



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

CAMPUS D'ALCOI

Control de la capacidad de una biblioteca pública en tiempo real.

MEMORIA PRESENTADA POR:

Luis Alberto Ehate Belope

Tutor: Pau Micó

GRADO EN INGENIERÍA INFORMÁTICA

Convocatoria de defensa: julio de 2019

Resumen

Para un usuario habitual de la Biblioteca de la EPSA, el acudir y encontrar un sitio libre es una tarea normal que se puede complicar dependiendo de la temporada académica. El problema se plantea cuando se aproximan los actos de evaluación, donde la biblioteca se satura con lo que su capacidad se convierte en un recurso limitado. Mediante el presente TFG se pretende aportar una posible solución al problema de saturación de la biblioteca anteriormente planteado. La propuesta pasa por que el usuario, en tiempo real y antes de acudir a la biblioteca, conozca su nivel de ocupación.

Palabras clave: Arduino, sensores, WiFi, conteo, tiempo real, biblioteca

Abstract

For a regular user of the EPSA Library, going and finding a free seat is a normal task that can become complicated depending on the academical season. The problem arises when the acts of evaluation will be taking place soon, which means that the library is saturated and that its capacity might become a limited resource. Through the present TFG is intended to provide a possible solution to the problem of saturation of the library previously raised. The proposal is that the user, in real time and before going to the library, will be able know its level of occupation.

Keywords: Arduino, sensors, WiFi, counting, real-time, library

Contenidos

1 Introducción

1.1 Antecedentes

1.2 Objetivos

1.3 Requerimientos

2 Anteproyecto

2.1 Estado del Arte

2.1.1 Dos maneras de contar

2.1.1.1 De manera voluntaria

2.1.1.2 De manera involuntaria (o anónima)

2.1.2 Sensores vs rastreo por dispositivo

2.1.3 Sensores

2.1.3.1 Análisis por video en 3D estéreo

2.1.3.2 Análisis por video en 2D monocular y fisheye video

2.1.3.3 Termografía

2.1.3.4 Haces infrarrojos

2.1.3.5 Tiempo de vuelo

2.1.3.6 Luz estructurada

2.1.3.7 Module V2 Camera

2.1.3.8 Sensores para asientos

2.1.3.9 Ultrasonidos

2.1.4 Rastreo personal basado en dispositivo

2.1.4.1 GPS

2.1.4.2 WiFi

2.1.4.3 BLE (Bluetooth Low Energy) Beacons

2.1.4.4 Ultra Wide Band

2.1.5 Otras tecnologías

2.1.5.1 En la Biblioteca de la EPSA

2.2 Estudio de propuestas

2.2.1 Propuesta 1

2.2.2 Propuesta 2

2.3 Justificación

2.3.1 Estimación de recursos

2.3.2 Impacto económico

2.3.2.1 Propuesta 1

2.3.2.2 Propuesta 2

2.3.2.3 Consumo estimado

2.3.3 Propuesta final

3 Implementación

3.1 Entorno de desarrollo

[3.2 Implementación práctica](#)

[3.2.1 Diseño del producto](#)

[3.2.2 Maquetas](#)

[3.2.3 Diagramas de funcionamiento](#)

[3.3. Pruebas](#)

[3.3.1 Del sistema](#)

[3.3.2 De integración de sistemas](#)

[4 Resultados](#)

[4.1 Migración al entorno de producción](#)

[4.2 Manual de uso](#)

[4.3 Estadísticas de explotación](#)

[4.3.1 Estudio comparativo](#)

[5 Conclusiones](#)

[5.1 Conclusiones personales](#)

[5.2 Futuras líneas de desarrollo](#)

[6 Bibliografía](#)

[7 Acrónimos](#)

[8 Anexos](#)

1 Introducción

En este documento se expondrá el procedimiento empleado y el resultado obtenido de la implementación de un sistema sencillo, capaz de contabilizar los accesos de usuarios en la Biblioteca de la EPSA y de proporcionarle al usuario mediante la combinación de tecnologías diversas los datos en tiempo real de su índice de ocupación.

1.1 Antecedentes

En los dos últimos cursos como estudiante de Ingeniería, y siendo muy asiduo a la biblioteca de la escuela, he constatado que, aparte de poder intuir que el hecho de tener la biblioteca altamente ocupada fuera indicativo de que se aproximaban evaluaciones en alguna de las titulaciones, los compañeros no disponían de una herramienta que les pudiera facilitar dicha información antes de acudir.

Es así como surge la idea de aportar por medio de este TFG a los compañeros de la Universidad una herramienta que les fuera de utilidad para averiguar el grado de ocupación de la biblioteca, y que del mismo modo, les sirviera de apoyo para la gestión de su tiempo.

1.2 Objetivos

Este proyecto tiene como finalidad diseñar e implementar un sistema heterogéneo y sencillo que contabilice los accesos a la biblioteca de la EPSA y facilite a los usuarios su índice de ocupación. Esta implementación consistirá en una combinación basada en:

- un *microcontrolador Arduino*, que, por medio de sensores capte el tráfico de usuarios y transmita dicha información a una base de datos.
- un *servidor de bases de datos*
- y un servidor web.

1.3 Requerimientos

La implementación del TFG plantea el desarrollo de una parte hardware (para el conteo de los usuarios) y una parte software (para la consulta de los usuarios). En base a esto, los requerimientos generales del proyecto son:

- dispositivo hardware de conteo de usuarios en tiempo real
- base de datos en la nube que permite guardar la información registrada por el hardware de conteo
- red o sistema de conexión entre el hardware y la base de datos
- aplicación web conectada a la nube que:
 - muestre la disponibilidad de los puestos de estudio en las dependencias de la biblioteca
 - facilite un histórico del índice de ocupación de la dependencia en cuestión (porcentajes de ocupación a lo largo de la temporada académica).

2 Anteproyecto

2.1 Estado del Arte

El conteo de personas es una analítica en cuya aplicación se hace uso de nuevas y de viejas tecnologías, o, la combinación de ambas.

Se utiliza en la actualidad para cuantificar el rendimiento de una ubicación, la optimización de las superficies y para la evaluación del comportamiento de las personas [1].

Dichas tecnologías se pueden agrupar en tres bloques:

- Sensores: (ej. cámaras de vigilancia).
- Rastreo basado en dispositivo. (ej. señal GPS de un smartphone).
- Otras tecnologías y tecnologías avanzadas: (ej. Reconocimiento facial).

La **complejidad** de cada solución a implementar siempre radicará en la **ubicación** de los objetos (interiores o exteriores), el **estado** de los objetos (en movimiento o estático), y de los **atributos** del objeto (tales como conteo por dispositivo o reconocimiento facial).

2.1.1 Dos maneras de contar

En general las acciones que derivan en contar personas solían estar clasificadas en dos subgrupos - una voluntaria, y, otra anónima - para los que presentaremos algunos ejemplos descriptivos:

2.1.1.1 De manera voluntaria

En la que se precisa alguna acción por parte del usuario. Dicha acción generalmente solía suponer un acto de identificación y/o solía contar con el consentimiento previo de la persona implicada.

Normalmente dichos actos solían consistir entre otros, en:

- Empleados accediendo a su puesto de trabajo o moviéndose de un departamento a otro (pasando su tarjeta **contactless** por el lector **RFID**, tecleando un pin de acceso o pasando su tarjeta por el lector de la banda magnética).
- Asistentes conciertos o grandes eventos: escáner de sus *entradas* o *pulseras* con *código QR* o código de barras llevado a cabo por el *puesto de control*.
- Accesos a estadios de fútbol: en los que además de presentar el pase, que será escaneado, hay que atravesar un tornio.

2.1.1.2 De manera involuntaria (o anónima)

En la que no se precisa ninguna acción por parte del usuario. Normalmente solían estar basados en sensores, y, se solían referir a la detección, reconocimiento y la predicción de comportamiento y la densidad de ocupación de los objetos dentro de su campo de visión (por ejemplo: entrar o salir por una puerta).

Dentro de este subgrupo se encuentran las soluciones basadas en procesamiento de imágenes, localización por la identidad inalámbrica de un dispositivo o por su señal GPS, el atravesado de los haces infrarrojos, etc.

2.1.2 Sensores vs rastreo por dispositivo

Ambas soluciones solían ser, por lo general, complementarias, ya que cada tecnología tiene sus ventajas y desventajas:

Con los **sensores**, nos interesa la precisión en el rastreo de todos los objetos que se encuentren dentro del campo de visión o alcance. Las soluciones basadas en sensores solían ser preferibles para cuantificar el *tráfico peatonal*.

Las **tecnologías inalámbricas**, son por el contrario distintas debido a su grado de precisión dependiente de la posición capturada de su posición real. Esto quiere decir que lo que se rastrea es el comportamiento de un dispositivo individual.

2.1.3 Sensores

Según la RAE, un sensor es un dispositivo que detecta una determinada acción externa, temperatura, presión, etc., y la transmite adecuadamente.

Lo que caracteriza los datos capturados por un sensor, es que él mismo es quien los proporciona. El objetivo principal de una solución basada en sensores es el de proporcionar una analítica en un punto determinado (en una dependencia). Los datos recogidos por el sensor serán enviados al servidor.

2.1.3.1 Análisis por video en 3D estéreo

Estos sensores ópticos están diseñados para capturar los objetos que se encuentran dentro de su campo de visión. Incluyen una cámara de alta resolución y un procesador para la captura del objeto en tres dimensiones.

Esta arquitectura proporciona precisión para situaciones de alta densidad de tráfico peatonal (aeropuertos y centros comerciales principalmente), gestión de colas y otros comportamientos complejos. Hay empresas que ofrecen soluciones en este ámbito como Eurecam y Xovis. El sensor *PC2 3D* de Xovis está valorado sobre los 895,95USD



Figura 1. Sensor Xovis PC2 3D

2.1.3.2 Análisis por video en 2D monocular y fisheye video

Los sensores monoculares capturan video a través de una cámara de una única lente. Una lente *fisheye* (ojo de pez), es una lente de ángulo amplio para grandes imágenes panorámicas.



Figura 2. Cámara fisheye



Figura 3. Visión cámara fisheye

El sensor procesa la imagen y cuenta los objetos que se mueven dentro de su campo de visión. En interiores, estos dispositivos pueden conseguir un 90% de precisión en el 90% de los establecimientos en los que son empleados.

El uso de estas cámaras y de los sensores monoculares está ampliamente extendido en el ámbito de la seguridad. Un proveedor muy destacado en la categoría es Panasonic, y el precio de su cámara WV-X451L está fijado en los **2.000,00€** sin IVA.

2.1.3.3 Termografía

Esta técnica detecta emisiones de calor de objetos en movimiento. Al no ser sensible a la luz, puede funcionar en cualquier espacio físico. El reto radica en aislar las huellas de calor entre personas y entorno.

Un proveedor de soluciones integrales destacado dentro del sector es **Irisys**, que aparte de optimizar superficies comerciales, desarrollan también productos para aplicaciones militares.

En la mayoría de los casos, se obtienen tasas del 95% de precisión, y son fáciles de instalar y de calibrar.



Figura 4. Gazelle 2 Dualview de Irisys



Figura 5. Imagen térmica con cámara Flir.

2.1.3.4 Haces infrarrojos

Los haces infrarrojos emiten una señal al ser atravesados. La ventaja está en su bajo coste y su simplicidad. La precisión para esta técnica supone un desafío.

Estos sensores no pueden reconocer la dirección en la que desplaza objeto o la persona. También tienen problemas a la hora de diferenciar si se trata de una o más personas. Además, solía contar de más o de menos, lo que amenaza la consistencia de los datos, motivo por el cual no se solían utilizar en análisis de datos profesionales.

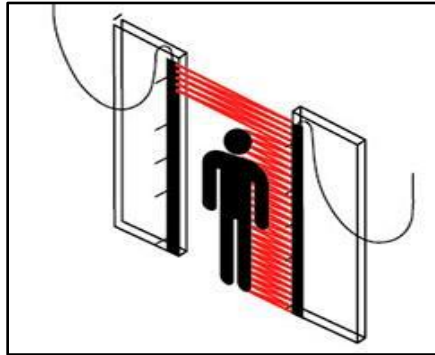


Figura 7. Barrera multihaz de infrarrojos

2.1.3.5 Tiempo de vuelo

El tiempo de vuelo utiliza la velocidad de la luz y produce una imagen en tridimensional aprovechando las características tridimensionales de los pasajeros.

Los sensores que hacen uso de esta tecnología, miden la distancia entre el sensor y el objeto detectado. Se producen cientos de puntos de imágenes para escanear el área de la puerta y producir imágenes 3D en tiempo real, lo que permitirá distinguir pasajeros (y la dirección de su movimiento) de objetos (mochilas, carritos, etc).

IRMA MATRIX [9] es una marca comercial de un sensor de Iris-GmbH utilizado para contabilizar por ejemplo, los pasajeros de un autobús de transporte público urbano.



Figura 8: Sensor IRMA MATRIX autobus

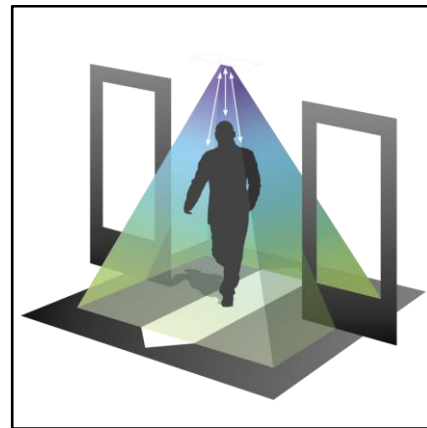


Figura 9: Sensor IRMA MATRIX puerta

2.1.3.6 Luz estructurada

La luz estructurada proyecta un patrón conocido en una escena. Un vector de luces incide sobre la superficie y calcula la profundidad y superficie de los objetos. Para ello se requiere un **escaneo estructurado de la luz en 3D**. Esta tecnología es utilizada por *Apple* y por *Amazon*.

2.1.3.7 Module V2 Camera

Es un sensor óptico de bajo coste para *Raspberry Pi*, para el cual hay un middleware desarrollado por *Enliteon* que implementa un paquete que soluciona el problema de conteo de personas [7]. Se puede obtener por **28,00EUR**.

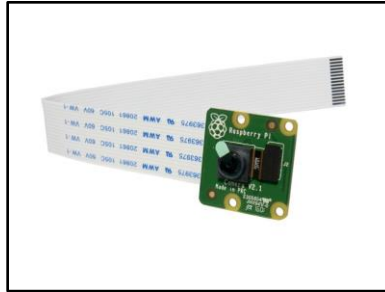


Figura 10. Module V2 Camera

2.1.3.8 Sensores para asientos

Son detectores de movimiento adherentes que funcionan con batería. Se conectan debajo de los asientos y transmiten una señal – normalmente por medio de una red de Internet de 4G - a un dispositivo cercano cada vez que detectan un movimiento.

- **Condeco** ofrece soluciones integrales basadas en **SaaS**. [4][10]
- Se trata de una solución económica, con el inconveniente de tener que sufrir una larga interrupción cuando hayan de sustituirse las baterías en instalaciones con un número elevado de dispositivos.



Figura 11. Sensores Condeco para asientos.

2.1.3.9 Ultrasonidos

Los sensores de ultrasonidos captan y hacen rebotar el sonido inaudible de las personas a medida que las mismas lo atraviesan. Solían contar con un emisor y un receptor.

Se puede obtener el *HC-SR04* [19] por un precio inferior a los **3€**, pero cuando se trate de soluciones profesionales como la **Toposens Alpha**[11], el kit alcanza un coste de **4.980,00€**



Figura 12: Sensor ultrasónico HC-SR04

2.1.4 Rastreo personal basado en dispositivo

Las soluciones de rastreo basadas en dispositivo se aplican asumiendo que el dispositivo representa el comportamiento de un individuo.

- Esta es la diferencia entre los sensores y los dispositivos. Los sensores captan información dentro de su alcance o campo de visión.
- La captura de señal inalámbrica de los dispositivos proporciona datos del rastro de una persona.



Figura 13: Rastreo por dispositivo

2.1.4.1 GPS

GPS es una red de satélites en órbita. El rastreo por GPS está embebido en las plataformas de **Apple** y **Android**. Entre los años 2015-2017 crecieron un 900% las búsquedas basadas en **Near Me (Google)**. Actualmente nuestros dispositivos móviles llevan integrado un GPS que puede reportar nuestra posición actual en un momento dado siempre que tengamos activada la utilidad que reporte nuestra ubicación.

2.1.4.2 WiFi

WiFi es una tecnología que sirve para interconectar de manera inalámbrica dispositivos electrónicos. Cada tarjeta o dispositivo de red cuenta con un identificador único de 48 bits (dirección MAC).

Si tenemos activa nuestra señal Wi-Fi, del mismo modo que podemos captar la señal de una red a nuestro alcance, la señal de nuestro dispositivo también puede ser captada dentro de un área determinada. Dicho esto, y, asumiendo que cada usuario es representado por un dispositivo, se podría determinar el número de personas que ocupan un espacio en base al número de direcciones MAC detectadas dentro de un rango de alcance.

Cisco Meraki ofrece soluciones profesionales basadas en rastreo Wi-Fi en el campo de la optimización de superficies comerciales[14].

2.1.4.3 BLE (Bluetooth Low Energy) Beacons

Bluetooth Low Energy (BLE) **Beacons** son dispositivos inalámbricos que periódicamente difunden información que, son recibidos por un dispositivo inteligente, y que, son utilizados para determinar la posición al respecto del mismo "beacon".[6].

Dichos dispositivos son ideales para una comunicación segura entre clientes. Es la tecnología preferida para el conteo de clientes en el interior de las tiendas.

Se enfrentan a dos grandes desafíos:

- En primer lugar, los clientes tienen que disponer de la aplicación del vendedor o escanear un código QR.
- En segundo lugar, las tecnologías WiFi y GPS están experimentando grandes mejoras en su uso en exteriores.

El ecosistema de los *Beacons* incluye: el **iBeacon de Apple** y **Eddystone de Google**. El precio del *iBeacon* ronda los **23 USD** de media.

2.1.4.4 Ultra-Wide Band

Se trata de una tecnología inalámbrica que, al igual que Wi-Fi, es capaz de detectar las señales de los diferentes dispositivos a su alcance, pero con la mejora de medir la distancia exacta a la que se encuentran los dispositivos con una precisión de 5-10 cm. Estos resultados se deben a la baja energía - al respecto de Wi-Fi y Bluetooth - que consume y las propiedades físicas de onda que emite [5].

Es utilizada en soluciones empresariales que pretenden ejercer un mayor control en productividad de sus empleados, facilitando datos precisos acerca de la ausencia en sus puestos de trabajo y la duración de la misma. Todo ello bajo la premisa de un dispositivo, un usuario.

Se augura una mayor implantación de ésta tecnología en pequeños dispositivos y en dispositivos inteligentes.

2.1.5 Otras tecnologías

2.1.5.1 En la Biblioteca de la EPSA

En la biblioteca de la Escuela Politécnica Superior de Alcoy de la Universitat Politècnica de València, hay implementado un sistema con paneles **EAS** que realiza dos funciones principales:

- Rastrear etiquetas RFID del material de la biblioteca.
- Contar los accesos al recinto.

Dicho sistema está implementado con tres paneles *ANT1520/680-B* [12][13] de *Feig Electronic GmbH* que utilizan, apoyados en un radar, varias antenas - en ambos sentidos- para clasificar el tráfico entrante y/o saliente, y, etiquetas RFID para libros y discos.

El software de monitorización se obtiene adquiriendo licencias para disponer de más o menos funcionalidades según lo requiera el comprador.

2.2 Estudio de propuestas

2.2.1 Propuesta 1

En primer lugar, damos por configurados los servidores - Apache y MySQL en un PC o en una Raspberry Pi - y la interconexión establecida entre componentes (arduino y servidores en la misma red).

Hemos programado nuestro sketch utilizando los sensores de presencia GH-718C, que aunque no tuvieran la precisión que se necesita para la implementación final, han resultado de gran utilidad para realizar pruebas.

Damos por configurados los servidores web y MySQL para la recepción y presentación de los datos:

1. Escribir un sketch que anotara los eventos a la base de datos usando la librería MySQL Connector/Arduino, desarrollada por el Dr. Charles Bell[16], para lo cual sería necesario adquirir un WiFi Shield[18] o un Ethernet Shield que es para lo que está realmente diseñada la librería.
2. Ello aumentaría considerablemente el número de conexiones contra la base de datos, ya que, cada vez que se accediera/abandonara la estancia, se abriría una conexión a la base de datos para ser anotar el dato (abrir conexión + realizar una operación de **inserción** en la base de datos + cerrar conexión).
3. El mayor inconveniente a la hora de tomar la decisión de no hacer uso del sensor GH-718C[20] ha sido que, aunque podamos limitar físicamente su ángulo de detección, el sensor tiene pulsos de 4 a 8 segundos, y un alcance de hasta 7m, motivos por los cuales se reduciría la precisión de nuestro cometido.

2.2.2 Propuesta 2

Dando del mismo modo por configurados los servidores web y de bases de datos- que se llevará a cabo en un PC o una Raspberry Pi-, y también, la interconexión de componentes (arduino y servidores dentro de la misma red).

1. Aprovecharemos directamente las capacidades de conectividad inalámbrica de nuestro Arduino Uno WiFi Rev.3 [17].
2. Cargaremos en el sketch un **cliente rest** que enviará peticiones HTTP al servidor, que nos servirá para enviar el dato del evento capturado por los sensores. Para ello se aprovecha un ejemplo disponible sólo para la librería “*Arduino Uno WiFi Dev Ed Library*” (*Rest Client*).
3. Los datos no se almacenarán directamente sobre la base de datos, esto se hará a través de un servicio API Rest que comprobará que los datos recibidos sean correctos antes de almacenarlos en la BBDD.

2.3 Justificación

2.3.1 Estimación de recursos

Para desarrollar nuestras propuestas, los recursos necesarios son:

A nivel de hardware:

- Un PC o una Raspberry Pi.
- Arduino Uno WiFi Rev3.
- Cables para las conexiones de los sensores.
- Dos sensores HC-SR04.
- Un router inalámbrico.

A nivel software:

- Un IDE de desarrollo o cualquier editor de texto.
- El IDE de arduino para desarrollar el sketch y cargarlo en el microcontrolador (imprescindible).
- El paquete XAMPP[22] para la instalación de los servidores Apache y MySQL.
- Navegador web.

2.3.2 Impacto económico

El software necesario para construir el proyecto se puede obtener de manera gratuita en las páginas web de sus proveedores, estos son: el entorno XAMPP, el IDE de Arduino, un editor de texto, o en su caso un entorno de desarrollo web y un navegador web.

El coste de los materiales hardware de cada una de las diferentes propuestas se detalla a continuación:

2.3.2.1 Propuesta 1

Artículo	Cantidad	Precio Ud. (€)	Subtotal (€)
Arduino Uno WiFi R3	1	39	39
Sensor HC-SR04	2	4	8
Raspberry Pi 3B	1	38.82	38.82
Router Inalámbrico	1	25	25
10 jumpers (cables)	1	3.4	3.4
Total			114.22

2.3.2.2 Propuesta 2

Artículo	Cantidad	Precio Ud. (€)	Subtotal (€)
Arduino Uno WiFi R3	1	39	39
Sensor GH-718C	2	5.4	10.8
Raspberry Pi 3B	1	38.82	38.82
Router Inalámbrico	1	25	25
10 jumpers (cables)	1	3.4	3.4
Total			117.02

2.3.2.3 Consumo estimado

El consumo energético anual estimado de nuestro circuito, suponiendo que estarán alojados nuestro servidor y base de datos en la nube, con un precio de 0,11€ kWh (precio medio 25/06/2019 a las 01:15) , y según la hoja de especificaciones de nuestro Arduino Uno WiFi[15] que cuenta con un consumo energético de **93 mA**, será de:

- 0.93mA donde, según la ley de Ohm $P = V \times I$; y alimentando el Arduino con 9V, entonces $P = 9V \times 0.093A = 0.837W$
- Potencia consumida: 0.000837kW
- Horas al mes(jornadas de 12h) = 12h al día x 20 días = 240 horas al mes
- Energía consumida al mes = 0.000837kW x 240h = 0.20088kWh x 0.11€ = 0,2220968€
- Al año, 0.20088kWh x 12 = 2.41056€

2.3.3 Propuesta final

Estudiadas ya las propuestas, y vistos los resultados que podemos obtener de ellas, se decide implementar la propuesta 2.

Con esta propuesta se aprovecharán mejor los recursos de los que disponemos sin la necesidad de añadir costes adicionales en material hardware como ya indicamos anteriormente (shields adicionales de comunicaciones, etc),y en la parte software, conseguimos implementar un sistema más desacoplado, explotando el modelo cliente-servidor por medio de una API rest para las peticiones HTTP sobre el servidor con sus operaciones básicas (GET, POST, PUT, DELETE). Y a modo de finalización, nuestro sistema derivará en una aplicación web a través de la cual consultamos los datos, generados por los eventos registrados por nuestros sensores, almacenados en nuestra BBDD.

3 Implementación

3.1 Entorno de desarrollo

Del sitio web oficial de Arduino, descargamos la última versión de su IDE [21] escogiendo la opción correcta para nuestro sistema operativo. Una vez descargado e instalado, podemos editar y cargar nuestro sketch en nuestro Arduino. [2]

A la hora de instalar el entorno XAMPP, seleccionamos únicamente los componentes necesarios para desarrollar nuestro proyecto: servidor web Apache, servidor de bases de datos MySQL (actualmente *MariaDB*) y el intérprete del lenguaje PHP.

En nuestro caso, nos hemos decantado por el editor *Visual Studio Code* [23] al ser gratuito por la posibilidad por tener la opción de añadir el complemento *xdebugger* para poder depurar y buscar errores en el código escrito en PHP[26].

3.2 Implementación práctica

La naturaleza del problema que nos hemos planteado resolver requiere, por sus características, de un sistema implementado con dos sensores como mínimo. La razón de este requerimiento se debe a que para acceder o abandonar la biblioteca deberá ser atravesada la única puerta habilitada para tal efecto.

3.2.1 Diseño del producto

Se crea una base de datos relacional "*arduino_db*", y dentro de ella la tabla que se utilizará, "*arduino_table*" que consta de tres columnas que consistirán en: **identificador** de tipo entero autoincremental, **acceso** de tipo carácter contenido de la petición HTTP, y por último la **fecha**, la cuál servirá para anotar el tiempo de anotación de cada registro.

3.2.2 Maquetas

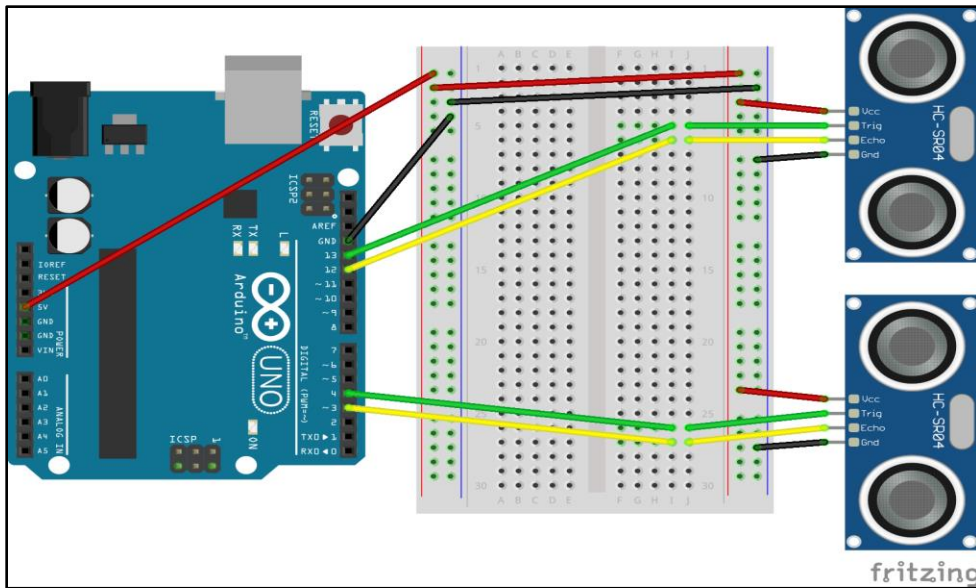
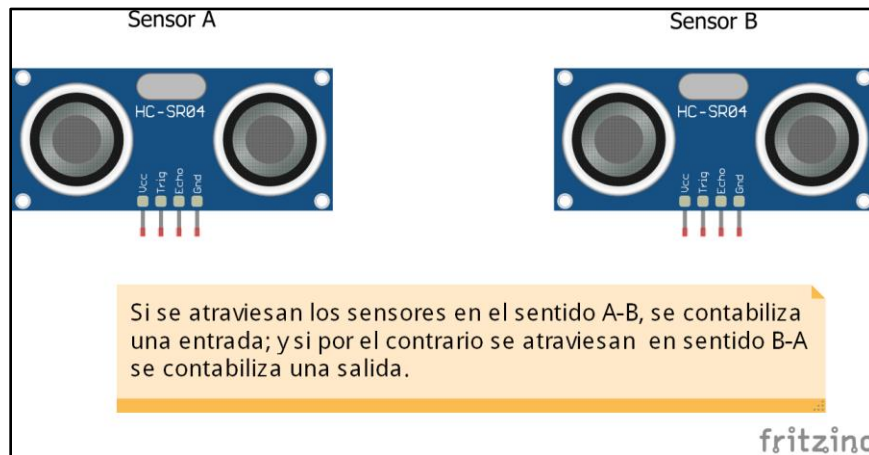
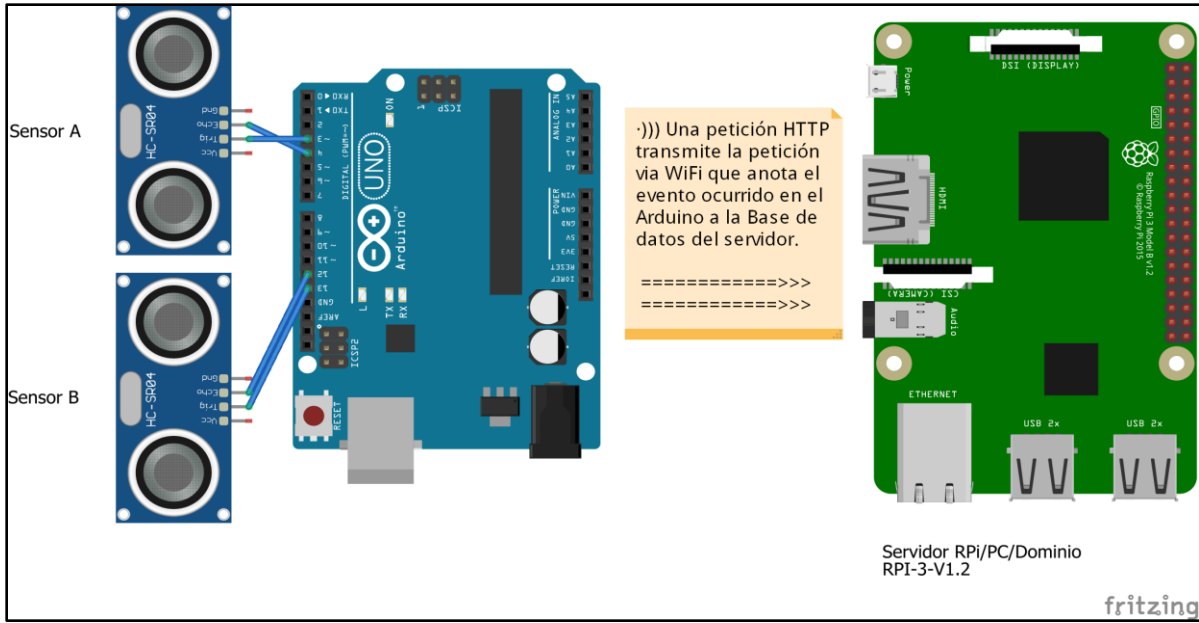


Figura 14: Maqueta de la propuesta 2

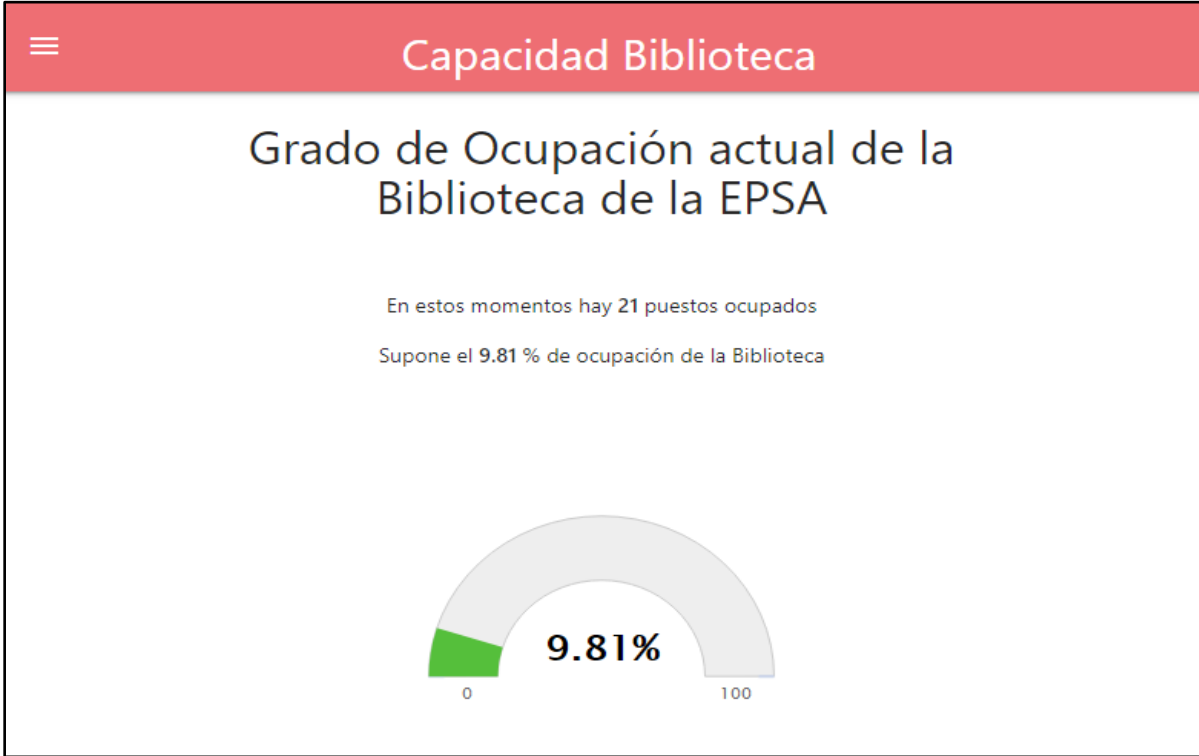
3.2.3 Diagramas de funcionamiento



Paso 1: Los sensores son atravesados en un sentido u otro.



Paso 2: La interacción se registra con por medio de un Cliente Rest al servidor de BBDD.



Paso 3: Un usuario que accede a la aplicación web que ejecuta consultas contra la BBDD.

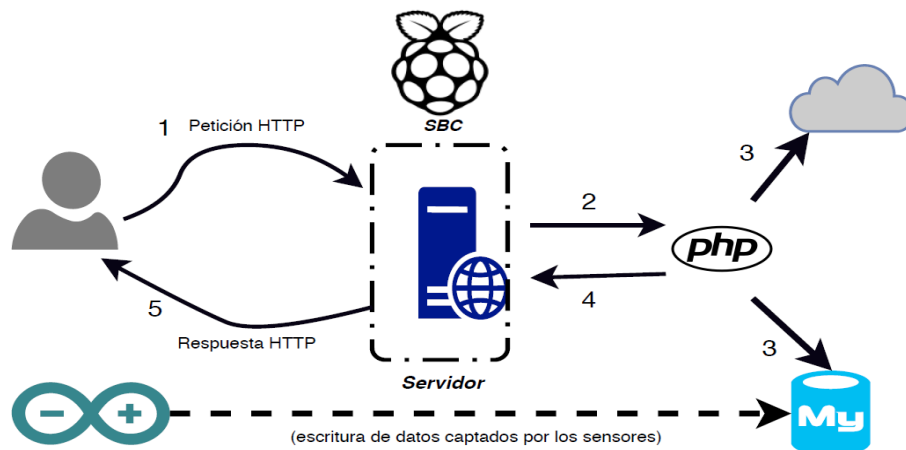


Figura 15. Esquema de una petición HTTP.

3.3. Pruebas

3.3.1 Del sistema

Tras cargar el sketch con la URL del servidor de destino - previa conexión de Arduino a la red correspondiente - se comprueba que los datos recogidos por los sensores del Arduino se anotan correctamente en el servidor: una “e” cuando se atraviesan los sensores en un sentido, y, una “s” cuando se atraviesan los sensores en sentido contrario.

	id	acceso	fecha
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	e	2019-06-18 21:32:49
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	s	2019-06-18 21:41:01
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	s	2019-06-18 23:01:28
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	e	2019-06-18 23:02:11

Figura 16: anotaciones en la base de datos

3.3.2 De integración de sistemas

Probamos que nuestro sistema es capaz de:

- Anotar las interacciones recogidas por los sensores en un servidor local o remoto.
- Diferenciar las interacciones captadas por los sensores en un sentido u otro.
- Sólo se permite realizar anotaciones en la base de datos a la asociación máquina/usuario con permisos para ello.
- No aceptar datos con un formato diferente al esperado {"e", "s"}.

4 Resultados

4.1 Migración al entorno de producción

En caso de tener que migrar al entorno de producción definitivo, se optaría por reemplazar la dirección IP actual que apunta a la Raspberry, por otra IP o dominio que apuntaría al servidor de producción.

Optaríamos también por darle un aspecto más discreto a nuestro PCB y los sensores metiéndolos en una cajita con aberturas para el correcto funcionamiento de los sensores. También sería necesario cambiar los sensores utilizados en el prototipo por unos de haz infrarrojo ajustables.

4.2 Manual de uso

Para hacer uso de la aplicación, tendremos que escribir en la barra de direcciones la dirección web del dominio, IP o URL en donde se encuentre alojado el archivo de nuestra aplicación.

En nuestro caso la hemos implementado en un servidor local: *localhost/arduino/index.php*

Una vez hayamos accedido, podremos navegar por dos documentos indexados en la barra de navegación superior:

- El principal *index.php* que nos mostrará el grado de ocupación actual de la biblioteca y
- *estadisticas.php* en el que se mostrarán datos de los accesos, porcentaje de ocupación, históricos, etc.

En el anexo aportamos también un pequeño manual para poner en marcha el sistema.

4.3 Estadísticas de explotación

4.3.1 Estudio comparativo

Nuestro prototipo, pese a no haber sido puesto en producción, en comparativa con otros sistemas podemos afirmar que:

- No se trata de un mero contador que nos muestra el aforo dentro de una estancia, guardamos información acerca de la ocupación de la estancia, y al mismo tiempo, ayudamos a los usuarios de la biblioteca y al personal en la toma de decisiones.
- En comparación con otros sistemas, como el de la biblioteca de la EPSA por poner un ejemplo, no precisamos de licencias adicionales, dado que estamos utilizando software libre a la hora de mostrar los análisis de nuestros datos sin costos adicionales.

5 Conclusiones

5.1 Conclusiones personales

La primera propuesta, pese a no haber sido posible de terminar de implementar por necesitar de hardware extra - WiFi shield para poder usar la librería MySQL Arduino/Connector - , había sentado ya las bases para la implementación de la propuesta final. Por una parte, de haber sido posible completarla, estaríamos despreciando las capacidades en conectividad que ya se disponían en el Arduino Uno WiFi, y por otra parte, podría existir la posibilidad de perder anotaciones debido a la latencia de las operaciones de abrir conexión + insertar dato en la tabla + cerrar conexión al depender éstas por completo de Arduino.

En la propuesta 2, se obtiene un prototipo que funciona como se espera, se obtiene la información generada de forma anónima de los usuarios consistente en el **qué**, el **cuándo** y **dónde**. Para el caso concreto de la biblioteca de la EPSA, se consigue con nuestro prototipo prescindir de la adquisición de licencias adicionales para poder realizar consultas más precisas acerca del tráfico peatonal registrado en la misma (accesos, salidas, evolución del grado de ocupación, etc).

Este TFG me ha permitido poner en práctica muchos de los conocimientos adquiridos en la carrera, siguiendo unas guías correctas de diseño, queriendo destacar entre ellas la del tratamiento de los datos.

5.2 Futuras líneas de desarrollo

En este apartado se mencionan varias posibilidades de mejora sobre la propuesta implementada:

- Modificar el sistema para que no se pierdan las cuentas en caso de pérdida de conexión con el servidor.
- Enviar un aviso al responsable de la biblioteca en caso de superarse un cierto grado de ocupación fijado previamente.
- Otra de las posibilidades contempladas como mejora del proyecto consistiría en poder analizar de manera más precisa los datos de ocupación consistentes en el **quién**, **cuándo**, **durante cuánto tiempo** ocupa las instalaciones de la biblioteca, permitiéndonos determinar qué rango de edades utiliza más la biblioteca, qué sexo utiliza más la biblioteca o cuál es la titulación con más alumnos utilizando la biblioteca por temporadas, etc.

6 Bibliografía

- [1] Max, Ronny – *Behaviour Analytics in Retail: Measure, Monitor and Predict Employee and Customer Activities to Optimize Store Operations and Profitably, and Enhance*. ISBN-13: 978-1491806272
- [2] Jody Culking y Eric Hagan: *Aprende electrónica con Arduino: Una guía ilustrada para principiantes sobre la Informática Física*. Ed. Marcombo, cop. 2019. ISBN 978-84-267-2659-9
- [3] Max, Ronny. (2019). “Technologies of People Tracking Behavior Analytics Retail” [online](#)
- [4] Farah, Andrew. (2018). “7 Technologies that count people” por Andrew Farah” [online](#)
- [5] Cornnell, Ciaran. (2015) “What’s The Difference Between Measuring Location by UWB, WiFi, and Bluetooth?” [online](#)
- [6] iBeaconTechnology 101: “What is iBeacon, how does it work, transmission range, iBeacon cost (and more)”. (2013) [online](#)
- [7] Scavix Software GmbH & Co. KG(2014). *Raspberry Pi as A Low-cost HD Surveillance Camera* (2014) [online](#)
- [8] Bosch Security - *Essential Video Analytics - People counting* (2017)(video) [online](#)
- [9] *Hoja características técnicas sensor IRMA MATRIX* (2016) [online](#)
- [10] *Condeco Workspace Occupancy Sensors* (2016) [online](#)
- [11] *Sensor de ultrasonidos -Toposens Alpha*. (2018) [online](#)
- [12] *Manual del ANT1520/680-B Crystal Gate* (2017) [online](#)
- [13] *Manual del ANT1710/690 Crystal Gate* (2017) [online](#)
- [14] *Análisis basados en WiFi - Cisco Meraki* (2019) [online](#)
- [15] *Arduino Uno WiFi Rev.3* [online](#)
- [16] *MySQL Connector/Arduino por el Dr. Charles Bell* [online](#)
- [17] *Microchip ESP8266EX* [online](#) [online](#)
- [18] *Arduino WiFi Shield* (2019) [online](#)
- [19] *Sensor de Distancia HC-SR04 | Proyectos con Arduino*(2019) [online](#)
- [20] *Sensor detector de movimiento GH-718C* [online](#)
- [21] *Arduino IDE* (2019) [online](#)
- [22] *Entorno XAMPP* (2019) [online](#)
- [23] *Editor Visual Studio Code* (2019) [online](#)
- [24] *Fritzing. Programa de diseño de prototipos electrónicos* (2016) [online](#)
- [25] *Framework Materialize para el diseño web* [online](#)
- [26] *Lenguaje PHP* [online](#)
- [27] *Lenguaje HTML* [online](#)
- [28] *Gráficos interactivos en JavaScript con HIGHCHARTS* (2019) [online](#)
- [29] *Stack Overflow* [online](#)
- [30] *Wikipedia* [online](#)

7 Acrónimos

- API: *Application Programming Interface*(interfaz de programación de aplicaciones).
- AR: Augmented Reality (realidad aumentada).
- CONTACTLESS: Tecnología que funciona por proximidad (sin contacto).
- EAS: Electronic Article Surveillance (vigilancia electrónica de artículos).
- GPL: General Public License.
- GPS: Global Positioning System (sistema de posicionamiento global).
- IDE: Integrated Development Environment (entorno integrado de desarrollo).
- IoT: Internet of Things (Internet de las cosas).
- Middleware: software que proporciona servicios al margen del S.O.
- PCB: Printed Circuit Board (placa de circuito impreso).
- RFID: Radio Frequency Identification (Identificación por radiofrecuencia).
- SBC: Single Board Computer (ordenador de placa reducida).
- SaaS: Software as a Service (software como servicio).
- Sketch: nombre que usa Arduino para definir los programas desarrollados por su IDE.
- Socket: asociación de IP y puerto.
- S.O.: sistema operativo.
- SSID: Service Set Identifier (nombre de la red WiFi).
- USB: Universal Serial Bus(bus de serie universal).
- RAE: Real Academia Española de la Lengua.
- RFID: Radio Frequency Identification (identificación por radiofrecuencia).

8 Anexos

- Anexo I: plano de acceso a la biblioteca.
- Anexo II: manual de puesta en marcha del prototipo.
- Anexo III: porciones del código que se sube al microcontrolador.
- Anexo IV: directorio de ficheros de la aplicación web.