

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Trabajo Fin de Máster

**GeoLink: Aplicación web para diseño de radioenlaces en entornos
suburbanos y rurales**



AUTOR: Pedro José Mendoza Rodríguez
DIRECTOR: Fernando Losilla López
Septiembre / 2017

AGRADECIMIENTOS

Quiero agradecer a mi familia, y en especial a mi madre, todo el apoyo incondicional demostrado todos estos años, por compartir tantas penas como alegrías, y por estar ahí siempre que me he caído, para levantarme y conseguir que nunca me rinda.

Por supuesto a todos los profesores que han sido parte de mi formación. Gracias a todos vosotros porque guardo un pedazo de cada uno en mis conocimientos. En especial a Fernando, por toda su ayuda depositada en este proyecto.

También a mi compañero de batallas José Luis. Gracias a ti he aprendido los valores de la constancia, propósito y superación. Desde pequeños has sido un ejemplo de luchador incansable, me has demostrado que la vida está hecha para los valientes, y una vez más tu apoyo me ha servido de gran ayuda para llegar hasta aquí.

Por último a mi pareja Marina, por arrojar un poco de luz en estos momentos difíciles, por ayudarme a sacar las fuerzas necesarias para llegar hasta el final y por todo tu apoyo incondicional que me demuestras a diario.

Índice

Lista de Figuras	6
1-Introducción	9
1.1- Introducción	10
1.2- Motivación.....	10
1.3- Estructura	11
2- Tecnologías empleadas	12
2.1- Tecnologías web	13
2.1.1- HTML5	13
2.1.2- CSS3	14
2.1.3- JavaScript.....	15
2.1.4- PHP	16
2.1.5- MySQL	17
2.2- Radioenlaces.....	18
2.2.1. Aspectos teóricos	19
2.2.1.1- Pérdidas en espacio libre	20
2.2.1.2- Zona de Fresnel	20
2.2.1.3- Atenuación por vegetación	23
2.2.2. Dispositivos y fabricantes	26
2.2.2.1- MikroTik.....	26
2.2.2.2- Cambium Networks.....	29
2.2.2.3- Albetia Sistemas	32
3.- Configuración de la Aplicación	35
3.1- Google Maps API	36
3.1.1- Primeros pasos	38
3.1.2- Google Places API Web Service	40
3.1.3- Google Maps Elevation API	41
3.1.4- Google Maps Geocoding API.....	43
3.1.5- Google Maps Geolocation API.....	44
3.1.5.1- Instalación certificado SSL.....	45
3.1.6- Google Maps Javascript API	47
3.2- Estructura de la página web	49
3.2.1- index.php.....	49
3.2.2- fibra.php	51
3.2.3- fibra2.php	52

3.2.4- nuevoemplazamiento.php	53
3.3.- Base de datos	54
3.3.1- phpmyadmin	54
4.- Resultados	57
4.1- Escenario 1	58
4.2- Escenario 2	60
4.3- Escenario 3	62
5.- Conclusiones y Líneas Futuras.....	63
5.1.- Conclusiones.....	64
5.2.- Líneas Futuras.....	65
6.- Referencias	66

Lista de Figuras

Figura 1- HTML5	13
Figura 2- CSS3	14
Figura 3- Javascript	15
Figura 4- PHP	16
Figura 5- MySql	17
Figura 6 - Estructura general conexión suscriptores-emisor (PMP).....	18
Figura 7- Zona de Fresnel	21
Figura 8 - Pérdidas por difracción.....	22
Figura 9- Trayecto radioeléctrico representativo en zona boscosa	24
Figura 10- Atenuación específica en zona boscosa	24
Figura 11 – MikroTik	26
Figura 12- RBMetal5SHPn.....	26
Figura 13- RFElements Sector Carrier Class 20dBi.....	27
Figura 14- Diagrama radiación RFElements Sector	27
Figura 15- Groove 52HPN	28
Figura 16- Panel direccional TetraAnt5	28
Figura 17- ePMP1000 GPS Sync Radio con Sector Antena	29
Figura 18- Force 180	30
Figura 19- Diagrama de radiación Force180	30
Figura 20- Force 200	31
Figura 21- Diagrama de radiación Force 200.....	31
Figura 22- Albentia AXS-BS-150-60	32
Figura 23- Diagrama de radiación Albentia AXS-BS-150-60	33
Figura 24- AXS-CPE150-15 y AXS-CPE150-RS.....	33
Figura 25- Conector RP-SMA	34
Figura 26- Diagrama de radiación AXS-CPE150-15.....	34
Figura 27- API de Servicios Web	37
Figura 28- Distintos servicios Google Maps API	37
Figura 29- Peticiones gratuitas por servicio de la API	38
Figura 30- Credenciales API	38
Figura 31- Ejemplo monitorización Javascript API	39
Figura 32- Ejemplo uso Google Places API	40
Figura 33- Ejemplo uso del servicio de elevación.....	41
Figura 34- Ejemplo uso geocodificación	43
Figura 35- Ejemplo de geolocalización	44
Figura 36- Geolocalización con HTTPS.....	46
Figura 37- Geolocalización sin HTTPS.....	46
Figura 38- Aspecto página principal	49
Figura 39- Cobertura de Fibra óptica en Totana	51
Figura 40- Herramienta para añadir nuevas calles.....	52
Figura 41- Ruta calles cobertura de fibra	52
Figura 42- Añadir emplazamiento	53
Figura 43- PhpMyAdmin	54
Figura 44- Interfaz de phpMyAdmin	55
Figura 45- Escenario 1 Figura 1.....	58

Figura 46- Escenario 1 Figura 2.....	58
Figura 47- Escenario 2 Figura 1.....	60
Figura 48- Escenario 2 Figura 2.....	61
Figura 49- Escenario 3 Figura 1.....	62
Figura 50- Escenario 3 Figura 2.....	62

1-Introducción

1.1- Introducción

A medida que las nuevas tecnologías van tomando parte en esta sociedad, las necesidades de los usuarios, a su vez, van siendo más exigentes. El uso de las Tecnologías de la Información y la Comunicación (TIC) está a la orden del día.

Estas necesidades nos hacen ser más exigentes con nosotros mismos, más creativos, nos hacen reinventarnos para dar un servicio más novedoso, intuitivo, útil, y a la vez competente.

Sin embargo, los usuarios suelen tener más problemas con la familiarización de dichas TIC, generalmente la tecnología avanza más rápido de lo que ellos pueden aprender. Aquí reside nuestra labor como desarrolladores. Debemos realizar una herramienta más intuitiva y fácil de manejar, de modo que el usuario medio pueda fácilmente acceder a la información y al manejo de la herramienta.

1.2- Motivación

A raíz de las necesidades de la empresa en la que actualmente estoy trabajando, surgió la necesidad de hacer una herramienta personalizada en la que poder estimar la cobertura mediante radioenlace de un cliente, de modo que no hubiese que desplazar un equipo para estimar la viabilidad de una prueba de cobertura en cierta ubicación, y así ahorrarse un desplazamiento y tiempo de trabajo.

Con este proyecto se pretende estimar, utilizando modelos teóricos, los parámetros necesarios para la ubicación de un radioenlace, así como el cálculo de la potencia recibida. De un modo visual y práctico, estimamos si en dicha ubicación existe visión directa con el nodo destino seleccionado.

Posteriormente, llegó a la empresa un nuevo servicio como es la fibra óptica. Con ello, se ha desarrollado otra herramienta con la que, traspasando la información mostrada en los planos de Autocad (ubicación de cajas de abonado, cajas de distribución, etc), el vendedor de tienda pueda de una manera sencilla y visual, saber en qué calles es viable una instalación de fibra óptica o no. De este modo, podrá hacerle un contrato al cliente con las condiciones estipuladas por la empresa sin necesidad de traspasar la orden a un técnico para comprar dicha cobertura.

1.3- Estructura

Este apartado está destinado a ofrecer una visión global de cada uno de los apartados en los que se encuentra dividida la memoria. Este documento se compone de seis capítulos.

El capítulo 1, titulado “Introducción”, ofrece una breve introducción al proyecto, se describe la motivación de este proyecto y se comentan brevemente los distintos servicios que ofrece este proyecto.

En el capítulo 2, “Tecnologías empleadas”, se explican las tecnologías utilizadas para el desarrollo de esta aplicación web, primeramente, las que se han empleado para desarrollar la propia web y, a continuación, las tecnologías que empleadas en el radioenlace y equipamiento.

En el capítulo 3, “Configuración de la Aplicación”, se explica la estructura seguida para la realización del proyecto, así como la funcionalidad de cada uno de los ficheros empleados en esta aplicación web. También se explica la configuración de la base de datos, así como los distintos servicios que nos ofrece la API de Google Maps.

En el capítulo 4, “Resultados”, se presenta un manual de uso, así como resultados prácticos obtenidos con la web. También se adjuntan capturas de dispositivos reales instalados en los emplazamientos de prueba testados.

El capítulo 5, “Conclusiones y Líneas Futuras”, pretende dar una valoración global del proyecto realizado. Además, se exponen algunas ideas futuras para mejorar la aplicación, así como la posibilidad de implementar servicios añadidos.

Y, por último, el capítulo 6 recoge las referencias bibliográficas consultadas para la realización de este TFM.

2- Tecnologías empleadas

2.1- Tecnologías web

2.1.1- HTML5

El lenguaje de Marcado para Hipertextos (HyperText Markup Language, versión 5) [HTML5] es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML, que, a su vez, es el elemento de construcción más básico de una página web y se usa para crear y representar visualmente una página web. HTML5 especifica dos variantes de sintaxis para HTML: una «clásica», HTML (text/html), conocida como HTML5, y una variante XHTML conocida como sintaxis XHTML5 que deberá servirse con sintaxis XML (application/xhtml+xml). Esta es la primera vez que HTML y XHTML se han desarrollado en paralelo.

HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las etiquetas <div> y , pero tienen un significado semántico, como por ejemplo <nav> (bloque de navegación del sitio web) y <footer>.

El lenguaje de marcador HTML, por sí solo, no tiene etiquetas para generar contenido dinámicamente. Aunque proporciona etiquetas para manejo de formularios, el lenguaje no puede procesar la información introducida por el usuario (se envía al servidor, y éste la interpreta y procesa). Para crear páginas web de contenido no estático, hay que complementar HTML con lenguajes y tecnologías adicionales. Esto es lo que hemos hecho en este proyecto, complementar HTML con técnicas como AJAX (Asynchronous JavaScript And XML), ejecutando scripts PHP en el servidor y funciones Javascript en el cliente.



Figura 1- HTML5

2.1.2- CSS3

HTML tiene etiquetas para definir el aspecto de los elementos de la página, pero su uso está desaconsejado. Las Hojas de Estilo en Cascada o CSS (Cascading Style Sheets) son un lenguaje utilizado para definir la presentación de un documento, mediante hojas de estilo. Las hojas de estilo se definen en un documento distinto al HTML, y se referencian desde este último.

Se pueden referenciar varias hojas de estilo desde un único HTML, y hacer que tengan efecto según el dispositivo en el que se visualiza el documento (por ejemplo, si se visualiza desde una tabla o un móvil).

Junto con HTML y JavaScript, CSS es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web, y GUIs para muchas aplicaciones móviles.

CSS [CSS] está diseñado principalmente para marcar la separación del contenido del documento y la forma de presentación de este, características tales como las capas o layouts, los colores y las fuentes. Esta separación busca mejorar la accesibilidad del documento, proveer más flexibilidad y control en la especificación de características presentacionales, permitir que varios documentos HTML compartan un mismo estilo usando una sola hoja de estilos separada en un archivo .css, y reducir la complejidad y la repetición de código en la estructura del documento.

En nuestro caso, para el diseño de las hojas de estilo, nos hemos apoyado del framework Bootstrap.



Figura 2- CSS3

2.1.3- JavaScript

JavaScript [JAVASCRIPT] (abreviado comúnmente JS) es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

JavaScript se diseñó con una sintaxis similar a C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo, Java y JavaScript tienen semánticas y propósitos diferentes.

En comparación con Java, JavaScript es un lenguaje muy libre de forma. No hay que declarar todas las variables, clases, y métodos. No hay que preocuparse de si los métodos son públicos, privados, o protegidos, y no hay que implementar interfases. Las variables, parámetros, y retornos de funciones no tienen que declararse explícitamente de un tipo dado.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. Actualmente es ampliamente utilizado para enviar y recibir información del servidor junto con ayuda de otras tecnologías como AJAX. JavaScript se interpreta en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML.



Figura 3- Javascript

2.1.4- PHP

PHP [PHP] es un acrónimo recursivo que significa PHP Hypertext Preprocessor (inicialmente PHP Tools, o, Personal Home Page Tools). Es un lenguaje de programación de uso general de código del lado del servidor, originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún coste.

PHP puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores. El número de sitios basados en PHP se ha visto reducido progresivamente en los últimos años, con la aparición de nuevas tecnologías como Node.JS, Golang, ASP.NET, etc.

El intérprete de PHP solo ejecuta el código que se encuentra entre sus delimitadores. Los delimitadores más comunes son `<?php` para abrir una sección PHP y `?>` para cerrarla. El propósito de estos delimitadores es separar el código PHP del resto de código, como por ejemplo el HTML.

Las variables se prefijan con el símbolo del dólar (\$) y no es necesario indicar su tipo. Las variables, a diferencia de las funciones, distinguen entre mayúsculas y minúsculas. Las cadenas de caracteres pueden ser encapsuladas tanto en dobles comillas como en comillas simples, aunque en el caso de las primeras, se pueden insertar variables en la cadena directamente, sin necesidad de concatenación.

Los comentarios se pueden escribir bien con dos barras al principio de la línea, o con una almohadilla. También permite comentarios multi-línea encapsulados en `/* */`.

En cuanto a las palabras clave, PHP comparte con la mayoría de otros lenguajes con sintaxis C las condiciones con `if`, los bucles con `for` y `while` y los retornos de funciones. Como es habitual en este tipo de lenguajes, las sentencias deben acabar con punto y coma (;).



Figura 4- PHP

2.1.5- MySQL

MySQL [MYSQL] es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

El sistema de base de datos operacional MySQL es hoy en día uno de los más importantes en lo que hace al diseño y programación de base de datos de tipo relacional. Cuenta con millones de aplicaciones y aparece en el mundo informático como una de las más utilizadas por usuarios del medio. El programa MySQL se usa como servidor a través del cual pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo.

La historia del MySQL (cuya sigla en inglés se traslada a My Structured Query Language o Lenguaje de Consulta Estructurado) se remite a principios de la década de 1980. Programadores de IBM lo desarrollaron para contar con un código de programación que permitiera generar múltiples y extendidas bases de datos para empresas y organizaciones de diferente tipo. Desde esta época numerosas versiones han surgido y muchas de ellas fueron de gran importancia. Hoy en día MySQL es desarrollado por la empresa Sun Microsystems.

Una de las características más interesantes de MySQL es que permite recurrir a bases de datos multiusuario a través de la web y en diferentes lenguajes de programación que se adaptan a diferentes necesidades y requerimientos.



Figura 5- MySqI

2.2- Radioenlaces

En este apartado vamos a explicar los aspectos a tener en cuenta a la hora de diseñar un radioenlace, ya sea punto a punto (PtP, Pt2Pt o sus variaciones) o punto a multipunto (a menudo abreviado como P2MP, PTMP, o PMP).

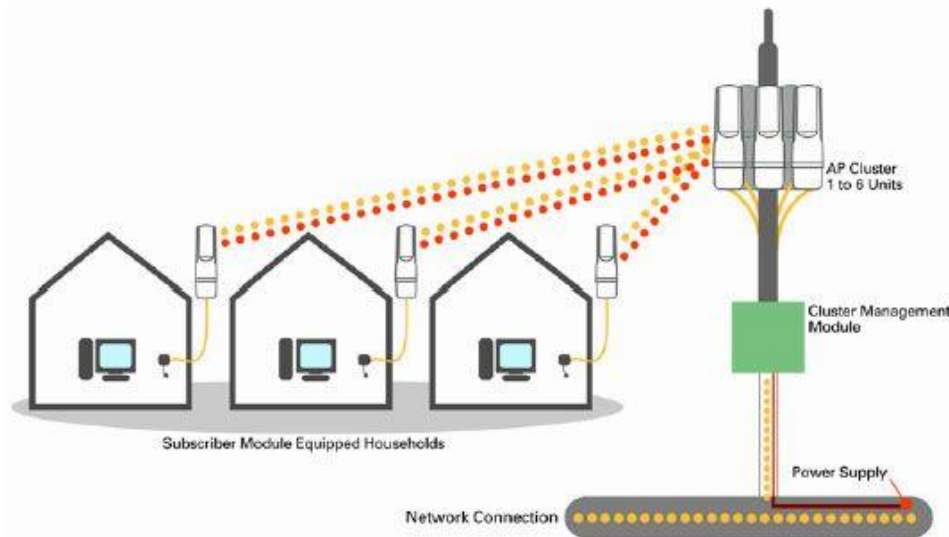


Figura 6 - Estructura general conexión suscriptores-emisor (PMP)

El término de **punto-a-punto [PTP]** se refiere a las telecomunicaciones inalámbricas entre dos terminales de comunicaciones de datos para Internet o Voz sobre IP a través de frecuencias de radio en la gama de varios gigahercios. También incluye tecnologías como el láser para las telecomunicaciones, pero en la mayoría de ocasiones el medio de transmisión es la línea de visión.

La señal de telecomunicaciones suele ser bidireccional, ya sea de acceso múltiple por división de tiempo (TDMA) o canalizado.

Punto a multipunto [PMP] de comunicación es un término que se utiliza en el ámbito de las telecomunicaciones, que se refiere a la comunicación que se logra a través de un específico y distinto tipo de conexión multipunto, ofreciendo varias rutas desde una única ubicación a varios lugares. Una conferencia puede ser considerada una comunicación punto a multipunto ya que existe solo un orador (transmisor) y múltiples asistentes (receptor).

El punto a multipunto de telecomunicaciones es el más típico utilizado en conexión inalámbrica a Internet y la telefonía IP a través de radiofrecuencias de gigahercios. Los sistemas P2MP han sido diseñados tanto como sistemas unidireccionales como bidireccionales. Un equipo o equipos que reciben las emisiones de varios transmisores y en conjunto utilizan una técnica de multiplexación por división en el tiempo para permitir el retorno de canales de tráfico.

En este proyecto vamos a trabajar con dispositivos que trabajan en la banda de los 5GHz, con un rango de frecuencias aproximado entre 4900-5875 MHz.

2.2.1. Aspectos teóricos

Entre los aspectos teóricos a tener en cuenta a la hora de diseñar un radioenlace, nos vamos a centrar en la frecuencia de funcionamiento, para estimar las posibles pérdidas en la transmisión, PIRE (potencia de transmisión del dispositivo emisor + ganancia de la antena emisora - pérdidas posibles debidas a la longitud del cable), la ganancia de la antena/dispositivo receptor o CPE (Equipo Local del Cliente) y por último en la altura estimada de éste último, para calcular el ángulo de inclinación del CPE.

Uno de los principales conceptos a la hora de diseñar un radioenlace son las pérdidas de propagación. Se define [ITU-R P.341-4] pérdida de transmisión (de un enlace radioeléctrico, expresado como L o A) como la relación, habitualmente expresada en decibelios, entre la potencia suministrada por el transmisor de un enlace radioeléctrico y la potencia suministrada al receptor correspondiente, en las condiciones reales de instalación, propagación y explotación.

La pérdida de propagación, para un trayecto terrenal con visibilidad directa, respecto a las **pérdidas en el espacio libre** es la suma de las contribuciones siguientes:

- atenuación debida a los gases atmosféricos,
- desvanecimiento por difracción debido a la obstrucción parcial o total del trayecto,
- desvanecimiento debido a la propagación por trayectos múltiples, la dispersión del haz y el centelleo,
- atenuación debida a la variación de los ángulos de llegada y de salida,
- atenuación debida a las precipitaciones,
- atenuación debida a las tormentas de arena y polvo.

Cada una de estas contribuciones tiene sus características propias en función de la frecuencia, la longitud del trayecto y la ubicación geográfica.

Estas pérdidas, salvo la segunda, podemos suponerlas nulas debido a que los dispositivos que utilizaremos disponen de control automático de potencia (ATPC) para regular este tipo de pérdidas y aproximarlas lo máximo posible a cero, además de reutilización de frecuencias para poder reducir interferencias y tener menos ruido.

En el siguiente apartado nos centraremos en las **pérdidas en espacio libre**, que son las pérdidas de transmisión para el trayecto de propagación de un rayo determinado, siendo igual a la pérdida básica de transmisión menos las ganancias de las antenas transmisora y receptora en las direcciones del trayecto del rayo.

2.2.1.1- Pérdidas en espacio libre

Tal y como podemos observar [ITU-R P.525-2], hay diferentes formas de calcular las pérdidas en espacio libre dependiendo al tipo de servicio particular que nos interese.

Cuando se trata de un enlace punto a punto, es preferible calcular la atenuación en el espacio libre entre antenas isotropas, denominada también pérdida básica de transmisión en el espacio libre (símbolos: L_{bf} o A_0) de la manera siguiente (1):

$$L_{bf}(\text{dB}) = 20 \cdot \log\left(\frac{4 \pi d}{\lambda}\right)$$

Dónde:

L_{bf} : pérdida básica de transmisión en el espacio libre (dB)

d : distancia

λ : longitud de onda

d y λ se expresan en las mismas unidades.

La ecuación anterior puede también escribirse en función de la frecuencia en vez de la longitud de onda (2):

$$L_{bf}(\text{dB}) = 32,4 + 20 \log f + 20 \log d$$

Dónde:

f : frecuencia (MHz)

d : distancia (km).

En nuestro caso vamos a utilizar (2), ya que la API nos proporciona la distancia entre dos puntos cualesquiera en Km, y la frecuencia es un parámetro de entrada que se pide al usuario y viene expresada en MHz.

2.2.1.2- Zona de Fresnel

Se llama zona de Fresnel [FRESNEL] al volumen de espacio entre el emisor de una onda -electromagnética, acústica, etc.- y un receptor, de modo que el desfase de las ondas en dicho volumen no supere los 180° .

La obstrucción máxima permisible para considerar que no hay obstrucción es el 40% de la primera zona de Fresnel. La obstrucción máxima recomendada es el 20%. Para el caso de radiocomunicaciones depende del factor K (curvatura de la tierra)

considerando que para un $K=4/3$ la primera zona de fresnel debe estar despejada al 100% mientras que para un estudio con $K=2/3$ se debe tener despejado el 60% de la primera zona de Fresnel.

Para establecer las zonas de Fresnel, primero debemos determinar la línea de vista de RF, que de forma simple, es la línea recta que une los focos de las antenas transmisora y receptora.

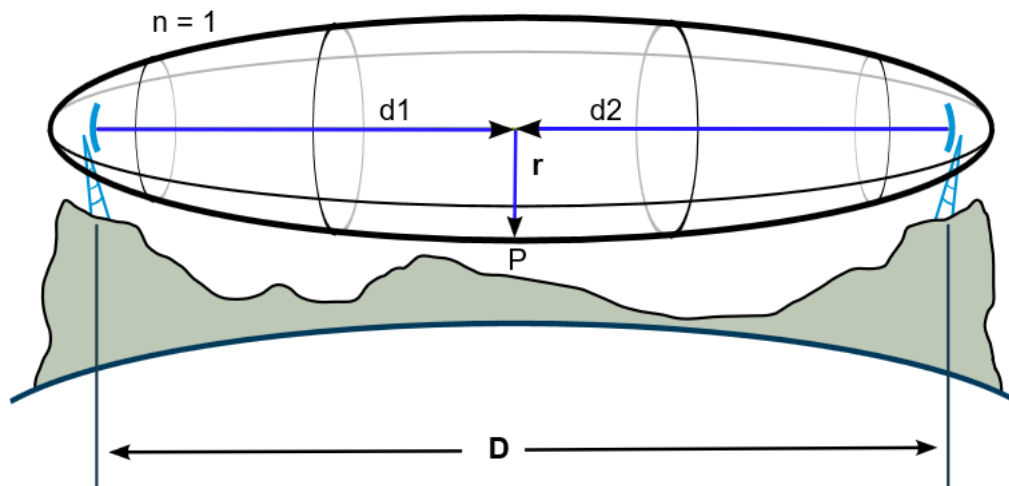


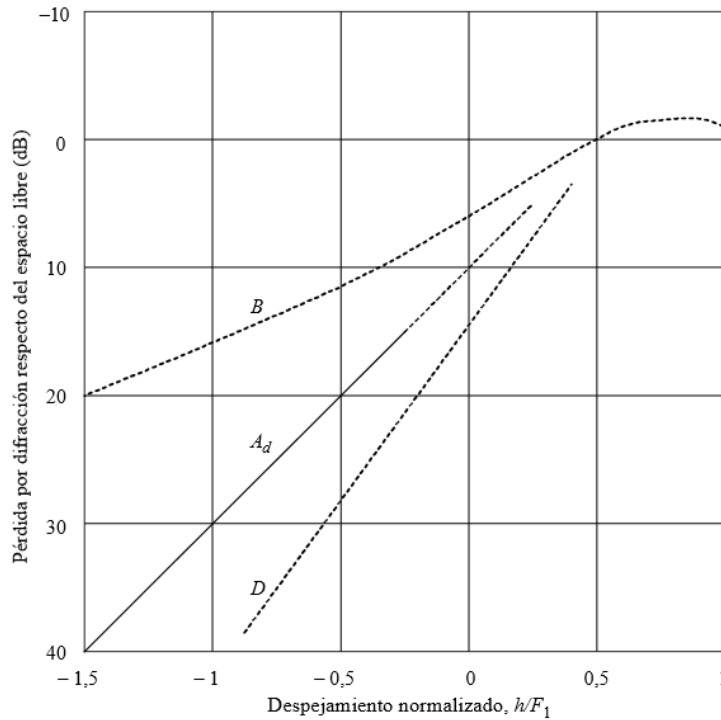
Figura 7- Zona de Fresnel

La pérdida por difracción [ITU-R P.530-7] dependerá del tipo de terreno y de la vegetación. Para un determinado despejamiento del trayecto del rayo, la pérdida por difracción variará desde un valor mínimo en el caso de un obstáculo único en arista (filo de cuchillo) hasta un valor máximo en el caso de una tierra esférica lisa. En la Recomendación UIT-R P.526 se examinan métodos para el cálculo de la pérdida por difracción en esos dos casos y también en el de trayectos a través de terreno irregular. En la Fig. 8 se muestran esos límites superior e inferior de la pérdida por difracción.

Las pérdidas por difracción en un terreno medio pueden aproximarse, para pérdidas mayores de unos 15 dB, mediante la fórmula:

$$A_d \text{ (dB)} = - 20 h / F_1 + 10 \quad (3)$$

Pérdida por difracción en el caso de existir obstáculos en trayectos radioeléctricos de microondas con visibilidad directa



- B*: curva teórica de pérdida por difracción en obstáculos en filo de cuchillo
- D*: curva teórica de pérdida sobre la tierra esférica lisa a 6,5 GHz y $k_e = 4/3$
- A_d*: curva empírica de la pérdida por difracción basada en la ecuación (2) para terreno intermedio
- h*: magnitud del despejamiento del trayecto radioeléctrico respecto de la superficie de la Tierra
- F₁*: radio de la primera zona de Fresnel

Figura 8 - Pérdidas por difracción

En la que *h* es la altura (m) del obstáculo más importante del trayecto por encima de la trayectoria de éste (*h* es negativa si la parte superior del obstáculo en cuestión está por encima de la línea de visibilidad directa), y *F1* es el radio del primer elipsoide de Fresnel (*r* en la Figura 7), indicado por (4):

$$F1 \text{ (m)} = 17,3 \sqrt{\frac{d1 \ d2}{f \ d}}$$

Dónde:

f: frecuencia (GHz)

d: longitud del trayecto (km)

d1 y *d2*: distancias (km) entre los terminales y la obstrucción del trayecto.

En la Fig. 8 se muestra también una curva, denominada A_d , basada en la ecuación (3). Esta curva, estrictamente válida para pérdidas superiores a 15 dB, se ha extrapolado hasta una pérdida de 6 dB para satisfacer la necesidad de los diseñadores de enlaces.

Este radio F_1 de Fresnel se calcula atendiendo a la recomendación [ITU-R P.526], evaluando la ecuación para $n=1$ (6):

$$R_n (m) = \sqrt{\frac{n \lambda d_1 d_2}{d_1 + d_2}}$$

Donde f es la frecuencia (MHz) y d_1 y d_2 son las distancias (km) desde el transmisor y desde el receptor al punto en que se evalúa el radio (m) del elipsoide.

2.2.1.3- Atenuación por vegetación

En ciertos casos, la atenuación debida a la vegetación puede ser importante, tanto para los sistemas terrenales como para los sistemas Tierra-espacio. Pero la gran diversidad de condiciones y tipos de follaje dificultan la elaboración de un procedimiento de predicción general. Además, existe una falta de datos experimentales convenientemente verificados.

Para un trayecto radioeléctrico terrenal [ITU-R P.833-8], uno de cuyos terminales está situado en un bosque o en una zona similar de vegetación extensa, la pérdida adicional debida a la vegetación puede describirse en base a dos parámetros:

- el índice de atenuación específica (dB/m) debida fundamentalmente a la dispersión de energía fuera del trayecto radioeléctrico, que se mediría en un trayecto muy corto; 2 Rec. UIT-R P.833-8

- la atenuación adicional total máxima debida a la vegetación en un trayecto radioeléctrico (dB) limitada por el efecto de otros mecanismos, entre ellos, la propagación de ondas de superficie por encima del medio vegetal y la dispersión dentro del mismo.

En la Fig. 9, el transmisor está fuera de la zona boscosa y el receptor está a una cierta distancia, d , dentro de la misma. La atenuación excesiva, A_{ev} , debida a la presencia de la vegetación viene dada por:

$$A_{ev} (dB) = A_m [1 - \exp(-d \gamma / A_m)] \quad (7)$$

siendo:

d : longitud del trayecto dentro de la zona boscosa (m)

γ : atenuación específica para trayectos en vegetación muy cortos (dB/m)

A_m : atenuación máxima cuando un terminal está dentro de una zona de vegetación de un tipo y profundidad específicos (dB).

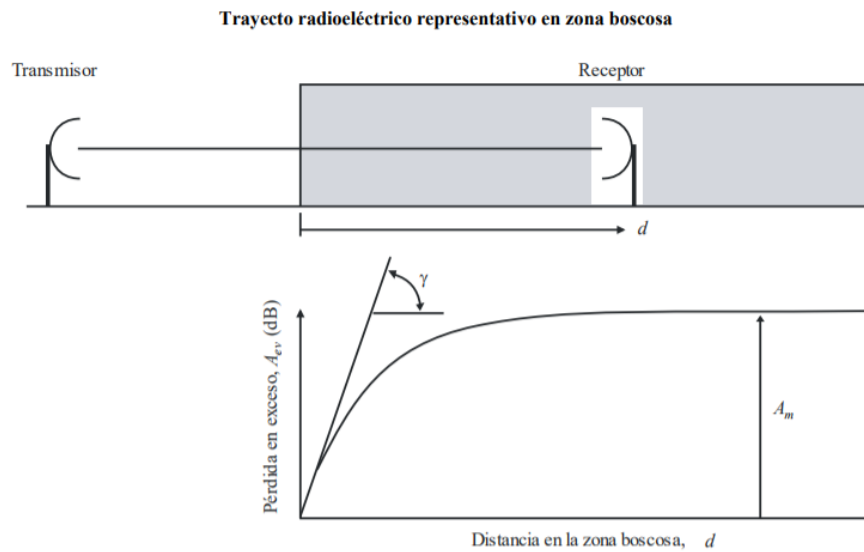


Figura 9- Trayecto radioeléctrico representativo en zona boscosa

El valor de la atenuación específica debida a la vegetación, γ dB/m, depende de la especie y la densidad de dicha vegetación. En la Fig. 10 se proporcionan valores aproximados de atenuación específica en función de la frecuencia obtenidos a partir de diferentes mediciones en la gama de frecuencias que va de 30 MHz a 30 GHz aproximadamente en zona boscosa. Por debajo de 1 GHz las señales polarizadas verticalmente tienen tendencia a experimentar una atenuación superior a la que experimentan las polarizadas horizontalmente, aunque esto se debe a la dispersión causada por los troncos de los árboles.

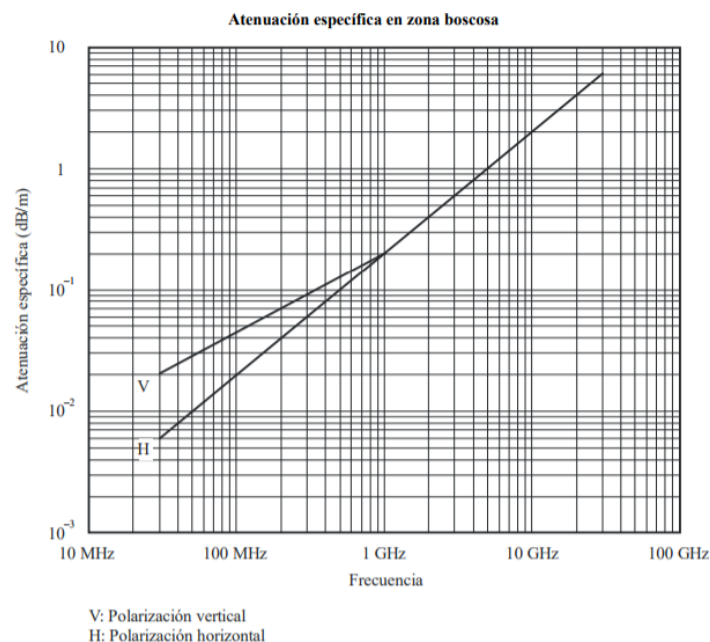


Figura 10- Atenuación específica en zona boscosa

Se hace hincapié en que la atenuación debida a la vegetación varía ampliamente debido a la naturaleza irregular del medio y a la gran variedad de especies, densidades y condiciones de humedad que se da en la práctica. Los valores que se muestran en la Fig. 10 deben considerarse únicamente como ejemplos.

A frecuencias del orden de 1 GHz, por ejemplo, la atenuación específica en zonas de árboles con hojas es normalmente un 20% superior (dB/m) que en las zonas de árboles sin hojas. También puede haber variaciones de la atenuación debido al movimiento del follaje, por ejemplo a causa del viento.

El valor de la atenuación máxima, A_m , limitada por la dispersión de la onda de superficie, depende del tipo y la densidad de la vegetación, así como del diagrama de radiación de la antena del terminal que se encuentra dentro de la vegetación y de la distancia en vertical entre la antena y el punto más alto de la vegetación.

Nosotros vamos a trabajar con frecuencias en torno a 5GHz, por lo que nuestra atenuación específica según la Figura 10 será aproximadamente de 1dB/m.

2.2.2. Dispositivos y fabricantes

A continuación vamos a comentar los distintos modelos de dispositivos utilizados en la empresa para hacer las pruebas de conexión e instalaciones reales, de diferentes fabricantes, para testear la precisión con la que la herramienta estima la potencia recibida de un radioenlace, teniendo en cuenta los aspectos teóricos anteriormente mencionados.

Vamos a comenzar comentando la marca MikroTik, por ser la que inicialmente se utilizaba en la empresa para instalaciones radioeléctricas. A continuación comentaremos Cambium Networks, la más utilizada actualmente debido a su eficacia para mantener una QoS y una elevada velocidad en cada conexión. Por último hablaremos de la más novedosa, la empresa Albentia Sistemas.

2.2.2.1- MikroTik

Mikrotiks Ltd., [MKT] conocida internacionalmente como MikroTik, es una compañía letona proveedora de tecnología disruptiva de hardware y software para la creación de redes. Mikrotik RouterOS es un software que funciona como un Sistema Operativo para convertir un PC o una placa Mikrotik RouterBOARD en un router dedicado. MikroTik se dedica principalmente a la venta de productos de hardware de red como routers denominados routerboards y switches también conocidos por el software que lo integra, denominado RouterOS y SwOS. La compañía fue fundada en el 1995, aprovechando el emergente mercado de la tecnología inalámbrica.

La cabecera de esta empresa, encargada de gestionar el tráfico de todos los clientes conectados mediante radioenlace, es de MikroTik, concretamente el modelo CCR1036-12G-4S-EM de Cloud Core [CCR1036].



Figura 11 – MikroTik

Entre los dispositivos punto a multipunto de esta tecnología, podemos destacar la RouterBoard Metal 5SHPN:



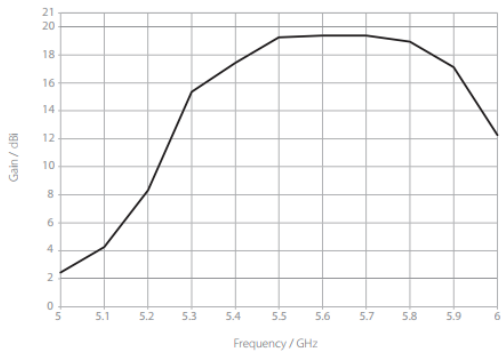
Figura 12- RBMetal5SHPn

Ésta RBMetal se conecta a una antena sectorial [RFESCC] con apertura de 100º horizontal y 5º vertical. El modelo es el siguiente:

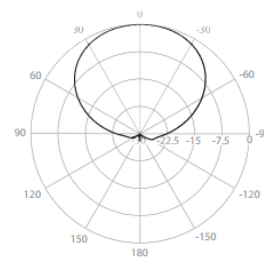


Figura 13- RFElements Sector Carrier Class 20dBi

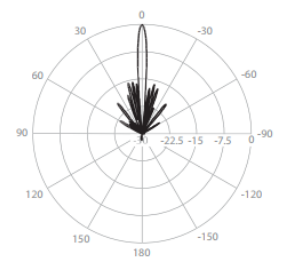
Gain H



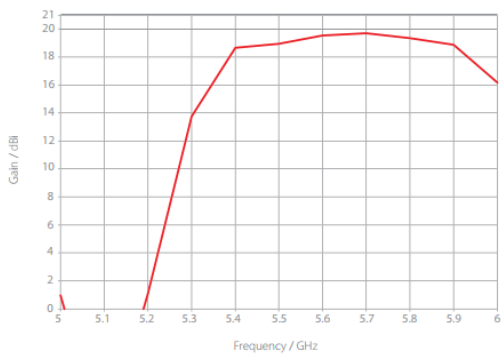
Azimuth Pattern H



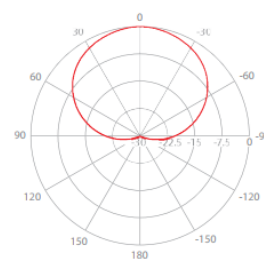
Elevation Pattern H



Gain V



Azimuth Pattern V



Elevation Pattern V

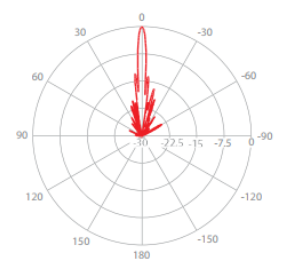


Figura 14- Diagrama radiación RFElements Sector

La figura 14 representa la ganancia en función de la frecuencia de trabajo, así como los diagramas de radiación tanto en polarización vertical como en horizontal.

En cuanto a dispositivos receptores, tenemos las Groove 52HPN [G52HPN]:



Figura 15- Groove 52HPN

Esta Groove modula a través del estándar wifi 2.4 GHz/5 GHz 802.11a/b/g/n, que además se conecta a un panel direccional [TETRAANT5] como por ejemplo:



Figura 16- Panel direccional TetraAnt5

Con este panel conectamos ambos dispositivos mediante un conector tipo N, estableciendo su ganancia de recepción en 23dBi.

2.2.2.2- Cambium Networks

Las soluciones de conectividad de Cambium Networks permiten a los operadores de redes adaptar las opciones de redes inalámbricas para satisfacer las necesidades inmediatas y escalar a medida que crecen. Su amplio catálogo incluye productos que utilizan tanto el espectro licenciado como el no licenciado, con capacidades diseñadas para cumplir con los requisitos de los casos de negocio.

Con un récord mundial de despliegue durante la pasada década, sus soluciones de servicio de banda ancha inalámbrica ePMP™ y Punto a multipunto (PMP) conectaron a millones de personas en todo el mundo. Gracias a un sistema de reutilización de frecuencias habilitada por la Sincronización GPS, su red de banda ancha inalámbrica puede evitar el cruce del espectro. Las soluciones punto a punto de distribución y acceso multipunto permiten a los operadores de redes ofrecer servicios de banda ancha inalámbrica de alta velocidad a clientes comerciales y residenciales. Las aplicaciones soportadas incluyen transferencia de datos y archivos, voz, video streaming y vigilancia por video. Estas redes de banda ancha inalámbrica punto a multipunto van desde pequeños despliegues para conectar áreas remotas hasta despliegues en toda la comunidad con miles de suscriptores.

Actualmente la empresa está gestionando más de 2200 clientes por aire, de los cuáles al menos 1500 están funcionando con este fabricante.

Como dispositivos AP emisores se emplean tanto la tecnología ePMP 1000, como la más novedosa ePMP2000. Puesto que posteriormente vamos a analizar un ejemplo con ePMP1000 [EPMP1000], vamos a explicar los dispositivos de esta tecnología. En los emisores tenemos ePMP 1000 GPS Sync Radio Integrated with a Sector Antenna. Empleamos sectores con 90º de ángulo de apertura y con una ganancia de 15dBi, lo que hace que regulemos el dispositivo GPS Sync Radio para gestionar la potencia de transmisión.



Figura 17- ePMP1000 GPS Sync Radio con Sector Antena

Como dispositivos receptores o CPEs, utilizamos principalmente 2 tipos, Force180 y Force200, ambos con 16 y 25 dBi de ganancia respectivamente.

En primer lugar tenemos el Force180, que es más común para instalaciones cercanas, ya que utilizamos el Force200 para las más lejanas o que requieran de una velocidad mayor. El Force180 [FORCE180] se puede usar como CPE, como AP (sin sincronismo) o como PTP. Posee una antena integrada de 16dBi y un ancho del haz a 3dB en azimut de 15º, mientras que en elevación es de 30º.



Figura 18- Force 180

El diagrama de radiación es el siguiente:

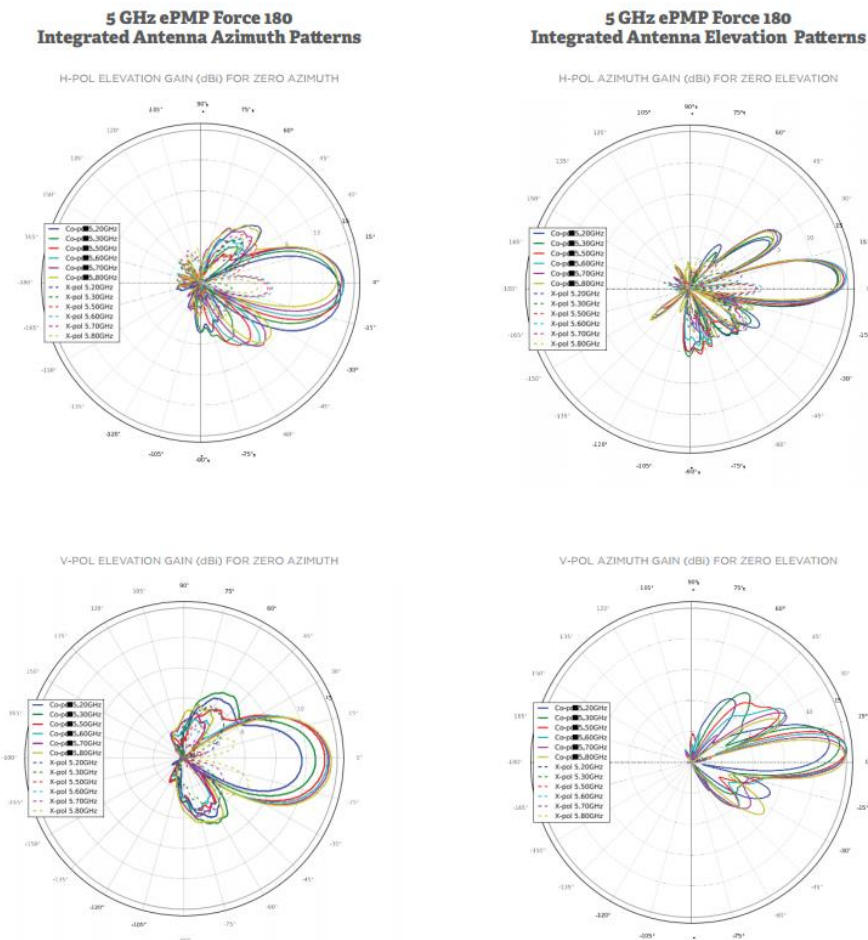


Figura 19- Diagrama de radiación Force180

Por su parte el Force 200 posee una ganancia de 25dBi en la banda de los 5Ghz y un ancho del haz a 3dB de tan sólo 7°. Éste es más potente que el anterior pero, a su vez, mucho más direccional.



Figura 20- Force 200

El diagrama de radiación es el siguiente:

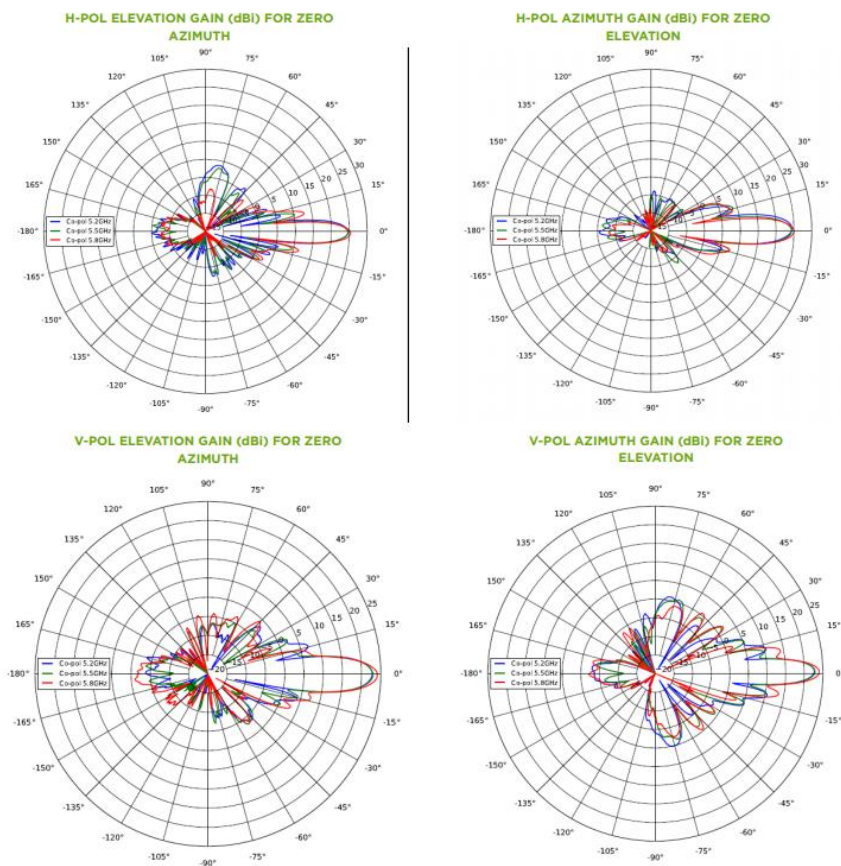


Figura 21- Diagrama de radiación Force 200

2.2.2.3- Albentia Sistemas

Albentia Systems es el proveedor líder del mercado español en soluciones de acceso inalámbrico de banda ancha. Fundada en 2004 y con sede en Madrid, Albentia Systems se apoya en una fuerte labor de I+D para desarrollar soluciones innovadoras basadas en la tecnología IEEE 802.16, para aplicaciones de acceso y transporte.

Dispone de un equipo de ingenieros con una contrastada experiencia en los mejores laboratorios de investigación del mundo y de un equipo dinámico que, de manera proactiva, acompaña a nuestros clientes en el despliegue de los sistemas de comunicaciones más avanzados de la industria.

Albentia forma parte del grupo BTESA (Broad Telecom S.A.), empresa líder en el mercado de difusión de TV, que lleva más de 30 años diseñando y fabricando transmisores de TV analógicos y digitales con fuerte presencia nacional e internacional. La experiencia del grupo BTESA supone un importante apoyo tanto logístico como financiero de una de las más sólidas compañías del mercado español.

Como dispositivos AP emisores tenemos 4 estaciones base AXS-BS-150-60.



Figura 22- Albentia AXS-BS-150-60

Estos dispositivos están gestionados como un único transmisor, con 4 sectores, etiquetados por colores. Una técnica novedosa que hace que el terminal de cliente o CPE se conecte automáticamente al radio que mejor señal obtenga. Los sectores poseen una antena integrada de 16dBi, y pueden emitir con una potencia máxima de transmisión de 23dBm. Nosotros estableceremos esta potencia en 14dBm puesto que emitimos con una PIRE máxima de 30dBm. A esto, sumamos los 60º de apertura que nos ofrece cada uno, y tenemos cubierta toda el área visible desde su emplazamiento, ya que por la parte trasera se ubica una montaña. Esta razón es la que nos llevó a colocar sectores de 60º en lugar de 90º.

La estación base AXS-BS-150-60 minimiza el uso espectral, al mismo tiempo que proporciona la mejor protección ante interferencias.

El diagrama de radiación es el siguiente:

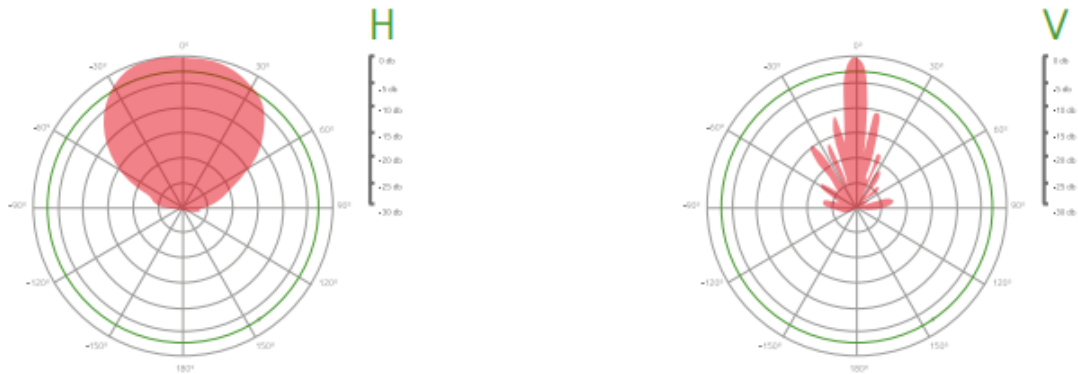


Figura 23- Diagrama de radiación Alentia AXS-BS-150-60

Entre los terminales finales de usuario o CPEs, empleamos 2 dispositivos, el AXS-CPE150-15, con una antena integrada de 15dBi, y el AXS-CPE150-RS, al que hay que acoplarle un panel direccional. Nosotros utilizaremos el panel de la figura 16, llegando a obtener una ganancia en recepción de 23dBi.



Figura 24- AXS-CPE150-15 y AXS-CPE150-RS

Ambos dispositivos son similares, solo que el AXS-CPE150-RS posee un conector RP-SMA para acoplarle el panel direccional al terminal.



Figura 25- Conector RP-SMA

Por último, cabe destacar que el diagrama de radiación del terminal AXS-CPE150-15 es el siguiente:

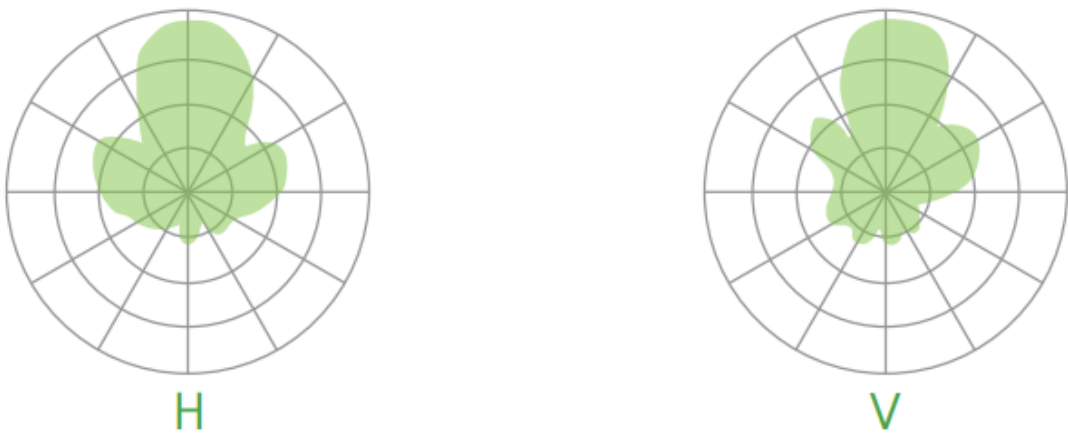


Figura 26- Diagrama de radiación AXS-CPE150-15

3.- Configuración de la Aplicación

3.1- Google Maps API

La interfaz de programación de aplicaciones, abreviada como API del inglés: Application Programming Interface, es un conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Igual que una interfaz de usuario permite la interacción y comunicación entre un software y una persona, una API facilita la relación entre dos aplicaciones para el intercambio de mensajes o datos.

Cada API está diseñada en un lenguaje de programación concreto y con unas especificaciones distintas que la definen (las APIs pueden incluir especificaciones para estructuras de datos y rutinas, clases de objetos o variables, a partir de las cuales se basa el uso de esa interfaz). Además, suele ser habitual que cada una de ellas disponga de documentación completa y eficaz (un conjunto de tutoriales, manuales y reglas de buenas prácticas para esa interfaz de programación).

Hay varios tipos de APIs [API], basadas en bibliotecas, basadas en clases, APIs de funciones en sistemas operativos y APIs de servicios web. Nosotros nos vamos a centrar en esta última.

Las APIs de servicios web son las interfaces de desarrollo de aplicaciones que permiten el intercambio de información entre un servicio web (software que da acceso a un servicio concreto a través de una URL) y una aplicación. Normalmente ese intercambio se produce a través de peticiones HTTP o HTTPS (la versión cifrada del protocolo HTTP). En la petición de la aplicación y respuesta, también en HTTP del servicio web, se contiene información de todo tipo tanto en los metadatos de la cabecera como en los del mensaje, normalmente en dos tipos de formatos muy usados: XML o JSON.

Hay cuatro tipos de API de servicios web habituales entre los desarrolladores: SOAP (Simple Object Access Protocol), un protocolo estándar de intercambio de información y datos en XML entre dos objetos; XML-RPC, un protocolo de llamada a procedimiento remoto que usa XML como formato de datos y llamadas HTTP como sistema de comunicación; JSON-RPC, mismo protocolo pero en formato JSON; y REST (Representational State Transfer), arquitectura de software para sistemas hipermedia en la World Wide Web; una API REST usa el protocolo HTTP.

En la página de desarrolladores de google [APIMAPS] tenemos toda la documentación correspondiente a la API de Google Maps. En nuestro caso, accederemos a la API de servicios web.

Google Maps para cada plataforma

Las Google Maps API están disponibles para Android, iOS, navegadores web y a través de servicios web HTTP.



Android



iOS



Web



Servicios web

[PRIMEROS PASOS](#)

Figura 27- API de Servicios Web

Una vez dentro, observamos las distintas opciones que nos ofrece esta herramienta, teniendo la opción de entrar a cada uno de sus servicios por separado.

The screenshot shows the Google Maps API documentation page. At the top, there are navigation links: 'Página principal', 'Documentación', and 'Tarifas y planes'. A search bar is present with the text 'Buscar'. Below the navigation, there are tabs for 'INFORMACIÓN GENERAL', 'GUÍAS', and 'SOPORTE'. The main heading reads 'Tu paquete de herramientas para el mundo real'. Below this, a paragraph states: 'Accede a interfaces HTTP, que proporcionan datos geográficos como geocodificación, indicaciones, elevación, sitio e información de zonas horarias.' There are two columns of service cards, each with a Google Maps icon and a brief description:

- Google Maps Geocoding API**: Realiza conversiones de direcciones a coordenadas geográficas.
- Google Places API Web Service**: Implementa el autocompletado y agrega información actualizada sobre millones de ubicaciones a tu sitio o aplicación.
- Google Maps Elevation API**: Datos de elevación para cualquier punto en el mundo.
- Google Maps Roads API**: Habilita la funcionalidad snap-to-road para rastrear de forma precisa rutas de navegación de GPS.
- Google Maps Geolocation API**: Encuentra una ubicación según información de torres celulares y nodos WiFi.
- Google Maps Directions API**: Calcula indicaciones entre varias ubicaciones.
- Google Maps Time Zone API**: Proporciona datos de zonas horarias para cualquier lugar del mundo.
- Google Maps Distance Matrix API**: Calcula el tiempo de viaje y la distancia para varios destinos.

Figura 28- Distintos servicios Google Maps API

3.1.1- Primeros pasos

Para empezar, en el apartado de tarificación [APIMAPSPET] vemos lo que nos ofrece la API gratuita para estimar, en función del volumen de peticiones esperado a la API, si es conveniente probar el plan Premium.

Web	ESTÁNDAR	PREMIUM
Google Maps JavaScript API Google Static Maps API Google Street View Image API	Gratis hasta 25 000 cargas de mapa por día. ³ USD 0,50 cada 1000 cargas de mapa adicionales, hasta 100 000 diarias, si está habilitada la facturación.	Precio basado en el volumen requerido. Funciones premium mejoradas: <ul style="list-style-type: none"> • Sin publicidad, garantizado. • Tamaño de imagen de hasta 2048 x 2048 píxeles. Para obtener más información, consulta Premium PlanLímites e índices de utilización .
Google Maps Embed API	Uso gratuito ilimitado.	---
Servicios web	ESTÁNDAR	PREMIUM
Google Maps Directions API Google Maps Distance Matrix API ⁴ Google Maps Elevation API Google Maps Geocoding API Google Maps Geolocation API Google Maps Roads API Google Maps Time Zone API	Gratis hasta 2500 solicitudes por día. USD 0,50 cada 1000 solicitudes adicionales, hasta 100 000 diarias, si está habilitada la facturación.	Precio basado en el volumen requerido. Funciones premium mejoradas: <ul style="list-style-type: none"> • Aumentos de consultas por segundo. • Se mejoraron las capacidades de Distance Matrix API y Roads API.² Para obtener más información, consulta Premium PlanLímites e índices de utilización .
Google Places API Web Service	150 000 solicitudes gratuitas por día (después de la activación de la tarjeta de crédito).	Precio basado en el volumen requerido. Para obtener más información, consulta Premium PlanLímites e índices de utilización .

Figura 29- Peticiones gratuitas por servicio de la API

Como podemos observar, Javascript API por ejemplo nos ofrece hasta 25.000 cargas de mapa por día. Los servicios de elevación, geocodificación y geolocalización nos ofrecen 2500 solicitudes por día, y el servicio de Google Places hasta 150.000 solicitudes diarias. Son cifras que para un usuario básico a nivel de empresa son más que suficientes, por lo que en este caso nos basta con lo que nos ofrece de manera gratuita esta API.

A continuación, nos vamos de nuevo a la página de la documentación. Una vez allí, pulsamos el botón “Obtener una clave” y registramos una cuenta de Gmail para empezar a usar la API. En mi caso, se me ha generado la clave **AlzaSyBLDCXFtLajMw5h4ugi-PeHf--HsopiG6M**, que utilizaremos posteriormente en nuestra web.

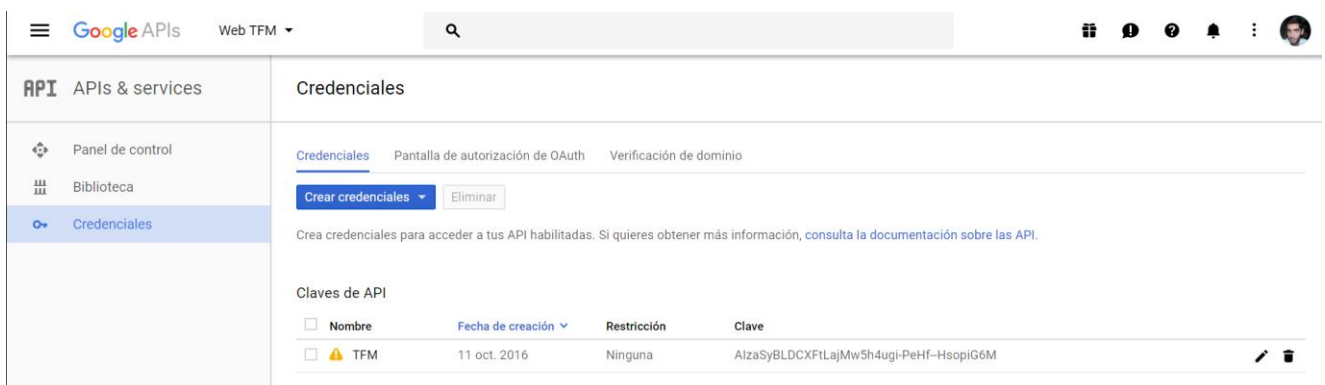


Figura 30- Credenciales API

En nuestro panel de APIs y servicios [APISERV], podemos editar nuestras credenciales, crear nuevas e incluso monitorizar el tráfico desde el panel de control.



Figura 31- Ejemplo monitorización Javascript API

En este ejemplo vemos que el día que más tráfico generamos fue el 26 de julio con hasta 109 cargas de mapa, muy por debajo del límite diario (25.000 cargas de mapa diarias) que nos permite la API de manera gratuita.

3.1.2- Google Places API Web Service

Este servicio implementa el autocompletado y agrega información actualizada sobre millones de ubicaciones a nuestra aplicación. En nuestro caso, nos sirve para buscar de un modo más rápido una dirección o una ubicación para movernos de una manera más rápida por el mapa.

En la documentación [PLACES], podemos ver más información acerca de este servicio. En nuestro proyecto, lo hemos empleado para búsquedas de sitios, autocompletado de direcciones, etc.

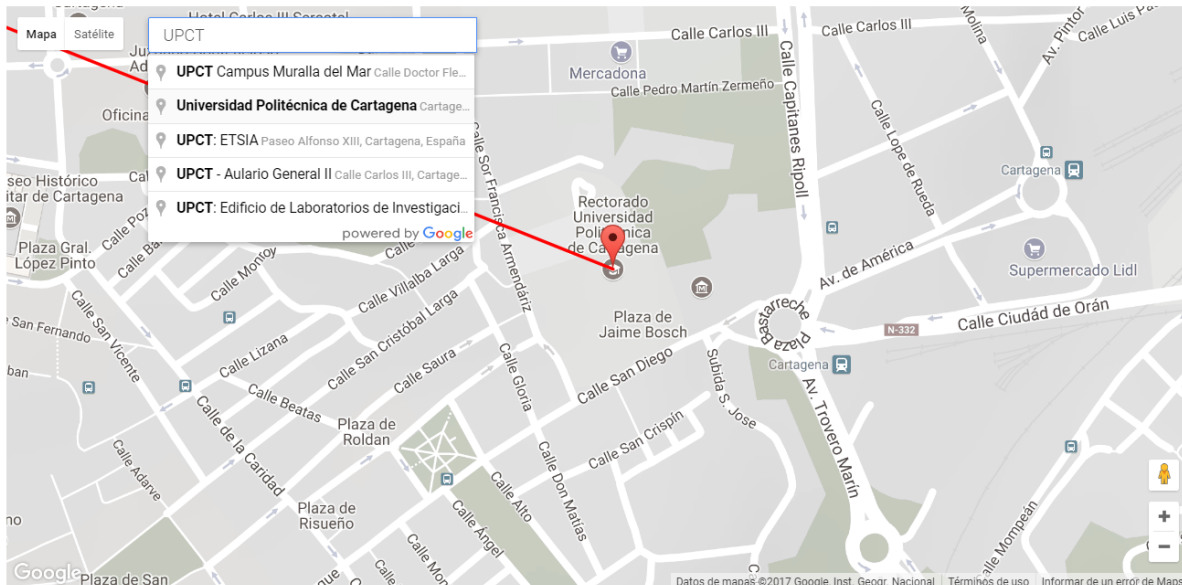


Figura 32- Ejemplo uso Google Places API

El servicio de autocompletado de sitios es un servicio web que devuelve predicciones de sitios en respuesta a una solicitud HTTP. La solicitud especifica una cadena de búsqueda textual y límites geográficos opcionales. El servicio se puede usar para proporcionar la funcionalidad de autocompletado para búsquedas geográficas basadas en texto al devolver sitios como negocios, direcciones y puntos de interés a medida que el usuario escribe.

Una solicitud de autocompletado de sitios es una solicitud HTTP URL como la siguiente:

```
https://maps.googleapis.com/maps/api/place/autocomplete/output?parameters
```

Donde *output* puede ser cualquiera de los siguientes valores:

- json (recomendado) indica el formato de salida en JavaScript Object Notation (JSON).
- xml indica el formato de salida como XML

3.1.3- Google Maps Elevation API

Este servicio nos devuelve datos de elevación para cualquier punto en el mundo. En el apartado “solicitar datos de elevación de varias ubicaciones y crear gráficos de elevación” [ELEVATION], podemos ver un ejemplo de cómo crear gráficos de elevación en función de una ruta especificada.

En nuestro caso, le vamos a pasar la ubicación del punto a punto que queremos calcular, desde la ubicación del cliente, hasta el emplazamiento del dispositivo emisor, ya sea del propio punto a punto, o bien de un sectorial de un emisor punto a multipunto.

El resultado de este proceso es el siguiente:

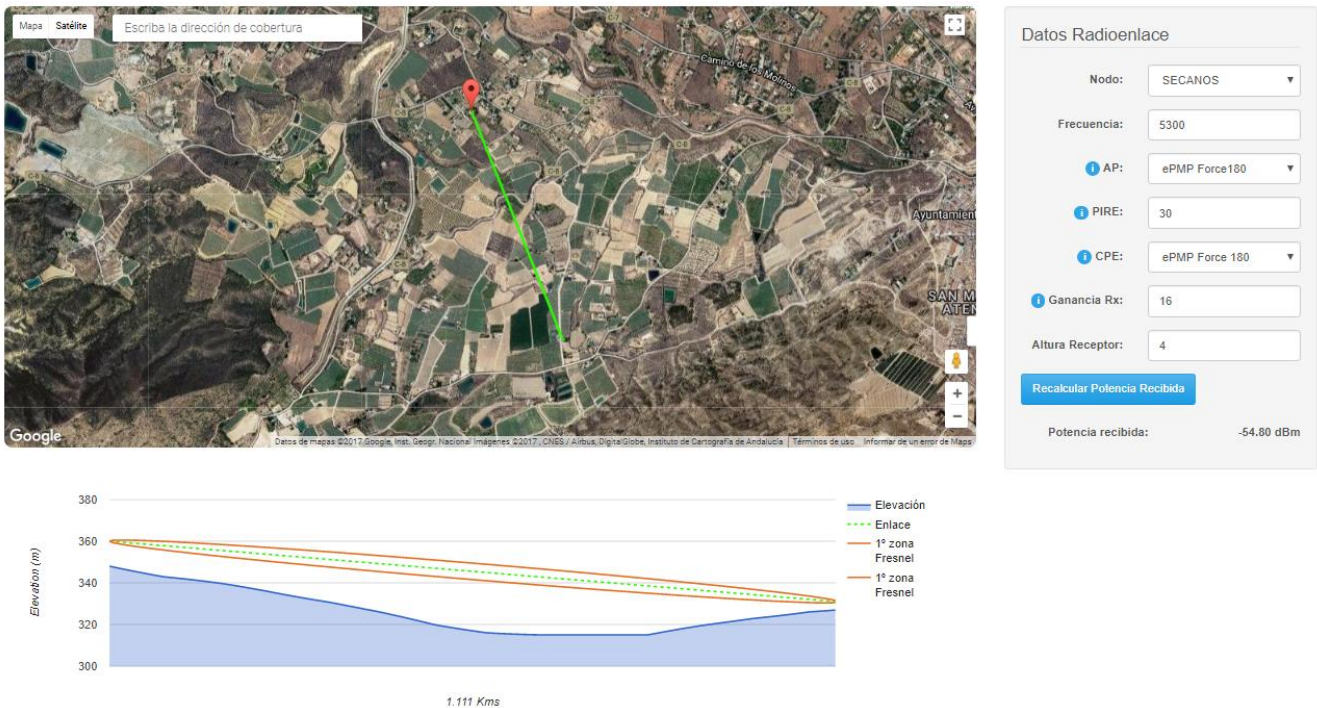


Figura 33- Ejemplo uso del servicio de elevación

Quando se realiza una petición de este tipo al servicio de elevación, la respuesta obtenida en formato JSON tiene la siguiente estructura:

```
{ "results" : [ {  
  "elevation" : 4411.941894531250,  
  "location" : {  
    "lat" : 36.5785810,  
    "lng" : -118.2919940  
  },  
  "resolution" : 19.08790397644043  
}, {  
  "elevation" : 1381.861694335938,  
  "location" : {  
    "lat" : 36.41150289067028,  
    "lng" : -117.5602607523847
```

```
},  
  "resolution": 19.08790397644043  
}, ...
```

Cada vez que hagamos una petición, llamaremos a la variable que generamos con el servicio de elevación:

```
// Create an ElevationService.  
elevator = new google.maps.ElevationService;
```

Y cuando llamamos a la función que lo invoca, tenemos que establecerle como parámetros tanto la ruta, como el número de muestras que queremos que nos devuelva. A mayor número de muestras, más precisión en el perfil de elevación pero más lenta es la consulta. Nosotros hemos utilizado un valor de 256 muestras para nuestro ejemplo.

```
// Create a PathElevationRequest object using this array.  
// Ask for 256 samples along that path.  
// Initiate the path request.  
elevator.getElevationAlongPath({  
  'path': path,  
  'samples': 256  
}, plotElevation);
```

La función `plotElevation`, es la que se encarga de dibujar en el cuadro correspondiente este perfil de elevación.

La variable `path` es un array que contiene la ubicación de las dos posiciones entre las cuales se hace la petición de este servicio:

```
path = [  
  {lat: 37.7675279927127, lng: -1.4985955742397437},  
  {lat: 37.70324, lng: -1.52571999999999784}];
```

En esta función `plotElevation`, obtenemos la variable *elevation* de cada punto de las 256 muestras que hemos solicitado, y la añadimos a la columna correspondiente del array de datos, en este caso a la segunda.

```
var data = new google.visualization.DataTable();  
data.addColumn('string', 'Dato'); //Sample  
data.addColumn('number', 'Elevación');  
data.addColumn('number', 'Enlace');  
data.addColumn('number', '1º zona Fresnel');  
data.addColumn('number', '1º zona Fresnel');
```

```
data.addRow(['', elevations[i].elevation, parseFloat(sample) + paso*i, ellipseA, ellipseB]);
```

`Elevations[i].elevation` contiene el valor de elevación en cada punto (iteración en un bucle), mientras que `sample` es el valor que va tomando la línea de visión (line of sight) del enlace, y los datos de la elipse corresponden a la curva calculada para la estimación del Fresnel.

3.1.4- Google Maps Geocoding API

Este servicio nos ayuda a realizar conversiones de direcciones a coordenadas geográficas [GEOCODING], dado que en este proyecto trabajamos principalmente con puntos, ubicaciones, especificadas con una latitud y longitud correspondientes.

La **geocodificación** es el proceso que convierte direcciones (como la dirección de una calle) en coordenadas geográficas (latitud y longitud) que puedes usar para disponer marcadores en un mapa o posicionar el mapa.

La **geocodificación inversa** es el proceso de conversión de coordenadas geográficas en direcciones en lenguaje natural. El servicio de geocodificación inversa Google Maps Geocoding API también te permite buscar la dirección por un id. de sitio determinado.

En la documentación podemos ver un extracto completo de la respuesta JSON tanto de la geocodificación, como de la geocodificación inversa. A continuación se muestra un ejemplo de uso de geocodificación, mediante el cual podemos obtener las coordenadas tanto de una calle como punto en concreto.

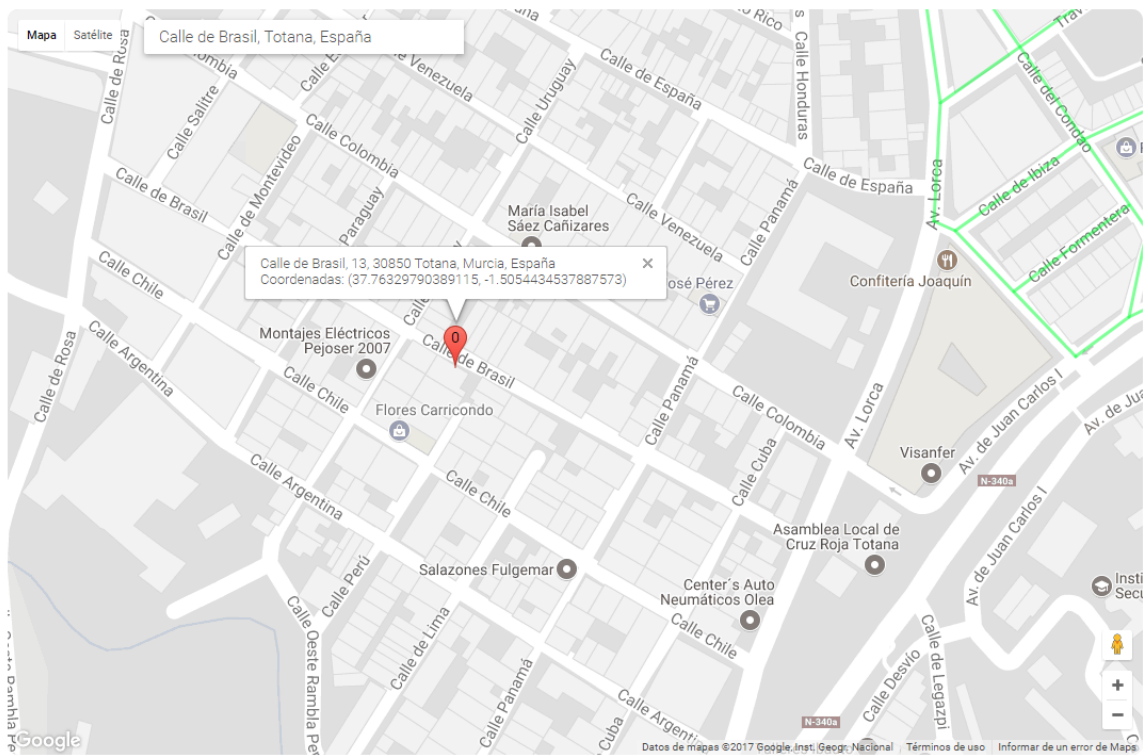


Figura 34- Ejemplo uso geocodificación

3.1.5- Google Maps Geolocation API

Google Maps Geolocation API [GEOLOCATION] devuelve una ubicación y radios de precisión en función de información acerca de torres celulares y nodos de WiFi que el cliente móvil pueda detectar. Este documento describe el protocolo utilizado para enviar estos datos al servidor y devolver una respuesta al cliente.

La comunicación se realiza a través de HTTPS usando POST. Tanto la solicitud como la respuesta poseen formato JSON, y el tipo de contenido de ambas es application/json.

Esta herramienta nos ha ayudado a implementar un servicio de localización más rápido y bastante preciso a la hora de estimar la ubicación de un radioenlace. Cargando la página principal intenta geolocalizarnos si es posible y ubicar dicha localización en el mapa, para desde ahí hacer una estimación de la potencia hasta el nodo emisor seleccionado.

Esto nos ha resultado bastante útil en la empresa a la hora de estimar a qué nodo es más viable conectar a un cliente desde su tejado sin tener que estar haciendo pruebas físicas buscando el mejor emplazamiento para el dispositivo receptor.

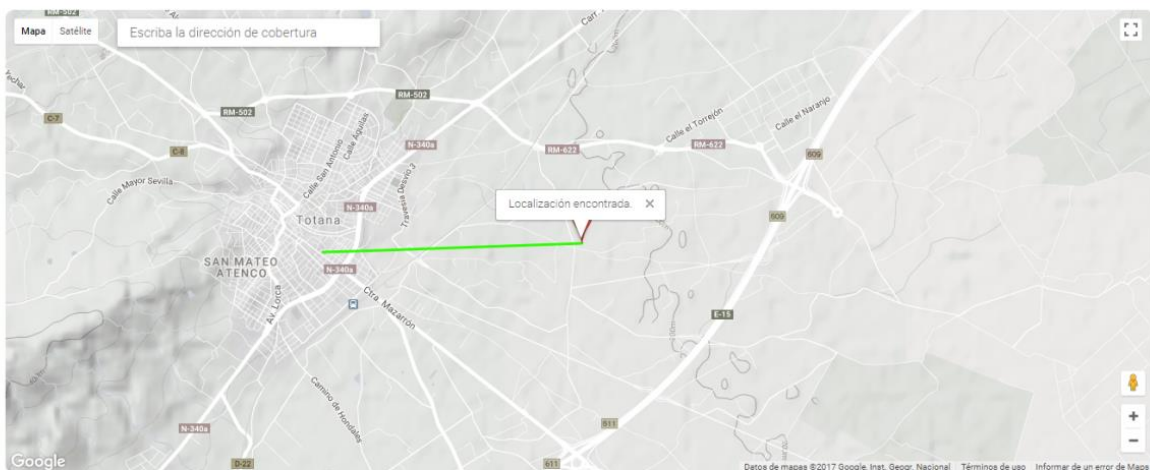


Figura 35- Ejemplo de geolocalización

Posteriormente explicaremos cómo incluir un certificado SSL a nuestra web, ya que ciertos servicios requieren HTTPS para todas las solicitudes de servicios web de Google Maps API que contengan datos del usuario o identificadores de desarrollador. Las solicitudes realizadas a través de HTTP que incluyan datos confidenciales se rechazarán. En este caso, como se trabaja con la ubicación actual de un usuario, la API requiere de una conexión segura HTTPS para intentar geolocalizar al usuario.

3.1.5.1- Instalación certificado SSL

La API de geolocalización sólo funciona en entornos seguros. Para ello, hemos de convertir nuestra página en segura (https) para poder utilizar esta herramienta.

SSL (Secure Sockets Layer) es un protocolo criptográfico que permite la comunicación segura entre un cliente (usuario con un navegador web) y un servidor.

Un certificado SSL es un certificado digital utilizado por el protocolo para el encriptamiento de la información. Este certificado es proporcionado por un proveedor autorizado (Verisign, Thawte, Comodo, etc.) y es enviado al cliente por el servidor con quien estamos estableciendo una conexión segura.

Hay muchos servicios que utilizan este protocolo, algunos ejemplos pueden ser: HTTPS, SMTPS, IMAPS, SSH, POP3S, etc.

El certificado que vamos a crear tiene el mismo nivel de encriptamiento que cualquiera de estos proveedores autorizados puede entregar, simplemente no aparecemos en la lista preestablecida en nuestro navegador, y por lo tanto desconfía del mismo.

Hemos instalado OpenSSL en el servidor que hemos utilizado para alojar la web. En primer lugar hemos ejecutado:

```
sudo apt-get install openssl
```

A continuación, hemos creado una llave privada, la cual nos será útil a la hora de crear el certificado. Una vez creado, nuestro certificado SSL dependerá de esta llave para la implementación del mismo en cualquier servicio que requiera una conexión segura. A continuación hemos creado una llave de 1024 bits.

```
openssl genrsa -out tfm.key 1024
```

El siguiente paso es crear un CSR (Certificate Signing Request). Un CSR es la base para un certificado SSL, en él se definen datos como el dominio, organización, ubicación, información de contacto, entre otros.

Para generar el CSR hemos ejecutado el siguiente comando, incluyendo en uno de los parámetros la llave privada que acabamos de crear:

```
openssl req -new -key tfm.key -out webtfm.csr
```

Al ejecutar este comando vienen una serie de preguntas las cuales hemos rellenado con los datos correspondientes del servidor web que aloja la página.

Finalmente queda generar el certificado SSL. Para ello vamos a necesitar tanto la llave privada como el CSR que acabamos de crear. Ejecutamos el siguiente comando:

```
openssl x509 -req -days 365 -in webtfm.csr -signkey tfm.key -out certfm.crt
```

Mediante el parámetro days le hemos establecido un año de expiración.

Una vez generado e instalado en nuestro servidor, ya podemos acceder mediante https a la web, pudiendo así obtener información de geolocalización.

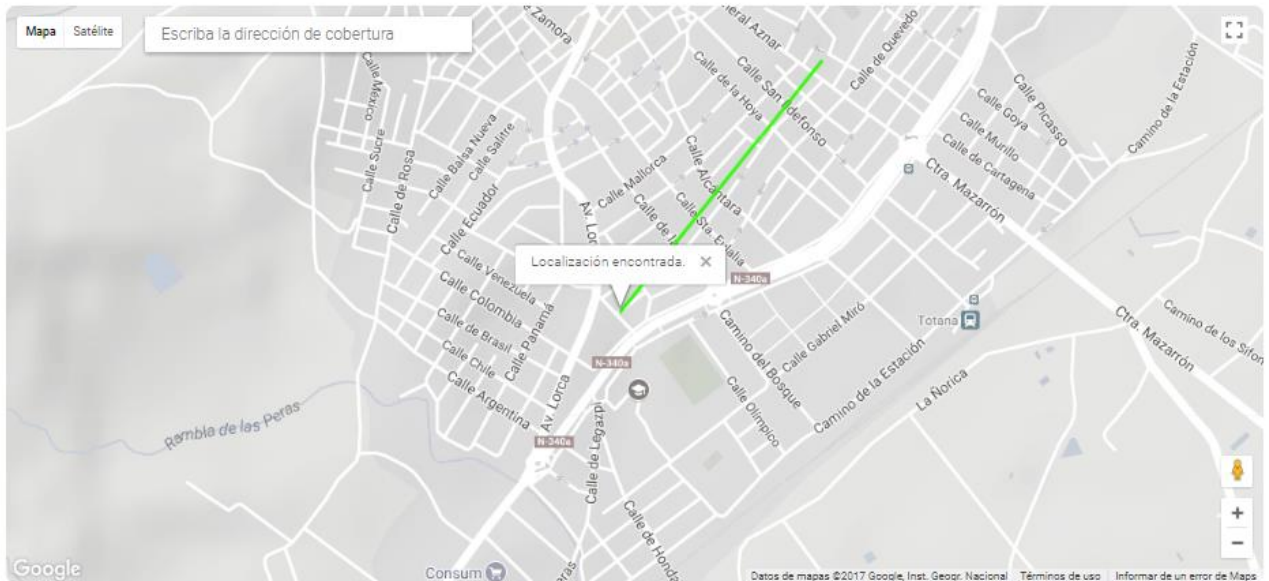


Figura 36- Geolocalización con HTTPS

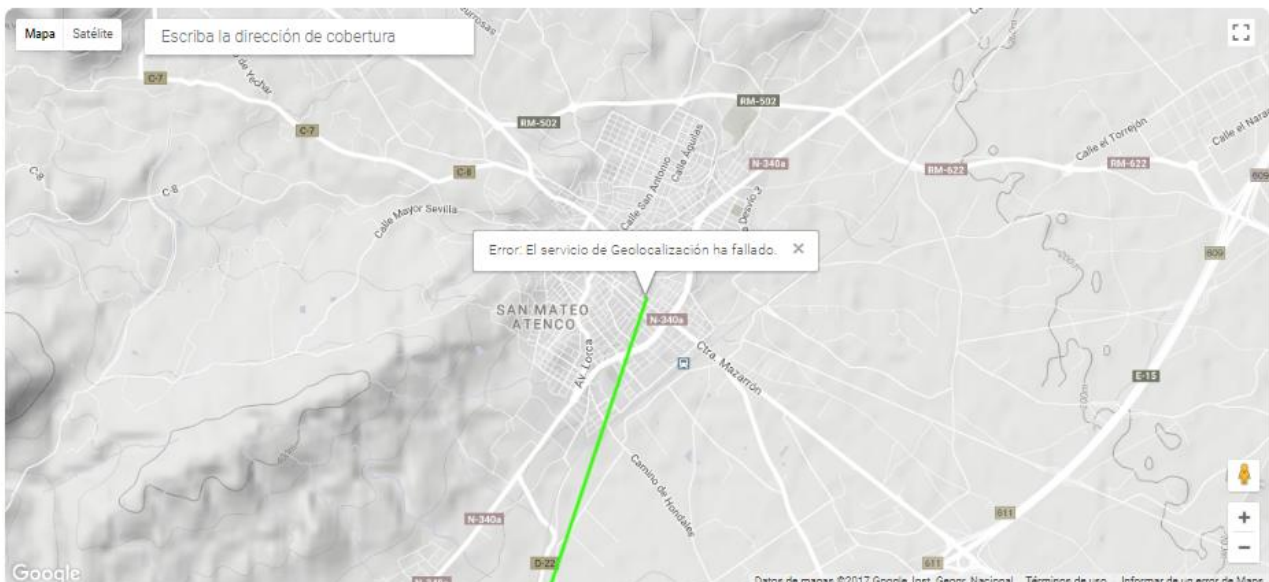


Figura 37- Geolocalización sin HTTPS

Como podemos observar, sin https el servicio de geolocalización falla, mientras que si entramos con https, nos localiza y nos ubica con el primer nodo de la lista.

3.1.6- Google Maps Javascript API

Esta es quizás la herramienta que más peso tiene en nuestra aplicación, ya que ha sido la que más juego nos ha dado a la hora de desarrollar los mapas en función de nuestras necesidades.

Con esta herramienta [JAVASCAPI] podemos desde cambiar dinámicamente la ubicación de un marcador sin tener que recargar la página ni el mapa, crear nuevos marcadores, calcular distancias entre dos puntos, hasta incluso usar MySQL y PHP para cargar datos desde una base de datos externa al mapa.

El uso de este servicio se resume básicamente a estos aspectos:

1. Se declara la aplicación como HTML5 a través de la declaración `<!DOCTYPE html>`.
2. Se crea un elemento `div` con el nombre "map" para que contenga el mapa.
3. Se define una función JavaScript que crea un mapa en el elemento `div`.
4. Se carga la Maps JavaScript API usando una etiqueta `script`.

Para cargar la API de Google Maps JavaScript, se usa una etiqueta `<script>` como la que aparece en el siguiente ejemplo:

```
<script async defer
  src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&
callback=initMap">
</script>
```

Como podemos observar, hace una llamada a la función `initMap` que es la función que se encarga de crear el mapa, añadir marcadores, escuchadores de eventos, etc.

La dirección URL que se incluye en la etiqueta `<script>` es la ubicación de un archivo de JavaScript que carga todos los símbolos y las definiciones que necesitas para usar la API de JavaScript. Se necesita esta etiqueta `<script>`.

El atributo `async` permite que el navegador represente el resto de tu sitio web mientras se carga la API de Javascript. Cuando la API esté lista, llamará a la función especificada usando el parámetro `callback`. El parámetro `key` contiene la clave de API de nuestra aplicación.

También se nos permite el uso de bibliotecas [JAVASBIB]. Las bibliotecas son módulos de código que proporcionan una funcionalidad adicional a la Maps JavaScript API, pero no se cargan a menos que lo solicites específicamente. Debemos especificar las bibliotecas adicionales para que se carguen dentro de la solicitud de arranque estableciendo un parámetro `libraries` y pasándole el nombre de las bibliotecas. Pueden especificarse varias bibliotecas como una lista de elementos separados por comas. Una vez descargadas las bibliotecas, se puede acceder a ellas a través del espacio de nombres `google.maps.biblioteca`.

Un ejemplo para la solicitud de varias bibliotecas es el siguiente:

```
<script type="text/javascript"
  src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&libraries=geometry,places">
</script>
```

Existen 4 tipos de bibliotecas, las cuales hemos empleado a lo largo de este proyecto:

- **drawing** proporciona a los usuarios una interfaz para que dibujen polígonos, rectángulos, polilíneas, círculos y marcadores en el mapa. Esta librería nos servirá para el dibujo de marcadores, y líneas entre dos puntos del mapa, así como una ruta de puntos en caso de ser necesaria.
- **geometry** incluye funciones de utilidades para calcular valores geométricos escalares (como la distancia y el área) sobre la superficie terrestre. Esta nos ha servido para calcular la distancia geodésica (recorrido más corto entre dos puntos) entre dos puntos mediante el uso de la función *computeDistanceBetween()* y pasándole como argumento dos objetos *LatLng*.
- **places** permite que tu aplicación busque sitios, como establecimientos, ubicaciones geográficas o puntos de interés destacados, dentro de un área definida. Mediante esta librería hemos implementado el autocompletado del cuadro de búsqueda de direcciones por ejemplo de la Figura 32.
- **visualization** proporciona representaciones visuales de datos, como los de mapas de calor y los del motor de Google Maps. Con esta librería hemos implementado el gráfico para visualizar el perfil de elevación obtenido entre dos puntos, como hemos podido observar en la Figura 35.

3.2- Estructura de la página web

La web desarrollada en este proyecto está compuesta principalmente por 4 apartados:

- La página principal, index.php, en la cual reside el corazón de este proyecto: geolocalización y estimación de los parámetros de un radioenlace

- La segunda página y tercera, fibra y fibra2.php, son una ampliación de los servicios que ofrece este proyecto, en las cuales se muestra un mapa de cobertura de fibra óptica en el municipio de Totana, en el cual trabajo actualmente.

- La cuarta y última página, nuevoemplazamiento.php sólo está visible para usuarios administradores, debido a que aloja un formulario con un mapa para geolocalizar y añadir a la base de datos nuevas ubicaciones para emplazamientos fijos de equipos.

Por último, se explicará el funcionamiento de otros scripts que hemos utilizado.

3.2.1- index.php

En la página principal, podemos escoger la ubicación para nuestro radioenlace.

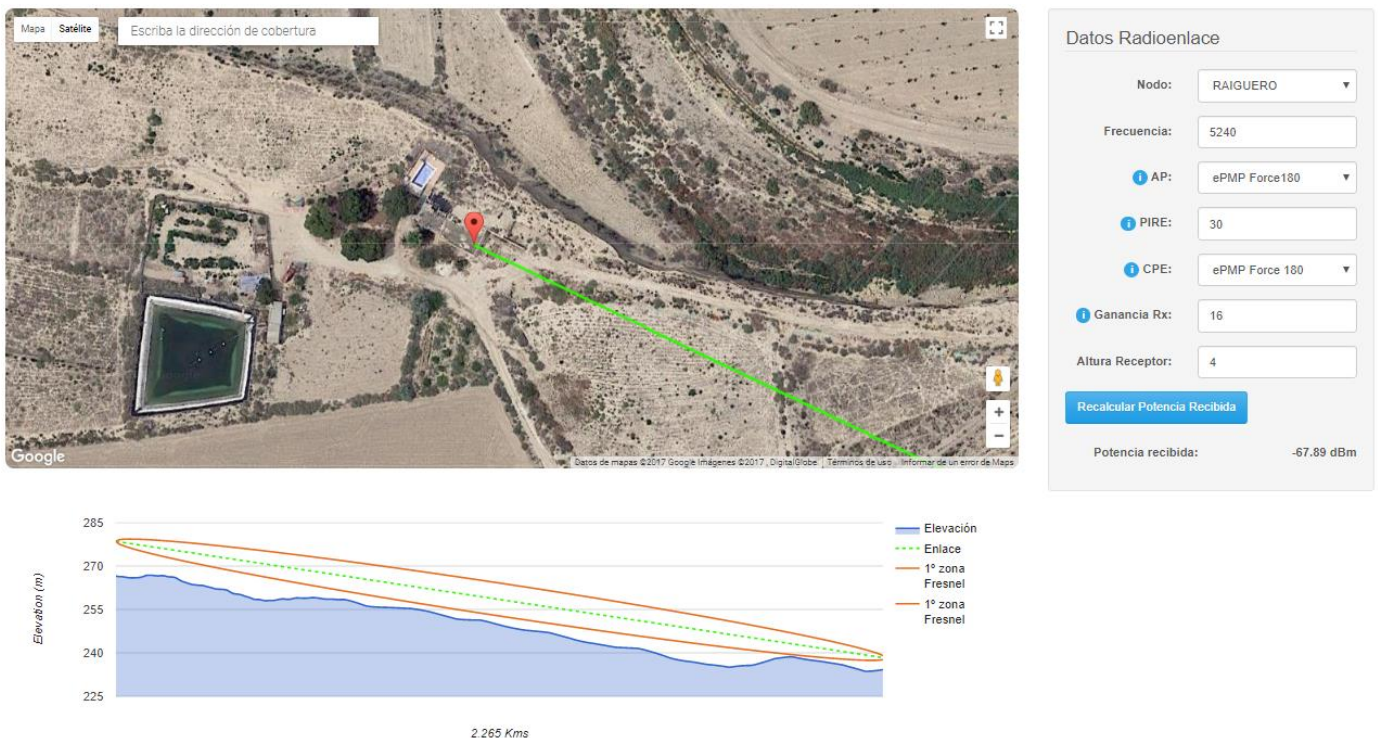


Figura 38- Aspecto página principal

El formulario que aparece en la parte derecha recoge los datos necesarios para estimar la potencia recibida en la ubicación receptora: PIRE – pérdidas en espacio libre calculadas – pérdidas por fresnel (si hubiere) + Ganancia antena receptora.

A la hora de cambiar el nodo emisor, se ejecuta una consulta a la base de datos donde se recogen la altura y las coordenadas de dicho emplazamiento. Al seleccionar el dispositivo emisor, se lee de la base de datos la potencia de transmisión, al igual que el receptor, que recoge la ganancia de la antena receptora.

Tanto si utilizamos el buscador para buscar una ubicación para el receptor, como si ejecutamos un doble click en la ubicación deseada (para facilitar el manejo desde un dispositivo móvil táctil) se añade un marcador en dicha ubicación, se dibuja la línea del punto A al punto B, y se ejecuta la función principal, en la cual se calcula la distancia, las pérdidas, el perfil de elevación y la potencia estimada.

Para cada marcador, se ha implementado un escuchador de eventos mediante el cual, para el objeto *marker*, se escucha el evento '*dragend*', para así estimar donde se ha dejado caer el marcador, obtener esas coordenadas y repetir el proceso. Esto nos sirve, poniendo el mapa en vista de satélite, para establecer la ubicación exacta del emplazamiento a la hora de estimar la dirección del radioenlace, así como ver la línea de visión exacta entre ambos dispositivos.

La función principal antes mencionada, es la encargada de estimar las pérdidas entre el punto A y B. También estima la elevación de cada uno de los puntos intermedios, establecidos como *samples* a 256 muestras. Una vez obtenidos estos datos de elevación, calculamos la pendiente de la recta para dibujar la línea de visión (line of sight, LOS), mediante la elevación del punto inicial y final, y dividido entre el número de muestras. A continuación, se calcula el radio de la primera zona de fresnel, la cual vamos a tener en cuenta en caso de estimar pérdidas. Con el radio, se calculan los puntos del elipsoide de Fresnel, despejando de la fórmula de la elipse, aplicando el semieje mayor como la distancia entre los dos puntos y el semieje menor como el radio de la primera zona de Fresnel. Por último se llama a la función que calcula la potencia recibida.

Al final del formulario se ha añadido un botón que invoca esta función en caso de modificar a mano alguno de los parámetros del formulario.

Al igual que el proceso de logueo, la función que se encarga de obtener los datos de coordenadas y altura de un emplazamiento de la base de datos es un script (*getPosition.php*) que se ejecuta mediante AJAX de forma asíncrona con el servidor. El cliente le envía los datos al servidor al seleccionar otro nodo, el servidor procesa localmente el script con los datos que le pasa el cliente y le devuelve una respuesta, con la cual trabajamos y actualizamos nuevamente con javascript nuestra web sin necesidad de recargar la página.

3.2.2- fibra.php

Este apartado es producto de las necesidades de la empresa así como del descubrimiento de hasta dónde somos capaces de llegar de forma creativa con la API de Google Maps.

Esta página básicamente contiene un mapa en el cual se muestra la cobertura de fibra a día de hoy de la empresa en la que actualmente trabajo. Para ello, se hace uso de la API de Javascript de nuevo, leyendo datos de ubicaciones guardadas (en este caso en un fichero .txt). Este fichero contiene las ubicaciones inicial y final, así como puntos intermedios de todas las calles en las que hay cobertura de fibra óptica.

Este procedimiento va leyendo en bucle del fichero, el cuál se lee y almacena en un array. Cada elemento del array contiene el punto o puntos de cada calle (cada punto está formado por unos datos de latitud y longitud). Muchos finales de calles coinciden con el inicio de la siguiente.

El resultado de este apartado es el siguiente:

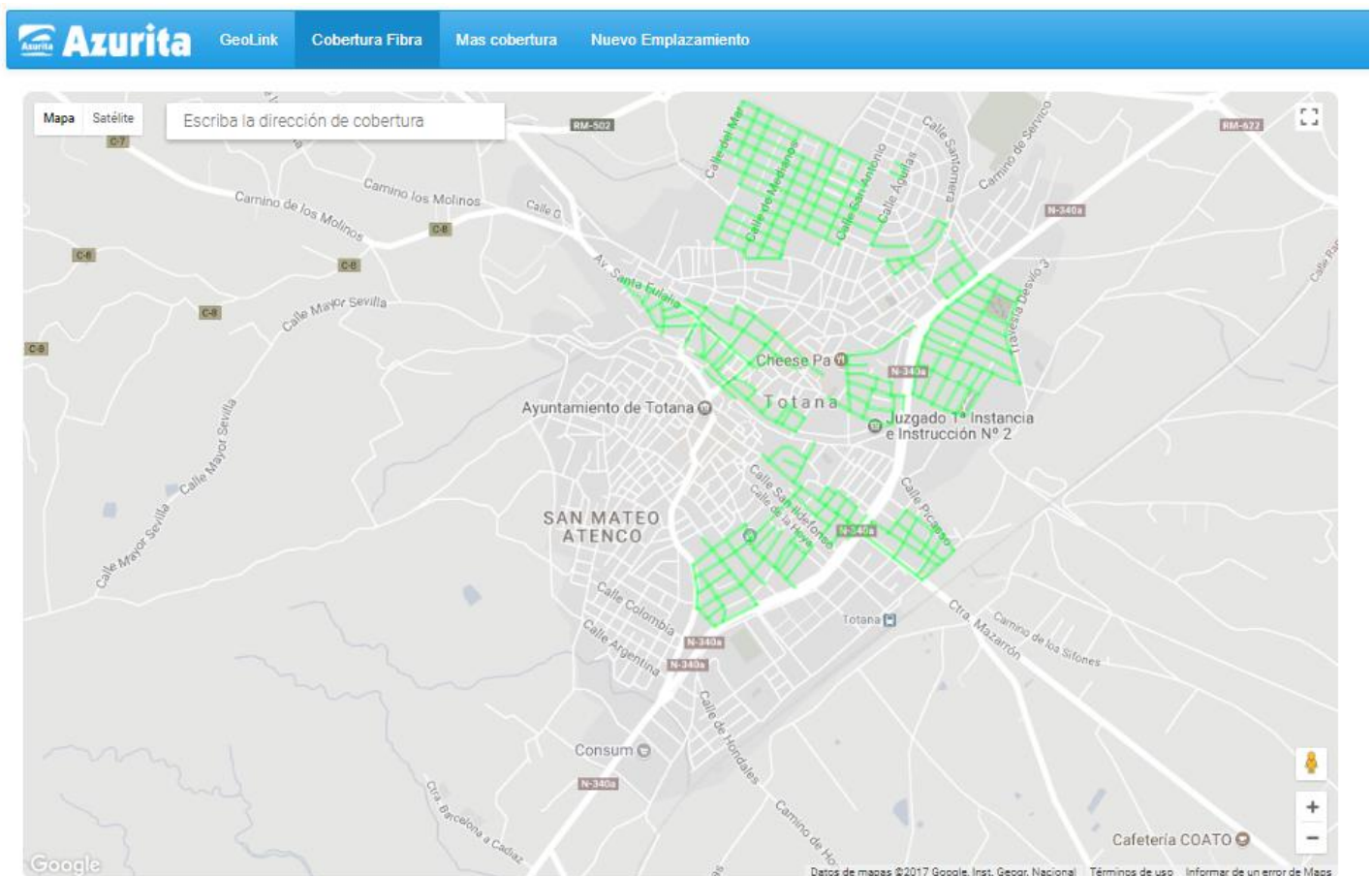


Figura 39- Cobertura de Fibra óptica en Totana

3.2.3- fibra2.php

Este apartado es más de lo mismo, pero a la inversa. Dado que ir obteniendo las coordenadas de cada calle es un proceso tedioso, mediante esta página se pretende facilitar la labor.

Se ha desarrollado una herramienta en la cual se van añadiendo marcadores al ritmo que se dan clicks en el mapa, geolocalizando dichos marcadores, de modo que van apareciendo numerados y, a su vez, se va dibujando en color verde una línea haciendo el trazado de uno a otro. Esto nos sirve para dibujar las zonas que vamos aprovisionando con cajas de abonado, de modo que se vean reflejadas, de un modo más directo y claro que en los planos de Autocad, las zonas en las que tenemos cobertura.

Un ejemplo en el cual se ve como añadir un trazado es el de la figura 24, en el cual señalamos el trazado de la Calle de Piqueras:

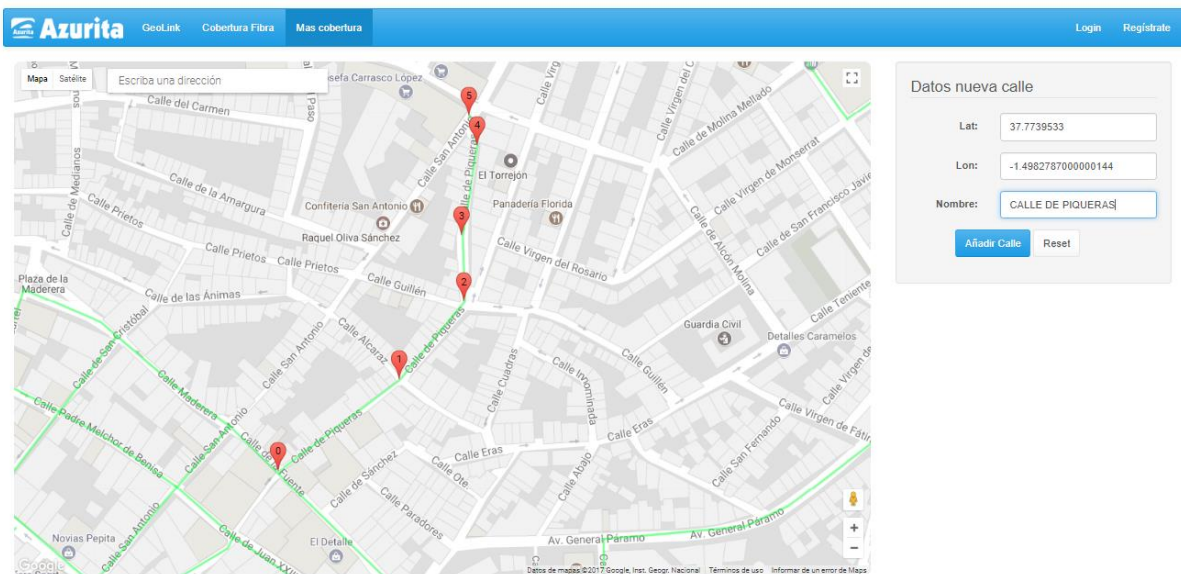


Figura 40- Herramienta para añadir nuevas calles

Una vez tenemos el trazado hecho, solo nos queda añadir un nombre a la calle en el formulario que se añadirá como comentario al fichero .txt, que luce así:

```
ruta_calles_fibra.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
[
// Salvador de Madariaga NODO 3
{lat: 37.77365486307383 , lng: -1.4922775569737041},
{lat: 37.77310363070961 , lng: -1.490870738345393},
],[ // C/ Menendez Pelayo
{lat: 37.77373881487161 , lng: -1.4914694469329106},
{lat: 37.77001453302046 , lng: -1.4940572026550853},
],[ // C/ Monseñor Tarancon
{lat: 37.77389867607792 , lng: -1.4918872955620373},
{lat: 37.773444971235065 , lng: -1.4907044413864696}
],[ // C/ Travesía paralela desvio y C/ Carmen conde
{lat: 37.773444971235065 , lng: -1.4907044413864696},
{lat: 37.77225769305028 , lng: -1.4912784341156566},
{lat: 37.77206032617463 , lng: -1.4912468783827535},
{lat: 37.77062621451834 , lng: -1.4907272401630962}
],[ // C/ Vicente Aleixandre
{lat: 37.7731031736504 , lng: -1.491914077875208},
{lat: 37.77340953214665 , lng: -1.4926838718624822}
],[ // C/ Antonio Machado
{lat: 37.77316465746428 , lng: -1.4930955909462682},
{lat: 37.77243002855098 , lng: -1.4912113391133062}
```

Figura 41- Ruta calles cobertura de fibra

3.2.4- nuevoemplazamiento.php

Esta página sólo está visible para usuarios que sean administradores. En ella, en un formato similar al apartado anterior, tenemos un mapa y un formulario.

El mapa está implementado de tal manera que a través de cada evento *click*, un único marcador se actualiza con los datos de geoposicionamiento obtenido, y queda escuchando eventos *dragend* para posibles movimientos de dicho marcador.

La finalidad es establecer el punto exacto en el que se ubica o ubicará un emplazamiento fijo, como un repetidor. De esta manera, podemos almacenar la ubicación todos los nodos a los que tenemos acceso para hacer conexiones, al igual que podemos hacer pruebas teóricas antes de desplazarnos al sitio.

Una vez hemos seleccionado la ubicación, queda ponerle un nombre y la altura a la que se elevarán los dispositivos en dicho emplazamiento (por ejemplo la altura del edificio más la de la torre).

Mediante Ajax, se envía una petición al servidor que ejecuta el script `subiremplazamiento.php` de manera asíncrona, recogiendo los datos y comprobando si ya existen o no en la base de datos. Una vez comprobado, si todos los datos son correctos, se inserta en la base de datos y se devuelve un mensaje de éxito.

Una vez hecho esto, ya tenemos a nuestra disposición dicha ubicación en la página principal `index.php`, para poder seleccionarla como punto de origen del radioenlace.

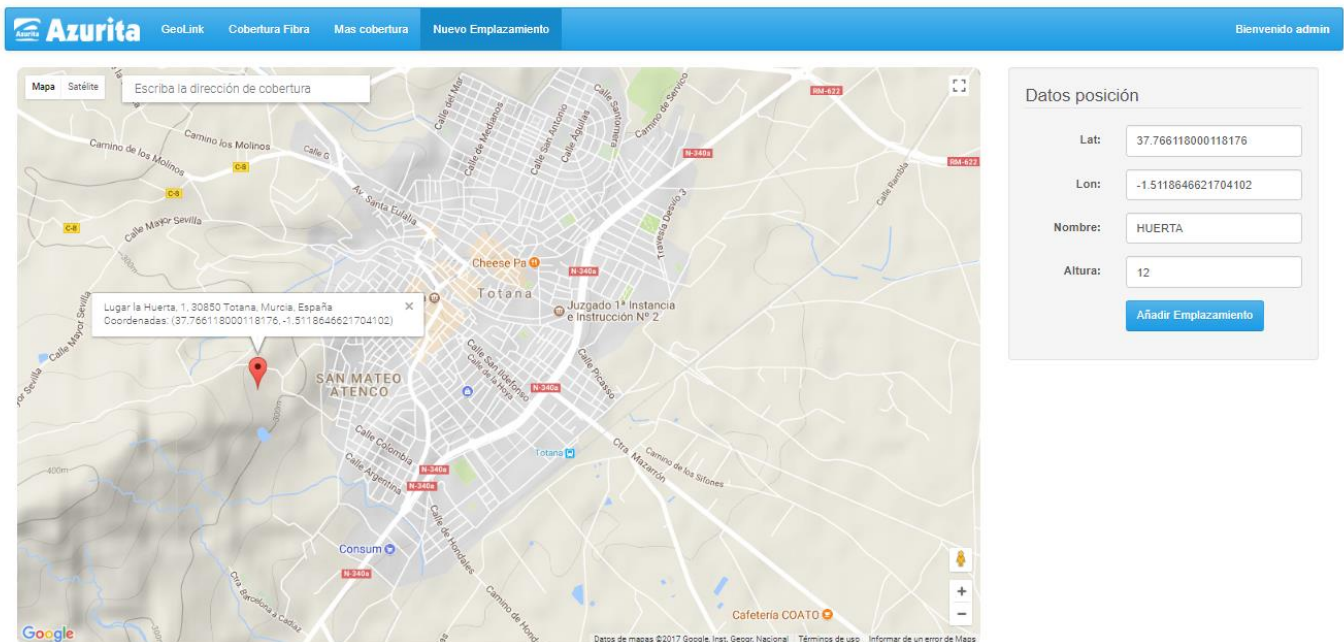


Figura 42- Añadir emplazamiento

3.3.- Base de datos

La base de datos está implementada mediante MySQL. Hemos escogido este tipo de datos relacional debido al tipo de uso que se le va a dar en nuestra aplicación.

Dado que no vamos a almacenar un gran volumen de datos ni vamos a generar consultas masivamente, MySQL ha demostrado responder de una forma rápida y eficaz a consultas espontáneas generadas desde nuestra aplicación. La idea principal era utilizar bases de datos NoSQL, como por ejemplo MongoDB u OrientDB. Bases de datos que generan estructuras de datos en documentos similares a JSON con un esquema dinámico.

Sin embargo, para almacenar datos de coordenadas, nombres, valores de potencias, etc, nos ha resultado mucho más sencillo trabajar con MySQL, almacenando estos datos en tablas relacionales.

Principalmente, se han creado 5 tablas:

- AP: Almacena las características de los dispositivos ubicados en el punto emisor del radioenlace.
- CPE: Almacena las características de los dispositivos ubicados en el punto receptor del radioenlace.
- NODOS: Almacena las coordenadas, nombre y altura de las ubicaciones de los emplazamientos fijos.
- USUARIOS: Almacena los datos de los usuarios.
- ORGANIZACIONES: Almacena los datos de las organizaciones.

3.3.1- phpmyadmin

phpMyAdmin es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet. Mediante esta herramienta podemos crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos, etc.



Figura 43- PhpMyAdmin

Mediante esta herramienta, vamos a gestionar los datos para modelarlos acorde a las necesidades de nuestra web.

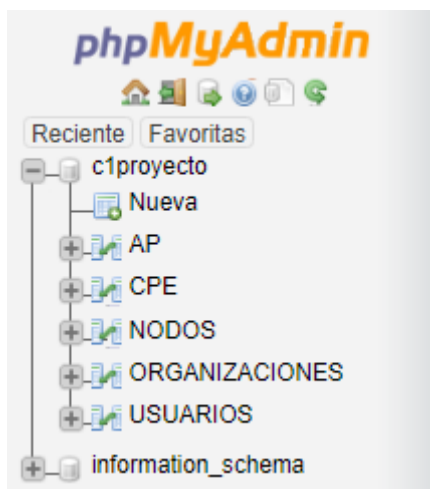


Figura 44- Interfaz de phpMyAdmin

El apartado APs está formado por la siguiente tabla:

NOMBRE_AP	PTX	RANGOFREQ	COMENTARIOS
-----------	-----	-----------	-------------

La columna NOMBRE_AP contiene el nombre del dispositivo transmisor que aparecerá primero en el formulario de nuestra página principal. PTX almacena el dato de la potencia de transmisión de dicho dispositivo. Este dato representa la PIRE del radioenlace, ya se almacena la potencia a la que transmite + la ganancia de la antena en caso de que la hubiere. RANGOPREQ almacena el rango de funcionamiento del dispositivo, en MHz. Y Por último, COMENTARIOS, almacena otros datos de interés, como por ejemplo un enlace a la hoja de características.

El segundo apartado, CPEs, es similar al anterior pero con dispositivos receptores. Está formado por la siguiente tabla:

NOMBRE	GRX	RANGOF	COMENTARIOS
--------	-----	--------	-------------

La columna NOMBRE guarda el nombre que se mostrará en el formulario principal, la ganancia GRX será la ganancia de recepción que se sumará al valor de la PIRE – pérdidas calculadas. RANGOF el rango de funcionamiento del dispositivo, y por último COMENTARIOS almacena otros datos de interés.

El tercer apartado contiene la tabla llamada NODOS, que guarda todos los datos de los emplazamientos almacenados por el usuario. Está formada por:

NOMBRE	LAT	LON	ALTURA	COMPANY	COMENTARIOS
--------	-----	-----	--------	---------	-------------

La columna NOMBRE incluye el nombre del emplazamiento denominado por el usuario. LAT Y LON almacenan las coordenadas de dicho emplazamiento. ALTURA, en metros, de la torre donde se encuentran los dispositivos en dicho emplazamiento. COMPANY almacena la organización o compañía a la que pertenece el usuario que lo creó, de este modo podemos separar de una manera sencilla los nodos de cada grupo de usuarios. Por último, COMENTARIOS contiene otros datos de interés.

A continuación tenemos la tabla ORGANIZACIONES, formada por:

NOMBRE	CLAVE	USUARIO	COMENTARIOS
--------	-------	---------	-------------

La columna NOMBRE se refiere al nombre de la organización o compañía, es la referencia a la compañía anteriormente mencionada en la tabla nodos. CLAVE es una contraseña de referencia unívoca a cada compañía, es decir, un usuario puede unirse a una organización, para poder tener acceso a sus nodos, solamente si sabe la contraseña. El USUARIO almacena el nombre del usuario administrador y creador de dicha compañía y, por último, COMENTARIOS contiene otros datos de interés.

Por último tenemos la tabla USUARIO, formada por:

ID	USERNAME	PASSWORD	EDAD	SEXO	ADMIN	COMPANY
----	----------	----------	------	------	-------	---------

La columna ID es un identificador unívoco del usuario. USERNAME y PASSWORD son las credenciales de autenticación del usuario. EDAD y SEXO son datos personales adicionales del usuario. ADMIN es una variable booleana que determina si el usuario es administrador y creador de una organización o simplemente pertenece a ella y, por último, COMPANY almacena el nombre de la compañía a la que pertenece dicho usuario.

4.- Resultados

4.1- Escenario 1

En primer lugar vamos a estimar la señal de un radioenlace con tecnología Mikrotik. Vamos a estudiar en concreto uno situado en el bloque de pisos de la Calle Legazpi Nº 22:

Radio Name	MAC Address	Interface	Uptime	AP	W...	Last Activit...	Tx/Rx Signal ...	Tx/Rx Rate
BELMA CAYO CARTORCENO	D4:CA:6D:1B:C5:EB	W	14d 08:15:22	no	no	0.000	-55/-58	54.0Mbps...
COMFORT TREGO POKU	D4:CA:6D:BD:D9:69	W	5d 12:31:38	no	no	0.020	-52/-52	54.0Mbps...
EDIFICIO C/ LEGAZPI Nº 22	00:0C:42:8A:00:46	W	16d 07:19:14	no	no	0.000	-49/-52	58.5Mbps...
ENCARNA MARTINEZ TUDELA	4C:5E:0C:BF:29:94	W	17d 08:31:56	no	no	0.000	-35/-37	54.0Mbps...
EXPLOTACIONES GANADERAS HONDALES	00:0C:42:61:04:A9	W	2d 05:16:56	no	no	0.010	-65/-71	54.0Mbps...
GAMBA DE ORO	4C:5E:0C:25:B8:29	W	2d 00:37:15	no	no	0.000	-41/-46	54.0Mbps...
GREGORIO MARTINEZ NAVARRO	4C:5E:0C:C3:CB:65	W	3d 06:49:34	no	no	0.020	-51/-53	72.2Mbps...
LUIS MILTON CALAPINA TIBAN	4C:5E:0C:23:A1:6F	W	3d 08:57:04	no	no	0.000	-30/-33	54.0Mbps...
LUIS ZAMORA CANOVAS	4C:5E:0C:60:62:C4	W	11d 04:02:12	no	no	0.010	-46/-48	54.0Mbps...
PEDRO CASANOVA VALENZUELA	D4:CA:6D:BF:B1:59	W	2d 09:20:13	no	no	0.030	-62/-63	65.0Mbps...
PEDRO JOSÉ HERNÁNDEZ ROMERO	D4:CA:6D:EE:90:37	W	18d 03:48:22	no	no	0.000	-60/-61	43.3Mbps...
SANTIAGO TUDELA GALLARDO	00:0C:42:6C:81:CE	W	17d 21:50:14	no	no	0.010	-51/-58	54.0Mbps...

Figura 45- Escenario 1 Figura 1

Como podemos observar en el apartado clientes del sectorial, la señal de dicho CPE es de -49/52 dBm. Vamos a ver en nuestra aplicación, configurando correctamente los parámetros, la potencia que estima será la siguiente:



Datos Radioenlace

Nodo:

Frecuencia:

AP:

PIRE:

CPE:

Ganancia Rx:

Altura Receptor:

Potencia recibida: -50.23 dBm

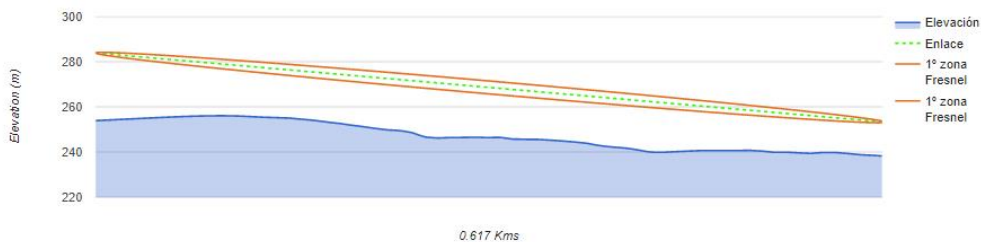


Figura 46- Escenario 1 Figura 2

Estableciendo la frecuencia de funcionamiento a 5640 MHz, la PIRE a 30dBm y la ganancia de recepción del panel direccional de la Groove 52HPn en 23dBi, obtenemos una estimación de la potencia recibida en -50.23dBm. Nos coincide con los -49/-52 (Tx/Rx) que nos devuelve el sector emisor correspondiente a dicho ejemplo y, por tanto, la predicción es bastante correcta aun estando en entorno urbano, ya que en este caso se trata de edificios altos y con buena visibilidad.

4.2- Escenario 2

En este segundo escenario vamos a estimar la señal de un radioenlace con tecnología Cambium Networks. Accediendo al dispositivo del cliente elegido para el estudio, podemos observar los siguientes valores:

Home

Device Name	[REDACTED]
Operating Frequency	5340 MHz
Operating Channel Bandwidth	20 MHz
Transmitter Output Power	29 dBm
Antenna Gain	16 dBi
Country	Other
Subscriber Module Mode	TDD
Subscriber Module Priority	Normal
Network Mode	Bridge
Downlink RSSI	-64 dBm
Downlink SNR	22 dB
Uplink MCS	MCS 12
Downlink MCS	MCS 12

Figura 47- Escenario 2 Figura 1

Estableciendo la PIRE a 30dBm del emisor, la frecuencia de trabajo a 5340 Mhz y la ganancia del receptor con un Force 180 a 16dBi, tenemos una señal de -64dBm. Vamos a ver qué resultado nos devuelve nuestra aplicación:



Datos Radioenlace

Nodo:

Frecuencia:

AP:

PIRE:

CPE:

Ganancia Rx:

Altura Receptor:

Recalcular Potencia Recibida

Potencia recibida: **-64.13 dBm**

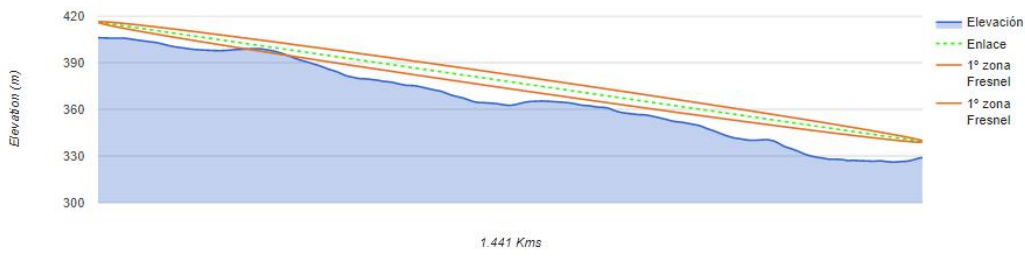


Figura 48- Escenario 2 Figura 2

La potencia recibida calculada es de -64.13 dBm. Nuevamente obtenemos un valor bastante fiel a la realidad.

4.3- Escenario 3

Por último, en este tercer escenario vamos a estimar la señal de un radioenlace con tecnología Alentia Systems. Accediendo al dispositivo del cliente elegido para el estudio, podemos observar los siguientes valores:

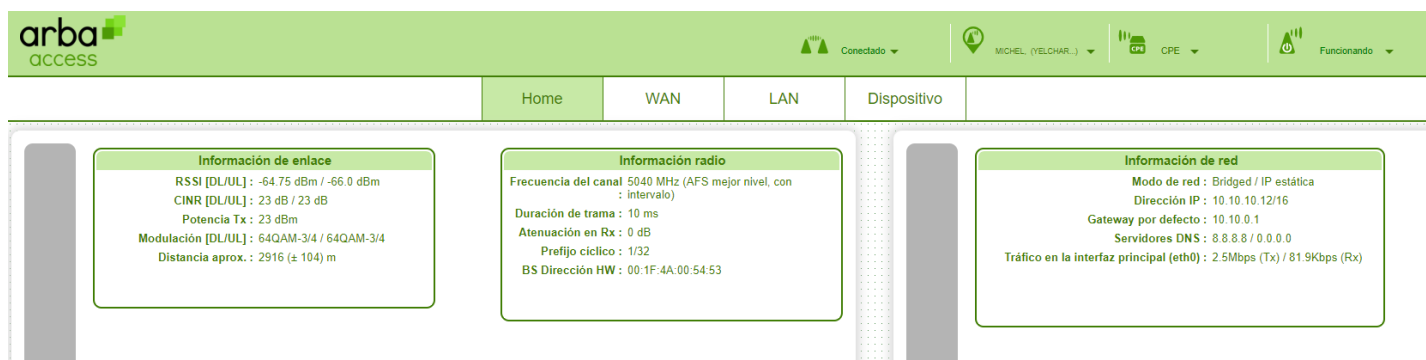


Figura 49- Escenario 3 Figura 1

Podemos destacar la frecuencia de funcionamiento, 5040 MHz, los 23 dB de la ganancia receptora y como siempre, la PIRE al máximo permitido en España, 30 dBm.

La señal recibida RSSI real es de -64.75/-66 dBm, y según nuestra aplicación:

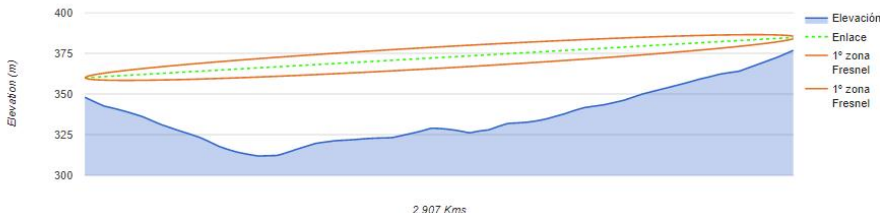
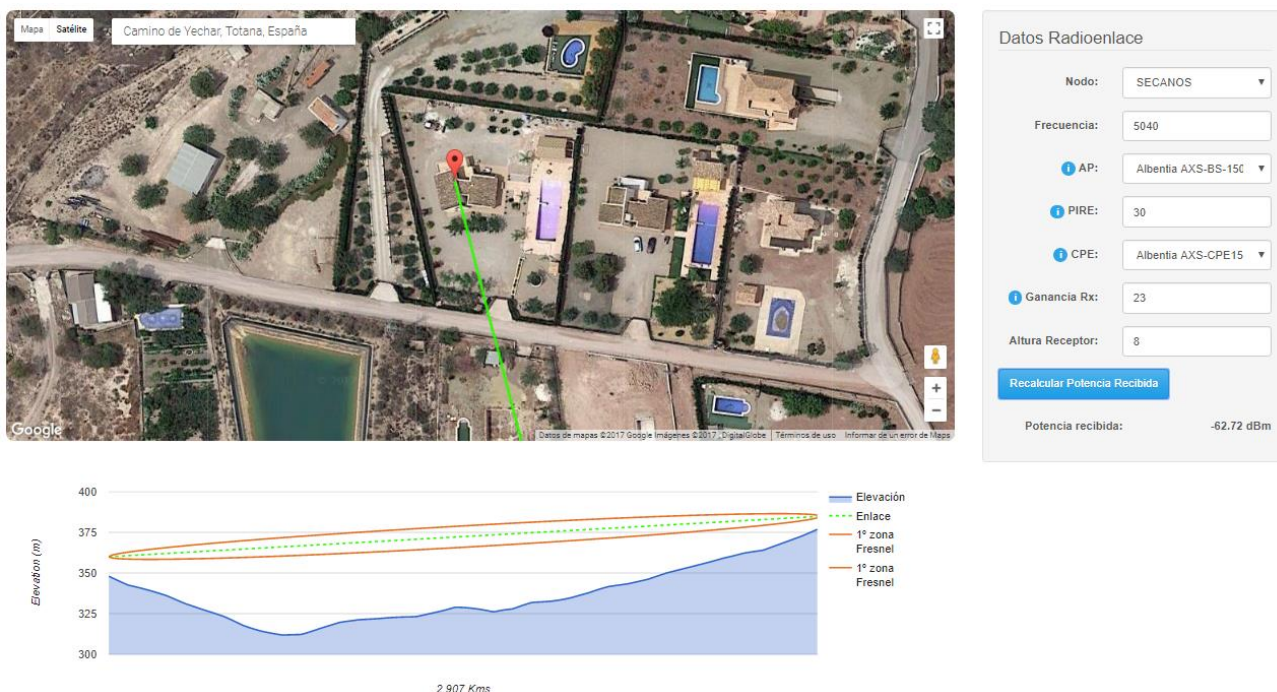


Figura 50- Escenario 3 Figura 2

Obtenemos una señal estimada de -62.72dBm. Esta diferencia la podemos achacar a un bloque de vegetación pinosa de aproximadamente 2-3m de espesor situada más adelante que, según vimos en el apartado 2.2.1.3 se estiman unas pérdidas de unos 2dB.

En este escenario también obtenemos de nuevo un caso de éxito en la estimación.

5.- Conclusiones y Líneas Futuras

5.1.- Conclusiones.

Con este proyecto se ha realizado una estimación teórico-experimental de la señal recibida en un radioenlace empleando tres tipos de tecnologías distintas aplicadas a casos reales.

En los tres casos, la estimación ha resultado bastante aproximada a la realidad, por lo que podemos concluir con éxito este proyecto para nuestro objetivo inicialmente planteado. Además, se ha añadido el apartado referente a la cobertura de fibra óptica que ha agilizado mucho tiempo a la hora de resolver las consultas de los clientes.

El otro objeto de estudio, a API de Google Maps, ha respondido satisfactoriamente combinada en este caso con tecnologías web, demostrando que puede dar mucho juego a la hora de realizar proyectos relacionados con las TIC.

Tanto los resultados obtenidos para la empresa, como mi valoración personal, han sido muy positivas a la hora del desarrollar nuevas capacidades de i+D en el ámbito de las TIC, combinando conceptos aprendidos en el Grado de Ingeniería Telemática, así como en el presente Máster de Ingeniería de Telecomunicaciones.

5.2.- Líneas Futuras.

A modo de conclusión se proponen las siguientes líneas futuras:

- Realizar ampliaciones o servicios extra a la página principal, como puede ser la configuración de la ubicación y dirección de antenas sectoriales en ciertos emplazamientos.
- Investigación de Google Maps Directions API, para poder desarrollar el punto anterior, calcular el área de apertura de los sectoriales en función de la dirección a la que apuntan, por ejemplo.
- Desarrollar la aplicación para otros dispositivos, como pueden ser Android o iOS, con el consecuente uso de la API de Google Maps para esta plataforma.
- Estudiar más en detalle la potencia recibida en casos de visión nula, con la posible aplicación de técnicas teórico-experimentales de difracción de ondas o reflexión.

6.- Referencias

[API] <https://bbvaopen4u.com/es/actualidad/que-es-una-api-y-que-puede-hacer-por-mi-negocio>

[APIMAPS] <https://developers.google.com/maps/?hl=es-419>

[APIMAPSPET] <https://developers.google.com/maps/pricing-and-plans/#details>

[APISERV] <https://console.developers.google.com/apis/credentials?project=web-tfm&authuser=1>

[CCR1036] <https://mikrotik.com/product/CCR1036-12G-4S-EM>

[CSS] https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada

[ELEVATION]

<https://developers.google.com/maps/documentation/elevation/intro?authuser=1#CreatingElevationCharts>

[EPMP1000] <http://www.globalys-usa.com/PDF/ficha-EPMP1000.pdf>

[FORCE180] <https://www.storewifidom.com/201b/31-force-180.html>

[FORCE200] <https://www.storewifidom.com/201b/34-force-200.html>

[FRESNEL] https://es.wikipedia.org/wiki/Zona_de_Fresnel

[GEOCODING]

<https://developers.google.com/maps/documentation/geocoding/start?authuser=1>

[GEOLOCATION] <https://developers.google.com/maps/documentation/geolocation/intro?hl=es-419>

[HTML5] <https://es.wikipedia.org/wiki/HTML5>

[ITU-R P.341-4] https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.341-4-199510-S!!PDF-S.pdf

[ITU-R P.525-2] https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.525-2-199408-I!!PDF-S.pdf

[ITU-R P.526] https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.526-13-201311-I!!PDF-S.pdf

[ITU-R P.530-7] https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.530-7-199708-S!!PDF-S.pdf

[ITU-R P.833-8] https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.833-8-201309-S!!PDF-S.pdf

[JAVASBIB]

<https://developers.google.com/maps/documentation/javascript/libraries?authuser=1&hl=es-419>

[JAVASCAPI]

<https://developers.google.com/maps/documentation/javascript/tutorial?authuser=1&hl=es-419>

[JAVASCRIPT] <https://es.wikipedia.org/wiki/JavaScript>

[MKT] <https://es.wikipedia.org/wiki/MikroTik>

[MYSQL] <https://es.wikipedia.org/wiki/MySQL>

[PHP] <https://es.wikipedia.org/wiki/PHP>

[PLACES] <https://developers.google.com/places/web-service/intro?authuser=1>

[PMP] https://es.wikipedia.org/wiki/Punto_a_multipunto

[PTP] https://es.wikipedia.org/wiki/Comunicaciones_punto-a-punto

[RFESCC] <https://www.rfelements.com/assets/Uploads/Datasheet-Sector-Carrier-Class-5-20.pdf>

[TETRAANT5] <https://www.interprojekt.it/elboxrf-tetraant-19dbi-p-1523.html>