de teléfonos cercanos?*. Recuperado de: https://www.androidsis.com/detectar-numeros-de-telefonos/

Cibernat (27 de Enero de 2017). *Computación en la nube.* Recuperado de http://cibernat.com/articulos/computacion-en-la-nube p32

Gerardo (25 de Septiembre de 2013). *Conoce los dispositivos y programas con que los gobiernos espían nuestros teléfonos.* Recuperado de: http://bitmovil.com/stingray-gossamer-programas/13018. p33

Web Sinologic (12 de Mayo de 2017). *OpenBTS: Cómo crear tu propia infraestructura GSM.* Recuperado de : https://blog.sinologic.net/2010-09/openbts-como-crear-tu-propia-infraestructura-gsm.html. p34

Alberto Relancio (9 de Octubre de 2017). *Wireshark, un gran analizador de protocolos.* Recuperado de: https://www.seas.es/blog/informatica/wireshark-un-gran-analizador-de-protocolos/. p36.

J. Steven Perry. (3 de Diciembre de 2012). *Introducción a la programación Java, parte 1: Conceptos básicos del lenguaje Java.* Recuperado de: https://www.ibm.com/developerworks/ssa/java/tutorials/j-introtojava1/index.html.

Cobo, A., Gómez, P., Pérez, D., y Rocha, R. (2005). *PHP y MySQL: Tecnologías para el desarrollo de aplicaciones web*. España: Ediciones Díaz de Santos.

Saúl Vázquez (20 de Octubre de 2013). *Conectarse automaticamente a sakis3g en Raspberry Pi (Script).* Recuperado de http://blogvazquezsaul.blogspot.com.es/2013/10/conectarse-automaticamente-sakis3g-en.html

GCF LearnFree.org. (14 de Febrero de 2017). *What is bluetooth?* Recuperado de: https://www.gcflearnfree.org/mobile-device-tips/what-is-bluetooth/1/

LINKSYS . (27 de Febrero de 2017). *¿Qué es MU-MIMO (o Wireless-AC de siguiente generación) y por qué la necesita?* Recuperado de: http://www.linksys.com/es/r/resource-center/qu%C3%A9-es-mu-mimo/

Sergio De Luz (7 Junio de 2015). *Wi-Fi AC con tecnología MU-MIMO: Todo lo que debes saber*. Recuperado de: https://www.redeszone.net/2015/06/07/wi-fi-ac-con-tecnologia-mu-mimo-todo-lo-que-debes-saber/

Francisco Javier Ruiz (15 de Diciembre de 2014). *Finlandia es el país lider en Cloud Computing de empresas en Europa.* Recuperado de: https://blog.dataprius.com/index.php/2014/12/15/finlanda-es-el-pais-lider-en-cloud-computing-de-empresas-en-europa/

Web Evilsocket.net (31 de Marzo de 2016). *How to Build Your Own Rogue GSM BTS for Fun and Profit.*Recuperado de: https://evilsocket.net/2016/03/31/how-to-build-your-own-rogue-gsm-bts-for-fun-and-profit/

Web Wikipedia (15 de Mayo de 2017). *Asterisk*. Recuperado de: https://es.wikipedia.org/wiki/Asterisk

Web Wikipedia (27 de Mayo de 2017). *Wireshark*. Recuperado de: https://es.wikipedia.org/wiki/Wireshark

Web Raspberry Shop (21 de Abril de 2017). *Raspberry PI.* Recuperado de: https://www.raspberryshop.es/

Web Cosas de móvil (10 de Mayo de 2017). *Smart-Watch/Phone & Accesorios.* Recuperado de: http://www.cosasdemovil.es/raspberry/sunfounder-rt5370-usb-wireless-network-wifi-adapter-for-raspberry-pi-with/

Web de Intel (7 de Abril de 2017). Wi-Fi diferentes protocolos y velocidades de datos. Recuperado de: https://www.intel.es/content/www/es/es/support/network-and-i-o/wireless-networking/000005725.html

10. Anexos

10.1. Anexo 1. Blog de vídeos de tipo tutorial realizados por el alumno durante el desarrollo del presente TFG.

Para mayor información consultar el siguiente enlace:
https://telemetricblog.wordpress.com/