



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES

TEMA:

**RED INALÁMBRICA PARA GEOLOCALIZACIÓN EN ESPACIOS
CERRADOS APLICADA AL MARKETING DIGITAL EN CENTROS
COMERCIALES**

Trabajo de graduación. Modalidad: Proyecto de Investigación, presentado previo a la obtención del título de Ingeniería en Electrónica y Comunicaciones.

LÍNEA DE INVESTIGACIÓN: Tecnologías de la Información y Comunicación.

AUTOR: Diego Antonio Villegas Chasiluisa

TUTOR: Ing. Mg. Víctor Santiago Manzano Villafuerte

Ambato – Ecuador

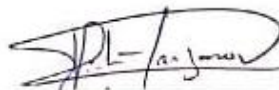
Enero 2020

APROBACIÓN DEL TUTOR

En mi calidad de tutor del Trabajo de Investigación sobre el tema: “RED INALAMBRICA PARA GEOLOCALIZACIÓN EN ESPACIOS CERRADOS APLICADA AL MARKETING DIGITAL EN CENTROS COMERCIALES”, del señor Diego Antonio Villegas Chasiluisa, estudiante de la carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, Enero de 2020

EL TUTOR



Ing. Mg. Santiago Manzano.

AUTORÍA

El presente Proyecto de Investigación titulado “RED INALAMBRICA PARA GEOLOCALIZACIÓN EN ESPACIOS CERRADOS APLICADA AL MARKETING DIGITAL EN CENTROS COMERCIALES”, es absolutamente original, autentico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Enero de 2020



Diego Antonio Villegas Chasiluisa

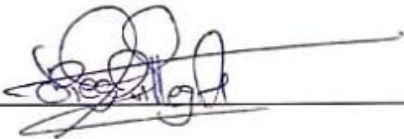
CC.: 0503239618

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, Enero de 2020



Diego Antonio Villegas Chasiluisa

CC.: 0503239618

APROBACIÓN DEL TRIBUNAL DE GRADO

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Santiago Altamirano e Ing. Paulina Ayala, revisó y aprobó el Informe Final del Proyecto de Investigación titulado **“RED INALAMBRICA PARA GEOLOCALIZACIÓN EN ESPACIOS CERRADOS APLICADA AL MARKETING DIGITAL EN CENTROS COMERCIALES”**, presentado por el señor Diego Antonio Villegas Chasiluisa de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.



Ing. Elsa Pilar Urrutia, Mg.

PRESIDENTA DEL TRIBUNAL



Ing. Santiago Altamirano

DOCENTE CALIFICADOR



Ing. Paulina Ayala

DOCENTE CALIFICADOR

DEDICATORIA

A mi madre,

Por ser esa persona que siempre ha estado a mi lado y me ha brindado su apoyo incondicional en todos los ámbitos de mi vida; por su amor, su paciencia y por el inmenso sacrificio que día a día hace para dar lo mejor a sus hijos.

A mi hermana,

Quien siempre confió en mí y ha sido esencial para que esta meta se haya vuelto realidad; por el esfuerzo y sacrificios que ha hecho al apoyarme durante mi formación profesional.

AGRADECIMIENTO

Agradezco principalmente a Dios, por darme la salud, la sabiduría y ha sabido guiar mis pasos para alcanzar esta meta tan importante en mi vida.

A mi núcleo familiar; agradezco infinitamente a cada uno de los miembros de la familia Villalva – Martínez, quienes hicieron de mí un hombre correcto y siempre me han brindado su confianza; sin su apoyo nada de esto habría sido posible.

Agradezco a la Universidad Técnica de Ambato, a la Facultad de Ingeniería en Sistemas Electrónica e Industrial, y a los docentes quienes compartieron sus conocimientos y sabiduría a lo largo de mi formación como profesional, en especial a mi docente tutor el Ingeniero Santiago Manzano.

A mis amigos con quienes compartimos gratos momentos en el transcurso de nuestra formación como profesionales, y a todas las personas que de una u otra forma me extendieron su mano durante este largo caminar.

A todos ustedes mi más profundo agradecimiento.

Diego Antonio Villegas Chasiluisa

ÍNDICE GENERAL

PORTADA.....	i
A. PÁGINAS PRELIMINARES.....	ii
APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR.....	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA.....	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL.....	viii
INDICE DE TABLAS Y FIGURAS	x
RESUMEN	xii
SUMMARY.....	xiii
B. CONTENIDO.....	1
CAPÍTULO I MARCO TEÓRICO.....	1
1.1. Antecedentes investigativos	1
1.1.1. Contextualización del problema.....	1
1.2. Fundamentación teórica	3
1.2.1. Centros comerciales	3
1.2.2. Marketing digital	4
1.2.3. Smartphone	4
1.2.4. Técnicas de geolocalización	5
1.2.5. Bluetooth BLE.....	8
1.2.6. BLE Beacons.....	14
1.2.7. Aplicaciones móviles.	18
1.2.8. Android Studio	19

1.2.9.	BLE en S.O. Android.....	22
1.2.10.	Java.....	23
1.3.	Objetivos.....	23
1.3.1.	Objetivo general.....	23
1.3.2.	Objetivos específicos.....	23
CAPÍTULO II METODOLOGÍA.....		24
2.1.	Materiales.....	24
2.1.1.	Smartphone.....	24
2.1.2.	Bluetooth Beacon IBKS105.....	24
2.1.3.	Android Studio IDE.....	25
2.2.	Métodos.....	26
2.2.1.	Modalidad de la investigación.....	26
2.2.2.	Recolección de información.....	26
2.2.3.	Procesamiento y análisis de datos.....	26
2.3.	Desarrollo del proyecto.....	27
2.3.1.	Requerimientos del sistema.....	27
2.3.2.	Análisis de componentes.....	27
2.3.3.	Implementación de red inalámbrica.....	32
2.3.4.	Configuración de dispositivos Beacon.....	34
2.3.5.	Calculo de distancia usando señales RF.....	37
2.3.6.	Diagrama de flujo de la app.....	40
2.3.7.	Desarrollo de la aplicación móvil.....	41
2.3.8.	Marketing de proximidad.....	48
2.3.9.	Pruebas de funcionamiento.....	49
CAPÍTULO III RESULTADOS Y DISCUSIÓN.....		53
3.1.	Análisis de datos.....	53
CAPÍTULO IV CONCLUSIONES Y RECOMENDACIONES.....		56

4.1. Conclusiones.....	56
4.2. Recomendaciones	57
C. MATERIALES DE REFERENCIA.....	58
Bibliografía	58
ANEXOS	62

INDICE DE TABLAS Y FIGURAS

Figura 1 Principio de trilateración	6
Figura 2 Localización wifi y bluetooth	8
Figura 3 Topología Broadcast	9
Figura 4 Topología de Conexión	10
Figura 5 Protocolo BLE	11
Figura 6 Canales de Radiofrecuencia.....	12
Figura 7 Ventanas de anuncio y escaneo.....	14
Figura 8 Interacción Beacon.....	15
Figura 9 Protocolos Beacon	17
Figura 10 Interfaz de usuario.....	21
Figura 11 Beacon iBKS105.....	25
Figura 12. Versión de Android Studio.	25
Figura 13. Implementación de red.	33
Figura 14. Ubicación de beacon 1.....	33
Figura 15. Ubicación de beacon 2.....	34
Figura 16. Aplicación de configuración	35
Figura 17. Configuración general	35
Figura 18. Configuración Eddystone UID.....	36
Figura 19. Diagrama de flujo para la app	40
Figura 20. Creación del proyecto en Android Studio	41
Figura 21. Estructura del proyecto.....	42
Figura 22. Permisos manifests.....	42
Figura 23. Librería Beacon.....	43

Figura 24. Condiciones para la inicialización de app	43
Figura 25. Interfaz gráfica MainActivity.....	44
Figura 26. Inicialización de BLE	45
Figura 27. Función de escaneo	45
Figura 28. Función de conversión Byte a HEX	45
Figura 29. ScanCallback, función de cálculos	46
Figura 30. Interfaz gráfica de localización	47
Figura 31. Codificación de notificaciones	47
Figura 32. Aplicación de campañas de marketing.....	48
Figura 33. 1ra medición dispositivo 1	49
Figura 34. 2da medición dispositivo 1	49
Figura 35. 1ra medición dispositivo 2	49
Figura 36. . 1ra medición dispositivo 3	50
Figura 37. Despliegue de dialogo de marketing 1	51
Figura 38. Dialogo de marketing 2 desplegada.....	51
Figura 39. Error absoluto	54
Tabla 1 Requisitos del sistema	21
Tabla 2. Características del smartphone	24
Tabla 3 Especificaciones técnicas iBKS105.....	25
Tabla 4 Comparación Accesibilidad, Precisión, Alcance	28
Tabla 5 Comparación: Seguridad, Costo, Consumo de energía	29
Tabla 6 Ventajas y Desventajas de los tipos de aplicaciones móviles	30
Tabla 7. Características físicas beacons	31
Tabla 8. Configuración de dispositivos	37
Tabla 9. RSSI - coeficiente de atenuación.....	39
Tabla 10. Distancias promedio obtenidas.....	50
Tabla 11. Despliegue de publicidad	52
Tabla 12. Errores absoluto y relativo	53
Tabla 13. Despliegue de publicidad	55

RESUMEN

El presente proyecto de investigación describe la implementación de un prototipo de red inalámbrica que conjuntamente con una aplicación móvil permitirá determinar la ubicación de un smartphone dentro de un centro comercial, esta información se utilizará para la optimización de campañas de marketing digital, mostrando al usuario a través de su dispositivo móvil información netamente relacionada a su posicionamiento en el interior de establecimientos comerciales, así se desplegará información publicitaria como anuncios, promociones, precios; o información de asistencia al cliente.

La red interna consiste en un conjunto de transmisores bluetooth estos dispositivos emiten constantemente su identificación la cual es recibida por los teléfonos móviles de los clientes a través de una aplicación de software libre Android, por medio de esta interfaz se identificará los dispositivos bluetooth y se desplegará en pantalla información que será de gran utilidad para los consumidores convirtiendo así a su propio smartphone en un asistente de compras, lo que se conoce hoy en día como marketing de proximidad.

Los transmisores Bluetooth Low Energy (BLE) más conocidos como Beacons tienen un reducido tamaño y su consumo de energía es mínimo por lo que es suficiente usar una batería de litio de 3v tipo moneda la cual tiene una vida útil de 3 años antes de ser reemplazada, estas características permiten ubicar los dispositivos en cualquier lugar ya que no presentan la necesidad de conectarse a la red eléctrica y tampoco necesitan de ningún otro tipo de cableado entre dispositivos ya que es únicamente el smartphone el que interactúa con ellos de forma inalámbrica, para ello el usuario deberá instalar la aplicación en su dispositivo móvil y activar la función bluetooth.

Como resultado se obtiene un sistema eficiente que permite ubicar dispositivos móviles dentro de un establecimiento comercial cerrado, mejorando la experiencia de los clientes al realizar sus compras y consecuentemente contribuyendo a los comerciantes a incrementar sus ganancias.

Palabras clave: Geolocalización, Android, Marketing digital móvil, Centros comerciales.

SUMMARY

This research project describes the implementation of a wireless network prototype which, together with a mobile application, will determine the location of a smartphone inside a shopping center, this information will be used for the optimization of digital marketing campaigns, showing the user through its cellphone information clearly related to its positioning inside commercial establishments, thus will display advertising information such as ads, promotions, prices; or customer support information.

The network prototype consists of a set of bluetooth transmitters, these devices constantly emit their identification which is received by the mobile phones of customers through an Android free application, through this interface the bluetooth devices will be identified and It will display information that will be very useful for consumers, thus turning their own smartphone into a shopping assistant, what is known today as proximity marketing.

The Bluetooth Low Energy (BLE) transmitters or “Beacons” are small and their energy consumption is minimal, so it is sufficient to use a coin-type 3v lithium battery which has a useful life of 3 years before being replaced, these features allow you to locate the devices anywhere since they do not have the need to connect to the electrical network and do not need any other type of wiring between devices because it is only the smartphone that interacts with them wirelessly, for this the user must install the application on their mobile device and activate the bluetooth function.

As a result, an efficient system is obtained that allows mobile devices to be located within a closed commercial establishment, improving the customer experience for making purchases and consequently contributing to merchants to increase their profits.

Keywords: Geolocation, Android, Mobile digital marketing, Shopping centers.

CAPÍTULO I

MARCO TEÓRICO

1.1. Antecedentes investigativos

1.1.1. Contextualización del problema.

Para el desarrollo del presente trabajo se han considerado investigaciones relacionadas con las tecnologías utilizadas para la localización de dispositivos móviles en espacios cerrados, esta información se ha obtenido de diferentes plataformas de investigación, bibliotecas de la Universidad Técnica de Ambato, repositorios digitales de Universidades y por lo tanto se evidencia lo siguiente:

Asle Fagerstrøm, Niklas Eriksson y Valdimar Sigurdsson en su artículo “Investigación del impacto de los servicios de un smartphone y una aplicación IoT en tiendas de comestibles” publicado en Octubre 2018 en Oslo-Noruega, analizan la importancia del Internet de las Cosas en el caso de elección de comestibles. Usando tecnología Bluetooth Beacon para geolocalización en interiores impulsan el contenido personalizado, como fechas de caducidad, promociones, ofertas y descuentos en tiempo real acerca de productos en tiendas. La aplicación móvil cuenta en su interfaz con una función de calificación así los usuarios luego de realizar el análisis del precio en tiempo real y la fecha de caducidad de un producto a través de su smartphone tienen la opción de calificar la utilidad de la misma, obteniéndose un 69,7% de votos favorables. [1]

En el año 2018 se publica un artículo científico “Monitoreo del uso del arnés en la construcción utilizando transmisores Bluetooth” en la ciudad de Málaga realizado por Jesús Gómez, Juan A. Fernández, Antonio López y Juan Carlos Rubio en el cual se utiliza la tecnología Bluetooth de bajo consumo de energía BLE en lugares de construcción, aquí se analiza la última versión de la tecnología bluetooth y sus ventajas como su bajo consumo de energía, tamaño y la posibilidad de enlazar esta tecnología con teléfonos móviles inteligentes. Con este sistema comprueban el uso del arnés de seguridad por parte de los obreros en lugares de construcción, la implementación de este prototipo elevó la tasa de uso del arnés de seguridad hasta en un 95% contribuyendo a la seguridad de los obreros. [2]

En la Universidad de Hamburgo Alemania Wolf Psdorfer y Walid Maalej realizan en abril del 2016 el artículo “Encuestas sensibles al contexto de geolocalización utilizando Bluetooth Beacons”, en este estudio los investigadores desarrollan un sistema de encuestas digitales, cuyo objetivo principal es adaptar dinámicamente las preguntas dependiendo de la ubicación de un usuario en un entorno cerrado. Debido a que es un espacio interior se utiliza transmisores inalámbricos mediante los cuales se realiza la localización de un usuario, esta información se envía a un servidor y este se encarga de adaptar las preguntas de una encuesta al contexto del usuario. La encuesta adaptada se presenta entonces en el dispositivo inteligente de los usuarios para que el la rellene a través de una aplicación. [3]

El artículo “Prototipo de aplicación para dispositivos móviles para el acceso a información turística de algunos puntos de interés de la ciudad” desarrollado en Bogotá - Colombia en el año 2015 por Daniel Mora y Diego Apolinar, se enfoca en el desarrollo de una aplicación móvil para el turista, usando tecnología Bluetooth de bajo consumo de energía, con el principio de funcionamiento de identificación de dispositivos, los autores proponen utiliza este principio para la localización de smartphones en exteriores y desplegar información turística relevante para ellos, si bien esta tecnología se usa en su mayoría en interiores no existe limitación alguna para usarlos en espacios abiertos como: lugares turísticos, museos, parques, plazas, etc. [4]

Poilabs un Start-Up tecnológico originario de Turquía es su investigación “Análisis de datos de geolocalización en el interior de centros comerciales” publicado en 2016, realiza innovación para el marketing de proximidad y la navegación en interiores

utilizando el principio de funcionamiento de detección de sensores bluetooth de bajo consumo de energía y plataformas digitales móviles, además el estudio se centra en la limitada accesibilidad que presentan los shopping center a personas con discapacidad visual, y como un sistema de geolocalización interno les ha ayudado a mejorar su movilidad e independencia. A su vez manifiestan la importancia de los smartphones como la herramienta principal para el desarrollo de marketing digital. [5]

1.2. Fundamentación teórica

1.2.1. Centros comerciales

La asociación española de centros comerciales AECC los define como un conjunto de establecimientos comerciales independientes, planificados y desarrollados por una o varias entidades, con criterio de unidad cuyo tamaño, variedad comercial, servicios compartidos y actividades complementarias están relacionadas con su entorno y que dispone de una imagen y gestión unitaria. [6]

Así mismo la “International Council of Shopping Centers ICSC” en su definición considera a los shopping center como un grupo de tiendas y otros establecimientos comerciales que ha sido planificado y desarrollado por uno o varios propietarios. Poseen un área de parqueadero y esta superficie comercial también oferta lugares de esparcimiento y ocio, a través de cines, áreas de paseo, restaurantes, u otros servicios. [7]

Ambas definiciones consideran estos shopping center como un predio cerrado y bien distribuido en el que se encuentra múltiples tiendas, esto los diferencia de los supermercados los cuales se consideran como una sola gran- tienda.

La doble funcionalidad “consumo y ocio” son la base del éxito alcanzado por los malls en los espacios urbanos ya que han logrado modificar la forma en que se realizan las compras gracias a su ambiente tranquilo lleno de escaleras eléctricas, elevadores de cristal, música relajante, aire acondicionado y muchas otras características; estos ofrecen en sus predios miles de productos lo que evita el desplazamiento de los clientes contribuyendo al ahorro del tiempo que dedican a sus compras. [8]

1.2.2. Marketing digital

El marketing digital móvil no es más que cualquier técnica de marketing que se realice a través de una red a la que los consumidores están conectados mediante un dispositivo digital, esto incluye todo tipo de ordenadores, teléfonos celulares inteligentes o smartphones, tablets, pc portátiles, etc.

La clave del marketing digital es la conectividad a internet, las empresas a través de redes sociales, páginas web y aplicaciones de terceros difunden contenido multimedia con información de sus productos o servicios a fin de conseguir nuevos clientes e incrementar sus ingresos. El marketing digital móvil se centra en la difusión de publicidad digital a través de dispositivos móviles en donde los nuevos teléfonos inteligentes o smartphones son la pieza clave, ya que el alto consumo de estos dispositivos hace innegable el atractivo de inversión para de este tipo de marketing. [9]

1.2.3. Smartphone

El teléfono móvil es un dispositivo electrónico inalámbrico cuya principal característica es su portabilidad, ya que la realización de llamadas no depende de ningún terminal fijo y no necesita ningún tipo de cableado para conectarse a la red telefónica móvil. En un principio su función era netamente la comunicación de voz aunque luego se agregaron funciones de mensajería instantánea, agenda, juegos, reproductor mp3 e incluso acceso a internet. [10]

Un smartphone o teléfono inteligente es un teléfono móvil con características similares a las de un computador personal, ya que además de realizar llamadas y enviar mensajes de texto permite la instalación de programas o aplicaciones para incrementar su funcionalidad. Entre sus principales características se destacan las pantallas táctiles y un sistema operativo que permite la instalación de aplicativos desarrollados por el fabricante o por terceros, cámaras fotográficas de alta calidad, software multimedia para reproducción de audio, video y principalmente cuentan con acceso a internet. [11]

Lo que hace inteligente a un teléfono móvil es su conectividad ya que permite el acceso a internet a través de la tecnología Wi-Fi o mediante las operadoras telefónicas de red móvil 4G LTE esto ha permitido que los usuarios tengan la posibilidad de tener acceso web en todo momento lo que dio paso a la creación de redes sociales, y plataformas

digitales en línea exclusivamente diseñadas para smartphones. Además cuentan con tecnologías como el GPS para geolocalización en tiempo real así como también conexión NFC y Bluetooth para transferencia de archivos. Estas funciones se ven complementadas con altas velocidades de procesamiento similares a las de un ordenador portátil lo que permite llevar a cabo múltiples funciones al mismo tiempo y baterías de larga duración que rondan los 4000mAh [10] [11]

1.2.4. Técnicas de geolocalización

La geolocalización es la determinación de la ubicación de una persona u objeto en el territorio mediante coordenadas de latitud, longitud y altura quedando plasmada en un mapa, esto en un modo netamente técnico ya que estas coordenadas se pueden interpretar como países, ciudades, y lugares específicos. [12]

Actualmente existen algunas tecnologías que permiten la determinar la geolocalización de un dispositivo móvil:

- Geolocalización GPS.
- Geolocalización RFID.
- Geolocalización por redes Wi-Fi (Wi-Fi Location).
- Geolocalización por bluetooth.

GPS

El sistema de posicionamiento global más conocido por sus siglas en inglés GPS, es un sistema de satélites que permite determinar la posición en cualquier lugar de la tierra las 24 horas del día. Consiste en un grupo de 24 satélites que envían ondas de radio a la superficie de la tierra. Un receptor GPS es un dispositivo electrónico que permite recibir las señales emitidas por los satélites y por medio de estas señales de radio puede calcular su posición a través de un grupo de números y letras que corresponden a un punto sobre un mapa. Fue desarrollado por el departamento de defensa de los Estados Unidos y su principio de funcionamiento genera algunas limitaciones. [13]

El principio de funcionamiento del GPS consiste en el cálculo de la distancia que separa el satélite del receptor y se basa en la propagación de las ondas

electromagnéticas. El tiempo en el que llega una señal satelital al receptor es directamente proporcional a la distancia recorrida, estas señales se propagan a una velocidad de 300.000 Km/s, esta señal contiene la hora exacta en la que es emitida y se compara en el receptor con la hora de llegada siendo capaz de obtener la distancia de separación. Los satélites forman superficies de cobertura con formas esféricas y gracias a su distribución alrededor de la tierra un receptor puede recibir las señales de por lo menos 3 satélites permitiendo encontrar un punto de intersección entre las áreas de cobertura determinando así la ubicación con un aceptable margen de error; esto se conoce como trilateración satelital. [14] [15]

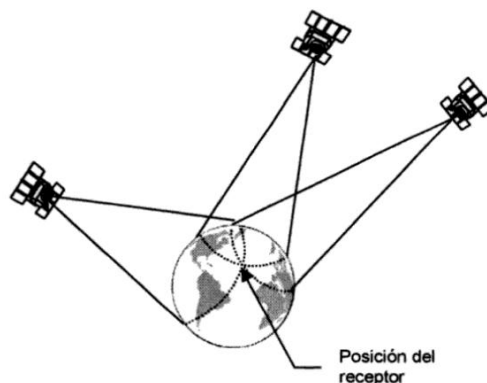


Figura 1 Principio de trilateración

Fuente: Guía práctica del GPS [16]

RFID

(Radio Frequency identification), es una forma de comunicación que usa radio frecuencia para identificar y rastrear objetos. Estos sistemas se componen de múltiples transmisores y un receptor con el cual interactúan con los transmisores. Por lo general funcionan con baterías y están destinados a la logística y gestión de almacenes, su diseño puede ser pequeño y en la mayoría de los casos son dispositivos pequeños y de bajo consumo de energía por lo que utilizan baterías. Pueden ser sistemas de comunicación activos o pasivos, según como estén configurados. Una de las mayores desventajas es que esta tecnología de comunicación no es compatible con dispositivos como smartphones por lo que se necesitaría de una interfaz de comunicación que sirva como medio para llegar a los móviles. [17]

Wi-Fi Location

Por su parte las redes Wifi se encuentran mayormente en espacios interiores tanto en lugares públicos como privados, esta tecnología permite el acceso a la red inalámbricamente. La ubicación de Wi-Fi se basa en el uso del protocolo de medición de sincronización fina (FTM) para medir el alcance entre el dispositivo móvil y los puntos de acceso (AP) cercanos. Los rangos se determinan utilizando intercambios de trama FTM de esta forma los dispositivos no necesariamente deben estar asociados (conectados) a un punto de acceso de la red. [17]

Utiliza el principio de trilaterización para determinar la ubicación de dispositivos al igual que el GPS emulando los satélites con los puntos de acceso, una de las limitaciones de este método de geolocalización es el consumo de energía ya que los AP necesariamente se deben conectar a la red eléctrica, así como también el cableado de red hacia el switch o router, además el margen de error es grande ya que los obstáculos como paredes, mobiliario e incluso los mismo usuarios constituyen un factor de interferencia. [18]

Bluetooth

Aplicando el principio de trilateración se podría utilizar cualquier sistema de comunicación inalámbrica para geolocalizar un dispositivo móvil, y una de las tecnologías inalámbricas mayormente extendidas en el mundo es Bluetooth. La tecnología bluetooth ha sido desarrollada para el intercambio de datos a cortas distancias de forma inalámbrica, el principal beneficio de esta tecnología es sin duda que puede trabajar con teléfonos móviles inteligentes (smartphone) como terminales.

En este caso la localización se basa en la colocación de transmisores bluetooth en un lugar fijo y es el terminal móvil el que determina su posición mediante el cálculo de las distancias del móvil hacia los transmisores detectados. Esta función actualmente no viene precargada por los SO móviles lo cual obliga a la instalación de una app en el terminal móvil para llevar a cabo el proceso de localización. [19]

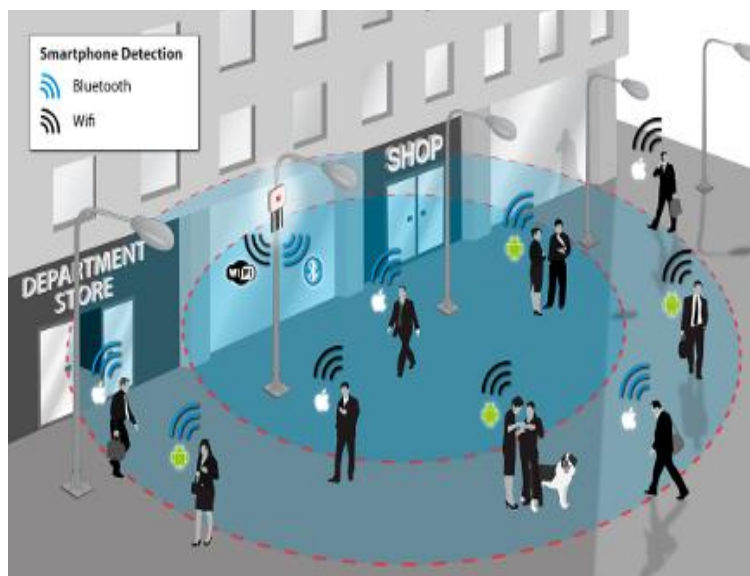


Figura 2 Localización wifi y bluetooth

Fuente: Wifi Tracker [19]

1.2.5. Bluetooth BLE

Bluetooth Low Energy es la nueva especificación 4.0 – 5.0 de la tecnología Bluetooth que ha sido desarrollado por Bluetooth Special Interest Group SIG. Utiliza la misma banda de frecuencia y su principal característica es el bajo consumo de energía lo que ha permitido aplicar esta tecnología en diferentes ámbitos. BLE se ha diseñado para la transmisión de pequeñas cantidades de datos en tiempos de transmisión muy pequeños los que garantiza un consumo de energía extremadamente bajo, su principal función es alertar su presencia a través de transmisiones recurrentes por lo que no está pensado para mantener largas conexiones entre dispositivos. [20]

Rango de operación.

El alcance real de cualquier dispositivo inalámbrico depende de factores como el diseño de la antena, la orientación, la carcasa o case, el lugar de trabajo, etc. En el caso de BLE está pensado en el uso en interiores por lo que el alcance de estos es bajo, esta tecnología permite la configuración de la potencia de transmisión que generalmente rodea los -30dBm – 0dBm siendo lo suficiente para dar una cobertura de 2 a 5 metros para alargar la vida útil de la batería. [20]

Topología de red.

Un dispositivo Bluetooth Low Energy puede comunicarse de dos maneras:

- a) Broadcasting.
- b) Conexión directa.

a) **Broadcasting.**

Mediante la transmisión Broadcasting se puede enviar datos a cualquier dispositivo receptor compatible que este dentro del rango de transmisión, el receptor escanea la información enviada por el dispositivo transmisor como se muestra en la Figura 3. Este mecanismo se utiliza principalmente para enviar datos de tal manera que cualquier dispositivo pueda recoger los datos transmitidos. [21]

Esta topología define dos roles separados: broadcaster y observador.

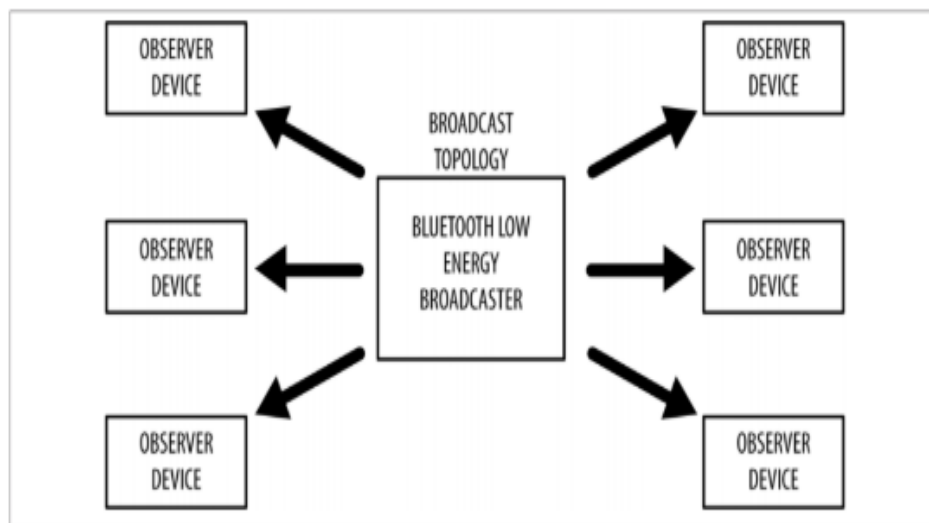


Figura 3 Topología Broadcast

Fuente: BLE Network Topology [22]

El broadcaster o transmisor es el encargado de emitir periódicamente datos para notificar su presencia hacia cualquier dispositivo que pueda recibir estos datos, con la característica de que no establece ninguna conexión. Mientras que el receptor se encarga de escanear repetidamente las frecuencias predefinidas para detectar cualquier dato no conectable que haya sido emitido por el broadcaster. [22]

b) **Conexión Directa**

Este método se utiliza si se necesita de la comunicación bidireccional entre dos dispositivos, o si se requiere enviar grandes cantidades de datos. La conexión entre dispositivos es permanente y se extiende hasta que uno de los dos decida terminar la conexión o salgan de la zona de cobertura.

Esta topología define dos roles el maestro y el esclavo.

El maestro central se encarga de buscar dispositivos en la frecuencia conectable una vez que identifica el dispositivo de destino establece la conexión e inicia con el intercambio de datos. Por otra parte el esclavo se conoce como un periférico que emite su identificación y tienen la capacidad de aceptar conexiones entrantes por parte de los dispositivos maestros, cuando estos dispositivos establecen una conexión dejan de emitir su disponibilidad hasta que esta finalice. [21]

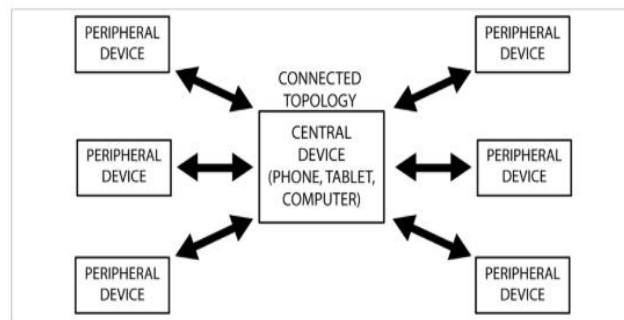


Figura 4 Topología de Conexión

Fuente: BLE Network Topology [22]

Protocolo BLE

La estructura de funcionamiento de un dispositivo Bluetooth Low Energy está dividido en tres bloques básicos: aplicación, host y controlador, cada uno de ellos está constituido por diferentes capas bien definidas que se estructuran de la siguiente manera. [23]

a) **Controlador**

- Interface de control de host (de controlador) (HCI)
- Capa de enlace (LL)
- Capa física (PHL)

b) Host

- Perfil de acceso genérico (GAP)
- Perfil de atributo genérico (GATT)
- Control lógico de enlace y Protocolo de adaptación (L2CAP)
- Protocolo de atributo (ATT)
- Administrador de seguridad (SM)
- Interface de control de host (de host) (HCI)

c) Aplicación

Como en otros protocolos el nivel de aplicación es el más alto y el responsable de contener la lógica, la interfaz con el usuario y el manejo de datos con todo lo relacionado al caso de uso real al cual se esté aplicando esta tecnología, por lo tanto su estructura dependerá del ámbito en el que se aplique el sistema. [23]

En la figura 5 se muestra la estructura clásica de un dispositivo BLE, siendo el bloque Host el que ha presentados mayores variaciones con respecto a las anteriores versiones de Bluetooth convencional, ya que este es el que permite la configuración de transmisión tipo broadcast.

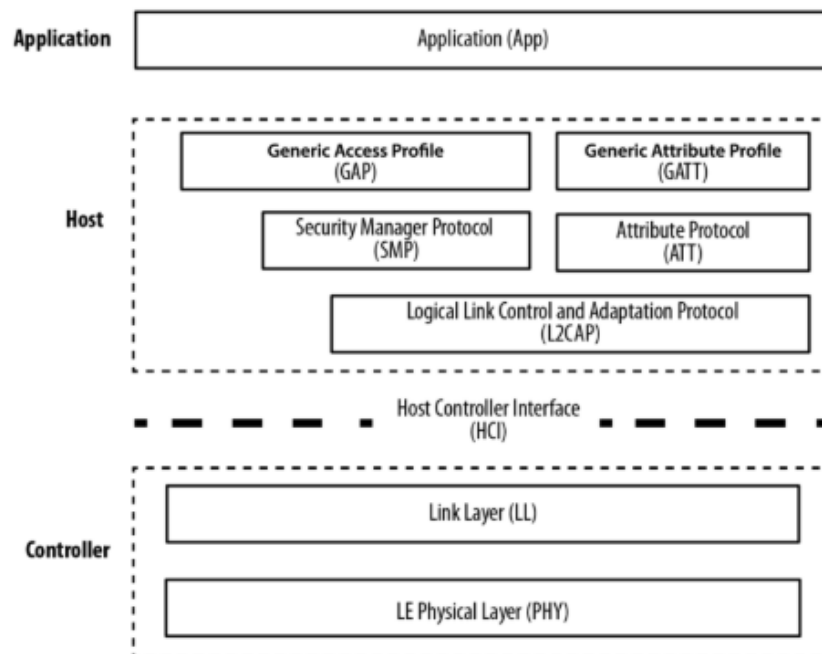


Figura 5 Protocolo BLE

Fuente: BLE structure. [21]

Banda de frecuencia

La capa física o Physical Layer (PHY) es la parte que contienen la circuitería de comunicación analógica, y se encarga de modular y demodular las señales para extraer la información que estas llevan.

En el caso de BLE utiliza la banda de comunicación de 2.4GHz ISM (Industria Científica y Médica) y divide en 40 canales desde 2400GHz hasta 2.4835GHz como se muestra en la figura 6, 37 de estos canales están destinados a conexión de datos y los últimos 3 son utilizados como canales de transmisión broadcast. [23]

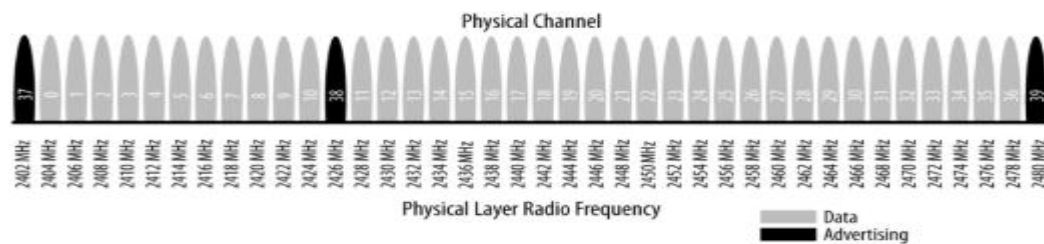


Figura 6 Canales de Radiofrecuencia

Fuente: Physical Layer [22]

El estándar utiliza la técnica llamada Spread Spectrum, en el que la conexión salta entre canales en cada evento de conexión utilizando la siguiente fórmula:

$$canal = (curr_channel + hop) \bmod 37$$

(Ecuación 1)

En donde el valor de hop es comunicado cuando la conexión es establecida y por lo tanto es diferente para cada nueva conexión, esta técnica minimiza el efecto de las interferencias de radio que se presentan en la banda de 2.4GHz ya que el Bluetooth clásico y el Wi-Fi aún prevalecen en el uso de esta frecuencia. [23]

Modulación

La modulación seleccionada para la transmisión inalámbrica es GFSK (Gaussian Frequency Shift Keying), siendo la misma modulación usada por el Bluetooth clásico y algunos otros dispositivos inalámbricos. La tasa de modulación para BLE está fijada en 1 Mbit/s que es, por lo tanto, el límite de rendimiento físico superior para esta tecnología. [23]

Dirección del dispositivo Bluetooth

Bluetooth Device Address o Dirección del dispositivo Bluetooth es el principal identificador de un dispositivo BLE similar a la dirección MAC Ethernet. La dirección está constituida por un número de 6 bytes cuya función principal es identificar de manera única un dispositivo de entre sus similares. [21]

Existen dos tipos de direcciones para identificar un dispositivo BLE estas son:

- **Dirección pública**

Esta es una dirección o identificador asignada por el fabricante y se registra en la Unidad de Registro de la IEEE y permanecerá permanente durante toda la vida útil de un dispositivo BLE.

- **Dirección aleatoria**

Esta dirección puede pre-programarse en el dispositivo o generarse aleatoriamente en tiempo de ejecución. Esta tiene muchos usos prácticos ya que puede ser cambiada por el administrador y será reconocible por los dispositivos receptores. [22]

Anuncio y Escaneo

BLE tiene solo un formato de paquete y dos tipos de paquetes: Advertising (anuncio de existencia) y Datos (conexión directa). Los paquetes de Advertising o Anuncio tienen dos propósitos:

- Transmitir datos para aplicaciones que no necesitan la sobrecarga de un establecimiento de conexión completo.
- Para descubrir esclavos y conectarse con ellos.

Cada paquete de anuncio puede transportar hasta 31 bytes de carga de datos incluida la dirección del dispositivo Bluetooth. Estos paquetes son simplemente transmitidos a ciegas por el dispositivo anunciante (broadcast) sin el conocimiento previo de la presencia de ningún dispositivo de escaneo.

Se envían a una velocidad fija definida que oscila entre 20 ms y 10s. Obviamente cuanto más corto sea el intervalo, mayor es la frecuencia con la que se transmiten los paquetes lo que aumenta la probabilidad de que esos paquetes sean recibidos por un dispositivo escáner. [21]

Los datos de anuncio usan un máximo de tres canales de frecuencia y el anunciante y el escáner no están sincronizados, el escáner recibirá un paquete de publicidad con éxito solo cuando se superpongan aleatoriamente, como se muestra en la Figura 7.

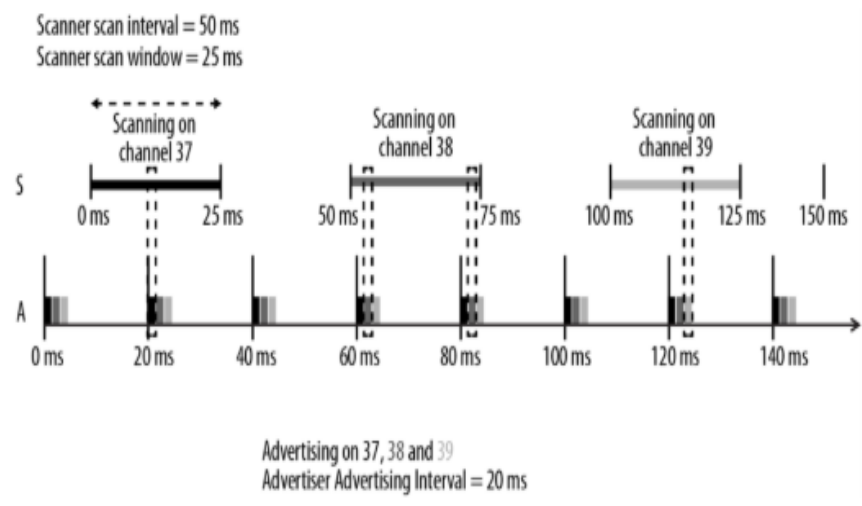


Figura 7 Ventanas de anuncio y escaneo

Fuente: Physical Layer BLE [23]

El intervalo de escaneo y los parámetros de la ventana de escaneo definen con qué frecuencia y durante cuánto tiempo un dispositivo receptor (escáner) escuchará posibles paquetes por parte de un dispositivo anunciante. De igual manera con el intervalo de Advertising (Anunciante), esos valores tienen un profundo impacto en el consumo de energía, ya que se relacionan directamente con la cantidad de tiempo que los circuitos de transmisión radio deben estar encendidos. [23]

1.2.6. BLE Beacons

Un beacon es un dispositivo electrónico que alerta a los dispositivos móviles, como teléfonos inteligentes (smartphones) y tabletas, de su presencia a través de la transmisión Broadcasting de Bluetooth Low Energy (BLE). Estos dispositivos emiten su identificación hacia el exterior de manera inalámbrica, cuando un dispositivo móvil está suficientemente cerca y recibe esta información, una aplicación desencadena una actividad o acción, es así que estos dispositivos únicamente utilizan la configuración Advertising y Scanning (Anuncio y Escaneo). [24]

Estos dispositivos están configurados de tal manera que no reciban conexiones directas con un receptor. Al evitar las conexiones permanentes se ahorra grandes cantidades de energía por lo que es suficiente una batería tipo moneda para darle autonomía al dispositivo, esto permite que su estructura física sea reducida y se pueda colocar en cualquier lugar. [25]



Figura 8 Interacción Beacon

Fuente: Accent Systems [26]

Los beacons se utilizan en su mayoría para realizar tareas de acuerdo a la ubicación que se les dé en un entorno, estas tareas se realizan a través de una aplicación de software en smartphones ya que son compatibles con la tecnología Bluetooth.

Protocolos Beacon

Al igual que cualquier otra tecnología, los Beacons funcionan con un conjunto de protocolos, se consideran a estos como un perfil que indica cómo se está transmitiendo información y no deben confundirse con los protocolos propios de la tecnología BLE. Esta información es manejada por un dispositivo que típicamente es un teléfono móvil, así para lograr una transmisión exitosa los dos extremos deben conocer la configuración de los datos del otro. El tipo de datos transmitidos por broadcast es definido por el tipo de protocolo que el beacon usa para transmitir su UUID (Universally Unique Identifiers). [25]

Debido a que estos dispositivos interactúan con smartphones los protocolos Beacon más utilizados obedecen a la penetración de los sistemas operativos móviles a nivel global, estos protocolos son **iBeacon** y **Eddystone** los cuales cuentan con el respaldo de Apple y Android (Google) respectivamente, estas gigantes compañías de tecnología móvil han logrado mayor penetración en el mercado, aunque en la última década Android se ha convertido en el Sistema Operativo Móvil de mayor consumo en el mercado. [25]

iBeacon

Apple iBeacon es un protocolo para dispositivos BLE Beacons, es considerado como una plataforma beacon nativa de Apple usada para determinar la ubicación de dispositivos. Se trata de una implementación cerrada y únicamente compatible con dispositivos que cuenten con el sistema operativo iOS, el paquete de datos de este protocolo cuenta con los siguientes componentes. [25]

- **UUID:** Identificación Universal Única
- **Major:** Cadena de 2Bytes de datos para identificar un grupo de beacons.
- **Minor:** Cadena de 2Bytes de datos para identificar beacons de forma individual.
- **Tx Power:** La potencia de transmisión se puede configurar.

Eddystone

Google Eddystone es un protocolo beacon de código abierto desarrollado por Google, y es compatible tanto con Android como con iOS. El proceso de configuración de un beacon se denomina aprovisionamiento y puede lograrse por una aplicación de Google ya sea en dispositivos Android o iOS. Eddystone a diferencia de iBeacon el protocolo Eddystone cuenta con tres tipos de paquetes de datos: [25]

a) Eddystone-UID

Eddystone-UID es un tipo de trama de difusión que consta de una ID de 16 bytes, de los cuales 10 bytes se utilizan para namespace el cual puede ser el mismo para un grupo específico de beacons y 6 bytes para identificar un beacon particular. Ambas partes del UID aseguran la que la identificación de un beacon o un conjunto de estos sea única.

b) Eddystone-URL

Eddystone URL es un paquete de datos configurado con el fin de direccionar a un usuario hacia cualquier sitio web a través de una dirección URL comprimida, para ello Google acorta los URL y cualquier dispositivo que identifique un beacon configurado con este protocolo podría decodificar el enlace URL y acceder a éste, pero obligatoriamente se necesita de acceso a Internet. [25]

c) Eddystone-EID

Este es otro tipo configuración de trama que tiene la particularidad de cambiar periódicamente, por ello se conoce como Identificador Efímero (Ephemeral Identifier siglas en ingles de EID). Su uso esta direccionado para aplicaciones de seguridad en donde se requiera la transmisión segura de un identificador, como en transferencias bancarias, compras, etc. [25]

d) Eddystone TLM

Con el fin de introducir estos dispositivos en el internet de las cosas existe una configuración de paquete de datos llamado Eddystone TLM el cual además de emitir su ID adjunta datos de telemetría tales como temperatura, humedad, batería etc.

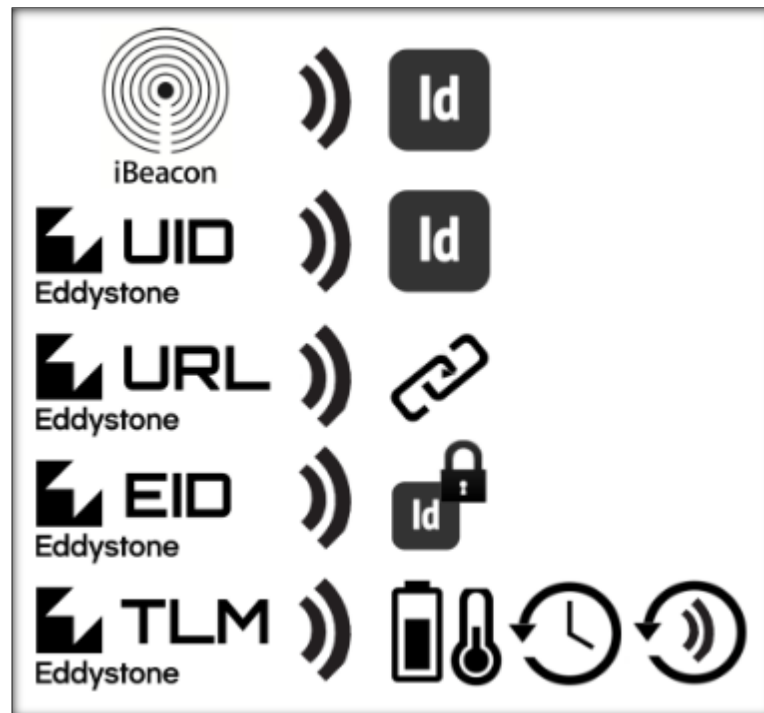


Figura 9 Protocolos Beacon

Fuente: Accent Systems [27]

Cada uno de estos paquetes tiene una estructura y su uso dependerá de la aplicación a la que esté destinado un beacon o un conjunto de ellos. Una vez seleccionada la modalidad de transmisión y la configuración del dispositivo la etapa de transmisión está lista para trabajar, la etapa de desarrollo de la aplicación móvil es la que tomara las decisiones una vez que haya detectado un paquete de datos proveniente de un beacon, para ello los fabricantes de estos dispositivos desarrollan SDK's, las cuales no son más que clases de código programado por el fabricante que facilitan el desarrollo de la aplicación simplificando los procesos que permiten a los teléfonos móviles interactuar con los dispositivos beacon. [24]

1.2.7. Aplicaciones móviles.

Las aplicaciones móviles o más conocidas como apps surgieron con el lanzamiento mismo de los teléfonos móviles, aunque es en la última década en la que el auge de las mismas ha revolucionado los móviles y se ha introducido el concepto de smartphone. Las aplicaciones no son más que programas de software diseñados para dispositivos móviles, podría decirse que los programas son para las computadoras lo que las apps son para los móviles. [28]

Junto con la aparición de los sistemas operativos para móviles Android, Apple, y Windows Phone se generaron en el mercado nuevos modelos de negocio basados en el desarrollo de software móvil, esto ha permitido el surgimiento de desarrolladores de apps así como también tiendas de aplicaciones como Google Play, App Store y Windows Phone Store, aunque esta última no ha tenido el impacto que si han logrado sus competidores ya mencionados. Para el desarrollo de una app móvil es importante definir un proceso ordenado el cual que empieza con la concepción de la idea, definición de funciones, diseño gráfico de la interfaz, desarrollo o codificación, y por último la publicación o distribución de la app ya terminada. [28]

Tipos de aplicaciones móviles

A nivel de programación o codificación, existen tres formas de desarrollar aplicaciones móviles y la elección de uno de estos métodos dependerá de las funciones concretas a realizar así como también de los recursos de hardware que se necesitarán del móvil; por ello las apps se clasifican en aplicaciones nativas, web e híbridas.

a) Aplicaciones Nativas

Este tipo de aplicaciones son desarrolladas exclusivamente con el software ofrecido por el sistema operativo en el que se va a ejecutar la misma, genéricamente se denomina Software Development Kit (SDK). De esta forma cada SO móvil cuenta con un diferente entorno de desarrollo y las apps nativas se programan específicamente para su plataforma. Una de las principales características de este tipo de apps radica en que no requieren del acceso a Internet para funcionar y también permiten utilizar recursos del hardware del teléfono móvil como la cámara, micrófono, GPS, acelerómetro entre otros. [28]

b) Aplicaciones web

La principal característica de este tipo de aplicaciones es que no se emplea un SDK, lo que permite programarlas independientemente del SO en el que se vaya a usar la aplicación, así estas apps se pueden utilizar en diferentes plataformas sin la necesidad de desarrollar un código diferente para cada una de estas plataformas. Para este tipo de apps típicamente se recurre a la programación web, en las que se utilizan herramientas como HTML, JavaScript y CSS, una gran desventaja de este tipo de apps es el limitado acceso a recursos del hardware del móvil. [28]

c) Aplicaciones híbridas

Las apps híbridas son una combinación entre las apps web y nativas, en su gran mayoría la etapa de desarrollo se realiza con el principio de app web es decir usando HTML, CSS JavaScript y una vez terminada esta etapa se empaqueta o compila de forma nativa para cada plataforma, obteniendo como resultado final una app nativa. Esta combinación permite acceder a capacidades del móvil igual que una app nativa normal y a su vez distribuir la aplicación en distintas plataformas sin la necesidad de desarrollarla completamente en otro SDK. [28]

1.2.8. Android Studio

Es el entorno de desarrollo integrado (IDE) oficial del sistema operativo Android, está destinado a la creación de aplicaciones y está sustentado por el editor de código IntelliJ lo que facilita la creación de aplicaciones gracias a su intuitivas herramientas, trabaja con lenguajes de programación Kotlin y Java aunque es compatible con C y sus

variantes. Android Studio cuenta con una interfaz de desarrollo simple y amigable para el trabajo del desarrollador, proporciona todas las herramientas necesarias y es un programa muy intuitivo lo que optimiza el tiempo del programador. [29]

Además este IDE cuenta con un asistente de inicio que ayuda al usuario a empezar su proyecto seleccionando la versión de Android que se va a utilizar, el tipo de dispositivo para el cual se desarrolla la app, y la opción de crear una actividad principal automáticamente dependiendo de las necesidades que tenga el proyecto.

Algunas de las funciones que facilitan el trabajo en este IDE se mencionan a continuación. [29]

- Cuenta con su propio administrador de dispositivos virtuales para emulación.
- El sistema de compilación está basado en Gradle.
- El entorno permite crear aplicaciones para todas las versiones del sistema operativo Android ya que se actualiza su versión de compilación automáticamente.
- Admite el uso de librerías o APIs externas y su inclusión es fácil y rápida.
- Los dispositivos virtuales permiten la emulación instantánea Instant Run lo que agiliza la simulación de cambios en el código sin la necesidad de compilar el APK nuevamente.
- Permite la creación de archivos APK firmados, lo que protege el código del desarrollador.
- El desarrollo de la interfaz gráfica de las apps se realiza mediante lenguaje XML.

A más de estas ventajas Android Studio cuenta con un sistema de actualización permanente el cual alertará al usuario sobre la existencia de nuevas versiones de compilación, nuevas versiones de Android, API's, cambio de políticas o nuevas actualizaciones de software lo cual permitirá al programador mantenerse siempre actualizado para el desarrollo de sus proyectos.

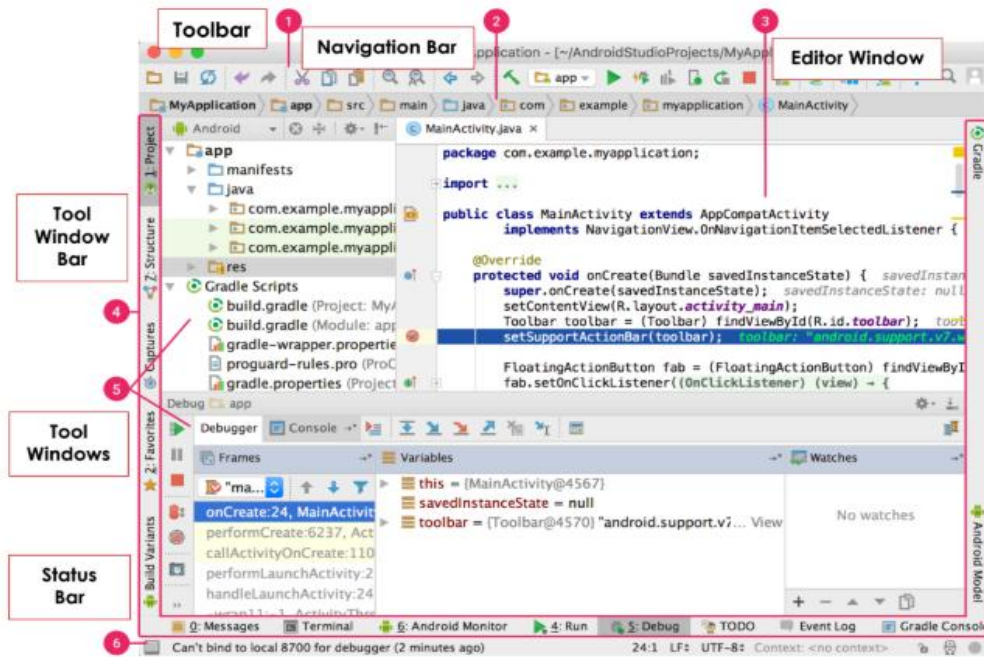


Figura 10 Interfaz de usuario

Fuente: Introducción a Android Studio [29]

Debido a su robustez Android Studio necesariamente debe ejecutarse en un ordenador orientado al desarrollo, por ello se recomienda utilizar este entorno en un computador con las siguientes características (tabla 1).

Tabla 1 Requisitos del sistema

Fuente: Android Studio Introduction [29]

Característica	Recomendación
Sistema Operativo	<ul style="list-style-type: none"> Windows 7/8/10 (64 bits) Mac OS 10 o posteriores.
RAM	Mínimo 3GB, Emulador mínimo 1GB
Almacenamiento	IDE 4GB, Emulador 2GB
Resolución de pantalla	Mínimo 1280x800.
Virtualización	Mínimo 2 Núcleos para Emulador

Android Studio 3.0 y las versiones posteriores admiten todas las funciones del lenguaje Java 7 y un subconjunto de funciones del lenguaje Java 8 que varían según la versión de la plataforma.

1.2.9. BLE en S.O. Android.

El sistema Operativo Android a partir del nivel de API 21 ha incorporado compatibilidad con tecnología Bluetooth de bajo consumo (BLE) característica con la cual las apps pueden interactuar con dispositivos bluetooth de esta nueva versión. Algunos de los casos de uso más frecuentes son los siguientes.

- Transferir volúmenes de datos de tamaño reducido entre dispositivos cercanos.
- Interactuar con sensores de proximidad como beacons, que permiten que el usuario tenga una experiencia personalizada en función de su ubicación actual.

Bluetooth (BLE) está diseñado para ofrecer un consumo de energía significativamente menor permitiendo que las apps de Android se comuniquen con dispositivos con requisitos de consumo más estrictos. [30]

Para el desarrollo de aplicaciones móviles Android existen algunos conceptos básicos que el IDE ha establecido para facilitar su implementación algunos de estos son:

Perfil de atributos genéricos (GATT)

El perfil GATT es una especificación para enviar pequeños fragmentos de datos, denominados "atributos", a través de un vínculo BLE, los perfiles de aplicaciones de bajo consumo actuales se basan en GATT. Un perfil es una especificación que define cómo funciona un dispositivo en una aplicación determinada. Cabe señalar que un dispositivo puede implementar más de un perfil. [30]

Protocolo de atributos (ATT)

GATT está basado en el protocolo de atributos (ATT). Por eso, también se denomina GATT/ATT. ATT usa la menor cantidad de bytes posible. Cada atributo se identifica de forma exclusiva mediante un identificador único universal (UUID), que es un formato estandarizado de 128 bits para un ID empleado para identificar información de manera exclusiva. [30]

1.2.10. Java

Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Esta plataforma ha sido fundamental para el funcionamiento de computadoras, teléfonos móviles, aplicaciones, sitios web y muchos otros dispositivos y plataformas. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras se han desarrollado gracias a la aparición de Java. Además una gran ventaja de Java radica en que es de libre acceso, su descarga es gratuita y constantemente se desarrollan nuevas actualizaciones que aportan a la seguridad, y fiabilidad de la plataforma.

Como lenguaje de programación Java se considera un lenguaje orientado a objetos, sencillo, portable, robusto, distribuido, securizado, independiente de las arquitecturas, eficaz, multi-hilo y dinámico. [31] [32]

Java está diseñado para el desarrollo de aplicaciones de alto rendimiento destinadas al más amplio rango de plataformas informáticas en el mundo. Java se ha convertido en el lenguaje de programación orientada a objetos más utilizado por la comunidad de desarrolladores a nivel mundial ya que permite combinar aplicaciones o servicios Java para crear otras aplicaciones o servicios personalizados, así como también escribir aplicaciones potentes y eficaces para teléfonos móviles. [33]

1.3. Objetivos

1.3.1. Objetivo general

Implementar un prototipo de red inalámbrica de geolocalización en espacios cerrados aplicado al marketing digital de proximidad en centros comerciales.

1.3.2. Objetivos específicos

- Analizar las tecnologías de geolocalización de dispositivos móviles en espacios cerrados.
- Implementar una red interna de transmisores para la geolocalización de smartphones en espacios cerrados.
- Realizar una aplicación móvil enfocada al marketing digital de proximidad utilizando el sistema de geolocalización.

CAPÍTULO II

METODOLOGÍA

2.1. Materiales

2.1.1. Smartphone

Para el desarrollo del proyecto se necesita de un teléfono móvil con sistema operativo Android versión 5 o posteriores (mínimo API 21 Lollipop), además el smartphone debe contar con tecnología Bluetooth BLE (Bluetooth 4.1 o posteriores). Las pruebas de funcionamiento se realizarán con un teléfono marca Huawei modelo P 2019, el cual cumple con las características mencionadas (tabla 2).

Tabla 2. Características del smartphone

Característica	Detalle
Android	9.1.0 (API 28)
Bluetooth	BLE (4.2) A2DP

2.1.2. Bluetooth Beacon IBKS105

Se trata de un dispositivo Bluetooth Low Energy que trabaja en modo Advertising o no conectable, su núcleo está basado en el semiconductor chip NRF51822, oficialmente fue el primer Beacon compatible con la tecnología iBeacon y Eddystone al mismo tiempo. Fue desarrollado específicamente para aplicaciones de proximidad, cuenta con una batería de tipo moneda CR2477 la cual en condiciones normales le puede dar una autonomía de hasta 46 meses. En la tabla 3 se detallan las características técnicas del Beacon IBKS105.

Tabla 3 Especificaciones técnicas iBKS105

Dimensiones	11.3x52.6Ø mm
Peso	24g
Chip	Nordic nRF51822
Distancia	50m
Batería	3V 1000 mAh Coin CR2477
Consumo de corriente	2.4 uA
Material del case	ABS
Color	Blanco
Temperatura de operación	-25 a 70°C
Protocolos	iBeacon Eddystone: URL, UID, TLM, EID



Figura 11 Beacon iBKS105

2.1.3. Android Studio IDE

Para el diseño y codificación de la aplicación móvil se utilizará el software Android Studio, la aplicación debe estar actualizada mínimo a la versión 3.2, debido a que las librerías Beacon varían constantemente y una librería obsoleta causaría problemas de rendimiento. La versión que se utilizará en el desarrollo de este proyecto es Android Studio 3.5.3.

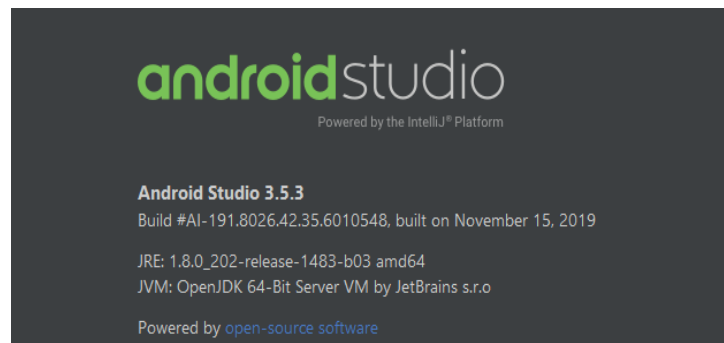


Figura 12. Versión de Android Studio.

2.2. Métodos

2.2.1. Modalidad de la investigación

- **Investigación Aplicada**

El presente proyecto se empleó Investigación Aplicada, debido a que se realizó un prototipo tangible funcional como producto de la investigación realizada.

- **Investigación bibliográfica-documental**

Esta se obtuvo a través de libros, revistas, artículos y base de datos científicas de la Universidad Técnica de Ambato que proporcionaron teorías, resultados, experimentaciones, instrumentos y técnicas usadas para llevar a cabo el proyecto.

- **Investigación de campo**

Se realizó la investigación de campo para el análisis y estudio del sistema de geolocalización, para la obtención de información acerca del marketing de proximidad.

2.2.2. Recolección de información

La recolección de información del presente proyecto se realizó a través de libros, artículos científicos, trabajos de investigación, revistas, y paginas oficiales de soporte tanto del hardware como del software que se utilizó en el proyecto además de la guía del docente tutor.

2.2.3. Procesamiento y análisis de datos

Para el procesamiento y análisis de datos se procedió de la siguiente manera:

- Se recopiló información bibliográfica afín al problema o bien información relacionada con la solución de experiencias similares al tema de investigación.
- Se analizaron las tecnologías que podrían utilizarse como solución al problema para realizar una adecuada elección.
- Se estructuró un proceso óptimo para el desarrollo del sistema.
- Se revisaron los datos obtenidos para comprobar la eficiencia del sistema y la prevención de fallas.

2.3. Desarrollo del proyecto

2.3.1. Requerimientos del sistema

Debido a que el proyecto de investigación pretende aplicar el sistema de geolocalización de dispositivos móviles en centros comerciales y posteriormente la utilización del mismo en la optimización de técnicas de marketing digital se han considerado algunas necesidades importantes para el sistema las cuales que se mencionan a continuación:

- Alta precisión, margen de error bajo, debido a que se aplicará en interiores y espacios reducidos.
- Estructura de red, simple en la que el teléfono móvil (usuario final) sea el único dispositivo recepción.
- Aplicación móvil de interacción de hardware nativa o híbrida.
- Bajo consumo de energía, tanto de la red de transmisión como de los receptores móviles.
- Cobertura de decenas de metros, focalizada a lugares cerrados y espacios reducidos.
- Versatilidad y escalabilidad.
- Seguridad de los datos de localización, privacidad de información.
- Bajo costo de implementación de red y software de desarrollo móvil.

A partir de estas necesidades se evaluarán y seleccionarán las tecnologías, técnicas, métodos, así como también materiales de hardware y software idóneos para la realización del prototipo y conseguir los requerimientos establecidos.

2.3.2. Análisis de componentes.

Tecnologías de geolocalización.

En las tablas 4 y 5 se muestra una comparación de tecnologías de geolocalización de dispositivos móviles. Las principales características a analizar son: accesibilidad, precisión, alcance, seguridad, costo y consumo de energía. A partir de esta comparación se seleccionará la tecnología que mejor se adapte a las necesidades y requerimientos del sistema, para conseguir un óptimo funcionamiento.

Tabla 4 Comparación Accesibilidad, Precisión, Alcance

Elaborado por: El investigador.

	Accesibilidad	Precisión.	Alcance
GPS	Los smartphones cuentan con esta tecnología ya incorporada.	Poca precisión en distancias cortas, nula conexión en espacios interiores o subterráneos. Interiores (10-15 metros)	Largo alcance, debido a la conexión satelital se puede extender hacia cualquier lugar en el exterior. (Km)
WIFI	Todos los móviles actuales están dotados de conexión wifi para acceso a internet.	La precisión de este método depende de la estructura de red disponible, por lo general es bajo ya que existen obstáculos en el medio de comunicación. Interiores (3-5 metros)	Alcance limitado, se podría cubrir mayor espacio con la instalación de repetidores AP. (10-60 metros)
RFID	Se requiere de múltiples actores físicos para interactuar con teléfonos móviles. Este se utiliza como un sistema de proximidad más no de geolocalización en sí.	De la calidad de los equipos de RF que se utilicen en este método dependerá la precisión del sistema, es decir antenas, receptores etc. Interiores (1-2 metros)	Alcance muy limitado debido a la estructura del sistema. (1-2 metros)
BLUETOOTH BLE	Tecnología disponible en todos los smartphones.	Alta precisión en espacios cerrados con el despliegue de mínimo tres transmisores. Interiores (1-2 metros)	Un solo transmisor podría cubrir un espacio de hasta 30m sin obstáculos.

Tabla 5 Comparación: Seguridad, Costo, Consumo de energía

	Seguridad	Costo	Consumo de energía.
GPS	La seguridad de datos depende de los receptores y su capacidad de mantener la información oculta.	La infraestructura necesaria para dar servicio GPS tiene un alto costo, aunque actualmente la red satelital es gratuita y de libre acceso para ser aplicada en smartphones.	El consumo de energía varía dependiendo de la situación de uso, en interiores en un móvil podría desgastar la batería hasta un 60% más rápido debido al forzamiento a buscar señales satelitales de potencia casi nula.
WIFI	Debido a la conexión a internet la seguridad de los datos compartidos es baja, aunque los protocolos WEP y WPA trabajan para evitarlo.	Si se requiere de alta precisión sería necesario la implementación de múltiples AP repetidores lo que aumentaría los costos de implementación.	En el caso de los puntos de acceso necesitan el suministro de energía de la red eléctrica de forma permanente lo que dificulta su libre ubicación. Para los receptores móviles no representa un consumo excesivo (10 – 20%) ya que esta tecnología se ha desarrollado pensando en el ahorro de energía para acceso a internet y redes.
RFID	La seguridad es limitada debido a que los equipos pueden ser replicados o clonados.	El costo de un sistema basado en RFID puede variar dependiendo de las necesidades del sistema, aunque para un solo receptor podría ser de bajo costo, pero con un rendimiento de localización limitado.	Los dispositivos transmisores móviles de esta tecnología por lo general trabajan con bajas cantidades de energía, por lo que utilizan pequeñas baterías. No es el caso de los receptores que necesitan de una fuente permanente de alimentación.
BLUETOOTH BLE	Los datos enviados pueden ser recibidos por cualquier dispositivo, aunque se puede administrar contraseñas de configuración.	Los actores para esta tecnología son únicamente dos el dispositivo transmisor y el móvil como receptor, por lo que el costo es bajo requiriendo únicamente los emisores bluetooth.	La tecnología bluetooth en su última versión prioriza el ahorro de energía (bluetooth BLE), los dispositivos transmisores de datos puede trabajar con baterías tipo moneda llegando a tener autonomía de años de uso. Para los receptores móviles su uso solo llegaría a desgastar del 2 a 3% de batería.

A partir de la comparación de las distintas tecnologías de geolocalización, y tomando en cuenta los requerimientos del sistema, se ha seleccionado la tecnología Bluetooth de bajo consumo de energía (BLE) para el desarrollo del proyecto, ya que esta es la que se adapta mejor a las necesidades establecidas previamente, principalmente por su reducido consumo de energía, versatilidad y el bajo costo de implementación, ya que en estos aspectos supera a las demás tecnologías cuando se requiere aplicaciones en espacios cerrados o interiores.

Aplicaciones móviles.

Una vez analizados los tipos de aplicaciones móviles en la fundamentación teórica, la tabla 6 muestra las ventajas y desventajas de estos tipos de apps.

Tabla 6 Ventajas y Desventajas de los tipos de aplicaciones móviles

Tipo de aplicación.	Ventajas	Desventajas
Nativa	<ul style="list-style-type: none"> • Acceso completo al dispositivo. • Presencia en tiendas de apps. • Actualización constante. 	<ul style="list-style-type: none"> • Desarrollo más costoso • Diferencia de herramientas para cada SO. • Código inutilizable para otra plataforma.
Web	<ul style="list-style-type: none"> • Código reutilizable. • Desarrollo más económico. • Disponible para todos los SO. 	<ul style="list-style-type: none"> • Conexión obligatoria a internet. • Acceso limitado a hardware del sistema. • No tiene presencia en tiendas de apps.
Híbrida	<ul style="list-style-type: none"> • El código base es reutilizable. • Presencia en tiendas de apps. • Distribuible en todas las plataformas. 	<ul style="list-style-type: none"> • Diseño visual diferente al del sistema operativo. • Tiempo de respuesta es menor que una app nativa. • Conexión a internet.

Debido a que se ha seleccionado la tecnología Bluetooth BLE la aplicación a desarrollarse requiere del acceso a los recursos de hardware del sistema, esta es la razón principal para desarrollar una aplicación tipo nativa. A su vez se ha establecido a Android como el sistema operativo base para el desarrollo de la app, debido a que este S.O. de mayor demanda en el mercado mundial y regional, además de que no requiere de licencia alguna para comenzar el desarrollo.

Beacons

Existe gran cantidad de fabricantes de bluetooth Beacons, en la tabla 7 se mencionan algunos fabricantes y las principales características de sus dispositivos, como medidas, peso, consumo de energía, licencia de uso de software, soporte técnico, y costos; cabe recalcar que los fabricantes ofrecen al mercado múltiples modelos y diseños de sus dispositivos y únicamente se ha tomado en cuenta a equipos que cuentan con el protocolo Eddystone y que permitan la configuración GATT.

Tabla 7. Características físicas beacons

Fabricante	Medidas	Peso	Batería	Licencia	Precio
Accent Systems	11.3x52Ø mm	24g	1 Coin CR2477	Libre > 5 dispositivos.	\$16 c/u
Blesh	14.8x60Ø mm	84g	1 Coin CR2032	Desarrollo limitado	---
Bluecats	27.8x76Ø mm	35.5g	2 Pilas AA	Libre > 3 dispositivos.	\$30 c/u
BlueUp	15x 40x40 mm	70.2 g	1 Coin CR2477	Desarrollo limitado.	---
Bluvison	8.35x35Ø mm	76g	1 Coin CR2032	Libre > 10 dispositivos	\$16 c/u
Estimote	25x69x46 mm	86g	1 Coin CR2477	Libre > 20 dispositivos.	\$25 c/u
Gimbal	86x77x25 mm	94g	4 Pilas AA	Libre > 15 dispositivos.	\$23 c/u
Kontakt.io	15x70Ø mm	35g	1 Coin CR2477	Libre > 3 dispositivos.	\$28 c/u
Minewtech	25x70x50 mm	65g	1 Coin CR2032	Libre > 10 dispositivos.	\$18 c/u
POI	30x60x45 mm	48g	2 Pilas AA	Desarrollo limitado	---
Radius	72x72x23 mm	109g	4 Pilas AA	Libre >10 dispositivos.	\$24 c/u

En términos de licencia se describe “desarrollo limitado” a los casos en los que el fabricante no permite utilizar libremente los dispositivos beacon.

Para el desarrollo del proyecto se ha seleccionado la marca Accent Systems, específicamente el dispositivo beacon iBKS105, ya que ofrece todos los perfiles de configuración de Eddystone necesario para trabajar con Android, a su vez cuenta con el protocolo GATT con el cual se pueden configurar múltiples perfiles para que trabajen al mismo tiempo, lo que resulta útil si se requiere que otra aplicación trabaje con los mismo dispositivos. Otra ventaja de esta marca es que desarrolla sus dispositivos para la región ibérica por lo que los costos y tiempos de envío son reducidos y el soporte técnico puede ser de gran ayuda para países de habla hispana.

2.3.3. Implementación de red inalámbrica

Una vez seleccionados los dispositivos de software para la implementación de la red inalámbrica se procede con el diseño de la misma, esto contempla la determinación del número de dispositivos necesarios y su ubicación en el lugar de interés. Para el diseño y la implementación de cualquier red inalámbrica se toman en cuenta principalmente dos aspectos fundamentales: la capacidad de la red y el área de cobertura.

Debido al principio de funcionamiento del sistema los dispositivos transmisores trabajan en modo broadcast, de esta forma no se realiza una conexión directa entre los Beacons y el Smartphone por lo tanto el ancho de banda de los dispositivos es un parámetro que no limitará el número de usuarios en la red. Por esta razón se desprecia el análisis de cantidad de transmisores basado en el ancho de banda y se determina únicamente analizando el área de cobertura.

Para el análisis del área a la que se debe dar cobertura es fundamental analizar algunos factores físicos como las capacidades de los transmisores: potencias de transmisión, ganancia de antenas; otras consideraciones importantes son la configuración de los espacios, obstáculos, materiales de construcción etc. Siguiendo el principio de funcionamiento de trilateración para la determinación de la ubicación de un usuario se requiere comunicación con mínimo tres transmisores, en tal virtud y analizando los factores físicos antes mencionados estos se han desplegado por el área de cobertura de la siguiente manera (figura 13).

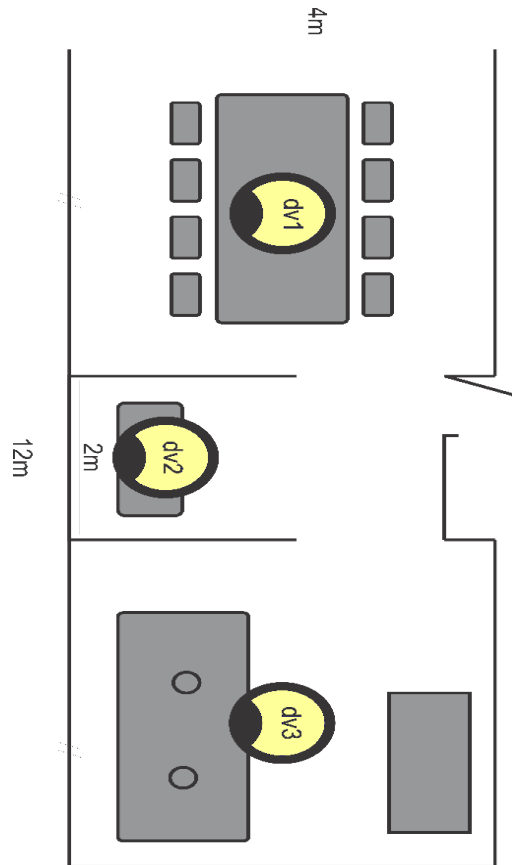


Figura 13. Implementación de red.

Las figuras 14 y 15 muestran la disposición física de los beacons en el espacio de análisis.



Figura 14. Ubicación de beacon 1



Figura 15. Ubicación de beacon 2

2.3.4. Configuración de dispositivos Beacon

Los dispositivos transmisores Bluetooth Beacons que se utilizarán para la implementación del prototipo, trabajarán bajo el protocolo Eddystone y se configurarán bajo el perfil UID, además de estos parámetros se debe configurar la potencia de transmisión, el intervalo de tiempo entre emisiones y el nombre del dispositivo; a través del software del fabricante (iBKSConfigTool), esta aplicación se encuentra disponible gratuitamente en las tiendas de apps y puede instalarse en cualquier dispositivo móvil que cuente con la tecnología Bluetooth BLE.

En la figura 16 se muestra la interfaz principal de la app de configuración.

Para iniciar con el proceso se debe quitar el seguro de la batería del dispositivo que se desea configurar para que este se energice y entre en funcionamiento, acercarlo al teléfono móvil y ejecutar la app de configuración. Inmediatamente la app inicia el escaneo e identifica los dispositivos disponibles para configurarlos. Una vez que se haya detectado el beacon deseado aparecerá en pantalla con su configuración de fábrica: nombre iBKS105 adicionalmente su dirección MAC y su perfil de emisión que por defecto es Eddystone URL.

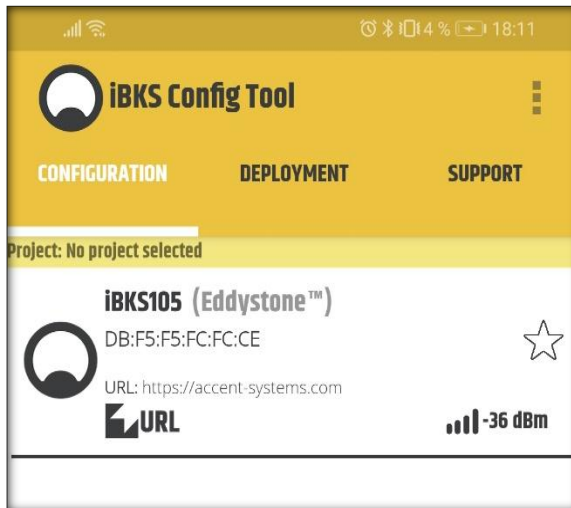


Figura 16. Aplicación de configuración

Para configurarlo se selecciona el dispositivo y la app mostrará los protocolos de emisión (Eddystone, iBeacon), configuración global y las opciones de cambiar el nombre o contraseña de seguridad (figura 17). Es recomendable cambiar el nombre de los dispositivos ya que será una forma rápida de identificarlos lo que facilitará su posterior ubicación en el área de aplicación; y establecer una contraseña de seguridad para restringir el acceso a terceras personas.

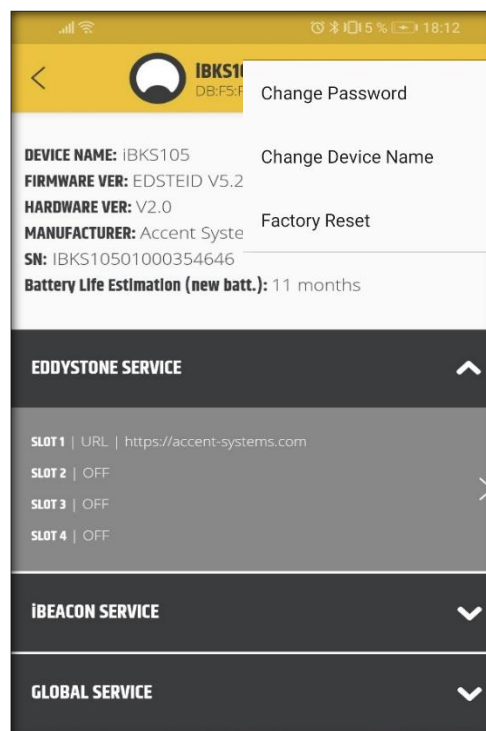


Figura 17. Configuración general

El dispositivo iBKS105 utilizar el SLOT1 con el perfil UID en el protocolo Eddystone por lo que se es recomendable desactivar los SLOTS restantes y el protocolo iBeacon que por defecto vienen activados; se desactivan estas funciones con la finalidad de ahorrar energía de la batería.

En el SLOT1 se establecerá el perfil UID. Posteriormente se debe establecer el intervalo de advertencia en milisegundos, este valor representa el tiempo que existe entre los mensajes emitidos por el dispositivo beacon; se debe tomar en cuenta que si este intervalo es muy reducido el dispositivo transmitirá el UID mayor cantidad de veces lo que representa mayor consumo de la batería; para el presente proyecto todos los beacons tendrán un intervalo de tiempo de 100ms (figura 18).

En el caso de la potencia de transmisión se analiza el rango de cobertura que se quiere lograr con el beacon, en el apartado de diseño de la red se establece que cada dispositivo tendrá una cobertura de 10 metros, el dispositivo cuenta con ocho niveles de potencia de transmisión disponibles, un nivel mayor de potencia asegurará mayor área de cobertura, para el prototipo se seleccionará +4dBm como se muestra en la figura 18.



Figura 18. Configuración Eddystone UID.

Cada valor de potencia de transmisión generará un valor distinto de potencia de recepción en el dispositivo que detecte los beacons; de acuerdo a pruebas de funcionamiento realizadas con +4dBm como potencia tx, en el receptor se obtiene una potencia promedio de -21dBm a cero metros de distancia, este valor servirá como referencia para el cálculo de las distancias y se almacena en el valor Cal. Tx Power. (figura 18)

Por último se establece el valor del identificador UID del beacon, este dato necesariamente debe tener 32 caracteres hexadecimales; se utiliza los primeros 20 caracteres como identificador de Instance, por lo general este valor es el mismo para el grupo de beacons que trabajan en un proyecto, mientras que los últimos 12 corresponden al Namespace y se configuran de tal forma que ninguno de los beacons del proyecto tengan el mismo Namespace.

Para el proyecto el valor de Instance será 00000000000000000004 y el Namespace varía desde su último carácter para cada dispositivo beacon. 000000000000X, como se muestra en la figura 18 el dispositivo1 (iBKSdv1) tendrá como Namespace 000000000001. La tabla 9 muestra un resumen de la configuración que se establece a los dispositivos beacon para la implementación del prototipo.

Tabla 8. Configuración de dispositivos

Nombre	UID	Potencia Tx	Pot. Cal. Rx 0m
iBKSdv1	0000000000000000000040000000000001	+4dBm	-21dBm
iBKSdv2	0000000000000000000040000000000002	+4dBm	-21dBm
iBKSdv3	0000000000000000000040000000000003	+4dBm	-21dBm

2.3.5. Cálculo de distancia usando señales RF

Para el cálculo de distancias entre transmisor y receptor se hará uso del indicador de potencia de la señal recibida RSSI (Received Signal Strength Indicator), los valores de RSSI cambian dependiendo de qué tan alejado se encuentre el dispositivo transmisor del receptor y debido a que no se aplicará en el espacio libre el modelo de propagación debe tener en cuenta las atenuaciones que sufre la señal.

El cálculo de distancias a través de niveles RSSI se trata de un modelo semi-empírico y ha probado ser una buena opción para el cálculo de distancias en algunos sistemas de comunicaciones (wifi, bluetooth, RF) y se encuentra definido por la ecuación 3.

$$RSSI = P_{tx} + G + 20 \log\left(\frac{c}{4\pi f}\right) - 10 \log(d)$$

(Ecuación 3)

En donde P_{tx} es la potencia de transmisión del beacon en dBm (+4dbm), G es la ganancia de las antenas de Tx y Rx, f la frecuencia de transmisión en el caso de BLE 2.44GHz, c velocidad de propagación en vacío 3×10^8 km/s y d que representa la distancia en metros. Este modelo de cálculo de la distancia involucra la Ganancia de las antenas de Tx y Rx datos que pueden variar dependiendo del móvil receptor. Por esta razón se utiliza una ecuación variante para el cálculo RSSI la cual se complementa con mediciones experimentales de atenuación de acuerdo a la ecuación 4.

$$RSSI = A - 10 n \log(d)$$

(Ecuación 4)

En donde A es el valor medido de potencia de recepción a un metro de distancia del dispositivo transmisor sin obstáculos, n el coeficiente de atenuación y d la distancia.

Para el proyecto se ha establecido una potencia de transmisión de +4dBm, colocando el receptor exactamente a 1m de distancia se ha obtenido un promedio de -67dBm valor que tomará A. El coeficiente de atenuación n puede tomar valores entre 2 y 4 se obtiene usando la ecuación 5.

$$n = -\left(\frac{RSSI - A}{10 \log(d)}\right)$$

(Ecuación 5)

Al igual que el valor de potencia de Rx a 1 metro de distancia se deben tomar más muestras a diferentes distancias lo que generará distintos valores de n, estos valores se promediarán para establecer un valor general que será utilizado para todos los cálculos de distancia. En la tabla 10 se muestran los valores de RSSI medidos en el dispositivo de prueba y el valor de n calculado para cada caso.

Tabla 9. RSSI - coeficiente de atenuación

Distancia	RSSI Promedio	Coeficiente de atenuación (n)
2 m	-75.3 dBm	2.7572
3m	-78 dBm	2.3054
4m	-81 dBm	2.3253
5m	-83 dBm	2.2890
6m	-86 dBm	2.4416
7m	-87 dBm	2.3665
8m	-87 dBm	2.2146
9m	-90 dBm	2.4102
10m	-91 dBm	2.4

El promedio de estos valores de n parciales se establecerá como n general.

$$n_g = \frac{n_2 + n_3 + n_4 + n_5 + n_6 + n_7 + n_8 + n_9}{9}$$

$$n_g = 2.3899$$

$$d = 10^{\left(\frac{-67 - RSSI}{10 * 2.389}\right)}$$

(Ecuación 6)

Reemplazando el valor n promedio en la ecuación 4, y despejando la distancia se genera la ecuación 6, la misma que será utilizada por la app y será capaz de realizar el cálculo de la distancia hacia el dispositivo beacon.

2.3.6. Diagrama de flujo de la app

Para facilitar el proceso de codificación de la aplicación móvil Android se ha desarrollado el siguiente diagrama de flujo.

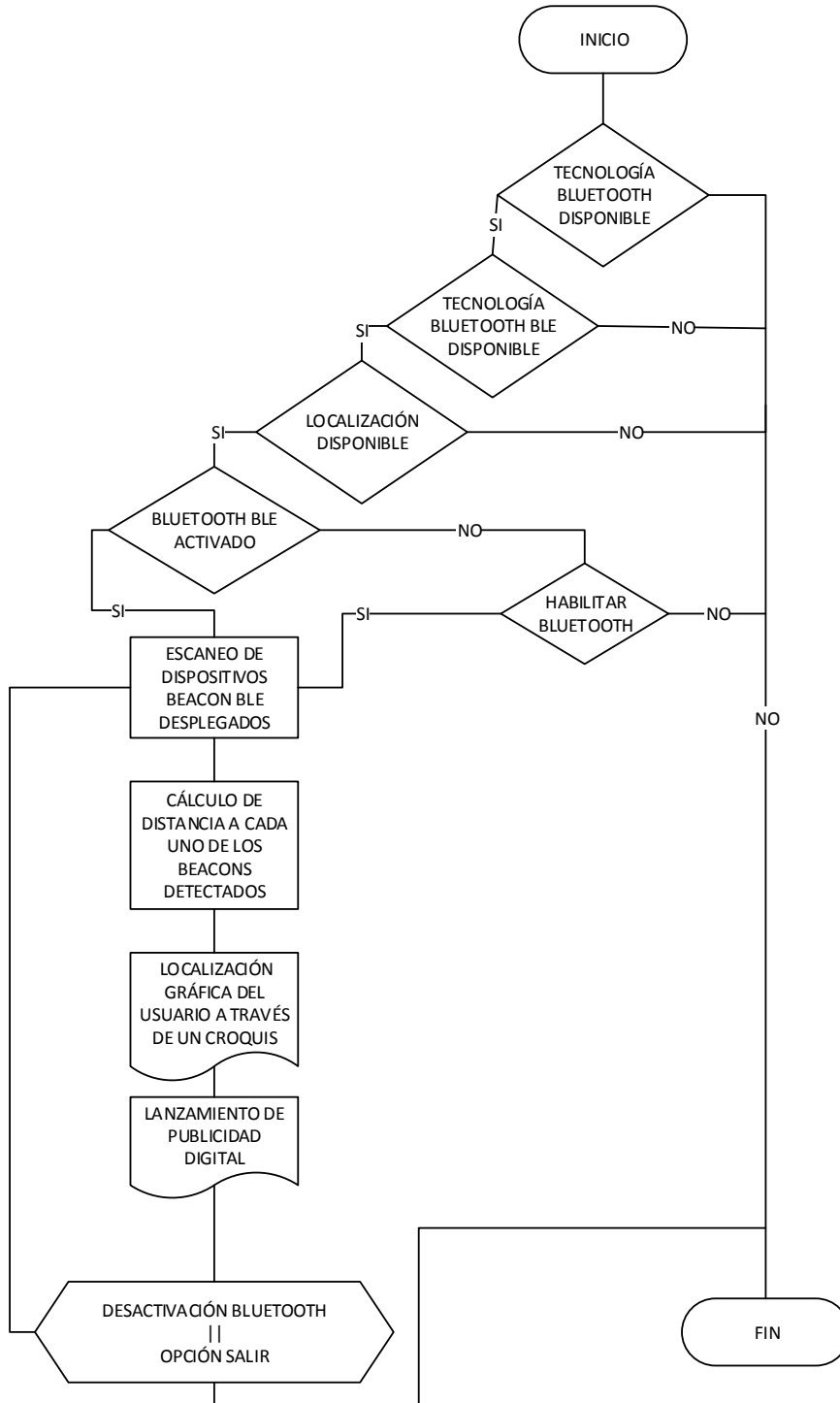


Figura 19. Diagrama de flujo para la app

2.3.7. Desarrollo de la aplicación móvil

Para iniciar con el desarrollo de la aplicación móvil se utilizará el asistente de proyectos de Android Studio, en el cual se le da un nombre al paquete de la aplicación y es de suma importancia establecer el nivel de compilación mínimo que tendrá la app; tal y como se ha manifestado en la fundamentación teórica el S.O. Android a partir de la versión de compilación (API) 21 ha incorporado compatibilidad con tecnología Bluetooth de bajo consumo (BLE), por esta razón al crear el proyecto se selecciona Lollipop 21 como se muestra en la figura 20.

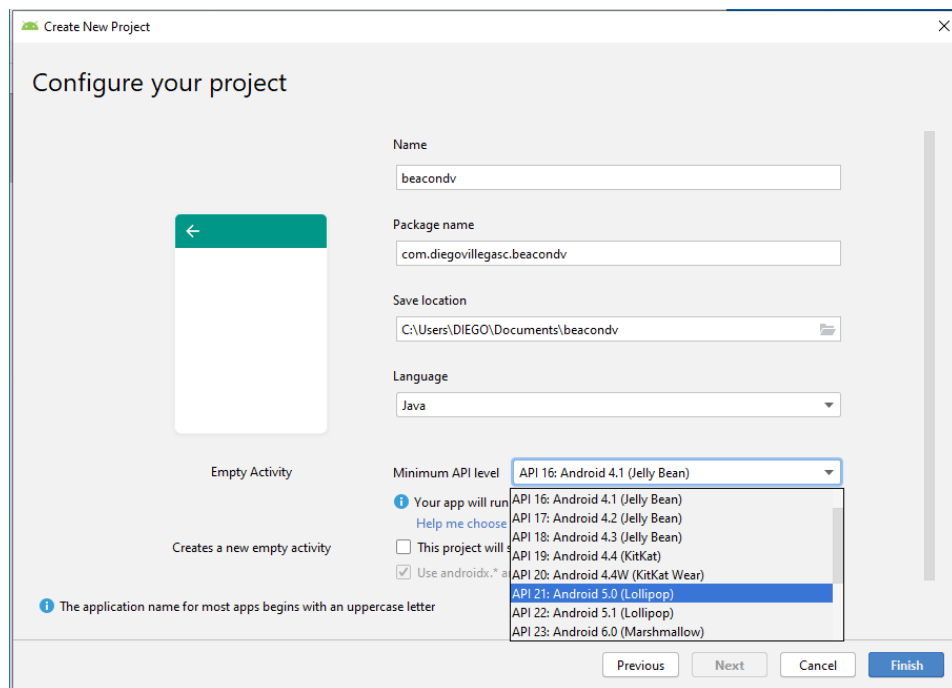


Figura 20. Creación del proyecto en Android Studio

Android Studio se encargará de incorporar las librerías oficiales fundamentales para el desarrollo de la aplicación y realizará una compilación Gradle de prueba, una vez que esta se haya ejecutado esta compilación el proyecto queda estructurado como se muestra en la figura 21. En ella se pueden distinguir cuatro bloques bien definidos: manifests, java, res, y el gradle, en cada uno de estos se llevará a cabo distintas implementaciones de código para el desarrollo de la aplicación.

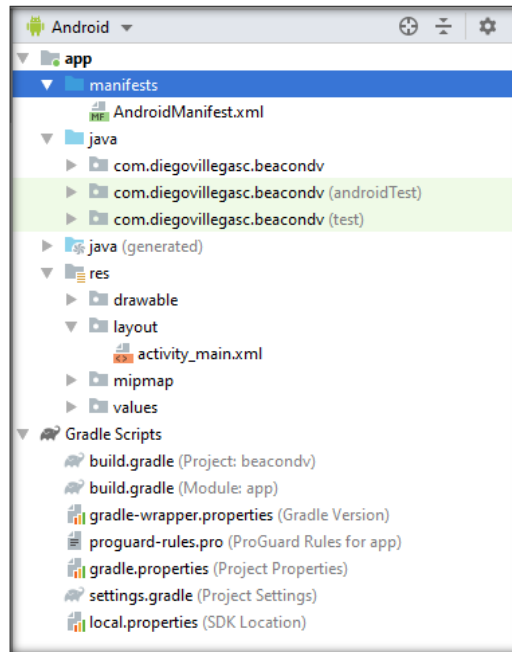


Figura 21. Estructura del proyecto.

Como primer paso es conveniente incorporar las dependencias y permisos necesarios para los propósitos de la aplicación, los cuales se incorporan en el archivo manifests. Para el presente proyecto de investigación se utilizan los permisos de localización, y de conexión bluetooth y estos se añaden como se muestra en la figura 22.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.diegovillegasc.beacondv">

    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

</manifest>
```

Figura 22. Permisos manifests

Además de los permisos para habilitar las funciones de hardware del teléfono se agrega la librería org.altbeacon:android-beacon en su última versión la cual está desarrollada para que el móvil sea compatible con el protocolo GATT y los perfiles de configuración Eddystone, esta dependencia se agrega mediante código en el archivo Gradle a nivel de modulo (figura 23).

```
dependencies {
    implementation fileTree(include: ['*.jar'], dir: 'libs')
    testImplementation 'junit:junit:4.12'
    implementation 'com.android.support:appcompat-v7:27.1.1'
    implementation 'com.android.support:design:27.1.1'
    implementation 'com.android.support:support-v4:27.1.1'
    implementation 'org.altbeacon:android-beacon-library:2.9'
```

Figura 23. Librería Beacon

Siguiendo el diagrama de flujo establecido, a partir de este punto se deben especificar las condiciones de trabajo y control de errores para la inicialización de la aplicación.

- El móvil debe contar con tecnología Bluetooth.
- El móvil debe contar con tecnología Bluetooth (BLE)
- El móvil debe tener habilitada la conexión Bluetooth y de no cumplirse esto la app permitirá la activación desde su interfaz.
- El dispositivo debe permitir el uso de la función de localización.

De no cumplirse alguno de los puntos antes mencionados la app no debe inicializarse, para la comprobación del cumplimiento o no de estas condiciones, el método OnCreate de la actividad principal de la app (MainActivity) realiza las consultas y según sea el caso cierra el flujo de la app o continúa con la ejecución. En la figura 24 se muestran las funciones de comprobación.

```
//Comprueba si el movil cuenta con tecnologia bluetooth
if (!getBaseContext().getPackageManager().hasSystemFeature(PackageManager.FEATURE_BLUETOOTH)) {
    //Mensaje de error
    Toast.makeText(this, "BLUETOOTH NO DISPONIBLE!", Toast.LENGTH_SHORT).show();
    //Si no soporta la tecnologia cierra la app
    finish();
}

//Comprueba si el moviel cuenta con bluetooth Low Energy
if (!getBaseContext().getPackageManager().hasSystemFeature(PackageManager.FEATURE_BLUETOOTH_LE)) {
    //Mensaje de error
    Toast.makeText(this, "BLUETOOTH BLE NO DISPONIBLE!", Toast.LENGTH_SHORT).show();
    //Si no soporta la tecnologia cierra la app
    finish();
} else {
    //Si el movil cuenta con bluetooth se inicia el servicio.
    mBTAdapter = BluetoothAdapter.getDefaultAdapter();
    //El adapter comprueba disponibilidad de conexión
    if (mBTAdapter == null) {
        Toast.makeText(this, "ERROR AL OBTENER BLUETOOTH ADAPTER!", Toast.LENGTH_SHORT).show();
        finish();
    } else {
        //Comprueba que el Bluetooth se encuentre habilitado
        if (!mBTAdapter.isEnabled()) {
            //Si no se encuentra encendido el Bluetooth lanza la actividad de configuración, para que este sea habilitado.
            Intent enableBTintent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
            startActivityForResult(enableBTintent, PERMISSION_REQUEST_COARSE_BL);
        }
    }
}
```

Figura 24. Condiciones para la inicialización de app

Una vez que se comprueban las condiciones de funcionamiento de hardware, la actividad principal habilita la función de iniciar la navegación. Como se muestra en la figura 25.



Figura 25. Interfaz gráfica MainActivity

El botón iniciar navegación lanza la actividad de escaneo de dispositivos BLE Beacon, esta tomara el nombre de NotificationActivity Ya que los dispositivos Beacon se encuentran configurados con el id del perfil UID, la actividad se encargará de escanear los beacons, identificarlos, calcular la distancia del móvil hacia los beacons, y de acuerdo a estas distancias realizar el cálculo de la posición en el espacio físico de análisis y por ultimo desplegar en pantalla el mensaje de Marketing Digital. Para ello se hace uso de métodos o funciones dentro del código, las cuales ayudarán a estructurar el proceso y su ejecución.

La primera función o método programado es el que permite la inicialización del servicio bluetooth en el teléfono móvil, este método se denomina `initBT()`, esta función principalmente establece el modo de escaneo del teléfono móvil este puede ser de baja latencia, o moderada latencia, por motivos de ahorro de energía se establecerá una latencia baja (`SCAN_MODE_LOW_LATENCY`) esta implementación se muestra en la figura 26.

```

private void initBT(){
    //Inicia el servicio bluetooth
    final BluetoothManager bluetoothManager = (BluetoothManager) getSystemService(Context.BLUETOOTH_SERVICE);
    mBluetoothAdapter = bluetoothManager.getAdapter();
    //Creacion de variables de configuracion de escaneo
    ScanSettings.Builder scanSettingsBuilder = new ScanSettings.Builder();
    //Establecer latencia baja.
    scanSettingsBuilder.setScanMode(ScanSettings.SCAN_MODE_LOW_LATENCY);
    scanSettings = scanSettingsBuilder.build();
    scanner = mBluetoothAdapter.getBluetoothLeScanner();
}

```

Figura 26. Inicialización de BLE

Cuando el servicio este levantado los recursos de hardware Bluetooth estarán a disposición de la app, por lo que se puede iniciar con el escaneo de beacons. Esta función se ejecuta a través del método StartLeScan() su codificación se muestra en la figura 27.

```

private void startLeScan(boolean endis) {
    if (endis) {
        //*****
        //INICIA EL ESCANEO
        //*****
        //Se inicia el proceso de deteccion con las configuraciones scanSettings creadas anteriormente
        scanner.startScan(filters: null, scanSettings, mScanCallback);
    }else{
        //Detiene el escaneo
        scanner.stopScan(mScanCallback);
    }
}

```

Figura 27. Función de escaneo

El escaneo iniciará inmediatamente y el móvil tendrá la capacidad de recibir señales pre configuradas para este protocolo, como ya se analizó en apartados anteriores la trama que receipta un móvil tiene una forma de vector o array de bytes y necesariamente debe convertirse en una cadena de caracteres lo cual se realiza mediante la función byteArrayToHex() (figura 28).

```

public static String byteArrayToHex(byte[] a) {
    StringBuilder sb = new StringBuilder(capacity: a.length * 2);
    for(byte b: a)
        sb.append(String.format("%02x", b & 0xff));
    return sb.toString();
}

```

Figura 28. Función de conversión Byte a HEX

La función de conversión de la trama a caracteres tipo hexadecimal permite obtener datos como el identificador UID, la potencia RSSI, el nombre del beacon y su dirección

MAC. Estos valores UID recibidos se comparan con los valores establecidos en el código para determinar el beacon emitió la señal, además del RSSI valor con el que se va a calcular la distancia. Esto se realiza utilizando una función ScanCallback, y es en esta función en la que se realizarán los cálculos de propagación para determinar distancias y la localización del móvil; además de que se identificarán los beacons para lanzar las notificaciones o cuadros de dialogo (showDialog) con las campañas de marketing digital.

```
private ScanCallback mScanCallback = new ScanCallback() {
    @Override
    public void onScanResult(int callbackType, ScanResult result) {
        super.onScanResult(callbackType, result);

        //Convert advertising bytes to string for a easier parsing.
        // GetBytes may return a NullPointerException. Treat it right(try/catch).
        String advertisingString = byteArrayToHex(result.getScanRecord().getBytes());
        //Print the advertising String in the LOG with other device info (ADDRESS - RSSI - ADVERTISING - NAME)
        Log.i(TAG, msg: result.getDevice().getAddress()+" - RSSI: "+result.getRssi()+"\t - "+advertisingString+" - "+result.getDevice().getName());
        Log.i(TAG, msg: "UID = "+ mUuid);
        String dv1="00000000000000000004-000000000001";
        String dv2="00000000000000000004-000000000002";
        String dv3="00000000000000000004-000000000003";

        if (advertisingString.contains(dv1.replace(" ", ""))) {
            detectado=1;
            double dvi_metros=0;

            if (result.getRssi() > -120) {
                if(counter == 0){
                    counter = 2;
                    dvi_metros=Math.pow((double)10, (double)((-75-result.getRssi())/(10*2.2)));
                    DecimalFormatSymbols separadoresPersonalizados = new DecimalFormatSymbols();
                    separadoresPersonalizados.setDecimalSeparator('.');
                    DecimalFormat formato2 = new DecimalFormat( pattern: "#.##", separadoresPersonalizados);
                    pot.setText(String.valueOf(result.getRssi())+" "+String.valueOf(formato2.format(dvi_metros)));
                    startAnim.intererrupt();
                    showDialog();
                }
            }
        }
    }
}
```

Figura 29. ScanCallback, función de cálculos

En la figura 29 se puede apreciar la comparación entre el UID recibido y los valores establecidos en la configuración de dispositivos, además del valor de RSSI resultante en el receptor. En este método se obtiene la distancia del móvil hacia todos los beacons detectados programando la ecuación 6 en código Java.

Al determinar las distancias existentes entre los dispositivos beacon desplegados se puede realizar la trilateración y ubicar un punto gráficamente en la pantalla de la aplicación de escaneo, como se muestra en la figura 30.

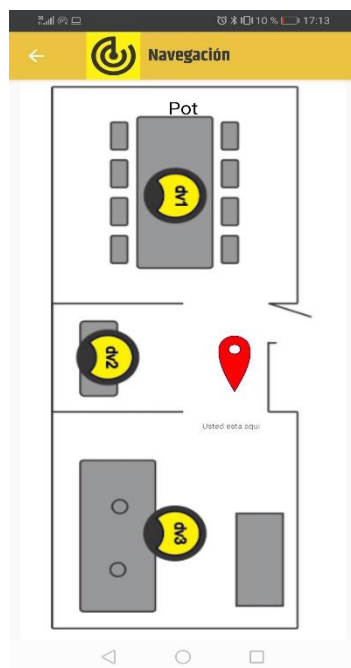


Figura 30. Interfaz gráfica de localización

De acuerdo al enfoque del proyecto, este sistema de localización se aplicará a el lanzamiento de técnicas de marketing digital, para ello una vez que el dispositivo ha logrado geolocalizar un usuario se requiere una actividad adicional la cual estará enfocada netamente a la implementación de marketing. Para ello se implementa la función `showDialog()`, cuya función será el lanzamiento de las actividades correspondientes cuando un beacon se ha detectado en cercanía.

```
private void showDialog(){
    final Dialog dialog = new Dialog( context: NotificationDemo.this);
    dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);
    dialog.setCancelable(false);
    dialog.setContentView(R.layout.custom_dialog_notification);

    TextView text = (TextView) dialog.findViewById(R.id.dialogMsg);
    text.setTypeface(Typeface.createFromAsset(getAssets(), path: "fonts/OpenSans-Light.ttf"));
    ImageView pubC= (ImageView)dialog.findViewById(R.id.pub);
    //The message introduced in settings dialog is showed in this dialog
    text.setText(msj1);
    if (detectado==1){
        pubC.setImageResource(R.drawable.mar1);
    }
    if (detectado==2){
        pubC.setImageResource(R.drawable.mar2);
    }
    if (detectado==3){
        pubC.setImageResource(R.drawable.mar3);
    }
}
```

Figura 31. Codificación de notificaciones

El cuadro de dialogo dialogMsg cuenta con su propia interfaz gráfica, en esta se puede añadir cualquier tipo de atributos de diseño como: videos, imágenes, texto, tablas de contenidos, enlaces web, enlaces a otras Activities e incluso redirigir al usuario a otras aplicaciones etc.; el presente proyecto de investigación se enfocará en utilizar estos recursos para la implementación de técnicas de marketing digital móvil de proximidad.

2.3.8. Marketing de proximidad

El proyecto de investigación se ha enfocado en el desarrollo de un sistema de geolocalización orientado a la optimización de campañas de marketing digital, una vez que la aplicación ha calculado las distancias existentes entre el usuario y el dispositivo beacon y se muestra la ubicación del usuario en el espacio, entonces la app esta apta para implementar campañas de marketing digital de proximidad. Como se ha analizado en la fundamentación teórica existen algunas técnicas para promover las ventas de un producto o servicio, en el proyecto de investigación se hará uso de publicidad tipo Intersticial o de pantalla completa, información que aparecerá en la app de localización cuando el usuario se encuentre realmente cerca de uno de los tres sitios de interés dispuestos en la implementación física de la red. Por ello se ha desarrollado tres anuncios de ejemplo, los cuales serán enviados a los móviles que se encuentren más próximos a los beacons dispuestos en el espacio de análisis. Como se muestra en las figuras 32.



Figura 32. Aplicación de campañas de marketing

El sistema de localización se ejecutará en segundo plano y sus funciones son independientes de la Activity dialogMsg la cual está destinada únicamente a la ejecución de las funciones de marketing.

2.3.9. Pruebas de funcionamiento

Para las pruebas de funcionamiento se ha considerado realizar 20 mediciones con la aplicación a diferentes distancias de forma aleatoria con un intervalo de 30 segundos entre cada medición, se ha desactivado la función de envío de notificaciones de marketing con el fin de únicamente realizar los cálculos de distancia hacia los 3 beacons. En las figuras 33 a la 36 se muestran los valores de distancia obtenidos.

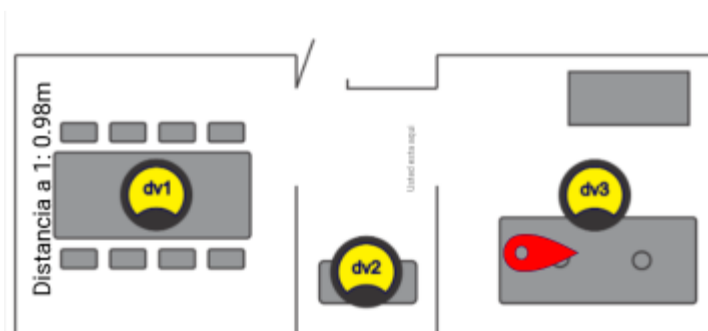


Figura 33. 1ra medición dispositivo 1

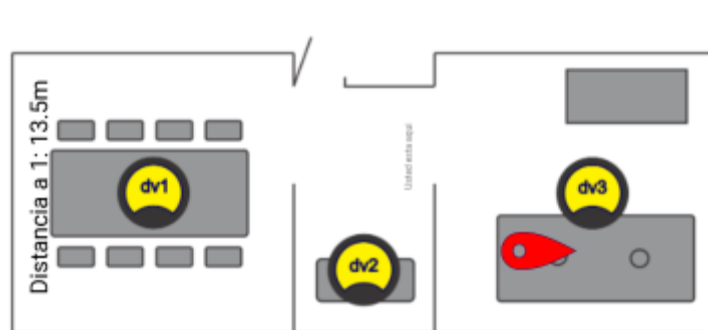


Figura 34. 2da medición dispositivo 1

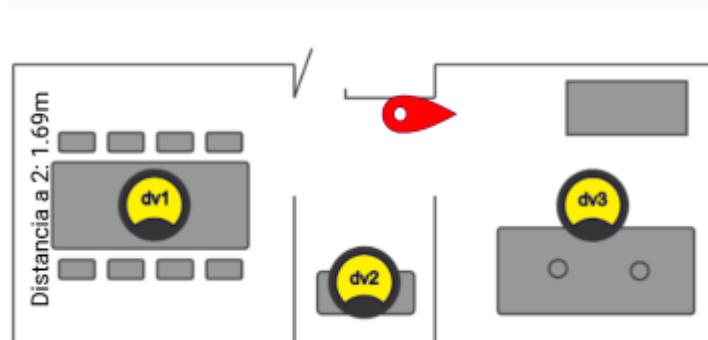


Figura 35. 1ra medición dispositivo 2

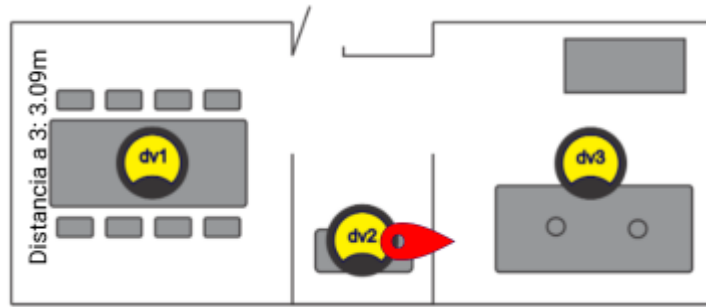


Figura 36. . 1ra medición dispositivo 3

Estas mediciones han sido promediadas para cada dispositivo. En la tabla 10 se muestran los valores promedio de distancia para los 3 beacons desplegados.

Tabla 10. Distancias promedio obtenidas

Distancia	RSSI Promedio	Distancia Calculada
2 m	-77,03	2,62930482
3m	-82,04	4,26139816
4m	-83,7	5,0007743
5m	-86,23	6,38174292
6m	-89,08	8,3991644
7m	-90,18	9,33857158
8m	-91,74	10,8537482
9m	-92,81	12,0328459
10m	-94,64	14,3538844

Una vez realizadas las pruebas de funcionamiento con respecto a la distancia, se realizaron pruebas del lanzamiento de notificaciones de marketing. (Figuras 38 y 39). Para ello se ha establecido mediante programación una proximidad de 1,5 metros cercanía del móvil al beacon para realizar el lanzamiento del cuadro de dialogo de marketing. Se han realizado 5 pruebas dentro del rango de distancia establecido y 5 pruebas fuera del mismo con intervalos de 30 segundos y la aplicación ha respondido de la siguiente manera (tabla 11).

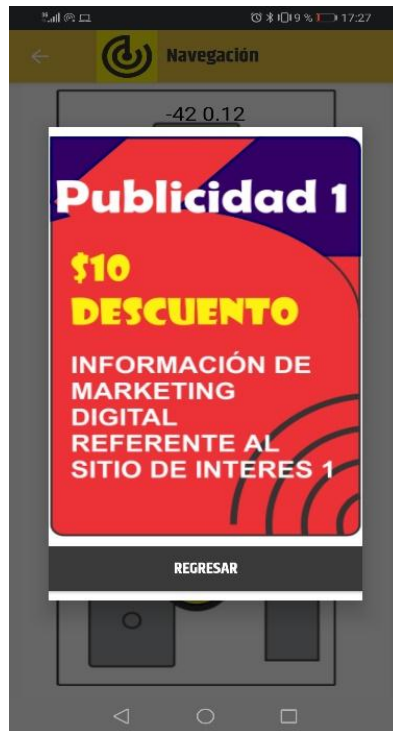


Figura 37. Despliegue de dialogo de marketing 1



Figura 38. Dialogo de marketing 2 desplegada.

Tabla 11. Despliegue de publicidad

Distancia	Muestra cuadro de dialogo (Publicidad)	Distancia	Muestra cuadro de dialogo (Publicidad)
0.20 m	SI-SI-SI-SI-SI	1.70 m	SI-NO-NO-SI-NO
0.25 m	SI-SI-SI-SI-SI	2.00 m	NO-NO-SI-NO-NO
0.40 m	SI-SI-SI-SI-SI	2.50 m	NO-SI-NO-NO-NO
0.50 m	SI-SI-SI-SI-SI	3.00 m	NO-NO-NO-NO-NO
0.75 m	SI-SI-SI-SI-SI	4.00 m	NO-NO-NO-NO-NO
0.60 m	SI-SI-SI-SI-SI	5.00 m	NO-NO-NO-NO-NO
0.80 m	SI-SI-SI-SI-SI	6.00 m	NO-NO-NO-NO-NO
1.00 m	SI-SI-SI-SI-SI	7.00 m	NO-NO-NO-NO-NO
1.25 m	SI-SI-NO-SI-SI	8.00 m	NO-NO-NO-NO-NO
1.50 m	SI-NO-SI-SI-NO	9.00 m	NO-NO-NO-NO-NO

CAPÍTULO III

RESULTADOS Y DISCUSIÓN

3.1. Análisis de datos

Una vez que se han desarrollado las pruebas de funcionamiento con los datos obtenidos se pueden determinar los errores absoluto y relativo que tendrá el sistema de localización.

Tabla 12. Errores absoluto y relativo

Distancia	Distancia Calculada	Error absoluto (metros)	Error relativo (%)
2 m	2,62930482	0,63	31,47
3m	4,26139816	1,26	42,05
4m	5,0007743	1,00	25,02
5m	6,38174292	1,38	27,63
6m	8,3991644	2,40	39,99
7m	9,33857158	2,34	33,41
8m	10,8537482	2,85	35,67
9m	12,0328459	3,03	33,70
10m	14,3538844	4,35	43,54

Se puede observar un claro aumento en el error cuando el smartphone se moviliza fuera del rango de los 5 metros de cercanía de alguno de los beacons. En la figura 35 se ilustra la tendencia del error absoluto calculado, este es relativamente bajo en los primeros metros de cobertura, no obstante puede llegar hasta los 5 metros de en distancias superiores a los 9 metros; en consecuencia las pruebas en movimiento arrojan un error relativo promedio de 34.72% y un error absoluto promedio de 2,14 metros.

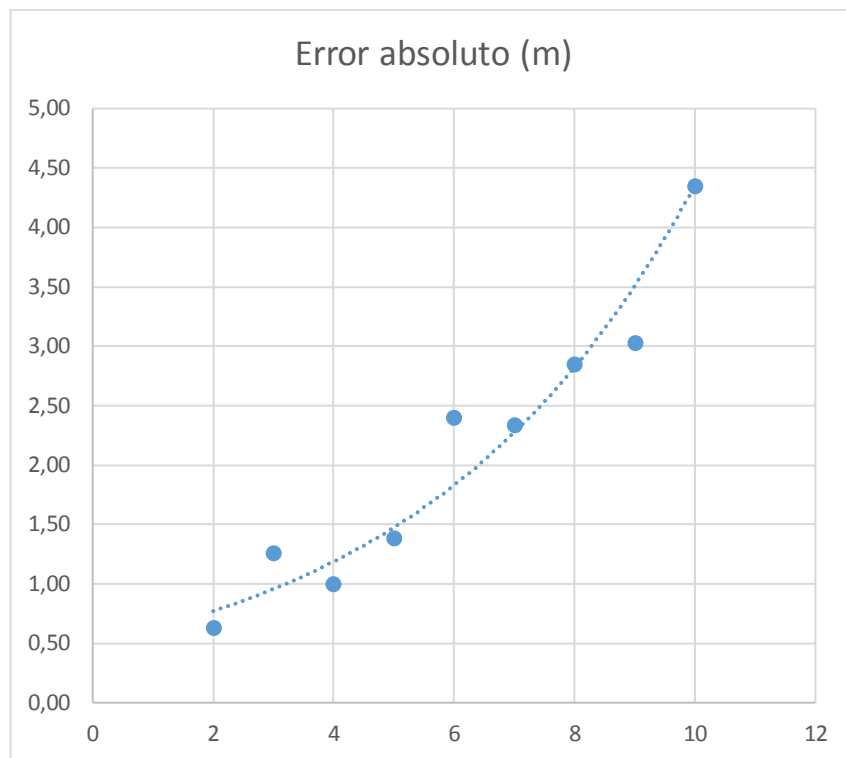


Figura 39. Error absoluto

A partir de los datos obtenidos en las pruebas de funcionamiento acerca del lanzamiento del cuadro de dialogo para marketing digital; se han expresado la aparición o no del cuadro de dialogo en términos de porcentaje con respecto a la distancia en la que el móvil se encuentra para detectar o no su proximidad inmediata hacia el beacon.

La tabla 13 contienen dos análisis, la distancia del móvil dentro del rango programado para la aparición del cuadro de dialogo y la distancia cuando el móvil se encuentra fuera del mismo.

Tabla 13. Despliegue de publicidad

Distancia dentro del rango	Muestra cuadro de dialogo (Publicidad)	Distancia fuera del rango	Muestra cuadro de dialogo (Publicidad)
0.20 m	100%	1.70 m	40%
0.25 m	100%	2.00 m	20%
0.40 m	100%	2.50 m	20%
0.50 m	100%	3.00 m	0%
0.75 m	100%	4.00 m	0%
0.60 m	100%	5.00 m	0%
0.80 m	100%	6.00 m	0%
1.00 m	100%	7.00 m	0%
1.25 m	80%	8.00 m	0%
1.50 m	60%	9.00 m	0%

Dentro del rango de distancia que se ha programado la aparición de notificaciones de marketing (1,5 metros), solo existió ausencia del mismo en 3 ocasiones de un total de 50 muestras, es decir dentro del rango de distancia el 94% de las veces se desplegó correctamente el cuadro de dialogo en el que se incluirá la información de marketing destinada al sitio de interés inmediatamente cercano al beacon (marketing de proximidad).

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- La tecnología Bluetooth BLE resulta ser la más adecuada para localización de dispositivos móviles en interiores, debido a que los beacons son versátiles, fáciles de instalar y principalmente por el ultra bajo consumo de energía el cual no sobrepasa los 2.5uA, lo que le da al sistema una autonomía prolongada de alrededor de 40 meses antes de que se agote la batería.
- La disposición de los beacons y la potencia de Tx se determinaron realizando mediciones de potencia en el Rx, esta se fijó en $+4dBm$ ya que con este valor de potencia no se presentaron mayores fluctuaciones en el receptor consiguiendo $\pm 2.5dBm$. a distancias menores a 5 metros.
- La medición de distancias realizadas mediante el modelo RSSI muestran un error absoluto promedio de 1.61 metros cuando no se presenta excesivo movimiento del terminal receptor.
- Al introducir movimiento, obstáculos o fenómenos físicos al sistema los valores de RSSI muestran grandes variaciones, en consecuencia el margen de error relativo en el cálculo de distancia aumenta de forma drástica al 34,72%, generando un error absoluto promedio de 2.14 metros, este error se refleja aún más notoriamente en distancias cercanas a los 10 metros.
- A pesar del significativo error que presenta la aplicación al momento de realizar los cálculos de distancias entre los 3 beacons, la app muestra un 94% de efectividad al momento de desplegar el cuadro de dialogo que contendrá la información de marketing digital; esta alta efectividad se debe a que el rango de detección hasta 5 metros presenta menores fluctuaciones.

4.2. Recomendaciones

- Se recomienda realizar las mediciones de atenuación en el mismo escenario en el que se implementará físicamente la red, tomando en cuenta todos los factores físicos que podrían intervenir en el entorno, ya que cualquier variación en el área de interés cambiará drásticamente los valores de distancia calculados por el sistema.
- Si se desea mayor precisión se recomienda desplegar un mayor número de transmisores pero con menor potencia de transmisión para que sus señales no se interfieran entre sí, de esta manera sin aplicar el principio de triangulación se puede reducir el margen de error para el cálculo de distancias.
- Para el desarrollo de la aplicación móvil de marketing se recomienda utilizar métodos o funciones que estén separadas del proceso de escaneo e identificación de los beacons, así como también la inclusión de funciones de seguridad para el usuario del dispositivo móvil.
- Desarrollar el código de la app a modo de Service en lugar de desarrollarlo como una Activity, de esta forma la app se ejecutaría en segundo plano así el usuario podrá utilizar su móvil normalmente, abrir otras app al mismo tiempo que navega por el área de interés y sigue recibiendo las notificaciones o cuadros de dialogo destinadas al marketing de proximidad.

C. MATERIALES DE REFERENCIA

Bibliografía

- [1] A. Fagerstrøm, NiklasEriksson y ValdimarSigurdsson, «Investigating the impact of Internet of Things services from a smartphone app on grocery shopping,» *Journal of Retailing and Consumer Services*, vol. 52, 2018.
- [2] G. Jesús, F. Juan, L. Antonio y R. Juan, «Monitoring harness use in construction with BLE beacons,» *Measurement*, vol. 131, pp. 329-340, 2019.
- [3] W. Posdorfer y W. Maalej, «Towards Context-aware Surveys Using Bluetooth Beacons,» *Procedia Computer Science*, vol. 83, 2016.
- [4] D. A. Daniel Mora, «DESARROLLO DE UN PROTOTIPO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES PARA EL ACCESO A INFORMACIÓN TURÍSTICA DETALLADA DE ALGUNOS PUNTOS DE INTERÉS DE LA CIUDAD,» 2015. [En línea]. Available: <https://repository.ucatolica.edu.co/bitstream/10983/2489/1/Documento%20FINAL.pdf>. [Último acceso: 13 09 2019].
- [5] E. Güray y E. Aktaş, «Poilabs,» Poilabs, 2016. [En línea]. Available: <https://www.poilabs.com/about>. [Último acceso: 15 09 2019].
- [6] «Asociación española de centros y parques comerciales,» AECC, 13 06 2012. [En línea]. Available: <https://www.aedecc.com/centros-comerciales/>. [Último acceso: 08 05 2019].
- [7] ICSC, «International Council of Shopping Centers,» ICSC, 2014. [En línea]. Available: <https://www.icsc.com/who-we-are/our-mission>. [Último acceso: 19 08 2019].

- [8] L. A. E. Gómez, Los centros comerciales, espacios postmodernos de ocio y de consumo, Barcelona, 2008.
- [9] A. Fonseca, Marketing Digital, Valencia, 2014.
- [10] A. B. Alonso, I. F. Artime y M. Á. Rodríguez, «UNIVERSIDAD DE OVIEDO,» 26 08 2016. [En línea]. Available: http://isa.uniovi.es/docencia/SIGC/pdf/telefonía_movil.pdf. [Último acceso: 15 06 2019].
- [11] J. M. R. Chillida, «informaTIC,» 04 2016. [En línea]. Available: <http://www.informeticplus.com/que-es-un-smartphone>. [Último acceso: 26 10 2019].
- [12] G. B. López, Geomarketing: geolocalización, redes sociales y turismo, Sevilla, 2014.
- [13] L. I. R. García, «PRINCIPIOS DEL G.P.S.,» Diciembre 1995. [En línea]. Available: <https://upcommons.upc.edu/bitstream/handle/2099/9657/Article007.pdf>. [Último acceso: 13 09 2019].
- [14] A. El-Rabbany, Introduction to GPS: The Global Positioning System, Norwood, 2002.
- [15] L. Letham, GPS fácil. Uso del sistema de posicionamiento global, Barcelona, 2001.
- [16] P. Correira, Guía práctica del GPS, Barcelona, 2002.
- [17] F. Andreu, I. Pellejero y A. Lesta, Fundamentos y Aplicaciones de Seguridad en Redes WLAN: Fundamentos y Aplicaciones de Seguridad, Barcelona, 2006.
- [18] «Wi-Fi alliance,» Wi-Fi ORG, 02 03 2017. [En línea]. Available: https://www.wi-fi.org/downloads-registered-guest/wp_Wi-

- Fi_CERTIFIED_Location_20170222.pdf/33466. [Último acceso: 08 10 2019].
- [19] «Be-Smart,» Wifi Tracker, 12 06 2018. [En línea]. Available: <https://be-smart.io/productos/wifi-tracker/>. [Último acceso: 23 09 2019].
- [20] N. Gupta, *Inside Bluetooth Low Energy*, Boston: Artech House, 2013.
- [21] T. Kevin, C. Carles, Akiba y D. Robert, *Getting Started with Bluetooth Low Energy: Tools and Techniques for Low-Power Networking*, Illinois, 2014.
- [22] K. Townsend, «Adafruit,» 1 Marzo 2019. [En línea]. Available: <https://cdn-learn.adafruit.com/downloads/pdf/introduction-to-bluetooth-low-energy.pdf>. [Último acceso: 20 08 2019].
- [23] Bluetooth SIG, *Bluetooth Core Specification*, Washington, 2019.
- [24] Accent Systems, «Accent Systems,» 15 02 2019. [En línea]. Available: <https://accent-systems.com/es/comprar-beacons/>. [Último acceso: 24 05 2019].
- [25] M. U. b. Aftab, *Building Bluetooth Low Energy Systems*, Birmingham: Packt Publishing, 2017.
- [26] R. Ligeró, «Accent Systems,» 10 01 2018. [En línea]. Available: <https://accent-systems.com/es/blog/8-cosas-que-deberias-saber-sobre-los-beacons/>. [Último acceso: 09 06 2019].
- [27] Accent Systems Beacons, «Accent Systems,» 26 02 2018. [En línea]. Available: <https://accent-systems.com/support/knowledge/getting-started/>. [Último acceso: 20 05 2019].
- [28] J. Cuello y J. Vittone, *Diseñando apps para móviles*, 2013.
- [29] AMC College, *Android Studio: Apps Development*, Sabah: ABSC, 2016.
- [30] Google Developers, «Android developers,» Google, 2017. [En línea]. Available:

<https://developer.android.com/reference/android/bluetooth/BluetoothGatt>.
[Último acceso: 09 26 2019].

- [31] Oracle, «Java,» Oracle, 28 05 2013. [En línea]. Available: https://www.java.com/es/download/faq/whatis_java.xml. [Último acceso: 02 10 2019].
- [32] T. Groussard, JAVA 7: Los fundamentos del lenguaje Java, Barcelona, 2012.
- [33] Oracle Technology Network, «Oracle,» 06 05 2013. [En línea]. Available: <https://www.java.com/es/about/>. [Último acceso: 24 08 2019].

ANEXOS

ANEXO 1. Código actividad principal.

```
import android.Manifest;
import android.annotation.TargetApi;
import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Build;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.Spannable;
import android.text.SpannableString;
import android.text.style.ForegroundColorSpan;
import android.util.Log;
import android.view.View;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    TextView prox;
    TextView notif;

    LinearLayout scanBtn;
    LinearLayout notifBtn;

    private static final int PERMISSION_REQUEST_COARSE_LOCATION = 1;
    private static final int PERMISSION_REQUEST_COARSE_BL = 2;

    //THIS METHOD RUNS ON APP LAUNCH
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        prox = (TextView) findViewById(R.id.proxTV);
        prox.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));
        notif = (TextView) findViewById(R.id.notifTV);
        notif.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));

        SpannableString s = new SpannableString("Navegación
Beacon");

        s.setSpan(new
com.accent_systems.ibkshelloworld.TypefaceSpan(this, "Khand-
Bold.ttf"), 0, s.length(),
Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
        s.setSpan(new
ForegroundColorSpan(Color.parseColor("#3a3c3e")), 0, s.length(),
```

```

Spannable.SPAN_INCLUSIVE_INCLUSIVE);
setTitle(s);

assert getSupportActionBar() != null;
getSupportActionBar().setLogo(R.mipmap.ic_launcher);
getSupportActionBar().setDisplayUseLogoEnabled(true);
getSupportActionBar().setDisplayHomeAsUpEnabled(true);

scanBtn = (LinearLayout) findViewById(R.id.proxBtn);
scanBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent startProxDemoActivity = new
Intent(MainActivity.this, ScanActivity.class);
        startActivity(startProxDemoActivity);
    }
});

notifBtn = (LinearLayout) findViewById(R.id.notifBtn);
notifBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent startNotifDemoActivity = new
Intent(MainActivity.this, NotificationDemo.class);
        startActivity(startNotifDemoActivity);
    }
});

checkLocBT();
initializeBluetooth();
}

private BluetoothAdapter mBTAdapter;

private void initializeBluetooth(){

    //Comprueba si el movil cuenta con tecnologia bluetooth
    if
(!getBaseContext().getPackageManager().hasSystemFeature(PackageManag
er.FEATURE_BLUETOOTH)) {
        //Mensaje de error
        Toast.makeText(this, "BLUETOOTH NO DISPONIBLE!",
Toast.LENGTH_SHORT).show();
        //Si no soporta la tecnologia cierra la app
        finish();
    }

    //Comprueba si el moviel cuenta con bluetooth Low Energy
    if
(!getBaseContext().getPackageManager().hasSystemFeature(PackageManag
er.FEATURE_BLUETOOTH_LE)) {
        //Mensaje de error
        Toast.makeText(this, "BLUETOOTH BLE NO DISPONIBLE!",
Toast.LENGTH_SHORT).show();
        //Si no soporta la tecnologia cierra la app
        finish();
    }else {

```

```

servicio.
        //Si el movil cuenta con bluetooth se inicia el
        mBTAdapter = BluetoothAdapter.getDefaultAdapter();
        //El adapter comprueba disponibilidad de conexión
        if (mBTAdapter == null) {
            Toast.makeText(this, "ERROR AL OBTENER BLUETOOTH
ADAPTER!", Toast.LENGTH_SHORT).show();
            finish();
        }else{
            //Comprueba que el Bluetooth se encuentre habilitado
            if (!mBTAdapter.isEnabled()) {
                //Si no se encuentra encendido el Bluetooth
                lanza la actividad de configuración, para que este sea habilitado.
                Intent enableBTintent = new
                Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
                startActivityForResult(enableBTintent,
                PERMISSION_REQUEST_COARSE_BL);
            }
        }
    }

    @TargetApi(23)
    private void checkLocBT(){
        //Comprueba el permiso de localizacion para API 23 o
        posteriores.
        if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){

            if(this.checkSelfPermission(Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED){
                //Si no se encuentra activada localización lanza la
                actividad de configuración, para que esta sea habilitada.
                requestPermissions(new
                String[]{Manifest.permission.ACCESS_COARSE_LOCATION},
                PERMISSION_REQUEST_COARSE_LOCATION);
            }
        }

        public void onRequestPermissionsResult(int requestCode, String
        permissions[], int[] grantResults) {

            switch (requestCode) {
                case PERMISSION_REQUEST_COARSE_LOCATION: {
                    if (grantResults[0] ==
                    PackageManager.PERMISSION_GRANTED) {

                        Log.d("TAG", "coarse location permission
                        granted");
                    } else {

                        finish();
                    }
                    return;
                }
            }
        }

        @Override
        protected void onActivityResult(int requestCode, int resultCode,

```

```

Intent data) {
    //Si el bluetooth se desactiva en algun instante se cierra
    la aplicacion
    if(requestCode == PERMISSION_REQUEST_COARSE_BL){

        if (resultCode == Activity.RESULT_CANCELED) {
            finish();
            return;
        }
        super.onActivityResult(requestCode, resultCode, data);
    }
}
}

```

Anexo 2. Codificación XML interfaz gráfica principal

```

<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

    <LinearLayout
        android:id="@+id/proxBtn"
        android:layout_width="match_parent"
        android:layout_height="90dp"
        android:layout_weight="1"
        android:elevation="5dp"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginBottom="8dp"
        android:clickable="true"
        android:orientation="horizontal"
        android:layout_gravity="center"
        android:gravity="center"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:background="@drawable/custom_button"
        android:visibility="gone"
    >

        <TextView
            android:id="@+id/proxTV"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:layout_marginTop="10dp"
            android:text="BUSCAR"
            android:textColor="#ffffff"
            android:textSize="35sp" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/layoutfachada"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_marginTop="8dp"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="8dp"
    >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:text="CENTRO COMERCIAL"
        android:layout_marginTop="8dp"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="8dp"
        android:layout_gravity="center_horizontal"
        android:textSize="30sp"
        android:textStyle="bold"
    ></TextView>

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/fachada"
        android:layout_marginTop="8dp"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="8dp"
    >
    </ImageView>

</LinearLayout>
<LinearLayout
    android:id="@+id/notifBtn"
    android:layout_width="match_parent"
    android:layout_height="90dp"
    android:elevation="5dp"
    android:layout_marginTop="8dp"
    android:layout_marginLeft="16dp"
    android:orientation="horizontal"
    android:layout_gravity="center"
    android:gravity="center"
    android:layout_below="@+id/layoutfachada"
    android:layout_centerHorizontal="true"
    android:background="@drawable/custom_button"
>
    <TextView
        android:id="@+id/notifTV"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="10dp"
        android:text="INICIAR NAVEGACIÓN"
        android:textColor="#ffffff"
        android:textSize="30sp" />
</LinearLayout>

</RelativeLayout>

```

Anexo 3. Código actividad Notification

```
package com.diegovillegasc.beacondv;

import android.app.Dialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothManager;
import android.bluetooth.le.BluetoothLeScanner;
import android.bluetooth.le.ScanCallback;
import android.bluetooth.le.ScanResult;
import android.bluetooth.le.ScanSettings;
import android.content.Context;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Bundle;
import android.support.v7.app.ActionBar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.text.Editable;
import android.text.Spannable;
import android.text.SpannableString;
import android.text.TextWatcher;
import android.text.style.ForegroundColorSpan;
import android.util.DisplayMetrics;
import android.util.Log;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;

import java.text.DecimalFormat;
import java.text.DecimalFormatSymbols;

public class NotificationDemo extends AppCompatActivity {

    String TAG = "NotificationDemo";

    private BluetoothAdapter mBluetoothAdapter;
    private BluetoothLeScanner scanner;
    private ScanSettings scanSettings;
    SharedPreferences mPrefs;
    SharedPreferences.Editor edt;
    String mUuid, dRange, mMsg, frameType;
    TextView editBtn;
    TextView pot;
    String msj1="-----";
    int ancho, alto;
    ImageView punto;
    TextView usted;
    int i=1;
    int detectado=0;
```



```

int rx,ry;

double dv1_metros=0;
double dv2_metros=0;
double dv3_metros=0;

double dist = 0.5, counter = 0;
Thread startAnim;

private Context mContext;

@Override
protected void onResume() {
    super.onResume();
    mPrefs = getSharedPreferences("iBKSDemo",
Context.MODE_PRIVATE);

    editBtn = (TextView) findViewById(R.id.editBtn);
    editBtn.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));
    pot = (TextView) findViewById(R.id.potencia);
    editBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startAnim.interrupt();

            showSettingsDialog();
        }
    });

    loadFilter();

    //image animation of scan
    startAnim = new Thread(loopAnimation);
    startAnim.start();

    //init Bluetooth adapter
    initBT();
    //Start scan of bluetooth devices
    startLeScan(true);
}

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    mContext = this;
    setContentView(R.layout.activity_notification);

getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON)
;

    assert getSupportActionBar() != null;
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);

    SpannableString s = new SpannableString("Navegación");

```

```

        s.setSpan(new
com.accent_systems.ibkshelloworld.TypefaceSpan(this, "Khand-
Bold.ttf"), 0, s.length(),
        Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
        s.setSpan(new
ForegroundColorSpan(Color.parseColor("#3a3c3e")), 0, s.length(),
Spannable.SPAN_INCLUSIVE_INCLUSIVE);
        setTitle(s);

        getSupportActionBar().setLogo(R.mipmap.ic_launcher);
        getSupportActionBar().setDisplayUseLogoEnabled(true);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        DisplayMetrics metrics = new DisplayMetrics();
        getWindowManager().getDefaultDisplay().getMetrics(metrics);
        ancho = metrics.widthPixels; // ancho absoluto en pixels
        alto = metrics.heightPixels; // alto absoluto en pixels

        punto = (ImageView) findViewById(R.id.punto);
        usted=(TextView) findViewById(R.id.menmen);

    }

    @Override
    protected void onPause() {
        super.onPause();
        startLeScan(false);
        startAnim.interrupt();
    }

    @Override
    protected void onStop() {
        super.onStop();
        startLeScan(false);
    }

    private void initBT(){
        //Inicia el servicio bluetooth
        final BluetoothManager bluetoothManager =
(BluetoothManager) getSystemService(Context.BLUETOOTH_SERVICE);
        mBluetoothAdapter = bluetoothManager.getAdapter();
        //Creacion de variables de configuracion de escaneo
        ScanSettings.Builder scanSettingsBuilder = new
ScanSettings.Builder();
        //Establecer latencia baja.

scanSettingsBuilder.setScanMode(ScanSettings.SCAN_MODE_LOW_LATENCY);
        scanSettings = scanSettingsBuilder.build();
        scanner = mBluetoothAdapter.getBluetoothLeScanner();
    }

    private void startLeScan(boolean endis) {
        if (endis) {
            //*****
            //INICIA EL ESCANEO
            //*****
            //Se inicia el proceso de deteccion con las
configuraciones scanSettings creadas anteriormente

```

```

        scanner.startScan(null, scanSettings, mScanCallback);
    }else{
        //Detiene el escaneo
        scanner.stopScan(mScanCallback);
    }
}

Thread loopAnimation = new Thread() {
    @Override
    public void run() {
        try {
            sleep(300);
            runOnUiThread(new Runnable() {
                @Override
                public void run() {

                }
            });
            while(true){
                if(counter > 0){
                    counter--;
                }
                sleep(400);
            }
        } catch (InterruptedException e) {
            return;
        }
    }
};

private ScanCallback mScanCallback = new ScanCallback() {
    @Override
    public void onScanResult(int callbackType, ScanResult
result) {
        super.onScanResult(callbackType, result);

        //Convert advertising bytes to string for a easier
parsing.
        // GetBytes may return a NullPointerException. Treat it
right(try/catch).
        String advertisingString =
byteArrayToHex(result.getScanRecord().getBytes());
        //Print the advertising String in the LOG with other
device info (ADDRESS - RSSI - ADVERTISING - NAME)
        Log.i(TAG, result.getDevice().getAddress()+" - RSSI:
"+result.getRssi()+"\t - "+advertisingString+" -
"+result.getDevice().getName());
        Log.i(TAG, "UID = "+ mUuid);
        String dv1="00000000000000000004-000000000001";
        String dv2="00000000000000000004-000000000002";
        String dv3="00000000000000000004-000000000003";

        double n=3;
        double A=-70;
        int dist_inmediata =1;

        if (advertisingString.contains(dv1.replace("-", ""))) {

```

```

detectado=1;

    if (result.getRssi() > -120) {
        if(counter == 0){
            counter = 2;
            dv1_metros=Math.pow((double)10, (double) ((A-
result.getRssi())/(10*n)));
            DecimalFormatSymbols
separadoresPersonalizados = new DecimalFormatSymbols();

separadoresPersonalizados.setDecimalSeparator('.');
DecimalFormat formato2 = new DecimalFormat("#.##",
separadoresPersonalizados);

pot.setText(String.valueOf(result.getRssi())+"
"+String.valueOf(formato2.format(dv1_metros)));
            if(dv1_metros<dist_inmediata){
                startAnim.interrupt();
                showDialog();
            }
        }
    }

if (advertisingString.contains(dv2.replace("-", ""))) {
    detectado=2;

    //pot.setText(String.valueOf(result.getRssi()));
    if (result.getRssi() > -120) {

        if(counter == 0){
            counter = 2;
            dv2_metros=Math.pow((double)10, (double) ((A-
result.getRssi())/(10*n)));
            DecimalFormatSymbols
separadoresPersonalizados = new DecimalFormatSymbols();

separadoresPersonalizados.setDecimalSeparator('.');
DecimalFormat formato2 = new DecimalFormat("#.##",
separadoresPersonalizados);

pot.setText(String.valueOf(result.getRssi())+"
"+String.valueOf(formato2.format(dv2_metros)));
            if(dv2_metros<dist_inmediata){
                startAnim.interrupt();
                showDialog();
            }
        }
    }

if (advertisingString.contains(dv3.replace("-", ""))) {
    detectado=3;

    //pot.setText(String.valueOf(result.getRssi()));
    if (result.getRssi() > -120) {

```

```

        if(counter == 0){
            counter = 2;
            dv3_metros=Math.pow((double)10, (double) ((A-
result.getRssi())/ (10*n)));
            DecimalFormatSymbols
separadoresPersonalizados = new DecimalFormatSymbols();

separadoresPersonalizados.setDecimalSeparator('.');
            DecimalFormat formato2 = new
DecimalFormat("#.##", separadoresPersonalizados);

pot.setText(String.valueOf(result.getRssi())+"
"+String.valueOf(formato2.format(dv3_metros)));

                startAnim.interrupt();
                showDialog();

            }
        }
    }
};

public static String byteArrayToHex(byte[] a) {
    StringBuilder sb = new StringBuilder(a.length * 2);
    for(byte b: a)
        sb.append(String.format("%02x", b & 0xff));
    return sb.toString();
}

private void showDialog(){
    final Dialog dialog = new Dialog(NotificationDemo.this);
    dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);
    dialog.setCancelable(false);
    dialog setContentView(R.layout.custom_dialog_notification);

    TextView text = (TextView)
dialog.findViewById(R.id.dialogMsg);
    text.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/OpenSans-Light.ttf"));
    ImageView pubC= (ImageView)dialog.findViewById(R.id.pub);
    //The message introduced in settings dialog is showed in
this dialog
    text.setText(msj1);
    if (detectado==1){
        pubC.setImageResource(R.drawable.mar1);
    }
    if (detectado==2){
        pubC.setImageResource(R.drawable.mar2);
    }
    if (detectado==3){
        pubC.setImageResource(R.drawable.mar3);
    }
}

```

```

        Button dialogButton = (Button)
dialog.findViewById(R.id.closeBtn);

dialogButton.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));
        dialogButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startAnim = new Thread(loopAnimation);
                startAnim.start();
                dialog.dismiss();
            }
        });

        dialog.show();
    }

    /** Dialog to set the trigger conditions of notification dialog
    **/
    private void showSettingsDialog() {
        final Dialog dialog = new Dialog(NotificationDemo.this);
        dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);
        dialog.setCancelable(false);
        dialog setContentView(R.layout.custom_dialog_settings);

        TextView title = (TextView)
dialog.findViewById(R.id.settTitle);
        title.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));

        TextView uuidTV = (TextView)
dialog.findViewById(R.id.settUuid);
        uuidTV.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));

        final Spinner spinneruid = (Spinner)
dialog.findViewById(R.id.spinneruid);
        title.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));
        spinneruid.setSelection(Integer.parseInt(frameType));

        spinneruid.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> arg0, View
arg1, int arg2, long arg3) {

                final int ft = arg0.getSelectedItemPosition();
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {

                        EditText uuidET = (EditText)
dialog.findViewById(R.id.uuid);
                        String uid;
                        if(ft == 0) {
                            uid = mPrefs.getString("uuid-ibeacon",
"00000000000000000000000000000000-0001-0001");
                        }
                    }
                });
            }
        });
    }

```

```

        else
            uid = mPrefs.getString("uuid-edstuid",
"00000000000000000000-000000000001");

            uuidET.setText(uid);
        }
    });
}
@Override
public void onNothingSelected(AdapterView<?> arg0) {
    //optionally do something here
}
});

final EditText uuid = (EditText)
dialog.findViewById(R.id.uuid);
    uuid.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/OpenSans-Regular.ttf"));

    uuid.setText(mUuid);

    TextView range = (TextView)
dialog.findViewById(R.id.settRange);
    range.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));

    final Spinner dstt = (Spinner)
dialog.findViewById(R.id.distSpinner);
    if (dRange.equals("Immediate")) {
        dstt.setSelection(0);
    } else if (dRange.equals("Near")) {
        dstt.setSelection(1);
    } else {
        dstt.setSelection(2);
    }

    TextView msgg = (TextView)
dialog.findViewById(R.id.settMsg);
    msgg.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));

    final EditText msgt = (EditText)
dialog.findViewById(R.id.msgg);
    msgt.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/OpenSans-Regular.ttf"));
    msgt.setText(mMsg);
    msgt.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence charSequence,
int i, int i2, int i3) {
        }

        @Override
        public void onTextChanged(CharSequence charSequence, int
i, int i2, int i3) {
        }

        @Override
        public void afterTextChanged(Editable editable) {
            if (null != msgt.getLayout() &&

```

```

msgt.getLayout().getLineCount() > 5) {
    msgt.getText().delete(msgt.getText().length() -
1, msgt.getText().length());
    }
    });

    final TextView err = (TextView)
dialog.findViewById(R.id.errorMsg);
    err.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/OpenSans-Light.ttf"));

    Button cancelButton = (Button)
dialog.findViewById(R.id.cancelBtn);

cancelButton.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));
    cancelButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startAnim = new Thread(loopAnimation);
            startAnim.start();
            dialog.dismiss();
        }
    });

    Button saveButton = (Button)
dialog.findViewById(R.id.saveBtn);
    saveButton.setTypeface(Typeface.createFromAsset(getAssets(),
"fonts/Khand-Bold.ttf"));
    saveButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            boolean isOK=true;
            edt = mPrefs.edit();
            if(spinneruid.getSelectedItemPosition() == 0) {
                if (uuid.getText().length() != 42) {
                    isOK=false;
                }
                else
                    edt.putString("uuid-
ibeacon",uuid.getText().toString());
            }else{
                if (uuid.getText().length() != 33){
                    isOK=false;
                }
                else
                    edt.putString("uuid-
edstuid",uuid.getText().toString());
            }

            if(isOK){
                edt.putString("frametype",
Integer.toString(spinneruid.getSelectedItemPosition()));
                edt.putString("range",
dstt.getSelectedItem().toString());
                edt.putString("msg", msgt.getText().toString());
                edt.commit();
                loadFilter();
                err.setVisibility(View.GONE);
            }
        }
    });
}

```



```

        startAnim = new Thread(loopAnimation);
        startAnim.start();
        dialog.dismiss();
    }else {
        err.setVisibility(View.VISIBLE);
    }
    }
});

dialog.show();
}

//get parameters saved on SharedPreferences
private void loadFilter(){
    frameType = mPrefs.getString("frametype","0");

    if(frameType.equals("0"))
        mUuid = mPrefs.getString("uuid-ibeacon",
"00000000000000000000000000000000-0001-0001");
    else
        mUuid = mPrefs.getString("uuid-edstuid",
"00000000000000000000-000000000001");

    dRange = mPrefs.getString("range", "Immediate");
    mMsg = mPrefs.getString("msg", "Hello World from Accent
Systems!");

    if(dRange.equals("Immediate")){
        dist = -38;
    }else if(dRange.equals("Near")) {
        dist = -60;
    }else if(dRange.equals("Far")){
        dist = -120;
    }
}
}
}

```

Anexo 4. Programación interfaz gráfica.

```
<?xml version="1.0" encoding="utf-8" ?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".NotificationDemo">

    <ImageView
        android:id="@+id/notifWave"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="10dp"
        android:src="@drawable/notif_wave"
        android:adjustViewBounds="true"
        android:scaleType="fitCenter"
        android:alpha="0.6"
        android:visibility="invisible"
        android:layout_above="@+id/editBtn"
        android:layout_centerInParent="true"/>

    <ImageView
        android:id="@+id/notifDevice"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:src="@drawable/cro2"
        android:adjustViewBounds="true"
        android:scaleType="fitCenter"
        android:layout_centerInParent="true"/>

    <TextView
        android:id="@+id/editBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="#ffffff"
        android:text="iBKS Settings"
        android:paddingBottom="14dp"
        android:paddingTop="14dp"
        android:paddingLeft="14dp"
        android:paddingRight="14dp"
        android:gravity="center"
        android:textSize="24sp"
        android:clickable="true"
        android:background="@drawable/custom_button"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        android:layout_marginBottom="16dp"
        android:layout_marginTop="5dp"
        android:visibility="gone"
    />

    <TextView
        android:id="@+id/potencia"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="#000000"
        android:text="Distancia a 3: 3.09m "
```

```
android:paddingBottom="14dp"  
android:paddingTop="14dp"  
android:paddingLeft="14dp"  
android:paddingRight="14dp"  
android:gravity="center"  
android:textSize="24sp"  
android:clickable="false"  
android:layout_alignParentTop="true"  
android:layout_centerHorizontal="true"  
android:layout_marginLeft="16dp"  
android:layout_marginRight="16dp"  
android:layout_marginBottom="16dp"  
android:layout_marginTop="5dp"  
android:visibility="visible"  
</>
```

```
<ImageView  
  android:id="@+id/punto"  
  android:layout_width="200px"  
  android:layout_height="200px"  
  android:layout_marginLeft="80dp"  
  android:layout_marginTop="380dp"  
  android:src="@drawable/punto"></ImageView>
```

```
</RelativeLayout>
```

Anexo 5. Implementación de la red beacon.





Se muestra el despliegue de los dispositivos beacon en el área de análisis.

IBKS105 Datasheet



ABSTRACT

iBKS105 Technical Data

AUDIENCE

This document is primarily focused for engineers or other users with a technical profile

FEATURES

- Advertising Beacon Device
- Bluetooth Low Energy®
- Full Eddystone & iBeacon compatible
- 100% Configurable Parameters
- Firmware update **Over The Air** (OTA)
- Waterproof optional kit
- Logo and color customizable (MOQ)
- Provided with your own configuration (MOQ 50 units)
- No tools needed for maintenance
- Additional optional sensors available



Table of Contents

FCC Statement.....	2
IC Statement.....	3
1. Specifications.....	4
2. IBKS105 Mechanical Draws.....	4
3. Battery Life Estimation.....	5
3.1 Estimation Scope.....	5
3.2 Common Use Cases.....	5
4. RX Power (dBm) vs Distance.....	7
Revision History.....	8

FCC Statement

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

FCC Caution:

Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

FCC Statement:

"This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help."

RSS-Gen & RSS-210 statement:

This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes : (1) l'appareil ne doit pas produire de brouillage, et (2)

L'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

RSS-102 Statement:

This equipment complies with Industry Canada radiation exposure limits set forth for an uncontrolled environment.

Cet équipement est conforme à l'exposition aux rayonnements Industry Canada limites établies pour un environnement non contrôlé.

IC Statement

This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes : (1) l'appareil ne doit pas produire de brouillage, et (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

This equipment complies with Industry Canada radiation exposure limits set forth for an uncontrolled environment.

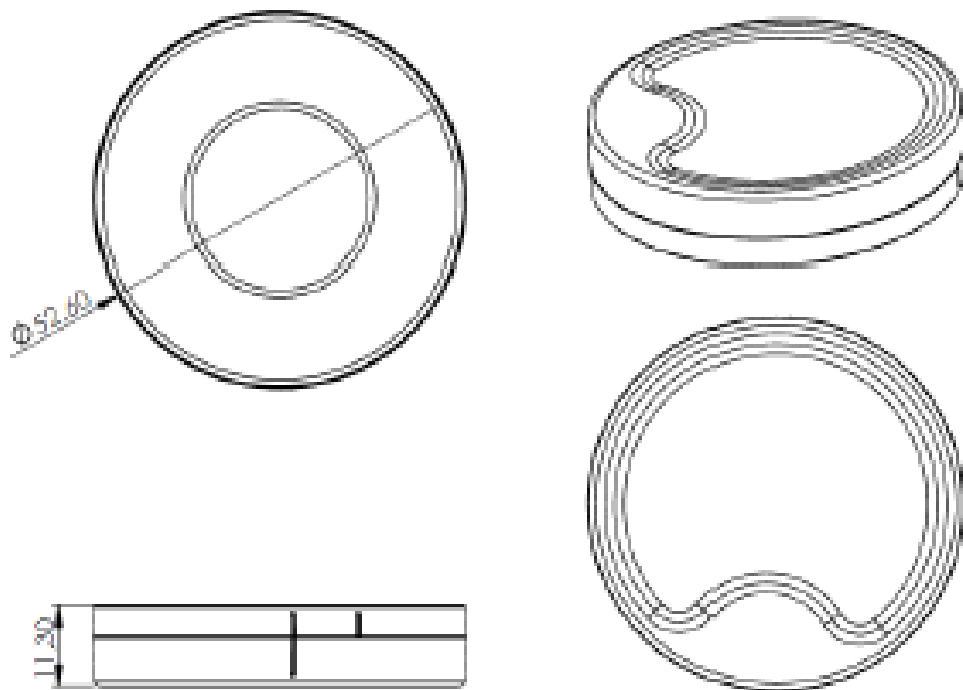
Cet équipement est conforme à l'exposition aux rayonnements Industry Canada limites établies pour un environnement non contrôlé.

1. Specifications

This section contains electrical, mechanical and software specifications for iBKS105

Dimensions	Ø52.6 x 11.3 mm	Case material	ABS
Weight	24g	Case finish	Matte white
Core	Nordic nRF51822	Fixing method	Double side sticker
Radio Protocol	Bluetooth® Low Energy	Operating Temperature	-25 to +60°C
Distance Range	Up to 50m	Storage Temperature	0 to +35°C
Battery	Coin Cell CR2477 3V - 1000mAh	Beacon Protocols	iBeacon Eddystone: UID, URL, TLM & EID
Optional Sensors	Hall Accelerometer	Firmware Update	OTA (Over The Air)
Idle Current Consumption	2.4µA	Certifications	CE, FCC, IC, Anatel

2. iBKS105 Mechanical Draws



All dimensions in mm

3. Battery Life Estimation

IMPORTANT:

Battery life estimation applies from firmware version EDSTEID V5.2016.06.29.1 onwards.

Notice all values shown in this chapter are just an estimation. Real battery life might differ depending on the environment where the iBKS105 is placed.

3.1 Estimation Scope

In order to estimate the battery life, the following configurable items have been considered:















- Number of slots enabled
- Type of slot: iBeacon or Eddystone (UID, URL, TLM and EID)
- TX power defined for each slot
- Advertising period defined for each slot
- Beacon mode: connectable or non-connectable.

3.2 Common Use Cases

The common use cases shown in the table have been estimated under the following conditions:

- All the slots enabled advertise at the same TX power.
- All the slots enabled, except Eddystone Telemetry (TLM), have the same advertising period: 950ms.
- The advertising period for Eddystone TLM is set to 60s.
- The iBKS is in always non-connectable mode.
- EID Rotation Period (K) is set to 10 which means that every 17 minutes approximately a new ephemeral ID is calculated.

☉ The Battery Capacity used is 1000mAh.

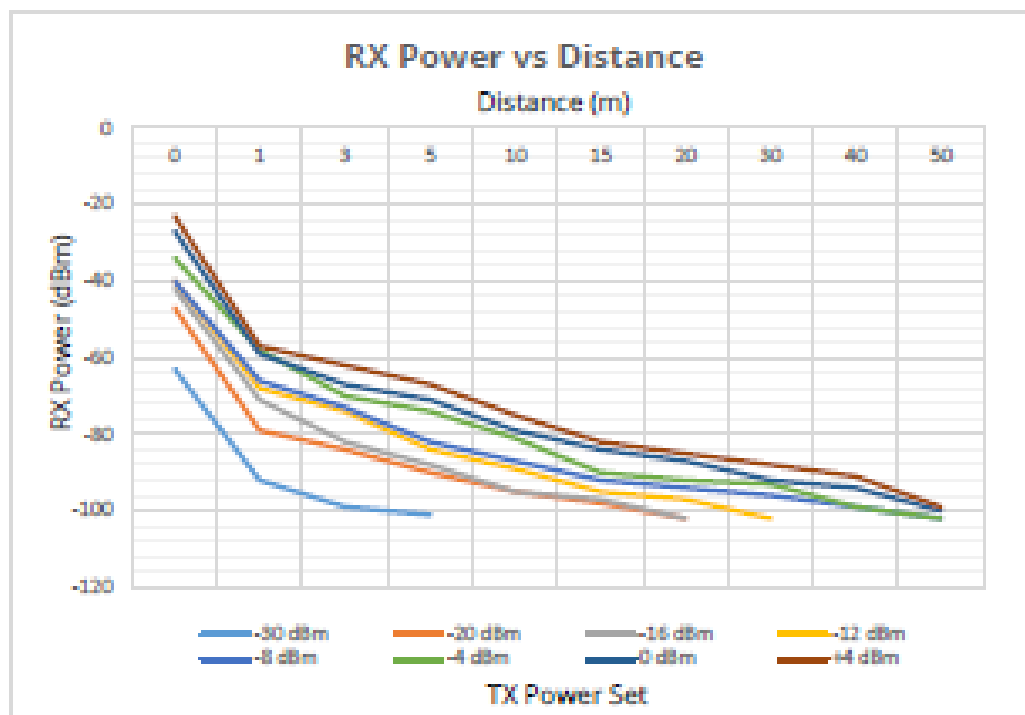
Slots Enabled	Tx Power (dBm)							
	-30	-20	-16	-12	-8	-4	0	+4
	46	46	45	44	43	41	38	31
 UID <small>iddystone</small>	45	45	44	43	42	40	37	31
 URL <small>iddystone</small>	45	45	44	43	42	40	37	31
 EID <small>iddystone</small>	48	48	47	46	45	44	41	34
 TLM <small>iddystone</small>	39	39	39	38	37	36	33	28
 URL <small>iddystone</small>	25	24	24	23	22	22	20	16
 UID TLM <small>iddystone iddystone</small>	39	38	38	37	36	35	32	27
 UID URL <small>iddystone iddystone</small>	24	24	24	23	22	21	19	16
 URL TLM <small>iddystone iddystone</small>	39	38	38	37	36	35	32	27
 EID URL <small>iddystone iddystone</small>	25	25	24	24	23	22	20	17
 EID TLM <small>iddystone iddystone</small>	41	41	40	39	39	37	35	30
 URL TLM <small>iddystone iddystone</small>	24	24	24	23	22	21	20	16
 UID URL TLM <small>iddystone iddystone iddystone</small>	24	24	23	23	22	21	19	16
 EID URL TLM <small>iddystone iddystone iddystone</small>	25	25	24	24	23	22	20	17

Notes: Battery Life in months. Battery Capacity: 1000mAh

4. RX Power (dBm) vs Distance

The following table and graph show the RX power received (dBm) in comparison with distance (m) for all configurable TX powers.

Distance (m)	TX Power (dBm)							
	-30	-20	-16	-12	-8	-4	0	+4
0	-63	-47	-42	-40	-40	-34	-27	-23
1	-92	-79	-71	-68	-66	-58	-59	-57
3	-99	-84	-82	-74	-73	-70	-67	-62
5	-101	-90	-88	-84	-82	-74	-71	-67
10		-95	-95	-89	-87	-81	-79	-75
15		-98	-97	-95	-92	-90	-84	-82
20		-102	-102	-97	-94	-92	-87	-85
30				-102	-96	-93	-92	-88
40					-99	-99	-94	-91
50					-102	-102	-100	-99



IMPORTANT:

This chapter applies from firmware version EDSTEID V5.2016.06.29.1 onwards.

Notice all values shown in this chapter have been tested in an open field of a city. Depending on the environment where the iBKS105 is placed these values might differ.



www.accent-systems.com

Accent Systems

Office
Avda. Francesc Macià 46-51, 7th floor
08208 Sabadell
Barcelona - Spain

Factory
Terra Alta, 1-3 (Pàr. Ind. Carí Corner)
08211 Castellat del Vallès
Barcelona - Spain