



*Sistema de geolocalización en tiempo real, aplicado al registro de recorridos del vehículo eléctrico, de la Universidad Católica de Cuenca*

*Geolocation system in real time, applied to the registration of electric vehicle routes, of the Catholic University of Cuenca*

*Sistema de geolocalização em tempo real, aplicado ao registro de rotas de veículos elétricos, da Universidade Católica de Cuenca*

Hermes Israel Álvarez-Pinos <sup>I</sup>  
[hialvarezp42@est.ucacue.edu.ec](mailto:hialvarezp42@est.ucacue.edu.ec)  
<https://orcid.org/0000-0003-3385-3037>

Carlos Alfredo Calle-Llguizaca <sup>II</sup>  
[cacallel43@est.ucacue.edu.ec](mailto:cacallel43@est.ucacue.edu.ec)  
<https://orcid.org/0000-0003-4307-9866>

**Correspondencia:** [hialvarezp42@est.ucacue.edu.ec](mailto:hialvarezp42@est.ucacue.edu.ec)

Ciencias de la tecnología  
Artículo de investigación

\***Recibido:** 20 de febrero de 2020 \***Aceptado:** 11 de mayo de 2020 \* **Publicado:** 31 de mayo de 2020

- I. Ingeniero en Electrónica, Universidad Católica de Cuenca, Cuenca, Ecuador.
- II. Ingeniero en Electrónica, Universidad Católica de Cuenca, Cuenca, Ecuador.

## Resumen

Este trabajo presenta un sistema de geolocalización, para el vehículo eléctrico de la Universidad Católica de Cuenca, que permitirá obtener los datos de su ubicación en tiempo real, y generará alarmas si excede a los límites dentro de sus zonas de recorrido. El prototipo desarrollado tributa al proyecto de investigación Smart UniverCity Cato 2.0, que pretende generar propuestas que impulsen al fomento de tecnologías de la información y comunicación, para obtener sistemas inteligentes que nos lleven al desarrollo de una ciudad inteligente, además, es inclusivo por tener la capacidad de expandirse en trabajos futuros que mejoren la automatización del vehículo. En el presente trabajo se muestran la metodología utilizada para el desarrollo y programación de los dispositivos de hardware y software utilizado, así como también, el desarrollo de una página web, la cual permite cumplir los objetivos planteados y determinar su registro de recorridos.

**Palabras claves:** Vehículo eléctrico; arduino; geolocalización; ciudad inteligente; alarma.

## Abstract

This work presents a geolocation system for the electric vehicle of the Catholic University of Cuenca, which will allow obtaining the data of its location in real time, and will generate alarms if it exceeds the limits within its travel areas. The developed prototype pays tribute to the Smart UniverCity Cato 2.0 research project, which aims to generate proposals that promote the promotion of information and communication technologies, to obtain intelligent systems that lead us to the development of a smart city, in addition, it is inclusive for having the Ability to expand into future jobs that improve vehicle automation. In this work, the methodology used for the development and programming of the hardware and software devices used is shown, as well as the development of a web page, which allows to meet the stated objectives and determine its record of routes.

**Keywords:** Electric vehicle; arduino; geolocation; smart city; alarm.

## Resumo

Este trabalho apresenta um sistema de geolocalização para o veículo elétrico da Universidade Católica de Cuenca, que permitirá obter os dados de sua localização em tempo real e gerará alarmes se exceder os limites dentro de suas áreas de viagem. O protótipo desenvolvido presta homenagem ao projeto de pesquisa Smart UniverCity Cato 2.0, que tem como objetivo gerar propostas que promovam a promoção de

tecnologias de informação e comunicação, obter sistemas inteligentes que nos levem ao desenvolvimento de uma cidade inteligente; além disso, é inclusivo por ter o Capacidade de expandir para trabalhos futuros que melhoram a automação de veículos. Neste trabalho, é mostrada a metodologia utilizada para o desenvolvimento e a programação dos dispositivos de hardware e software utilizados, bem como o desenvolvimento de uma página da Web, que permite atender aos objetivos estabelecidos e determinar seu registro de rotas.

**Palavras-chave:** Veículo elétrico; arduino; geolocalização; cidade inteligente; alarme.

## Introducción

Con el pasar del tiempo, el desarrollo tecnológico en la industria automovilística ha crecido, alcanzando su modernización, el mismo que permite el crecimiento de vehículos inteligentes, los cuales han evolucionado para satisfacer las necesidades de la sociedad actual. En los últimos años ha incrementado el desarrollo de este tipo de vehículos, en 2018 el porcentaje de venta incrementó un 28,6%, prorrogando con un índice creciente en los años subsiguientes [1]. Esta creciente venta se debe a la innovación que se ha ejecutado, la misma que va desde controles de cruceo inteligentes, hasta sistemas de rastreo en tiempo real, lo que brinda un modo de conducción tangible, siendo reflejado en el compromiso que tienen las empresas, con la creación de vehículos eléctricos, inteligentes y seguros.

Se estima que el valor promedio de un vehículo eléctrico oscila entre 14,990 y 34,990 dólares, según las características del modelo [2], además, la seguridad para los propietarios es importante conociendo que el índice delictivo creciente en el país [3], haciendo indispensable la protección del patrimonio de las personas, mediante una planeación preventiva, para disminuir las probabilidades de robo.

Esta investigación hace referencia a la implementación de un dispositivo de geolocalización y el desarrollo de una página web, el cual se implementó en el vehículo eléctrico de la Universidad Católica de Cuenca, con la intención de brindar seguridad al usuario y entregar notificaciones en tiempo real de uso del automóvil, conociendo que, entre los años 2015 y 2017 se han recuperado 2,363 vehículos de 13,271 reportados como robados [2].

Un problema que padece la sociedad ecuatoriana es que el índice delictivo se ha desarrollado conforme se declina la situación económica del país, entre ellos, se puede deducir que el

problema de desempleo en Ecuador, va de la mano con el hurto [3], es por esto, que el robo de vehículos es uno de los delitos que suscita mayor connotación social, siendo esta una afección que hostiga al país. Una forma de combatir este problema, es el método de planeación preventiva a través del uso de la tecnología, para este caso particular, se utiliza geolocalización vehicular para su ubicación en tiempo real. [4]

Según [4], la geolocalización indica la posición de un teléfono móvil, vehículo, o persona, la característica principal de este tipo de sistema es monitorear la ubicación del objeto a rastrear en tiempo real, brindando seguridad al mismo, estos sistemas de planeación preventiva se consideran como una solución viable, para reducir la probabilidad de hurto.

Los sistemas de geolocalización, mediante una dirección de red IP, permiten a las empresas suministrar contenido relevante de forma sencilla, pues a más de mantener la confidencialidad, se adapta a la ubicación del usuario cuando se necesite[4]. Estos sistemas utilizan información de coordenadas geográficas, para determinar la ubicación exacta en tiempo real, en efecto, se considera revelador la consumación de este mecanismo de geolocalización, en el vehículo eléctrico de la Universidad Católica de Cuenca, que permita monitorear su ubicación en tiempo real.

El presente trabajo se compone de cinco secciones. En la primera sección se expone un panorama general del problema y un estudio del estado del arte. En la segunda sección se explica la metodología utilizada para la solución de cada tarea ejecutada. Los resultados del desarrollo de la investigación se exponen en la tercera sección, mientras que la discusión de la investigación se detalla en la cuarta sección, y, finalmente las conclusiones, se presentan en la quinta sección.

## **Materiales y Métodos**

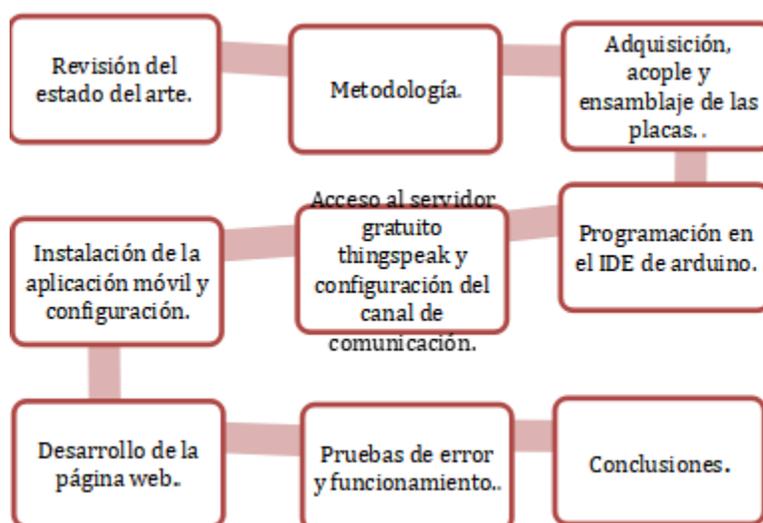
La investigación en su desarrollo utilizó un método científico, y la revisión del estado del arte y literatura, para desarrollar un dispositivo, a través de pruebas de ensayos y error, logrando disponer de un sistema que funcione correctamente en base a los requerimientos planteados en el objetivo.

Durante el desarrollo se definió 9 fases distribuidas de la siguiente manera:

1. Revisión del estado del arte.
2. Metodología.

3. Adquisición, acople y ensamblaje de las placas.
4. Programación en el IDE de arduino.
5. Acceso al servidor gratuito ThingSpeak y configuración del canal de comunicación.
6. Desarrollo de la página web.
7. Instalación de la aplicación móvil y configuración.
8. Pruebas de error y funcionamiento.
9. Conclusiones.

**Figura 1.** Metodología.



Fuente: Autor.

## Resultados

La arquitectura propuesta en este trabajo asume la medida de exponer un mecanismo móvil de geolocalización, que impulsa a conocer la ubicación exacta del vehículo eléctrico; a continuación, en la Figura, se observa el montaje de las placas Arduino, en la estructura de acrílico.

**Figura 2.** Estructura montaje y cableado cableado.



**Fuente:** Autor.

Ya consumado el acoplamiento, cableado y refuerzo de las placas, se procede a encapsular estos elementos, en la

Figura . Se muestra el encapsulado de los equipos electrónicos.

**Figura 3.** Encapsulado de los equipos electrónicos.



**Fuente:** Autor.

El desarrollo del software se ha seccionado, considerando los dispositivos que intervienen en el proyecto. Para arduino el software está enfocado en programación por funciones, respetando la estructura propia del código de programación. Para la comunicación dentro de la red GPRS entre equipos en tiempo real, se usa como servidor el software gratuito de ThingSpeak, la página web desarrollada en lenguaje HTML y la aplicación para teléfono móvil. Adicionalmente para la comunicación por sms se usa la red GSM.

## **Programación en Arduino**

Como cualquier programa tiene su estructura y su esquema de funcionamiento, este caso no es la excepción, ya que se ha desarrollado un software que contiene librerías, declaración de variables y funciones adicionales al void setup y el void loop.

Sabiendo que arduino tiene sus dos funciones imprescindibles, void setup y el void loop. Se desarrolló una estructura basada en bucles de repetición encargados de inicializar todas las herramientas necesarias del sim 808. Estos bucles de repetición fueron programados en funciones adicionales a las principales, pero son llamadas desde el void set up, el cual tiene la característica principal de ejecutarse una sola vez desde que se inicia el dispositivo arduino. Además, el void setup también es el encargado de inicializar las librerías y el puerto serial para comunicarse con el sim 808.

Una vez inicializado el dispositivo en su totalidad y con datos correctos por parte del GPS, se procede a unirse a la red de telefonía móvil para enviar datos como velocidad, latitud, longitud, distancia del punto de referencia. Todos estos datos son obtenidos y calculados mediante funciones implementadas en el programa. Estas funciones son llamadas desde la función principal void loop la cual tiene la característica particular de ejecutarse repetitivamente, durante todo el tiempo que este encendido el dispositivo.

El funcionamiento iterativo general se describe en la Figura 4. Se aprecia un diagrama de flujo que explica el funcionamiento que realiza el sistema para obtención de datos.

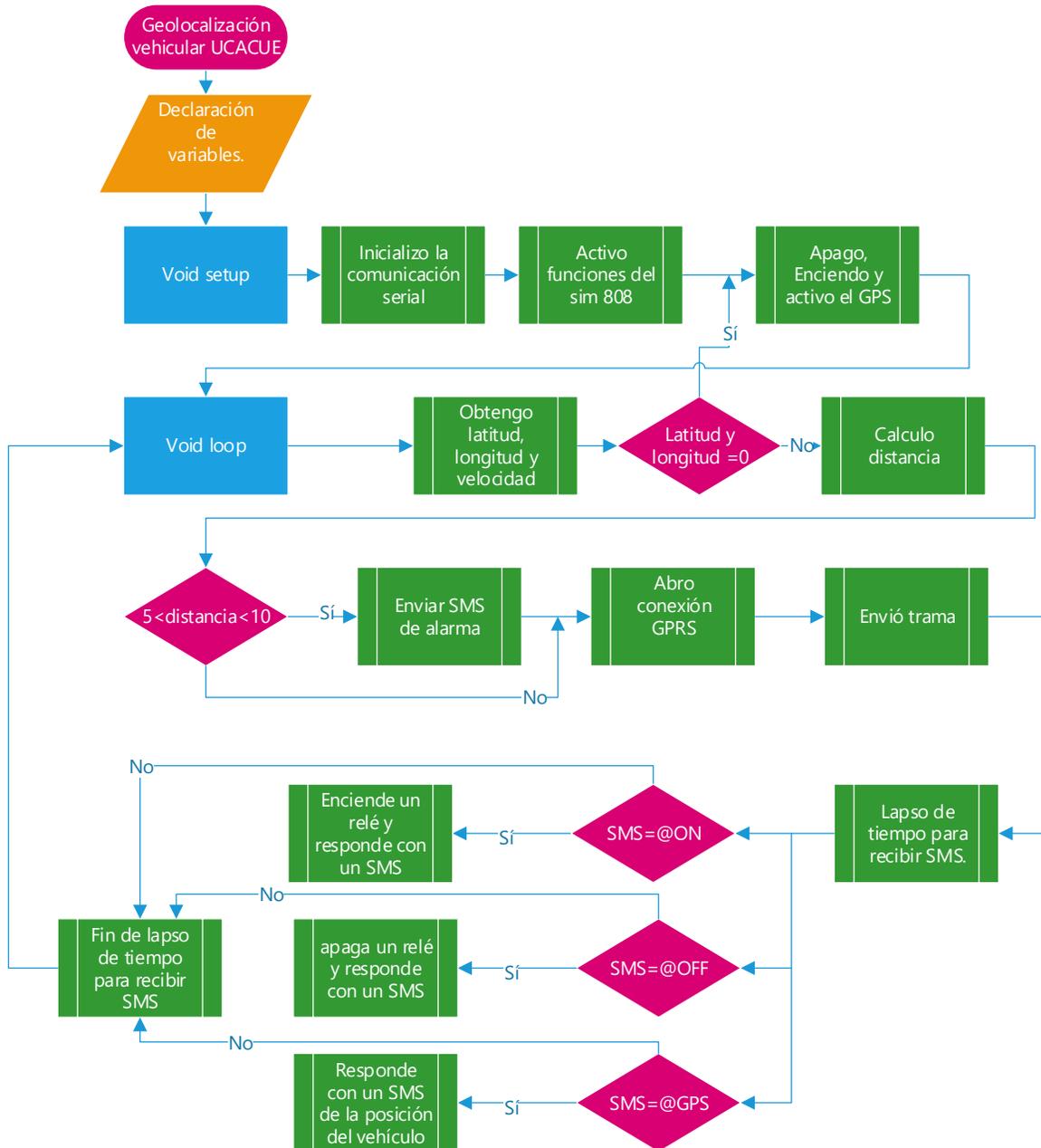


Figura 4. Diagrama de flujo del respectivo funcionamiento del dispositivo.

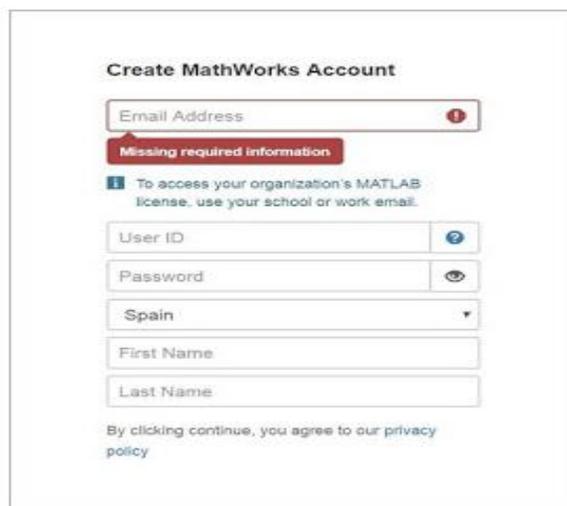
Fuente: Autor.

### Introducción a Thingspeak

ThingSpeak es un servidor web que admite subir datos mediante el uso de un canal específico designado. Los datos se almacenan en campos y pueden ser visualizados y analizados en vivo. Al mismo tiempo los datos se pueden leer mediante el uso de una clave api, que serán

imprescindibles al momento de trabajar con la aplicación móvil y la página web. Un aspecto a considerar es que solo permite subir o leer datos con una frecuencia mínima de hasta 15 segundos por canal en la versión gratuita[5]. Para trabajar con este servidor web es necesario iniciar sesión con la cuenta de MathWorks, la Figura 5 muestra el registro en ThingSpeak mediante la cuenta de MathWorks.

**Figura 5.** Registro en ThingSpeak mediante MathWorks

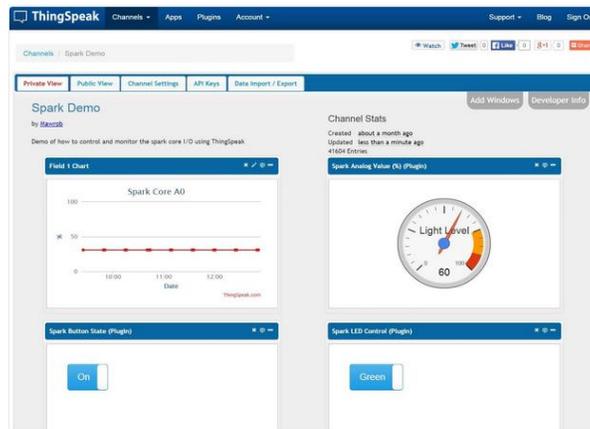
The image shows a web form titled "Create MathWorks Account". It contains several input fields: "Email Address" (with a red error message "Missing required information" and a note "To access your organization's MATLAB license, use your school or work email."), "User ID", "Password", a dropdown menu set to "Spain", "First Name", and "Last Name". At the bottom, there is a checkbox for "By clicking continue, you agree to our privacy policy".

Fuente.[5]

### Uso del ThingSpeak

Básicamente, ThingSpeak es una plataforma de Internet de las cosas, la cual recoge y almacena datos de los sensores en la nube, y facilita el desarrollo de aplicaciones (IoT), brindando la capacidad de visualizar los datos y actuar sobre estos [5], la Figura 6 presenta la app de acceso a través de <https://thingspeak.com/>, una vez dentro de esta página web se ingresa a “Sign Up” mediante una cuenta gratuita en MathWorks cómo se mencionó anteriormente.

**Figura 6.** Página web de ThingSpeak.



Fuente.[5]

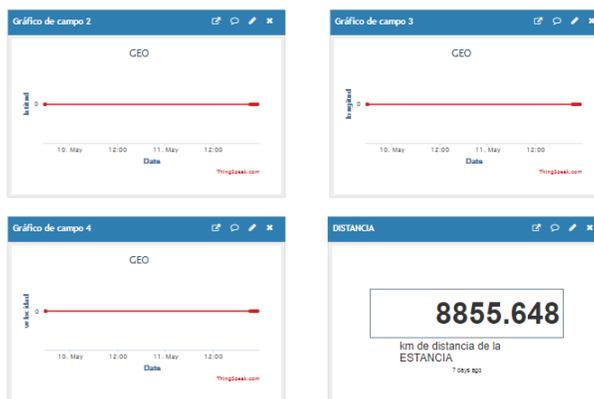
### Crear un canal

Para la creación de un canal y configurarlo se sigue los siguientes pasos:

Una vez ingresado a la plataforma de ThingSpeak escogemos la opción nuevo canal, y procedemos a llenar los datos requeridos. Los campos hacen referencia a los espacios de memoria destinados por el servidor para recibir los datos y almacenarlos, en este caso específico usamos los campos dos, tres y cuatro para agregar datos de latitud longitud y velocidad respectivamente. Existen otros requerimientos que se llenaran a conveniencia de uso como por ejemplo las vistas públicas o privadas.

Es indispensable el uso de claves API, para subir y leer datos del servidor. Una vez configurado el servidor y programado arduino se obtiene datos en el servidor como se ilustra en la Figura 6.

**Figura 7.** Canal creado en ThingSpeak.



Fuente. [6]

### Modificar canal

Cuando se requiera modificar, se accede a la opción configuraciones de canal, luego a los parámetros establecidos anteriormente, para la modificación de cualquier campo o propiedad, [7], a continuación, en la

Figura 8 se observa los campos que se pueden modificar.

Figura 8. Information en My Channel.

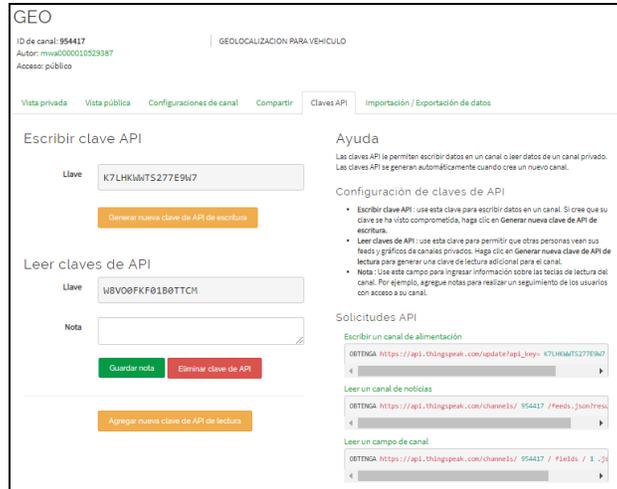
Configuraciones de canal	
Porcentaje completo	sesenta y cinco%
Canal ID	954417
Nombre	GEO
Descripción	GEOLOCALIZACION PARA VEHICULO
Campo 1	alarma <input checked="" type="checkbox"/>
Campo 2	latitud <input checked="" type="checkbox"/>
Campo 3	longitud <input checked="" type="checkbox"/>
Campo 4	velocidad <input checked="" type="checkbox"/>
Campo 5	contador de paquetes <input checked="" type="checkbox"/>

Fuente. [7].

Acceso a las claves api del servidor.

Para acceder a las claves API del servidor se dirige a la pestaña API key como se observa en la **¡Error! No se encuentra el origen de la referencia.** y escoge a conveniencia de sus requerimientos.

Figura 9. Acceso a las claves api del servidor.



Fuente. [10].

## Visualización de datos en Google Maps

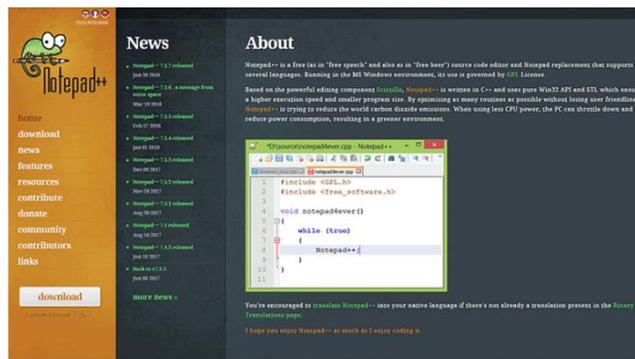
Para la visualización de los datos en la página web, se usaron las siguientes herramientas. Un editor de código para lenguaje html, la clave de lectura de datos del servidor ThingSpeak y Google Maps con su respectivo api key que permita su uso gratuito.

Para el desarrollo del programa se optó por usar Notepad++, el cual es un editor de código para lenguaje html.

## Editor de código para lenguaje html

Notepad++ es un editor de código gratuito que permite crear o modificar archivos que luego serán usados en una página web [10]. Para acceder a la descarga se accede sitio web <https://notepad-plus-plus.org>, la **¡Error! No se encuentra el origen de la referencia.** presenta la página oficial de Notepad++.

Figura 10. Página oficial de Notepad++



Fuente.[10].

Esta metodología exige la revisión de la carpeta de descargas, e iniciará el instalador en el escritorio del ordenador, cuando este solicite cambios en el equipo seleccione el idioma que desee la instalación, [10]. Para este caso en particular se optó por el idioma en español, como se presenta en la **¡Error! No se encuentra el origen de la referencia.**

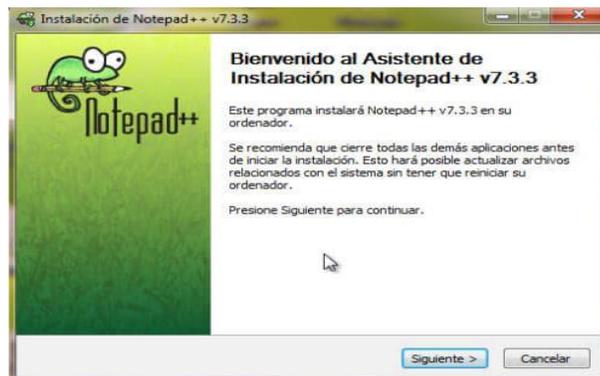
**Figura 11.** Selección de idioma de instalación.



Fuente. [10].

Una vez realizada la selección del idioma, se procede a realizar la instalación, la **¡Error! No se encuentra el origen de la referencia.** indica la instalación de Notepad++.

**Figura 12.** Instalación de Notepad



Fuente.[10]

Como se ha mencionado en el apartado anterior, Notepad++ es un editor de código para Windows, es semejante al editor de texto, pero entre otras cosas permiten guardar archivos con diferentes extensiones, la **¡Error! No se encuentra el origen de la referencia.** muestra el código de extensiones.

**Figura 13.** Código de extensiones en Notepad.

```

MATrix LAboratory (*.m)
MMIXAL file (*.mms)
Nimrod file (*.nim)
extended crontab file (*.tab;*.spfl)
MSDOS Style/ASCII Art (*.rfo)
Nullsoft Scriptable Install System script file (*.nisi;*.nsh)
Script source file (*.sxc)
Objective-C source file (*.mm)
Pascal source file (*.pas;*.pp;*.p;*.inc;*.lpr)
Perl source file (*.pl;*.pm;*.plx)
PHP Hypertext Preprocessor file (*.php;*.php3;*.php4;*.php5;*.phps;*.phpt;*.phtml)
PostScript file (*.ps)
Windows PowerShell (*.ps1;*.psm1)
Properties file (*.properties)
PureBasic file (*.pb)
Python file (*.py;*.pyw)
R programming language (*.r;*.s;*.spius)
REBOL file (*.r;*.reb)
registry file (*.reg)
Windows Resource file (*.rc)
Ruby file (*.rb;*.rbw)
Rust file (*.rs)
Scheme file (*.scm;*.smd;*.es)
Smalltalk file (*.st)
spice file (*.scp;*.out)
Structured Query Language file (*.sql)
Motorola S-Record binary data (*.mot;*.srec)
Swift file (*.swift)
Tool Command Language file (*.tcl)
Tektronix extended HEX binary data (*.tek)
TeX file (*.tex)
Visual Basic file (*.vb;*.vbs)
bd2tags file (*.12)
Verilog file (*.v;*.sv;*.vh;*.svh)
VHSIC Hardware Description Language file (*.vhdl;*.vhdl)
Visual Prolog file (*.pro;*.cl;*.i;*.pack;*.ph)
eXtensible Markup Language file (*.xml;*.xaml;*.xal;*.xalt;*.xsd;*.xul;*.kml;*.svg;*.mxml;*.xsmi;*.wsdl;*.xfl;*.diff;*.xbl;*.xbl;*.sitemap;*.gml;*.gpx;*.plist)
YAML Ain't Markup Language (*.yaml;*.yml)

```

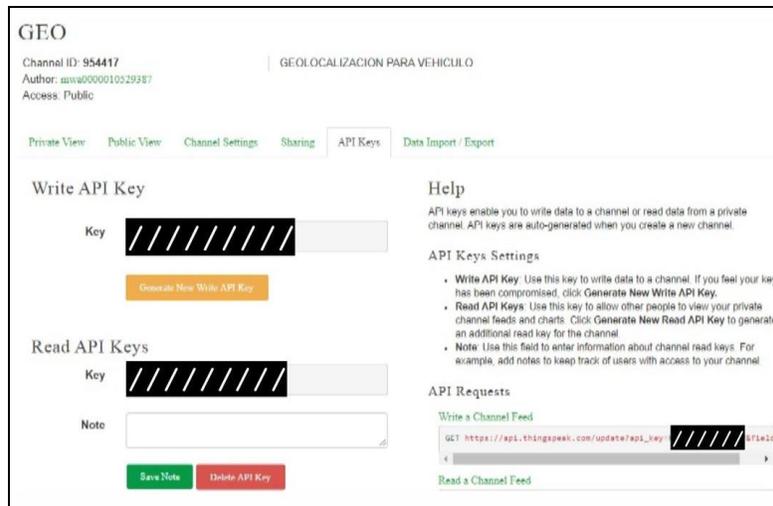
Fuente. [10]

## Obtención de clave api del servidor ThingSpeak y de Google Maps

Cabe recalcar que el uso de claves api es imprescindible para el funcionamiento correcto de la página web. En la programación html se requiere el uso de dos claves api, la primera hace referencia al acceso de los datos del servidor y la segunda que permite ingresar a la aplicación de Google Maps para el uso de sus mapas.

Para el acceso a la clave api del servidor de ThingSpeak es necesario crear un canal y dirigirse a la pestaña claves api, donde se seleccionará la clave necesaria para el tipo de requerimiento, para este caso específico se usó la clave api para la lectura de los datos de un canal, la figura 14 indica la obtención de clave del servidor ThingSpeak.

**Figura 15.** Claves del servidor ThingSpeak.



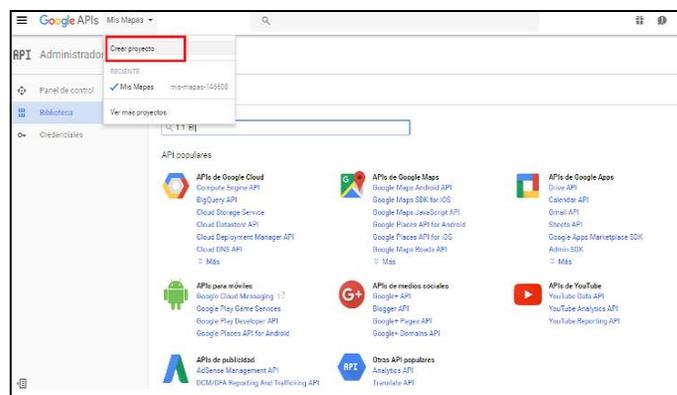
Fuente.[6]

Relativamente para obtener la clave api de Google es un tanto más complejo como se puede observar a continuación en este documento.

### Generar clave API de Google Maps y habilitarla

Para insertar una proyección de Google Maps en nuestro sitio, usando la API es necesario conseguir una clave gratuita. Para adquirirla es imprescindible el uso de una cuenta en GMAIL y luego dirigirse a la consola desarrolladora de Google, una vez en la consola, se procede a crear un nuevo proyecto y aceptar los términos de uso, la figura 16 presenta la creación del proyecto.

Figura 17. Creación del Proyecto.



Fuente. [6]

Posteriormente se procede a llenar los campos, como se detalla en la **¡Error! No se encuentra el origen de la referencia.**

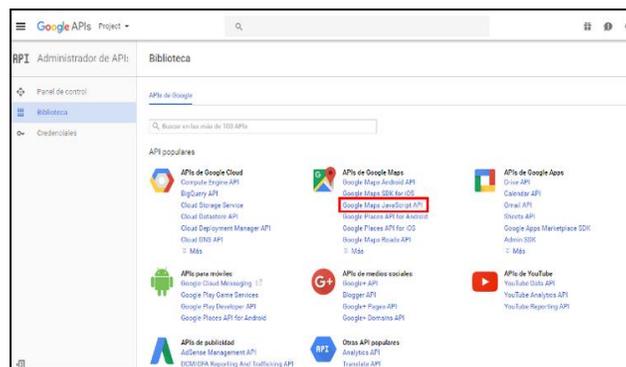
**Figura 18.** Campos a llenar en la creación del Proyecto



Fuente.[6]

Por consiguiente se agrega la API que se va a utilizar, en este caso se manejará JavaScript de Google Maps, [7] la **¡Error! No se encuentra el origen de la referencia.** enseña la creación de la selección de la API de JavaScript de Google Maps.

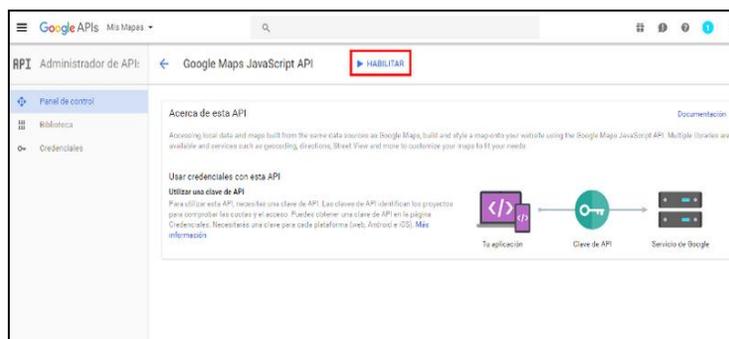
**Figura 19.** Selección de la API de JavaScript de Google Maps



Fuente.[6]

Una vez seleccionada, se la habilita. **La;Error! No se encuentra el origen de la referencia.** Figura 11 indica la creación de las credenciales.

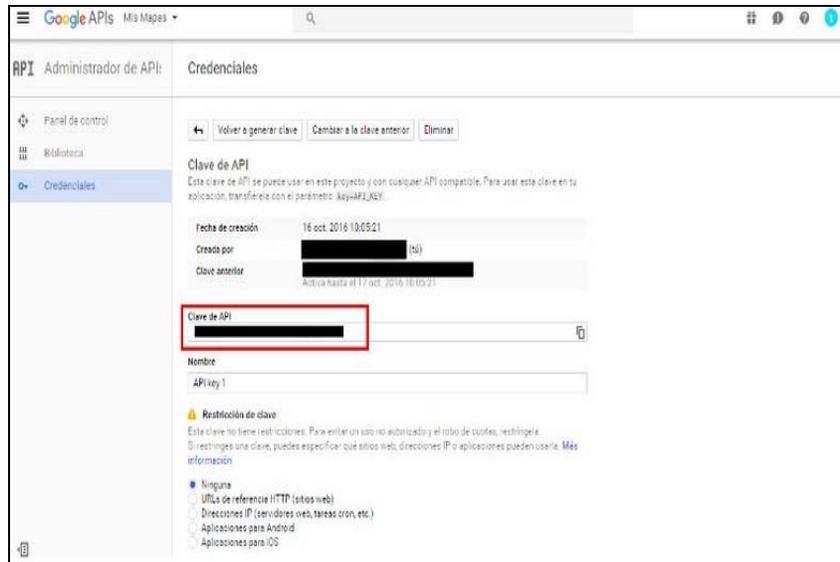
**Figura 20.** Creación de las credenciales



Fuente. [6].

Finalizado el proceso de creación, se asignará una clave que incorpora la API, la **¡Error! No se encuentra el origen de la referencia.** detalla este proceso.

**Figura 21.** Clave API asignada.

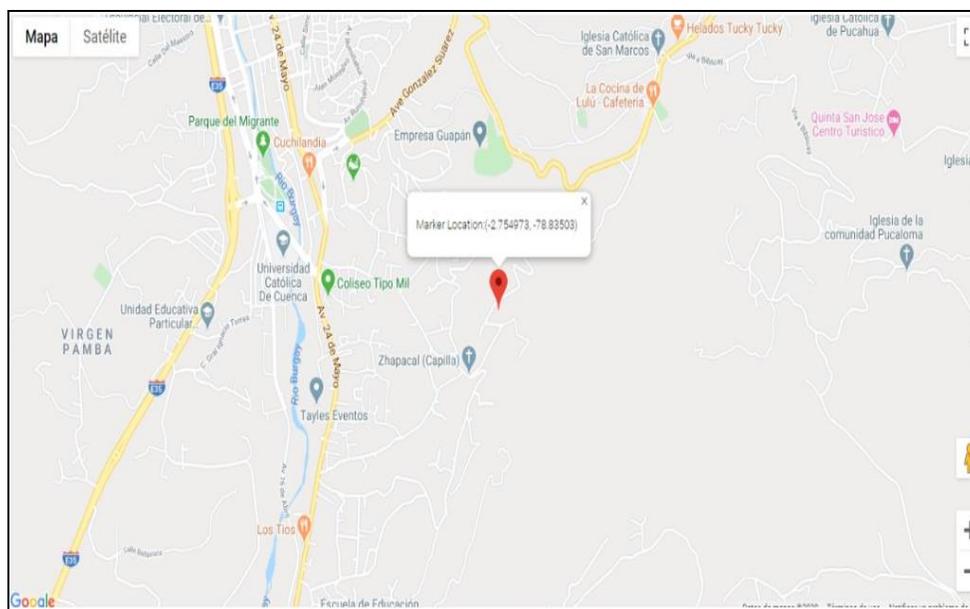


Fuente.[6]

En definitiva, ya con la clave API, se procede a adherir en la plataforma web que se está utilizando, a la hora de crear el proyecto se debe considerar las restricciones necesarias para el funcionamiento de la página WEB.

Como resultado, para visualizar la ubicación del vehículo en el mapa de Google se creó un archivo con extensión “HTML” en donde se implementó las claves obtenidas anteriormente. Todo el código a ejecutar tendrá que ir entre las etiquetas <HTML>, un archivo de este tipo cuenta con dos campos fundamentales: head y body, en el head o cabecera se describe información para el navegador como por ejemplo el nombre de la página y en el body se implementan las funciones que debe cumplir, además del contenido de la página.[7]. La Figura 13 visualiza la localización del vehículo.

Figura 22. Geolocalización en Google Maps usando las herramientas API.



Fuente.[11]

### Instalación del widget para Android

La instalación de esta aplicación ocupa suficiente ancho de banda y consume recursos del sistema, es por esta razón que se utilizan equipos de gama media en adelante para el funcionamiento correcto de este software, si el equipo Android a usar es de gama baja puede que funcione lento, [8]. Para añadir widgets a la pantalla se siguen los siguientes pasos:

- Seleccione la aplicación o pestaña widgets, ahí se visualizarán las opciones.
- Desplácese por la lista de widgets, deslizándose hacia la izquierda/derecha.
- Mantenga pulsado el widgets de la aplicación y arrástrelo a la ubicación que desee añadirlo.

Para mover un widgets manténgalo pulsado hasta que note una vibración y a continuación arrástrelo a otro punto de la pantalla, se ajusta el tamaño pulsándolo y arrastrándolo hacia una esquina, la Figura 23 visualiza la aplicación de play store ThingSpeak.

Figura 23. Aplicación de play store ThingSpeak.



Fuente.[5]

### Configuración del widgets para Android

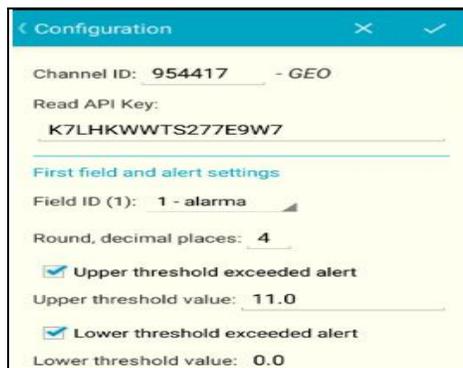
Esta aplicación está destinada a verificar la conexión y el estado actual de la cuenta, cada instancia de este software pauta el último valor de campo real del canal, haciendo que los valores se actualicen automáticamente cada 30 segundos, para obtener el nombre y el campo [9].

Se lo obtiene configurando el ID del canal, para lograrlo se asignan los siguientes pasos:

- Pulsar de manera sostenida la pantalla en blanco.
- Pulsar la pestaña widgets.
- Encuentre las opciones de los widgets.
- Manténgalo pulsado y arrástrelo a la pantalla de inicio.

A continuación, en la **¡Error! No se encuentra el origen de la referencia.** se muestra la configuración del widget.

Figura 24. Configuración del widget.

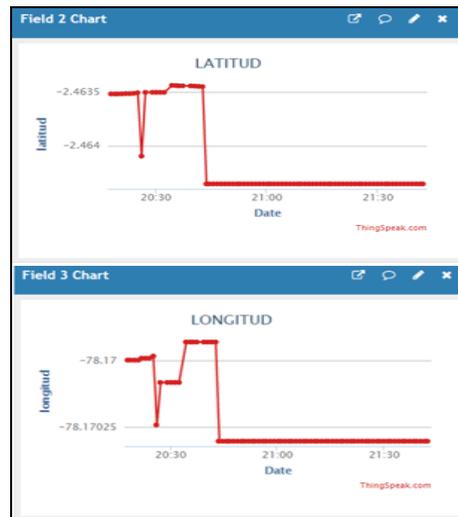


Fuente.[9]

## Discusión

El desarrollo del dispositivo se lo realizó en arduino y la tarjeta sim 808, obteniendo mediante coordenadas geográficas los datos de longitud, latitud, distancia, etc., cómo se presenta en la **¡Error! No se encuentra el origen de la referencia..**

**Figura 25.** Coordenadas visualizadas en ThingSpeak.



Fuente.[5].

La codificación en Arduino está orientada a calcular la distancia entre puntos específicos programados previamente y el vehículo a ser rastreado, una de las características fundamentales de este proyecto es el uso del módulo de telefonía móvil GPRS y GSM, para la transmisión y recepción de datos, el cual ayuda al monitoreo constante, los datos son enviados mediante la red GPRS hacia el servidor web, el mismo que recibe la información, la almacena y brinda estos datos a otras aplicaciones que permitirán ver la ubicación del vehículo a través de la geolocalización en tiempo real,[7].

### Envío de datos hacia el dispositivo

Una de las características a destacar en esta investigación es el uso de forma adicional de la tecnología GSM, la misma que en un determinado tiempo, permite enviar un mensaje de texto que pueden contener @GPS, @ON, @OFF hacia el sim 808, el cual al verificar que sea un texto correcto mediante arduino, da una respuesta y reenvía un mensaje dependiente de la solicitud que recibió.

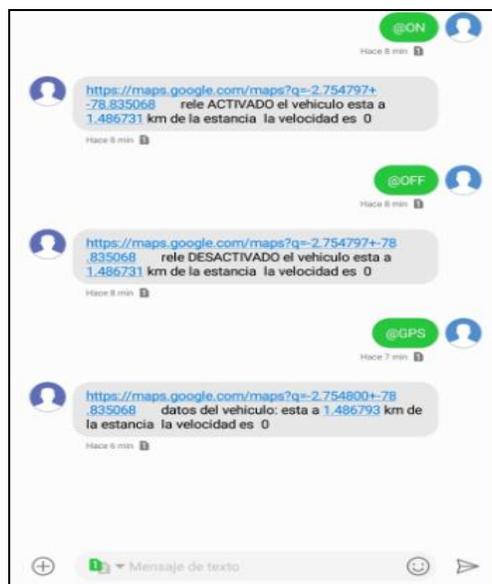
En el caso de que el mensaje de texto que ingreso fue @GPS, el módulo responde con una cadena de caracteres compuesta por velocidad, distancia al punto de referencia y la latitud con la

longitud concatenada a la página web de Google Maps, donde se visualiza la ubicación del vehículo en tiempo real.

En el caso donde los mensajes de texto, que reciba el sim 808 sean @ON o @OFF el módulo activará o desactivará un pin de arduino y devolverá un mensaje de respuesta con la ubicación en Google Maps, la velocidad y la distancia existente entre el punto de referencia y el vehículo en ese instante.

Teniendo esta funcionalidad en cuenta, se procede a activar o desactivar un elemento en el vehículo mediante un relé conectado al pin de la placa arduino, este relé puede ser usado para manipular el encendido del vehículo y poder inmovilizarlo o permitir su uso a conveniencia del personal administrativo de la universidad. La Figura 26 indica la comunicación mediante mensajes de texto.

**Figura 26.** Comunicación mediante de mensajes de texto.



Fuente.[11].

### Configuración de alarmas

Usualmente se usan para identificar cuando el vehículo pasa los límites permitidos, están configuradas en el “IoT ThingSpeak Monitor Widget”, una vez aplicada la programación se procede a activar la alerta, indicando el radio máximo permitido de circulación, dado en kilómetros [5]. Si este excede el límite se activa la alarma, la **¡Error! No se encuentra el origen de la referencia.**5 indica un ejemplo con la distancia de la Estancia de Posgrados de la Universidad Católica de Cuenca hasta el punto en el que se encuentra el vehículo.

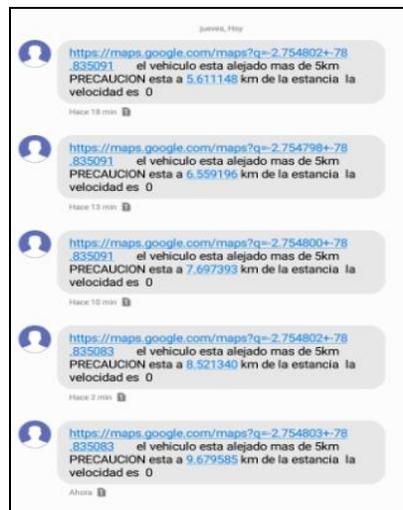
**Figura 25.** Visualización del widget en el teléfono móvil.



Fuente. [9].

Adicionalmente esta implementado otro sistema de alarma mediante el uso de la res GSM a través de mensajes de texto. Estos mensajes son enviados automáticamente desde el dispositivo cuando el vehículo excede un radio de 5 kilómetros, específicamente son enviados por cada kilómetro que se incremente al salir del radio, como se observa en la figura 26.

**Figura 26.** Visualización de alarmas por SMS en el teléfono móvil.



Fuente. [9].

## Conclusiones

Este sistema de geolocalización se fundamenta principalmente en la ubicación del vehículo en tiempo real, concatenado con la tecnología GSM, permitiendo como resultado el fortalecimiento de su seguridad y posicionamiento, dotando de acceso al personal de la Universidad Católica de Cuenca con datos e información en tiempo real.

Se investigaron diferentes alternativas para el desarrollo de la aplicación, siendo Arduino la plataforma que posee mejores prestaciones, en virtud de que cuenta con características que ayudan al manejo de las coordenadas geográficas (latitud y longitud), además, se integra fácilmente con la tecnología de comunicación GSM, actuando bidireccionalmente en la dotación de datos de ubicación, velocidad y distancia del vehículo eléctrico con respecto a sus puntos de control.

Una característica adicional que se introdujo en el sistema, es la de la activación de una alarma, la cual está diseñada para mantener el coche dentro del rango asignado, si este excede el radio para el cual fue programado se activarán los protocolos de seguridad, que alertarán a los encargados del monitoreo y control, que el vehículo no se encuentra dentro de los límites permitidos para tomar decisiones al respecto.

Una ventaja de usar el módulo sim 808 es que este trabaja con la red GPRS para la transmisión de datos, a bajo costo. Para el uso de la página web lo primordial es utilizar plataformas de Google Maps, mediante las herramientas API.

## Referencias

1. Gonzalo Garcia, “Los vehículos híbridos y eléctricos alcanzan el 5,2% de cuota de mercado - Sector - Híbridos y Eléctricos | Coches eléctricos, híbridos enchufables,” 02/27/2019, 2019. [Online]. Available: <https://www.hibridosyelectricos.com/articulo/sector/vehiculos-electricos-hibridos-cuota-mercado-global/20190227134756026009.html>. [Accessed: 02-May-2020].
2. El Telegrafo decano de la prensa nacional, “El Telégrafo - Noticias del Ecuador y del mundo - En 2 años y medio, la Policía registró el robo de 13.271 carros,” 2017. [Online]. Available: <https://www.eltelegrafo.com.ec/noticias/judicial/12/en-2-anos-y-medio-la-policia-registro-el-robo-de-13-271-carros> . [Accessed: 02-May-2020].

3. El tiempo Diario de Cuenca., “EL TIEMPO - Se reduce la tasa de crímenes en el Azuay,” Cuenca, p. 20, 04-Jan-2016.
4. [xatakamovil, “Las 16 mejores aplicaciones de geolocalización en Android,” 2014. [Online]. Available: <https://www.xatakamovil.com/espacio-sony/las-15-mejores-aplicaciones-de-geolocalizacion-en-android>. [Accessed: 02-May-2020].
5. ThingSpeak, “IoT: ejemplos de aplicación, comunicaciones y plataformas,” Aplicacion de comunicaciones y desarrollo, 2018. [Online]. Available: <http://iotrookies.blogspot.com/>. [Accessed: 02-May-2020].
6. [Ruben Alcaraz, “Cómo obtener una clave API para Google Maps,” 2017. [Online]. Available: <http://www.rubenalcaraz.es/pinakes/informatica/como-obtener-una-clave-api-para-google-maps/>. [Accessed: 02-May-2020].
7. [Arduino, “Thingspeak | Aprendiendo Arduino,” Aprendiendo Arduino, 2018. [Online]. Available: <https://aprendiendoarduino.wordpress.com/2018/11/23/thingspeak/>. [Accessed: 02-May-2020].
8. Motorola Global, “Configuración y uso de widgets,” configuracion y uso de widgest, 2017. [Online]. Available: [https://motorola-global-es-latam.custhelp.com/app/answers/indevice\\_detail/a\\_id/99447/p/30,6720,9146](https://motorola-global-es-latam.custhelp.com/app/answers/indevice_detail/a_id/99447/p/30,6720,9146). [Accessed: 02-May-2020].
9. [Evernote, “Como configurar el Widget de Evernote en Android – Ayuda & Aprendizaje de Evernote,” Evernote Config. del equipo, 2018.
10. Notepad++, “Notepad++,” 2019. [Online]. Available: <https://notepad-plus-plus.org/>. [Accessed: 02-May-2020].
11. R. Gsm, “4 Red GSM 4.1,” 2004.

## References

1. Gonzalo Garcia, “Hybrid and electric vehicles reach 5.2% of market share - Sector - Hybrids and Electric | Electric cars, plug-in hybrids, ”02/27/2019, 2019. [Online]. Available: <https://www.hibridosyelectricos.com/articulo/sector/vehiculos-electricos-hibridos-cuota-mercado-global/20190227134756026009.html>. [Accessed: 02-May-2020].

2. The Telegraph dean of the national press, "The Telegraph - News from Ecuador and the world - In 2 and a half years, the Police recorded the theft of 13,271 cars," 2017. [Online]. Available: <https://www.eltelegrafo.com.ec/noticias/judicial/12/en-2-anos-y-medio-la-policia-registro-el-robo-de-13-271-carros>. [Accessed: 02-May-2020].
3. The weather Diario de Cuenca., "EL TIEMPO - The crime rate in Azuay is reduced," Cuenca, p. 20, 04-Jan-2016.
4. [xatakamovil, "The 16 best geolocation applications on Android," 2014. [Online]. Available: <https://www.xatakamovil.com/espacio-sony/las-15-mejores-aplicaciones-de-geolocalizacion-en-android>. [Accessed: 02-May-2020].
5. ThingSpeak, "IoT: Application Examples, Communications and Platforms," Communications and Development Application, 2018. [Online]. Available: <http://iotrookies.blogspot.com/>. [Accessed: 02-May-2020].
6. [Ruben Alcaraz, "How to get an API key for Google Maps," 2017. [Online]. Available: <http://www.rubenalcaraz.es/pinakes/informatica/como-obtener-una-clave-api-para-google-maps/>. [Accessed: 02-May-2020].
7. [Arduino, "Thingspeak | Learning Arduino," "Learning Arduino, 2018. [Online]. Available: <https://aprendiendoarduino.wordpress.com/2018/11/23/thingspeak/>. [Accessed: 02-May-2020].
8. Motorola Global, "Widget Configuration and Use," Widget Configuration and Use, 2017. [Online]. Available: [https://motorola-global-es-latam.custhelp.com/app/answers/indevice\\_detail/a\\_id/99447/p/30.6720,9146](https://motorola-global-es-latam.custhelp.com/app/answers/indevice_detail/a_id/99447/p/30.6720,9146) [Accessed: 02-May-2020].
9. Evernote, "How to configure the Evernote Widget on Android - Help amp; Evernote Learning," "Evernote Config. of the team, 2018.
10. Notepad ++, "Notepad ++," 2019. [Online]. Available: <https://notepad-plus-plus.org/>. [Accessed: 02-May-2020].
11. R. Gsm, "4 GSM Network 4.1," 2004.

## Referências

1. Gonzalo Garcia, "Veículos híbridos e elétricos alcançam 5,2% de participação de mercado - Setor - Híbridos e Elétricos | Carros elétricos, híbridos plug-in, "27/02/2019, 2019. [Online].

- Disponível: <https://www.hibridosyelectricos.com/articulo/sector/vehiculos-electricos-hibridos-cuota-mercado-global/20190227134756026009.html>. [Acesso em: 02 de maio de 2020].
2. O decano do Telegraph da imprensa nacional, “The Telegraph - Notícias do Equador e do mundo - em dois anos e meio, a Polícia registrou o roubo de 13.271 carros”, 2017. [Online]. Disponível: <https://www.eltelegrafo.com.ec/noticias/judicial/12/en-2-anos-y-medio-la-policia-registro-el-robo-de-13-271-carros> [Acesso em: 02 de maio de 2020].
  3. O clima Diario de Cuenca., “EL TIEMPO - A taxa de criminalidade em Azuay é reduzida”, Cuenca, p. 20 de janeiro de 2016.
  4. [xatakamovil, “Os 16 melhores aplicativos de geolocalização no Android”, 2014. [Online]. Disponível: <https://www.xatakamovil.com/espacio-sony/las-15-mejores-aplicaciones-de-geolocalizacion-en-android>. [Acesso em: 02 de maio de 2020].
  5. ThingSpeak, “IoT: Exemplos de Aplicativos, Comunicações e Plataformas”, Aplicativo de Comunicação e Desenvolvimento, 2018. [Online]. Disponível: <http://iotrookies.blogspot.com/> [Acesso em: 02 de maio de 2020].
  6. [Ruben Alcaraz, “Como obter uma chave de API para o Google Maps”, 2017. [Online]. Disponível: <http://www.rubenalcaraz.es/pinakes/informatica/como-obtener-una-clave-api-para-google-maps/> [Acesso em: 02 de maio de 2020].
  7. [Arduino, “Thingspeak | Learning Arduino, ”Learning Arduino, 2018. [Online]. Disponível: <https://aprendiendoarduino.wordpress.com/2018/11/23/thingspeak/> [Acesso em: 02 de maio de 2020].
  8. Motorola Global, “Configuração e uso de widgets”, Configuração e uso de widgets, 2017. [Online]. Disponível: [https://motorola-global-es-latam.custhelp.com/app/answers/indevice\\_detail/a\\_id/99447/p/30,6720,9146](https://motorola-global-es-latam.custhelp.com/app/answers/indevice_detail/a_id/99447/p/30,6720,9146). [Acesso em: 02 de maio de 2020].
  9. [Evernote, “Como configurar o Evernote Widget no Android - Help amp; Evernote Learning ”, Evernote Config. da equipe, 2018.
  10. Notepad ++, “Notepad ++”, 2019. [Online]. Disponível: <https://notepad-plus-plus.org/>. [Acesso em: 02 de maio de 2020].
  11. R. Gsm, “4 GSM Network 4.1”, 2004.

©2020 por el autor. Este artículo es de acceso abierto y distribuido según los términos y condiciones de la licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional (CC BY-NC-SA 4.0) (<https://creativecommons.org/licenses/by-nc-sa/4.0/>).