

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

ESCUELA DE SISTEMAS



DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE
INGENIERA EN SISTEMAS Y COMPUTACIÓN

“Desarrollo de un prototipo de aplicativo móvil para la localización de mascotas en tiempo real mediante el uso de tecnología GPS y geofencing”

AUTORA: CLAUDINE ALEJANDRA CADENA LLIGÜÍN

DIRECTOR: MSC, DAMIÁN ANÍBAL NICOLALDE RODRÍGUEZ

QUITO, 2022

AGRADECIMIENTOS

A Dios, a mis padres en especial a mi madre, a mi tío, a mi familia, a mis amigos, a mis maestros y a todas mis personas especiales.

DEDICATORIA

A mi abuelo Luis Ll., a mi tío Luis.

RESUMEN

Prototipo de aplicativo móvil para rastreo de mascotas en tiempo real con tecnología GPS e implementación de cercas virtuales o geovallas. Para su desarrollo se utilizó la metodología ágil (XP) o Programación Extrema que permitió agilizar la codificación y documentación del proyecto dividiéndolo en iteraciones integrales que abarcan desde el análisis, diseño, hasta las pruebas de aceptación.

Los IDEs empleados fueron: Android Studio y Visual Studio con .NET Framework, en cuanto al hardware en un principio se hizo uso de un rastreador GPS TK-102 de Ablegrid que posteriormente fue remplazado por celulares debido a fallas técnicas, sin embargo, a pesar del cambio de dispositivos el análisis, diseño y desarrollo no se vieron afectados, manteniendo el proyecto tal como se lo venía ejecutando.

Como resultado de esta disertación se ha conseguido elaborar un prototipo de aplicación móvil que incluye la implementación de un corral virtual, basado en la arquitectura REST, pudiendo comprobarse que las técnicas, herramientas y metodologías aplicadas fueron las adecuadas para la culminación del proyecto.

Se puede concluir que la elaboración de prototipos de apps móviles permite visualizar en primera fase como sería su funcionamiento a futuro como modelo de negocio, procurando establecer cuáles serían sus limitaciones, cuellos de botella y fortalezas.

Como recomendación se plantea realizar un estudio técnico del hardware a utilizar para evitar complicaciones en la codificación de cualquier sistema.

INTRODUCCIÓN

Este proyecto se enfoca en el desarrollo de un prototipo de aplicativo móvil para la localización en tiempo real de cualquier tipo de mascotas con ayuda de la tecnología GPS y geofencing, buscando ser de fácil acceso para el público aprovechando recursos tecnológicos del dominio de todos.

El primer capítulo explica toda la sustentación teórica en la que se fundamenta este trabajo.

El segundo capítulo analiza los antecedentes en el país en cuanto a iniciativas para el rastreo de mascotas ya implementadas en el Ecuador, dando paso al levantamiento de requerimientos.

En el tercer capítulo se evidencia todo el proceso de desarrollo del prototipo con sus respectivas fases aplicando la metodología elegida: Extreme Programming.

Por último, en el cuarto capítulo, se reflejarán los resultados de esta disertación apoyados de toda la documentación necesaria para su correcta configuración y posterior uso.

ÍNDICE

AGRADECIMIENTOS.....	1
DEDICATORIA.....	2
RESUMEN.....	3
INTRODUCCIÓN.....	4
ÍNDICE.....	5
ÍNDICE DE FIGURAS.....	8
ÍNDICE DE DIAGRAMAS.....	9
ÍNDICE DE TABLAS.....	10
1. CAPÍTULO 1 - FUNDAMENTOS TEÓRICOS.....	11
1.1. Hardware.....	11
1.1.1. Smartphone.....	12
1.1.2. Ablegrid 2G rastreador GPS.....	12
1.2. Redes de comunicación.....	14
1.2.1. Redes de telefonía móvil.....	15
1.2.1.1. Sistema Global de Comunicaciones Móviles (GSM).....	15
1.2.1.2. Servicio General de Paquetes de Radio (GPRS).....	16
1.2.2. Sistema de Posicionamiento Global (GPS).....	16
1.2.3. Enlaces satelitales.....	19
1.2.4. Protocolos de comunicación.....	19
1.2.4.1. Protocolo de la Asociación Nacional de Electrónica Marina (NMEA).....	20
1.2.4.2. Protocolo de Transferencia de Hipertexto (HTTP).....	20
1.2.4.3. Protocolo de Control de Transmisión / Protocolo de Internet (TCP/IP).....	20
1.3. Software.....	21
1.3.1. Aplicativo móvil.....	21
1.3.2. Geofencing.....	22
1.3.3. Lenguajes de programación.....	23
1.3.3.1. Java.....	24
1.3.3.2. C#.....	26
1.3.4. Bases de datos.....	27
1.3.5. Gestores de Bases de Datos.....	28
1.3.5.1. SQL Server.....	31
1.3.6. IDE.....	33
1.3.6.1. Android Studio.....	34
1.3.6.2. Visual Studio.....	35

1.3.7.	Framework	35
1.3.7.1.	.NET Framework	36
1.3.8.	APK.....	37
1.4.	Servicios Web	38
1.4.1.	Modelo arquitectónico REST	40
1.5.	Sistemas de información geográfica.....	41
1.5.1.	Servidores GIS	42
1.5.2.	API de Google Maps	42
1.6.	Metodologías ágiles	44
1.6.1.	Programación Extrema o XP	44
1.6.1.1.	Historias de usuario.....	47
1.6.1.2.	Fase de análisis	47
1.6.1.3.	Fases de codificación y pruebas.....	48
2.	CAPÍTULO 2 - REQUERIMIENTOS	50
2.1.	Situación actual	50
2.2.	Levantamiento de Requerimientos.....	51
2.2.1.	Detalles del sistema	51
2.2.2.	Arquitectura de la aplicación (con dispositivos móviles).....	53
2.2.3.	Historias de usuario	54
2.3.	Diagramas de casos de uso.....	56
2.3.1.	Caso de uso General	56
2.3.2.	Caso de uso F0 – Administrar Sesión.....	56
2.3.3.	Caso de uso F1 – Administrar Mascotas	57
2.3.4.	Caso de uso F2 – Administrar Ubicación	59
2.3.5.	Caso de uso F3 – Administrar Geovalla.....	60
3.	CAPÍTULO 3 - DESARROLLO	61
3.1.	Planificación de las iteraciones	61
3.2.	Primera iteración.....	62
3.2.1.	Análisis.....	62
3.2.2.	Diseño.....	62
3.2.2.1.	Interfaz gráfica	62
3.2.3.	Codificación y pruebas	64
3.3.	Segunda iteración	65
3.3.1.	Análisis.....	65
3.3.2.	Diseño.....	65
3.3.2.1.	Interfaz gráfica	65
3.3.3.	Codificación y pruebas.....	67

3.4.	Tercera iteración	68
3.4.1.	Análisis.....	68
3.4.2.	Diseño.....	68
3.4.2.1.	Interfaz gráfica	68
3.4.3.	Codificación y pruebas.....	70
3.5.	Cuarta iteración	74
3.5.1.	Análisis.....	74
3.5.2.	Diseño.....	74
3.5.2.1.	Interfaz gráfica	74
3.5.3.	Codificación y pruebas.....	76
4.	CAPÍTULO 4 – EJECUCIÓN.....	77
4.1.	Configuración del dispositivo de rastreo GPS.....	77
4.2.	Puesta en marcha – Funcionamiento en móviles.....	81
I.	CONCLUSIONES.....	83
a.	Conclusiones del desarrollo de la disertación.....	83
b.	Conclusiones técnicas	84
II.	RECOMENDACIONES.....	85
III.	BIBLIOGRAFÍA	86

ÍNDICE DE FIGURAS

<i>Figura 1-1</i> GPS Tracker TK-102 (vista exterior).....	13
<i>Figura 1-2</i> GPS Tracker TK-102 (vista interna).....	14
<i>Figura 1-3</i> Distribución de Satélites GPS por Bloques	18
<i>Figura 1-4</i> Detectar la entrada de un móvil en un área prohibida.....	23
<i>Figura 1-5</i> Índice TIOBE para enero 2022.....	26
<i>Figura 1-6</i> Sistema Gestor de Base de Datos	29
<i>Figura 1-7</i> Plataformas de Bases de Datos más populares	30
<i>Figura 1-8</i> Actualizaciones disponibles de SQL Server.....	32
<i>Figura 1-9</i> .NET 5.....	37
<i>Figura 1-10</i> Patrón de interacción de la arquitectura de Web Services.....	39
<i>Figura 1-11</i> Cliente de API REST.....	41
<i>Figura 1-12</i> Componentes de un SIG.....	42
<i>Figura 1-13</i> Iteraciones de XP.....	46
<i>Figura 2-1</i> Arquitectura prototipo aplicativo móvil	53
<i>Figura 3-1</i> Prototipo Interfaz Gráfica - Ingreso.....	63
<i>Figura 3-2</i> Prototipo Vista en Móvil - Ingreso	63
<i>Figura 3-3</i> Prototipo Interfaz Gráfica - Lista Mascotas.....	66
<i>Figura 3-4</i> Prototipo Vista en Móvil - Lista Mascotas	66
<i>Figura 3-5</i> Prototipo Interfaz Gráfica – Geolocalización	69
<i>Figura 3-6</i> Prototipo Vista en Móvil - Geolocalización.....	69
<i>Figura 3-7</i> Tarjeta GSM SIM Ablegrid.....	70
<i>Figura 3-8</i> Limitaciones configuración geovalla en Modo de Texto SMS.....	71
<i>Figura 3-9</i> Prototipo Interfaz Gráfica - Geofencing.....	75
<i>Figura 3-10</i> Prototipo Vista en Móvil - Geofencing	75
<i>Figura 4-1</i> Configuración - Inicio	78
<i>Figura 4-2</i> Configuración - APN	78
<i>Figura 4-3</i> Configuración - SMS	79
<i>Figura 4-4</i> Configuración - GPS	79
<i>Figura 4-5</i> Configuración - Ubicación GPS mapa.....	80
<i>Figura 4-6</i> Configuración - Valores coordenadas.....	80

ÍNDICE DE DIAGRAMAS

<i>Diagrama 2-1 Caso de Uso General</i>	56
<i>Diagrama 2-2 Caso de Uso - F0. Adm. Sesión</i>	56
<i>Diagrama 2-3 Caso de Uso - F1. Adm. Mascotas</i>	57
<i>Diagrama 2-4 Caso de Uso - F2. Adm. Ubicación</i>	59
<i>Diagrama 2-5 Caso de Uso - F3. Adm. Geovalla</i>	60

ÍNDICE DE TABLAS

<i>Tabla 2-1 Historia de Usuario # 1</i>	54
<i>Tabla 2-2 Historia de Usuario # 2</i>	54
<i>Tabla 2-3 Historia de Usuario # 3</i>	55
<i>Tabla 2-4 Historia de Usuario # 4</i>	55
<i>Tabla 2-5 Flujos básicos y alternativos F0 Adm. Sesión</i>	57
<i>Tabla 2-6 Flujos básicos y alternativos F1 Adm. Mascotas</i>	58
<i>Tabla 2-7 Flujos básicos y alternativos F2 Adm. Ubicación</i>	60
<i>Tabla 2-8 Flujos básicos y alternativos F3 Adm. Geovalla</i>	61
<i>Tabla 3-1 Plan de Iteraciones</i>	61
<i>Tabla 3-2 Criterio de Aceptación # 1</i>	62
<i>Tabla 3-3 Tarjeta CRC # 1</i>	64
<i>Tabla 3-4 Prueba de Aceptación # 1</i>	64
<i>Tabla 3-5 Criterio de Aceptación # 2</i>	65
<i>Tabla 3-6 Tarjeta CRC # 2</i>	67
<i>Tabla 3-7 Prueba de Aceptación # 2</i>	67
<i>Tabla 3-8 Criterio de Aceptación # 3</i>	68
<i>Tabla 3-9 Cuadro comparativo funcionalidades TK-102</i>	72
<i>Tabla 3-10 Tarjeta CRC # 3</i>	73
<i>Tabla 3-11 Prueba de Aceptación # 3</i>	73
<i>Tabla 3-12 Criterio de Aceptación # 4</i>	74
<i>Tabla 3-13 Tarjeta CRC # 4</i>	76
<i>Tabla 3-14 Prueba de Aceptación # 4</i>	76

1. CAPÍTULO 1 - FUNDAMENTOS TEÓRICOS

En este capítulo se muestra toda la sustentación teórica en la que se basa esta disertación, abarcando temas de hardware, software, redes y protocolos de comunicación, bases de datos, metodologías de desarrollo y arquitectura de aplicaciones.

1.1. Hardware

Se define como el conjunto de todos los dispositivos tangibles que conforman una computadora, siendo la unidad central de procesamiento (CPU) la parte más importante. El CPU hace las veces del cerebro de la máquina, encargándose de manejar y procesar adecuadamente todo lo que llega y sale de la computadora.

“Una computadora es una máquina que trabaja con grandes cantidades de datos o información, procesándolos de manera muy rápida. Una computadora completamente habilitada requiere tanto del hardware como el software, siendo el hardware la parte física de la misma” (Small, 2018).

La clasificación del hardware varía de autor en autor, pero a simples rasgos se lo puede clasificar en tres grupos:

- Entrada: Teclado, mouse, micrófono, escáner.
- Salida: Impresoras, pantallas.
- Dispositivos de Almacenamiento: Disco duro, memoria RAM y ROM.

El uso de la palabra hardware cada vez es más extenso, abarcando mayor cantidad de dispositivos en el mercado. El término hardware no se usa solamente en referencia a las partes físicas de una computadora, sino también para todos aquellos dispositivos tales como celulares, tablets, relojes inteligentes, equipos GPS, entre otros.

1.1.1. Smartphone

Los smartphones se han convertido en una necesidad para los consumidores dadas sus diferentes aplicaciones y usos, desde el uso de redes sociales, e-mail, acceso al sistema de mapas con la ayuda del GPS, descarga de juegos online y offline, calendario, entre otros.

Un smartphone es un teléfono celular inteligente, o teléfono celular con una computadora en su interior, que incluye algo de hardware extra. En consecuencia, los programadores pueden escribir software de aplicaciones para ellos; este software de aplicaciones es llamado APP. Un smartphone tiene los típicos componentes de una computadora regular: CPU, memoria, dispositivo de almacenamiento, sistema operativo, y otros dispositivos como cámara, acelerómetro, o GPS. (Franceschi, 2017)

Actualmente, la mayoría de las personas tienen celulares inteligentes para uso particular o profesional, fomentando que las ventas tengan un incremento sustancial en los últimos años y dando lugar a un sin número de variedades y modelos. Los smartphones pueden tener diferentes sistemas operativos tales como Android de Google o iOS de Apple, distintas resoluciones de pantalla, diferente capacidad de almacenamiento, entre otras características disponibles en el mercado.

1.1.2. Ablegrid 2G rastreador GPS

Un GPS Tracker o también conocido como rastreador GPS, es un pequeño dispositivo electrónico encargado de recibir datos GPS y transmitirlo en coordenadas en tiempo real a un servicio de monitoreo.

Este rastreador GPS de última generación está diseñado para el seguimiento/monitoreo avanzado, con tecnología GPS, GSM y GPRS. El rastreador GPS está diseñado para usarse en cualquier vehículo, mascota,

persona o pertenencias sujetas a pérdida, para rastrear su ubicación, siendo las posibilidades de rastreo en vivo ilimitadas. (Ablegrid Corp, 2019)

En la Figura 1-1 se muestra el diseño externo del rastreador GPS TK-102.

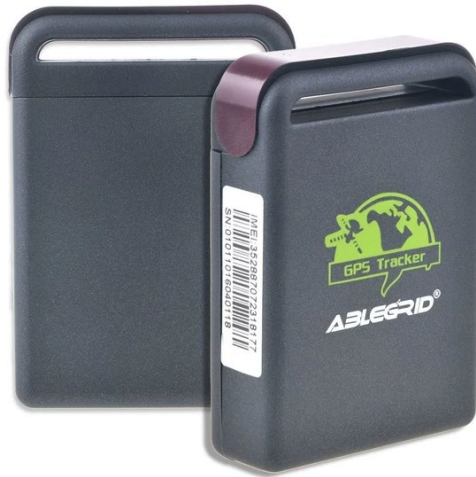


Figura 1-1 GPS Tracker TK-102 (vista exterior)

Fuente: Ablegrid Corp. (2019)

En la Figura 1-2 se muestra la vista interna del rastreador GPS, donde se encuentra la ranura en donde se colocará el chip de la operadora móvil.



Figura 1-2 GPS Tracker TK-102 (vista interna)

Fuente: Ablegrid Corp. (2019)

- Para el desarrollo de este prototipo se utilizará un rastreador GPS modelo TK-102, cuyas especificaciones se detallan a continuación (Ablegrid Corp, 2019):
 - Red: GSM/GPRS
 - Chip GPS: SIRF3
 - Sensibilidad GPS: -159dBm
 - Duración de batería: 96 horas
 - Color: Negro
 - Temperatura de Almacenamiento: -40°F a +85°F

1.2. Redes de comunicación

“El concepto de redes en el mundo de la informática consiste en la capacidad de enlazar dos o más dispositivos digitales para que estos puedan hacer uso compartido de sus datos, y están conformadas tanto de software como de hardware” (Pérez Arbesú, 2021).

“Las redes de datos son redes de telecomunicaciones donde los dispositivos actúan como nodos y se encuentran conectados en red compartiendo datos entre sí por medio de conexiones de datos, las conexiones representan los enlaces ya sean físicos o inalámbricos” (Pérez Arbesú, 2021).

“Existen conexiones globales y locales en donde las redes juegan el papel de plataforma para los servicios, permitiendo conectarse entre sí al apoyarse en el uso de datos, video, imágenes y más” (Instituto de Ciencias Básicas e Ingeniería de la UAEH, 2018).

1.2.1. Redes de telefonía móvil

“Las comunicaciones móviles siguen el principio general de la telefonía: Conectar dos usuarios remotos a través del equipo de red de un operador responsable de la gestión del servicio” (Radio Waves Orange, 2022).

Esta red de radio distribuida por tierra a través de celdas, en la que cada una de ellas incluye un transceptor de ubicación fija llamado estación base. La unión de dichas celdas permite la cobertura de radio en áreas geográficas de gran tamaño. (Icy Science, 2022)

Actualmente, se dispone de la quinta generación de redes móviles o más conocida como 5G, que comenzó a implementarse en 2019.

1.2.1.1. Sistema Global de Comunicaciones Móviles (GSM)

Conocido como Sistema Global de Comunicaciones Móviles, incorpora tecnología digital y por lo mismo cualquier cliente de GSM puede conectarse a través de su teléfono con su ordenador y puede hacer, enviar y recibir mensajes por e-mail, faxes, navegar por Internet, acceso seguro a la red informática de una compañía (LAN/Intranet), así como

utilizar otras funciones digitales de transmisión de datos, incluyendo el Servicio de Mensajes Cortos (SMS). (Samsung, 2021)

- **Arquitectura**

La conexión se realiza a través de ondas de radio, en donde los teléfonos actúan como estaciones móviles, los cuales poseen un identificador único o IMEI y necesitan de una tarjeta SIM para su correcto funcionamiento; la SIM se encarga de informarle a la estación base cual es el usuario que se está comunicando a través de la red. (VIU, 2018)

- **Tarjeta SIM**

“Se trata de un chip que contiene información sobre las estaciones base y también del usuario, incluyendo la operadora de red y tipo de contrato” (VIU, 2018).

1.2.1.2. Servicio General de Paquetes de Radio (GPRS)

El Servicio General de Paquetes de Radio “es el método de transferencia de datos en las redes de telefonía móvil 2G, fue lanzado en 2001 por el Sistema Global de Comunicaciones Móviles para proporcionar acceso a Internet a los usuarios de dispositivos móviles” (Verizon, 2022).

“GPRS colaboró con la introducción del protocolo de comunicaciones móviles WAP” (VIU, 2017).

1.2.2. Sistema de Posicionamiento Global (GPS)

El GPS o Sistema de Posicionamiento Global es un sistema de navegación basado en satélites, que permite determinar la posición de cualquier objeto, persona o animal a nivel mundial. Fue lanzado por el Departamento de Defensa de los

Estados Unidos en el año de 1978 y actualmente cuenta con 30 satélites en órbita totalmente operativos.

Los satélites GPS se encuentran a 20.000 km de altitud, formando una constelación, y se necesitan al menos 24 de ellos para que el sistema se encuentre funcionando adecuadamente. Están organizados en 6 órbitas que rodean la Tierra y cada una se encuentra dividida en 4 espacios iguales, garantizando que cada usuario final pueda visibilizar al menos 4 satélites al mismo tiempo desde cualquier punto del planeta.

Se los puede clasificar por bloques de acuerdo con su generación (orden de lanzamiento, características y otras especificaciones provistas por el Centro de Navegación de los Estados Unidos), como se detalla en la Figura 1-3:

SATÉLITES HEREDADOS		SATÉLITES MODERNIZADOS		
BLOQUE IIA	BLOQUE IIR	BLOQUE IIR-M	BLOQUE IIF	GPS III/IIIF
0 operativo	7 operativos	7 operativos	12 operativos	4 operativos
<ul style="list-style-type: none"> ▪ Código de adquisición aproximada (C/A) en la frecuencia L1 para usuarios civiles ▪ Código P(Y) preciso en frecuencias L1 y L2 para usuarios militares ▪ Vida útil del diseño de 7,5 años ▪ Lanzado en 1990-1997 ▪ Último dado de baja en 2019 	<ul style="list-style-type: none"> ▪ Código C/A en L1 ▪ Código P(Y) en L1 y L2 ▪ Monitoreo de reloj a bordo ▪ Vida útil del diseño de 7,5 años ▪ Lanzado en 1997-2004 	<ul style="list-style-type: none"> ▪ Todas las señales heredadas ▪ 2ª señal civil en L2 (L2C) SABER MÁS ➔ ▪ Nuevas señales militares de código M para mejorar la resistencia a los atascos ▪ Niveles de potencia flexibles para señales militares ▪ Vida útil del diseño de 7,5 años ▪ Lanzado en 2005-2009 	<ul style="list-style-type: none"> ▪ Todas las señales de bloque IIR-M ▪ 3ª señal civil en frecuencia L5 (L5) SABER MÁS ➔ ▪ Relojes atómicos avanzados ▪ Precisión, intensidad de la señal y calidad mejoradas ▪ vida útil del diseño de 12 años ▪ Lanzado en 2010-2016 	<ul style="list-style-type: none"> ▪ Todas las señales de bloque IIF ▪ 4ª señal civil en L1 (L1C) SABER MÁS ➔ ▪ Fiabilidad, precisión e integridad de la señal mejoradas ▪ Sin disponibilidad selectiva MÁS INFORMACIÓN ➔ ▪ vida útil del diseño de 15 años ▪ IIIF: reflectores láser; carga útil de búsqueda y rescate ▪ Primer lanzamiento en 2018

Figura 1-3 Distribución de Satélites GPS por Bloques

Fuente: National Coordination Office for Space-Based Positioning, Navigation, and Timing. (2022)

“Los satélites GPS orbitan la Tierra dos veces por día, transmitiendo una señal única y parámetros orbitales, los receptores GPS utilizan esta información y trilateración para calcular la ubicación exacta de un usuario” (Garmin Ltd., 2017).

La propagación de la señal y el propio sistema GPS están afectados por errores sistemáticos o desviaciones y por errores aleatorios, podemos clasificar esos errores según su procedencia en tres grupos, errores asociados al satélite (errores en el reloj del satélite), con el medio de propagación (errores

atmosféricos) y con el receptor (retardo instrumental). (Berné Valero, J., Anquela Julián, A. y Garrido Villén, N., 2013)

Los dispositivos GPS receptores de las señales enviadas por el satélite son muy precisos, sin embargo, cuando se necesita realizar algunas correcciones a los datos recibidos, se hace uso del DGPS o GPS Diferencial o del Sistema de Aumentación Basado en Satélites (WAAS). Ambos sistemas buscan complementar el sistema de satélites GPS proveyendo mayor precisión en los cálculos de posicionamiento.

Actualmente, la tecnología GPS se encuentra disponible en todos los smartphones, incrementado sus aplicaciones que varían desde rastreo de mascotas, aplicaciones de citas, Uber, Waze, entre otras. Una de las herramientas más conocidas es Google Maps que en un inicio era privada, pero “algunos desarrolladores descubrieron la manera de hackearla para incorporar los mapas en sus propios sitios web, y esto llevo a Google a lanzar la API públicamente en 2005” (Nolasco Valenzuela, 2016).

1.2.3. Enlaces satelitales

“Es el canal por el cuál serán enviadas y recibidas las señales transmitidas de la estación terrestre al satélite y de este a la estación terrestre”. (Arias, 2017)

Los enlaces entre estaciones terrenas y los satélites o entre satélites están constituidos por radiación electromagnética dirigida en forma de haces, similares en algunas de sus características, a los enlaces entre estaciones ubicadas sobre la superficie terrestre. Existen tres tipos de enlaces: enlace de subida de las estaciones terrenas a los satélites, enlace de bajada de los satélites a las bases terrenas y enlace intersatelital. (UNAM, s.f.)

1.2.4. Protocolos de comunicación

“La interconexión de sistemas o redes de computadoras son la base de las comunicaciones hoy en día y están diseñadas bajo múltiples protocolos de

comunicación” (KIO, 2021). Existen distintas clasificaciones y tipos de protocolos, dentro de los más conocidos se encuentran los siguientes:

1.2.4.1. Protocolo de la Asociación Nacional de Electrónica Marina (NMEA)

Un GPS entrega sus datos a través de un protocolo llamado NMEA (Asociación Nacional de Electrónica Marina). Pudiendo comunicarse la mayoría de estos receptores mediante este protocolo. El GPS devuelve una línea de datos, denominada “trama”, con sus respectivos separadores reglamentarios y en algunos casos letras o símbolos de identificación. (De Castillo, 2020)

1.2.4.2. Protocolo de Transferencia de Hipertexto (HTTP)

Hypertext Transfer Protocol, protocolo que nos permite realizar una petición de datos y recursos, como pueden ser documentos de Lenguaje de Marcas de Hipertexto (HTML). Es la base de cualquier intercambio de datos en la Web, y un protocolo de estructura cliente-servidor. Las peticiones son enviadas por una entidad: el agente del usuario, cada petición individual se envía a un servidor, el cual la gestiona y responde. Entre cada petición y respuesta, hay varios intermediarios, normalmente denominados proxies. (Mozilla, 2022)

1.2.4.3. Protocolo de Control de Transmisión / Protocolo de Internet (TCP/IP)

Se trata de dos protocolos diferentes TCP (Protocolo de Control de Transmisión) e IP (Protocolo de Internet) que trabajan en conjunto, encargándose de diferentes funciones:

IP es la parte que obtiene la dirección a la que se envían los datos. TCP se encarga de la entrega de los datos una vez hallada dicha dirección IP.

Lo que hace TCP/IP es descomponer cada mensaje en paquetes que se vuelven a ensamblar en el otro extremo y divide las distintas tareas de comunicación en capas, siendo el propósito de las capas crear un sistema estandarizado, sin que los distintos fabricantes de hardware y software tengan que gestionar la comunicación por su cuenta. (Avast, 2021)

1.3. Software

Generalmente el término software se puede definir como el conjunto de aplicaciones y programas que utiliza un sistema informático.

Se pueden distinguir dos tipos de softwares principalmente:

- **Software de Aplicación:** Está diseñado para el beneficio de los usuarios al realizar una o más tareas. Tenemos como ejemplo a Microsoft Word, Power Point, Access, entre otros.
- **Software del Sistema:** Es aquel que opera directamente en dispositivos de hardware de la computadora, proporcionando una plataforma para ejecutar una aplicación, tal es el caso de los sistemas operativos (Linux, Windows, etc.).

1.3.1. Aplicativo móvil

Según un estudio de la compañía Criteo en el año 2016:

Después de haber analizado el comportamiento y las tendencias de compra a través de dispositivos móviles a partir de 1700 millones de transacciones online en más de 3300 empresas, las apps se sitúan como el canal más eficiente en comercio móvil. (Pintado Blanco & Sánchez Herrera, 2017)

Se trata de aplicaciones informáticas diseñadas para ser ejecutadas en dispositivos móviles, permitiendo al usuario llevar a cabo distintos tipos de tareas con fines

educativos, profesionales, entretenimiento, etc.; en un inicio estas ya venían incluidas en el sistema operativo de los teléfonos celulares, sin embargo, estaban enfocadas en mejorar el rendimiento de este con funcionalidades tales como calendario, calculadora, alarma, entre otras.

El auge de las apps surge con el ingreso del iPhone al mercado, dando lugar a la aparición de las tiendas de aplicaciones como Apple Store o Android Market (ahora conocido como Google Play), en donde la mayoría de aplicativos se distribuyen gratuitamente.

Actualmente, existen varias clasificaciones para los aplicativos móviles, una de las más populares es por el tipo de desarrollo y diseño, donde se conocen las apps nativas (su codificación es específica para cada sistema operativo), apps híbridas (mezcla lo mejor de las apps nativas y aplicaciones web) y aplicaciones web.

1.3.2. Geofencing

La palabra geofencing es un acrónimo de dos palabras: “fence” que significa cercamiento o perímetro, y el prefijo “geo” que indica que este perímetro está construido a partir de datos geográficos, entonces se puede definir geofencing como la delimitación de un área geográfica usando un perímetro virtual. Las aplicaciones de geofencing operan e interactúan con diferentes servicios basados en la ubicación utilizando el Sistema Global de Navegación por Satélite (GNSS) y tecnologías de comunicación. (Nait Sidi Moh y otros, 2013)

Como se ejemplifica en la Figura 1-4, una geovalla es capaz de detectar la entrada, salida y el tiempo de permanencia de cualquier vehículo, persona o mascota que se encuentre en sus límites preestablecidos.



Figura 1-4 Detectar la entrada de un móvil en un área prohibida

Fuente: Nait Sidi Moh, A., Ait-Cheik-Bihi, W., Bakhouya, M., Gaber, J., & Wack, M. (2013)

Una geovalla implica establecer cercas virtuales alrededor de entidades geográficas del mundo real, puede ser creada dinámicamente o con límites predefinidos. En una geovalla dinámica, los límites se crean en función de un punto elegido y se pueden crear sobre la marcha, a diferencia de una geovalla estática que usa límites predeterminados (zonas de peligro, centro de la ciudad, etc.). (Abdishakur & Jayakrishnan, 2019)

La integración de técnicas de geofencing o geovallas para seguimiento en tiempo real en dispositivos móviles ha visto un incremento significativo en los últimos años; sus aplicaciones son variadas, pueden ser usadas en seguridad, negocios, entre otras. Sin embargo, su uso más popular es con fines de marketing.

1.3.3. Lenguajes de programación

“Son aquellos que se utilizan para escribir programas; constan de secuencias de instrucciones que se codifican y, a su vez, son traducidas para que las computadoras los entiendan” (Arroyo Díaz, 2019a)

Dado que los lenguajes de programación conocidos son muchos y difieren según su forma de trabajar, se pueden diferenciar claramente los siguientes:

- Lenguajes de bajo nivel: los programas desarrollados con este tipo de lenguajes solo pueden ser ejecutados en el sistema para el que fueron diseñados; por lo tanto, no pueden migrar a otros computadores, siendo necesario reescribirlos.
- Lenguajes de alto nivel: se trata de lenguajes más universales, cuya principal ventaja es que son de fácil entendimiento para los usuarios; pueden ejecutarse en diferentes plataformas sin realizar modificaciones en el código.

Otra clasificación conocida es por la forma como se ejecutan, distinguiéndose los lenguajes compilados y los interpretados; los primeros necesitan de un compilador que traduce todo al mismo tiempo antes de ejecutar cualquier instrucción, mientras que los interpretados traducen y ejecutan el código en directo.

De acuerdo con Arroyo Díaz (2019a), también es posible clasificarlos en otros dos grandes grupos de lenguajes:

- Imperativos: secuenciales como Ensamblador, estructurados como BASIC, procedurales como C o Pascal y orientados a objetos por ejemplo, C++, Java, Ruby.
- Declarativos: lógicos como Prolog, funcionales como ejemplo, Haskell, Erlang, Lisp.

Actualmente, los lenguajes con mayor popularidad para este año son C, Java, Python, C++ y C#.

1.3.3.1. Java

Java es un lenguaje de programación orientado a objetos que se basa en la sintaxis de C y C++. En el año 1991 Sun Microsystems desarrolló este

lenguaje como parte de un proyecto para decodificadores dirigido por James Gosling, sin embargo, no sería hasta 1995 que se comercializó por primera vez en su plataforma.

“Un objetivo clave de Java es poder escribir programas que se ejecuten en una variedad de sistemas computacionales y dispositivos controlados por computadora. A eso se le conoce algunas veces como «escribir una vez, ejecutar en cualquier parte»” (Deitel & Deitel, 2016).

Java es 100% orientado a objetos, pues intenta trasladar la naturaleza de los objetos de la vida cotidiana a la programación, es decir, que los objetos tienen un estado, un comportamiento y propiedades. Los objetos combinan datos, comportamiento e identidad. En consecuencia, pensar de esta manera tiene sus ventajas pues dividimos a los objetos en elementos cada vez más pequeños. (Arroyo Díaz, 2019b)

Actualmente, Java es parte de Oracle, después de adquirir Sun Microsystems en el 2010. La última versión estable (Java SE 17) fue lanzada el 14 de septiembre del año pasado, con su última actualización 17.0.2. De acuerdo con el índice TIOBE representado en la Figura 1-5, Java ocupa el tercer lugar hasta enero del 2022.



febrero de 2022	febrero de 2021	Cambio	Lenguaje de programación	Calificaciones	Cambio
1	3	▲	 Pitón	15,33%	+4,47%
2	1	▼	 C	14,08%	-2,26%
3	2	▼	 Java	12,13%	+0,84%
4	4		 C++	8,01%	+1,13%
5	5		 C#	5,37%	+0,93%
6	6		 básico visual	5,23%	+0,90%
7	7		 JavaScript	1,83%	-0,45%
8	8		 PHP	1,79%	+0,04%
9	10	▲	 lenguaje ensamblador	1,60%	-0,06%
10	9	▼	 sql	1,55%	-0,18%

Figura 1-5 Índice TIOBE para enero 2022

Fuente: TIOBE Software BV. (2022)

1.3.3.2. C#

“Es un lenguaje de programación orientado a objetos y orientado a componentes, proporciona construcciones de lenguaje para respaldar directamente estos conceptos, lo que lo convierte en un lenguaje natural en el que crear y usar componentes de software” (Wagner, B., Costa Santos, R., Sharkey, K., Coulter, D., Pine, D., Schonning, N., Addie, S., Kulikov, P., Wenzel, M., Victor, Y., Latham, L. y Onderka, P., 2021).

Es una evolución que Microsoft realizó, tomando lo mejor de los lenguajes C y C++, y ha continuado añadiéndole funcionalidades, tomando de otros lenguajes, como java, algo de su sintaxis evolucionada y con el tiempo adaptó las facilidades de la creación de código que tenía otro de sus lenguajes más populares, Visual Basic,

haciéndolo tan polivalente y fácil de aprender como éste, sin perder ni un ápice de la potencia original de C. (BE Software, 2020)

1.3.4. Bases de datos

De acuerdo con Deitel & Deitel (2016), una base de datos “es una colección de datos organizados para facilitar su acceso y manipulación. El modelo más popular es la base de datos relacional, en la que los datos se almacenan en simples tablas. Una tabla incluye registros y campos”.

“Llamamos base de datos (o bases de datos) a un conjunto de datos dispuestos con el objetivo de proporcionar información a los usuarios y permitir transacciones como inserción, eliminación y actualización de datos” (Benítez & Arias, 2017).

Una base de datos es también un modelo del mundo real y, como tal, debe poder servir para toda una gama de usos y aplicaciones. Esta definición es especialmente interesante puesto que introduce un nuevo concepto: el modelo de la base de datos; y este modelo representa una arquitectura organizativa que impone un esquema de acceso a los datos. (González, 2011)

Actualmente, el modelo de bases de datos más empleado es el relacional:

- Relacional: “propuesto por E.F. Codd hacia 1970. En este modelo, la propia base de datos es suficientemente potente para buscar por cualquier criterio y suficientemente inteligente como para aprovechar los índices cuando es pertinente; y además posee las siguientes prestaciones” (González, 2011):
 - Abstracción sobre el sistema de almacenamiento físico
 - Relaciones reflexivas y recursividad
 - Tablas y vistas
 - Cardinalidad
 - Procedimientos almacenados y triggers

Sin embargo, existen otros tipos de modelos:

- Jerárquico

En este modelo se tiene una entidad o tabla como pieza central para el manejo de datos. Se trata, pues, de una estructura arbórea; se empieza por el tronco y se va hacia las ramas. Es decir, se accede de forma jerarquizada, por niveles, hasta la información que nos interesa. (González, 2011)

- En red: en este modelo se origina la redundancia de campos, aumentando la complejidad en el acceso a cada uno de los componentes, mediante la codificación de claves compuestas.

También se pueden clasificar las bases de datos según su uso, en las que se destacan las Bases de Datos Transaccionales (OLTP) empleadas en transacciones masivas de datos, las Bases de Datos Orientadas a Objetos (OOP), los Sistemas de Información Geográfica (SIG) o Bases de Datos Geográficas, que almacenan datos sobre superficies o puntos geográficos; entre otras.

1.3.5. Gestores de Bases de Datos

Los Sistemas Gestores de Bases de Datos (SGBD) o DBMS (Database Management System) es un software de aplicación usado para acceder, crear y manejar bases de datos. Con la ayuda de los Sistemas Gestores de Bases de Datos, se puede fácilmente crear, recuperar y actualizar data en una base de datos. Un SGBD consiste en un grupo de comandos para manipular la base de datos y actuar como una interfaz entre los usuarios finales y la base. (Kappagantula, 2019)

En la Figura 1-6 se muestra la forma en la que interactúa un Sistema Gestor de Bases de Datos con la base de datos y la parte del cliente donde se recupera la información.

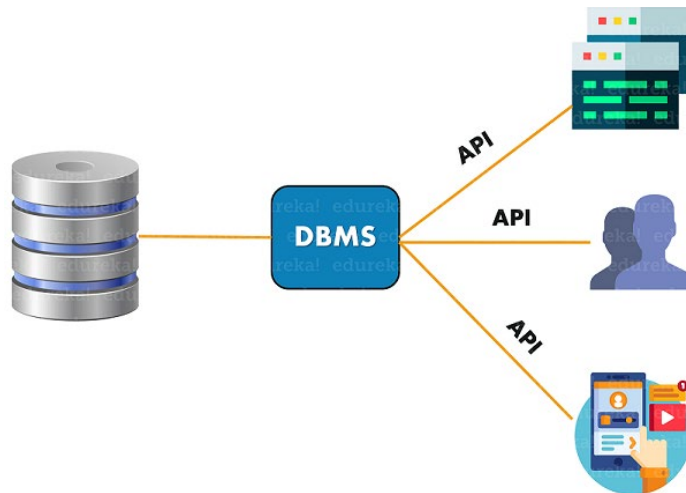


Figura 1-6 Sistema Gestor de Base de Datos

Fuente: Kappagantula, S. (2019)

Existen varios SGBD en el mercado, siendo de vital importancia diferenciar que tipo de gestor es necesario según la base de datos a utilizar. Los principales son (SolarWinds, 2019):

- SGBD Relacional: se basan en bases de datos relacionales y muchas veces se refiere a bases de datos SQL; este tipo de gestor trabaja bajo la premisa ACID, haciendo alusión a sus principales ventajas como atomicidad, consistencia, aislamiento y durabilidad.
- SGBD NoSQL: son mucho menos estructurados que los otros gestores, y están centrados en documentos en lugar de tablas. No es necesario que los datos encajen en filas y columnas bien definidas. Están orientados a colecciones de datos grandes.
- Base de Datos en Memoria: conocidos como IMDB, se caracterizan por almacenar datos en la RAM, proporcionando extrema velocidad para datos comprimidos y no estructurados.
- Otros DBMS: Sistemas gestores en columna o basados en la nube.

Los sistemas gestores, para que se desempeñen óptimamente, es necesario que tengan las siguientes características y funciones (Hueso Ibáñez, 2015):

- Catálogo o diccionario de datos.
- Garantizar la seguridad de los datos.
- Asegurar que la base se actualice correctamente.
- Tener capacidad de integración con software.
- Controlar el ingreso al servidor y datos.

Hoy por hoy, el uso de los sistemas gestores de base de datos ha visto un crecimiento exponencial dada la alta demanda de las empresas en cuanto a la correcta manipulación de datos; en cuanto a los SGBD más populares hasta el 2019, se mantuvieron MySQL, PostgreSQL y SQL Server en los primeros lugares como se indica en la Figura 1-7.

Plataformas de bases de datos más populares				♥ Amor	♥ Miedo	★ Desear	
	2019	2018	%Cambio	2019	2018	2019	2018
mysql	52%	59%	-7%	54%	49%	46%	51%
postgresql	36%	33%	3%	70%	62%	30%	38%
Servidor MS SQL	34%	42%	-8%	58%	52%	43%	48%
SQLite	30%	20%	10%	56%	48%	45%	52%
MongoDB	26%	26%	0%	60%	55%	41%	45%
redis	20%	19%	1%	71%	sesenta y cinco%	29%	36%
MariaDB	17%	14%	3%	59%	53%	41%	47%

Figura 1-7 Plataformas de Bases de Datos más populares

Fuente: Explore Group. (2019)

1.3.5.1. SQL Server

Originalmente creado por Sybase a finales de los 80, en el año 1989 Microsoft se une con Sybase para crear una nueva versión de SQL Server para IBM, dando origen a Microsoft SQL Server e introduciéndolo como uno de los gigantes de las bases de datos. “SQL Server es un sistema de gestión de base de datos relacional, lo que le confiere una gran capacidad de gestionar los datos, conservando su integridad y coherencia” (Gabillaud, 2015).

Siendo sus principales funciones almacenar y recuperar data usada por otras aplicaciones, es necesario diferenciar entre las distintas versiones (actualizaciones disponibles) y ediciones que se encuentran actualmente en uso, con el fin de determinar la versión idónea de SQL Server a correr en el sistema según las necesidades del usuario.

Actualmente, la versión más reciente liberada es la 15.0 o SQL Server 2019, pero también se encuentran vigentes las versiones 2012, 2014, 2016 y 2017. Otro aspecto fundamental por considerar es que existen cinco ediciones de SQL Server y “estas ediciones no solo tienen diferentes niveles de funcionalidad, sino también diferentes consideraciones de licencia” (Carter, 2019):

- Enterprise: edición con todas las funciones de SQL Server para sistemas empresariales y aplicaciones críticas.
- Standard: funcionalidad básica de base de datos y BI, dirigida a sistemas de nivel departamental y aplicaciones no críticas.
- Web: solo está disponible para proveedores de servicios que alojan sitios web públicos que utilizan SQL Server.
- Developer: una edición con todas las funciones al nivel de la edición Enterprise, pero diseñada para usarse en desarrollo y no permitida para sistemas de producción.

- Express: la versión gratuita y más simple de SQL Server, diseñada para aplicaciones pequeñas con requisitos de datos locales.

Las diferentes versiones de SQL Server con sus últimas actualizaciones se encuentran representadas en la Figura 1-8 a continuación:

Versión	Paquete de servicio más reciente	Última RDA	Última actualización acumulativa	Información completa de la versión	Orientación general
Servidor SQL 2019	Ninguna	RDA ² (15.0.2080.9 - enero de 2021)	15 u.m. para 2019 ² (15.0.4198.2 - enero de 2022) 8 u.m. + GDR ² (15.0.4083.2 - enero de 2021)	Compilaciones de SQL Server 2019 ²	Instalación de SQL Server 2019
Servidor SQL 2017	Ninguna	RDA ² (14.0.2037.2 - enero de 2021)	28 u.m. para 2017 ² (14.0.3430.2 - enero de 2022) 22 u.m. + GDR ² (14.0.3370.1 - enero de 2021)	Compilaciones de SQL Server 2017 ²	Instalación de SQL Server 2017
Servidor SQL 2016	SP3 ² (13.0.6300.2 - septiembre de 2021) SP2 ² (13.0.5026.0 - abril de 2018) SP1 ² (13.0.4001.0 - noviembre de 2016)	GDR para SP2 ² (13.0.5103.6 - enero de 2021) GDR para SP1 ² (13.0.4259.0 - julio de 2019) GDR para RTM ² (13.0.1745.2 - enero de 2018)	17 u.m. para 2016 SP2 ² (13.0.5888.11 - marzo de 2021) 15 u.m. + GDR para SP2 ² (13.0.5865.1 - enero de 2021) 15 u.m. + GDR para SP1 ² (13.0.4604.0 - julio de 2019) 15 u.m. para SP1 ² (13.0.4574.0 - mayo de 2019)	Compilaciones de SQL Server 2016 ²	Instalación de SQL Server 2016

Figura 1-8 Actualizaciones disponibles de SQL Server

Fuente: Konidena, R., Xu, S., Yang, C., Spradling, G., Yu, H., Iqbal, S., . . . Assaf, W. (2022)

1.3.6. IDE

“El Entorno de Desarrollo Integrado (IDE), es un conjunto de herramientas de programación que nos permite desarrollar diferentes actividades como: programar, ejecutar, compilar, corregir, buscar errores, entre otras” (Lujan, 2015a). Permitiendo a los programadores consolidar los diferentes aspectos de escribir un programa de computadora.

Los IDE combinan la edición de código fuente, creación de ejecutables y depuración de errores en una sola plataforma.

Características (Red Hat Inc., s.f.):

- Editor de Código Fuente: “ayuda a escribir el código del software con funciones como el resaltado de la sintaxis con indicaciones visuales, el relleno automático específico del lenguaje y la comprobación de errores a medida que se escribe el código” (Red Hat Inc., s.f.).
- Automatización de Compilación Local: poniendo como ejemplo a Java (lenguaje compilado), antes de que se ejecuten los programas, el código fuente del archivo .java debe ser transformado por el compilador en un ejecutable .class. Una vez compilado, el programa puede ser ejecutado desde el terminal. (Codecademy, s.f.)
- Depurador: permite a los programadores examinar diferente tipo de variables e inspeccionar el código deliberadamente, mostrando la ubicación de los errores en el código original de forma gráfica.

En el caso del desarrollo de aplicaciones móviles, un factor clave es la selección del IDE adecuado según las necesidades del producto a codificar. Como ejemplo, si se diseña una app móvil para su uso en iOS y Android, se necesita un IDE que brinde soporte para ambos sistemas operativos; en cambio sí se necesita crear una app

nativa exclusivamente para Android no sería necesario, siendo las opciones más interesantes para trabajar Android Studio o Eclipse.

1.3.6.1. Android Studio

Conocido como el IDE oficial de Google para el sistema operativo Android, que ha reemplazado a Eclipse. “Cuenta con diferencias notables en comparación con Eclipse, esto no hace ni más difícil ni más sencillo el desarrollo de una aplicación, en realidad cambian algunas cosas como la compilación, uso de librerías o paquetes externos, entre otras” (Lujan Castillo, 2019b).

Su primera versión estable fue lanzada en diciembre del 2014, las versiones posteriores a esta fueron introducidas cada dos meses y actualmente se encuentra en la 4.1 que salió en octubre del año 2020, cinco meses después de su último lanzamiento.

En las últimas versiones de Android Studio, dentro de las mejoras innovadoras en el pulido de características y estado del sistema, se encuentran (Williams E. , 2019) :

- Estado del Sistema
 - Mejora de velocidad del IDE
 - Reportes de uso de memoria y configuración
 - Reducción de excepciones
 - Interfaz de usuario congelada
 - Acceso a archivos de entrada / salida

- Pulido de Características
 - Aplicación de cambios

- Mejoras de proyectos
- Editor de diseño
- Soporte para dispositivos Google Pixel
- Actualización de la plataforma IntelliJ 2019.1

Android Studio utiliza el lenguaje Java, aunque también soporta otros lenguajes de programación como C++ y Kotlin, el último siendo el preferido por Google para el desarrollo de apps móviles.

1.3.6.2. Visual Studio

“Es una plataforma de lanzamiento creativa que se puede usar para editar, depurar y compilar código, y luego publicar una aplicación” (Lee, T., Petersen, T., Anand, M., Martens, J., Hogenson, G., Parente, J., Jacobs, M. y Robertson, C., 2022). Funciona en los sistemas operativos Windows y MAC, y cuenta con tres ediciones: Community, Professional y Enterprise.

Actualmente, su última versión es la 2022, convirtiéndose en el primer IDE de 64 bits elaborado por Microsoft.

1.3.7. Framework

Es una solución de alto nivel para la reutilización de piezas de software, un paso adelante en la reutilización simple basada en bibliotecas que permite compartir funciones comunes y lógica genérica de una aplicación de dominio. También asegura un mejor nivel de calidad del producto final, ya que una parte importante de la aplicación ya se encuentra dentro del framework y, por lo tanto, ya ha sido probada. (Salas-Zárate, M., Alor-Hernández, G., Valencia-García, R., Rodríguez-Mazahua, L., Rodríguez-González, A. y López Cuadrado, J., 2015)

Así como existen varios lenguajes de programación, igualmente existen algunos tipos de frameworks de programación, “estos frameworks se adaptan a una gran variedad de propósitos diferentes y resuelven una amplia cantidad de problemas distintos” (Schults, 2020):

- Frameworks de aplicaciones web: para el desarrollo de apps web o sitios web dinámicos.
 - Angular
 - Laravel
 - ASP.NET

- Frameworks de Data Science: enfocados en el análisis de datos haciendo uso de machine learning.
 - Apache Spark
 - TensorFlow
 - PyTorch

- Frameworks de desarrollo móvil: para el desarrollo de apps móviles.
 - Xamarin
 - Ionic
 - Flutter

1.3.7.1. .NET Framework

.NET es un software que se ejecuta en la plataforma del sistema operativo Windows. Proporciona al usuario las herramientas, bibliotecas y los servicios para desarrollar web services y aplicaciones de software mucho más rápido y fácil; el usuario puede acceder a su información desde cualquier dispositivo y en cualquier lugar. (Agarwal, 2019)

Comenzó como la iniciativa de Microsoft llamada Servicios de Windows de Próxima Generación (Next Generation Windows Services) que 3 años de trabajo más tarde, en agosto de 2002, libera .Net Framework 1.0.

Catorce versiones más adelante en abril del 2019, lanza su última versión estable .Net Framework 4.8, manifestando que no existirían futuros lanzamientos en vista que se daría paso a .NET 5, estableciendo la nueva generación de .NET al unificar .NET Framework, .NET Core y Mono desde noviembre del 2020 tal como se visibiliza en la Figura 1-9.

.NET - Una plataforma unificada



Figura 1-9 .NET 5

Fuente: Díaz Alcolea, C. (2020)

1.3.8. APK

El Paquete de Aplicación de Android (APK) o Android Application Package, es un paquete generado por el desarrollador en Android Studio al finalizar la codificación de la app que incluye todo lo necesario para ejecutar la aplicación ya sea para el ambiente de pruebas o para el usuario final, pudiendo diferenciarse dos tipos de APK:

- Sin firma electrónica o APK de depuración: orientado al testing en dispositivos virtuales o simuladores, o en un celular de prueba.
- APK con firma electrónica: firmado por un certificado digital, se encuentra perfectamente habilitado para subir la aplicación a Google Play o para ser ejecutado directamente en un celular, garantizando seguridad en la instalación y su autoría por parte del programador, por lo que las actualizaciones de versión deberán seguir siendo firmadas con el mismo certificado.

Funciona a manera de un archivo .exe, por lo que para su instalación basta con tener el archivo .apk descargado en el móvil.

1.4. Servicios Web

Es un conjunto de estándares de intercambio de datos que permiten la comunicación entre productos, aunque se encuentren en entornos operativos diferentes. Esta capacidad es similar a navegar por Internet en un ordenador personal; se puede acceder a todos los sitios web remotos, independientemente de si estén alojados en Solaris, AIX, Windows, etc. De la misma manera, los servicios web posibilitan que los productos se comuniquen por HTTP con distintos servidores independientemente del entorno operativo. (Broadcom Inc. , 2019)

Existen dos organizaciones que son responsables de la definición de los estándares y de la arquitectura de los servicios web, OASIS y W3C, logrando diferenciar tres componentes principales de la arquitectura de los servicios de acuerdo con su interacción:

- Proveedor
- Consumidor
- Registro del Servicio

Las diferentes interacciones y llamadas realizadas por un servicio web SOAP se plasman en la Figura 1-10.



Figura 1-10 Patrón de interacción de la arquitectura de Web Services

Fuente: Pascual, J. R. (2019)

De acuerdo con Pascual (2019), el patrón de interacción de los servicios web se puede delinear de la siguiente manera:

- El proveedor del servicio genera el contrato de integración WSDL, donde se describirá la especificación del Servicio Web y se registra este WSDL en el directorio UDDI o Registro del Servicio.
- El consumidor del servicio solicita un Servicio Web, a través del UDDI lo localiza en el Registro del Servicio y dispone de su ubicación física.
- El consumidor, o cliente del servicio, a través de la especificación del descriptor (WSDL) envía una solicitud (Request) para un servicio particular al Web Service Listener, que se encarga de recibir y enviar los mensajes en formato SOAP.
- El proveedor del servicio recibe el mensaje SOAP del consumidor y ejecuta la operación relacionada con dicha solicitud. El resultado de esta se devuelve como mensaje SOAP al consumidor.

- El consumidor recibe el SOAP con la respuesta y/o resultado de la operación y lo procesa.

Es necesario recurrir a estándares para formar la base tecnológica de un servicio web, siendo los más relevantes: Lenguaje de Marcado Extensible (XML), Protocolo Simple de Acceso a Objetos (SOAP), Transferencia de Estado Representacional (REST) y Lenguaje de Descripción de Servicios Web (WSDL).

1.4.1. Modelo arquitectónico REST

Los servicios web RESTful se basan en la arquitectura de servicios Transferencia de Estado Representacional (REST) o Representational State Transfer, caracterizada por orientar el desarrollo de servicios web a recursos del sistema y utilizada para sistemas distribuidos.

REST es un conjunto de principios arquitectónicos que se ha ido posicionado como uno de los modelos de diseño predominantes para servicios web desde el año 2000, en el que fue publicado por Roy Fielding en su disertación académica para la Universidad de California.

Este tipo de desarrollo sintetiza el proceso de codificación de aplicaciones ya que “no es necesario realizar una programación exclusiva para cada plataforma, es suficiente con exponer el servicio web para ser consumido por los clientes” (Haro, Guarda, Zambrano Peñaherrera, & Ninahualpa Quiña, 2019), mejorando la portabilidad y permitiendo escalar aplicaciones sin mayor complicación.

Para que una aplicación se considere implementada en el marco de las directrices típicas de RESTful, es necesario que cumpla con ciertas pautas arquitectónicas (Tehreem-naem, 2021):

- Arquitectura cliente – servidor.
- Utilizar métodos HTTP de manera explícita (get, post, put, delete).

- Comunicación cliente – servidor sin estado.
- Cada operación necesita el método URI (fragmentos de URL), que contiene el nombre del recurso y debe estar en formato de directorios.
- Las representaciones emiten una respuesta en formato JSON (formato preferido para mensajes) o XML.
- Restricción del sistema en capas.
- Código según se solicite.

En la Figura 1-11 se muestra la comunicación cliente – servidor de un servicio web RESTful.

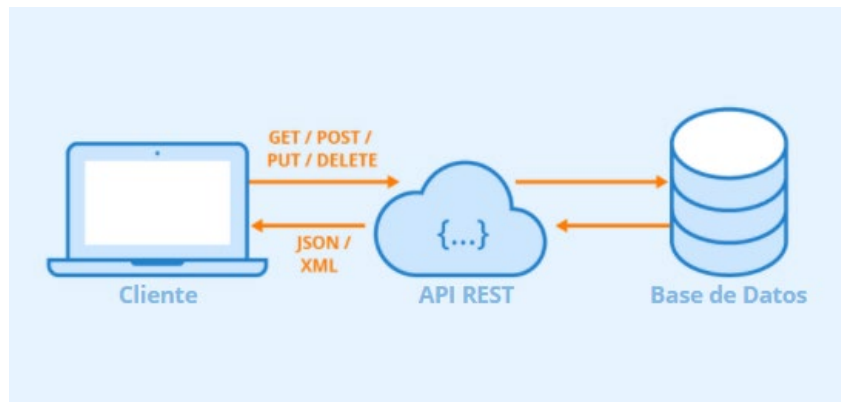


Figura 1-11 Cliente de API REST

Fuente: Tehreem-naem, I. A. (2021)

1.5. Sistemas de información geográfica

Un SIG “es un sistema de información diseñado para trabajar con datos referenciados mediante coordenadas espaciales o geográficas, es tanto un sistema de base de datos con capacidades específicas para datos georreferenciados, como un conjunto de operaciones para trabajar con esos datos” (García, 2021).

“La tecnología de los SIG aplica la ciencia geográfica a través de herramientas para la comprensión y la colaboración. Ayuda a las personas a alcanzar un objetivo común: ganar

inteligencia accionable a partir de todo tipo de datos” (Aeroterra, 2022), haciendo uso de datos geográficos, mapas y análisis espaciales ejemplificados en la Figura 1-12.

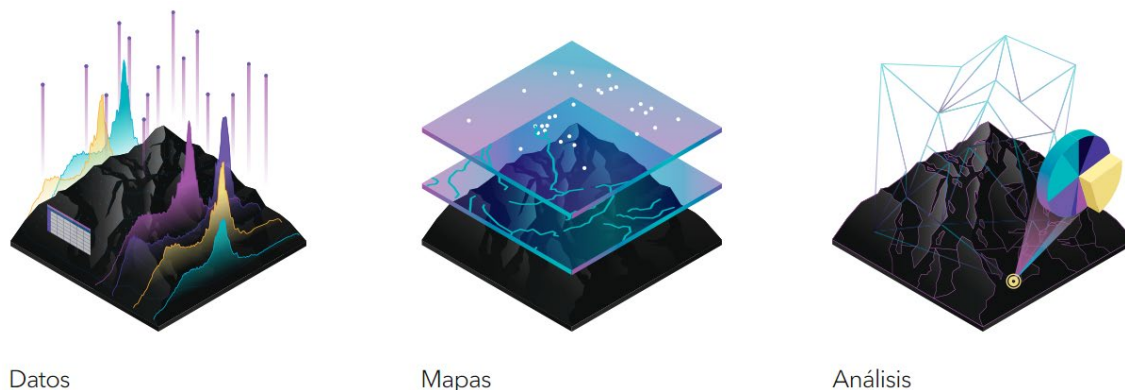


Figura 1-12 Componentes de un SIG

Fuente: Aeroterra. (2022)

1.5.1. Servidores GIS

Son aquellos que integran datos con referenciación geográfica y datos alfanuméricos, permitiendo la generación de mapas completos, tanto para su representación gráfica como para su análisis de forma intuitiva. Así, toda la información geográfica se concentra en un solo servidor desde el cual se extraen y explotan mediante herramientas de GIS, navegadores Web, dispositivos móviles, aplicaciones a medida, entre otros. (Servicio Web del Gobierno Vasco, 2021)

1.5.2. API de Google Maps

Nos podemos referir a API como una Interfaz de Programación de Aplicaciones.

“La propuesta de Google es simplificar el uso de Google Maps para integrar sus mapas en aplicaciones y servicios de terceros” (Cubel, 2018).

De acuerdo con Google (2022), la API de Google Maps se subdivide en tres grandes grupos:

- Mapas: focalizado en crear experiencias ágiles y personalizadas que lleven el mundo real a sus usuarios con mapas estáticos y dinámicos, imágenes de Street View y vistas de 360°.
 - API Mapas Dinámicos
 - API Vista de calle dinámica
 - API Elevación
 - API Mapas Incrustados
 - API Mapas Estáticos
 - API Vista estática de la calle

- Rutas: ayuda a sus usuarios a encontrar la forma ideal de ir de la A a la Z con datos completos y tráfico en tiempo real.
 - API Direcciones
 - API Matriz de distancia
 - API Carreteras

- Lugares: los usuarios pueden descubrir el mundo con datos completos de sus lugares de interés, permitiéndoles encontrar lugares específicos usando números de teléfono, direcciones y más.
 - API Autocompletar
 - API Geocodificación
 - API Geolocalización
 - API Detalles del lugar
 - API Fotos del lugar
 - API Búsqueda de lugares
 - API Zona horaria

1.6. Metodologías ágiles

“Permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno” (Trejo Medina, 2017).

Mejoran el enfoque de desarrollo de software haciéndolo más flexible y con capacidad de mejorar constantemente, influyendo en la reducción de costes y tiempos de entrega del proyecto e incrementando la calidad; cuentan con un manifiesto a manera de guía que posibilita considerar el desarrollo de software desde otro punto de vista sin imponer un conjunto de prácticas.

1.6.1. Programación Extrema o XP

Esta metodología “tiene como objetivo principal que un equipo de desarrollo pueda producir software de mejor calidad de forma constante y a su vez promover una buena calidad de vida para el equipo” (López Mendoza, 2020).

Fue creada por Kent Beck, formalizando su concepto en 1999 en el libro Programación Extrema Explicada, siendo un compendio de todas las mejores prácticas para el desarrollo de software usadas hasta esa fecha utilizadas en un nivel extremo.

Considera 4 variables principales para su ejecución: coste, tiempo, calidad y alcance; propone realizar iteraciones en ciclos que van de una a tres semanas e involucran las fases de análisis, diseño, codificación y pruebas en cada iteración.

Consta de 5 valores, 15 principios y 24 prácticas para la correcta aplicación de la metodología, tomando en cuenta que “los valores sin prácticas son difíciles de

interiorizar y las prácticas sin valores son actividades de memoria sin un propósito” (López Mendoza, 2020).

- Valores: comunicación, simplicidad (hacer solo lo necesario lo más simple posible), retroalimentación, coraje, respeto.
- Principios
 - Principios principales: realimentación rápida, asumir simplicidad, cambio incremental, abrazar el cambio, trabajo de calidad.
- Prácticas
 - Pequeños releases
 - Desarrollo guiado por pruebas
 - Refactorización
 - Programación en parejas
 - Integración continua
 - Historias de usuario
 - Estándares de codificación

En la Figura 1-13 se puede visualizar que en cada iteración de XP se completa un ciclo entero que incluye análisis, diseño, desarrollo y pruebas.



Figura 1-13 Iteraciones de XP

Fuente: Wells, D. (2000)

Antes de empezar con las iteraciones es necesario contar con un Plan de Lanzamiento en el que se identifiquen que Historias de Usuario se van a desarrollar en cada iteración.

La duración de cada iteración determina la velocidad del proyecto que se reflejará en el Plan de Iteraciones.

Durante la fase de desarrollo se pueden identificar nuevas historias de usuario que pueden modificar la velocidad del proyecto agregando nuevas funcionalidades.

Tanto las tareas que quedaron pendientes en una iteración, como las nuevas Historias de Usuario y las Pruebas de Aceptación fallidas, pueden hacer que la Planificación de Iteraciones propuesta varíe.

En cada nueva iteración se corrigen los errores y se pueden incluir nuevas funcionalidades a la versión final del proyecto hasta acabarlo.

La culminación de todo el proyecto depende de que todas las Pruebas de Aceptación sean exitosas.

1.6.1.1. Historias de usuario

Las historias de usuario son parte de la poca documentación requerida por la metodología XP, enfocándose en la persona como el usuario objetivo y la acción que va a realizar para obtener un resultado; respondiendo estas tres interrogantes: ¿a quién?, ¿para qué? y ¿por qué?.

Las pautas para la correcta elaboración de una historia de usuario puntualizan:

- Los roles de usuario no se consideran como una persona, es necesario que la historia incluya el nombre de una persona real.
- Evitar el uso de lenguaje ambiguo y de los conectores “y” u “o”, ya que de ser utilizados estarían dando origen a una nueva historia de usuario.
- Cada una solo se ocupa de una acción en específico.

1.6.1.2. Fase de análisis

Siguiendo con la fase de análisis aplicando la metodología Extreme Programming, es necesario definir los criterios de aceptación para cada historia de usuario planteada en la fase de levantamiento de requerimientos y posteriormente verificar que los criterios se cumplan en las pruebas de aceptación, validando que la historia de usuario se encuentre completa una vez pasada la prueba.

Los criterios de aceptación se enfocan en comprobar que lo propuesto en una historia de usuario funcione adecuadamente, pueden ser escritos de diferentes formas por lo que usan diferentes sintaxis:

- Triple A
 - **1. Arrange** (ordenar / organizar) → al sistema
 - **2. Act** (actuar) → acción
 - **3. Assert** (afirmar) → resultados

- SEAT
 - **1. Setup** (configuración)
 - **2. Execute** (ejecutar)
 - **3. Assert** (asegurar)

- Gherkin
 - **1. Given** (dado) → estado del sistema
 - **2. When** (cuando) → evento es provocado
 - **3. Then** (entonces) → debería ser el resultado

En este proyecto se usará la sintaxis Gherkin y en consecuencia los criterios de aceptación emplearan la siguiente plantilla:

- **1. Given** → Estado inicial o contexto del sistema
- **2. When** → Evento provocado por el usuario
- **3. Then** → Comportamiento esperado del sistema

1.6.1.3. Fases de codificación y pruebas

La metodología XP considera una parte importante a la codificación y pruebas de cualquier sistema a desarrollar, tanto así que plantea como parte vital el realizar test antes de empezar a codificar cualquier unidad o parte de este por más pequeña que esta sea, ya que considera que subestimar alguna de sus funcionalidades puede acarrear un mayor impacto tanto en tiempo, recursos y costos en las siguientes iteraciones.

La Programación Extrema propone que antes de desarrollar cualquier componente se cuente con Tarjetas de Clase – Responsabilidad – Colaboración (CRC) y Pruebas de Unidad, y que al finalizar la codificación posteriormente a la verificación mediante Pruebas Unitarias se realicen Pruebas de Aceptación basadas en las Historias de Usuario planteadas previamente en la fase de levantamiento de requerimientos.

- Tarjetas de Clase – Responsabilidad – Colaboración (CRC) o Class Responsibility Collaborator Diagram: como sus siglas lo indican muestra cuales son las clases principales del sistema y sus diferentes responsabilidades, así como que otras clases colaboran con las mismas con el fin de completar su funcionalidad principal.
- Pruebas de Unidad: O Unit Test, se realizan antes de empezar con la codificación, pueden realizarse de manera automatizada con cualquier software disponible para realizar Pruebas Unitarias Automatizadas tal es el caso de JUnit o de manera manual, no es necesario ser presentado en la documentación.
- Pruebas de Aceptación: esta clase de pruebas se basan en las Historias de Usuario y los Criterios de Aceptación formulados en la fase de Análisis, cuyo objetivo es verificar que la parte del sistema a ser testada arroja el resultado deseado.

2. CAPÍTULO 2 - REQUERIMIENTOS

Este capítulo evalúa la situación actual previa al desarrollo del prototipo, muestra todos los requerimientos del sistema que necesita para su correcta puesta en marcha incluyendo la arquitectura de la aplicación. Se elaboran las historias de usuario y los diagramas de casos de uso.

2.1. Situación actual

Actualmente, en el Ecuador la tecnología para el rastreo de mascotas va creciendo en el mercado, aún, cuando los servicios ofertados por varias compañías y emprendimientos del sector no satisfagan del todo las necesidades de los usuarios.

La oferta va desde empresas para rastreo de vehículos que usan su plataforma ya previamente implementada en combinación con aplicaciones para aprovechar los recursos disponibles y adaptarlos a la geolocalización de mascotas (Finder Rastreo Satelital - Cuenca), collares con códigos QR que funcionan a manera de cédula e incluyen toda la información del dueño para que si esta se extravía cualquier persona pueda escanear el código e informar al propietario (PetNet), collares con dispositivos GPS integrados para la localización que igualmente hacen uso de una app móvil (Smart Track - Guayaquil), el sistema de cedulación para animales domésticos constituido por 3 fases: emisión de una cédula que incluye información de la mascota y del dueño, una placa con código QR y la implantación de un microchip en el animal para rastrearlo fácilmente (qrSIUM).

Las opciones en el país son amplias y todas las anteriormente mencionadas son netamente industria ecuatoriana, incluso tenemos aplicaciones móviles utilizadas específicamente para reportar mascotas extraviadas (Anipal) o como es el caso de Kiarama donde se puede buscar una familia para hospedar a cualquier animal doméstico.

Sin embargo, estos servicios ofrecidos por diferentes compañías del país tienen algunas limitaciones, algunos necesitan adquirir un plan mensual para proveer de una app de

rastreo y dar seguimiento personalizado al usuario, otros en cuanto a funcionalidad no son tan intuitivos para el cliente, datos de los usuarios quedan expuestos, exigen demasiada información por parte de los propietarios de las mascotas, solo funcionan en caso que el animal doméstico se pierda, los costos a veces son elevados y a falta de información las personas desconocen de estos productos.

Las diversas alternativas presentadas para satisfacer las demandas del mercado ecuatoriano deben considerar varios aspectos: facilidad de adquisición en cuanto a costos, que el usuario use tecnología a su alcance tal es el caso de los smartphones combinado con collares que incluyan un pequeño dispositivo de rastreo que no sea invasivo para el animal, sistema no focalizado en canes sino que incluya a cualquier tipo de mascota, para uso continuo y no se brinde el servicio solo por un periodo limitado de tiempo, no requerir mayor mantenimiento, fácil de usar y que cumpla con sus funciones de rastreo, notificaciones y alarma integrada si es el caso.

Este prototipo de aplicación móvil busca ser una herramienta que aproveche los sensores incluidos en los celulares conjuntamente con la tecnología de chips GPS, que sea eficiente y proporcione rastreo en tiempo real de cualquier mascota añadiendo un plus: geofencing, asequible para todo el público, con miras al futuro de convertirse en una app móvil, satisfaciendo las brechas del mercado de geolocalización para animales de compañía, inclusive pudiendo sus funciones ser extrapoladas a rastreo de personas u objetos.

2.2. Levantamiento de Requerimientos

2.2.1. Detalles del sistema

La elaboración de este prototipo de app móvil comprende 3 partes: hardware, software y la metodología de desarrollo.

En cuanto a hardware se necesita un smartphone con soporte mínimo del sistema operativo Android Nougat 7.0 con su actualización 7.1.1 hasta su más reciente y definitiva versión: Android 12, que se encontraba habilitada en 22 dispositivos de 11 marcas hasta agosto de 2021 en su primera beta y ahora puede ser utilizada en 144 modelos, en un inicio se utilizó un dispositivo GPS Tracker marca Ablegrid modelo TK-102 con una duración de batería de 96 horas que incluye un chip GPS SIRF3 para redes GSM/GPRS con sensibilidad -159dBm, que después fue reemplazado por un celular Huawei P30 Lite y un LG G5.

- Huawei P30 Lite (Pérez, 2021):
 - “Pantalla: 6,15" IPS/LCD FullHD+” (Pérez, 2021)
 - “Procesador: Kirin 710” (Pérez, 2021)
 - RAM: 4 GB
 - “Almacenamiento: 128GB + microSD” (Pérez, 2021)
 - Sistema Operativo: Android 10 Pie con EMUI 10.0.0

- LG G5 (Penalva, 2017):
 - “Pantalla: IPS Quantum Display 5,3 pulgadas” (Penalva, 2017)
 - “Procesador: Snapdragon 820” (Penalva, 2017)
 - “RAM: 4 GB DDR4” (Penalva, 2017)
 - Almacenamiento: 32 GB
 - “Sistema Operativo: Android 8 Oreo” (Penalva, 2017)

El IDE por usar para el desarrollo del prototipo es Android Studio 4.1 con el SGBD SQL Server, la codificación se realizará con Java para la app y con C# para los servicios web RESTful.

La metodología ágil por aplicar es Extreme Programming, cada iteración tendrá un tiempo de duración de 1 semana, comprenderá de 4 historias de usuario para el levantamiento de requerimientos que serán posteriormente evaluadas con sus respectivas pruebas de aceptación en la fase de desarrollo; para la elaboración de

las historias de usuario fue necesario recopilar información de varias personas propietarias de mascotas que evidenciaron diferentes necesidades a ser cubiertas.

Considerando que se realizó un cambio en el dispositivo de rastreo, el prototipo sigue manteniéndose de la misma manera en cuanto a análisis, diseño y funcionalidades, los cambios se vieron reflejados en la codificación para recuperar la ubicación de la mascota y la geovalla.

El prototipo a diseñar permitirá que el usuario final rastree su mascota en tiempo real y si es necesario levantar una valla para delimitar el perímetro geográfico en donde se va a encontrar la misma.

2.2.2. *Arquitectura de la aplicación (con dispositivos móviles)*

La siguiente Figura 2-1 representa la arquitectura del prototipo de aplicativo móvil, siguiendo los lineamientos de la arquitectura REST.

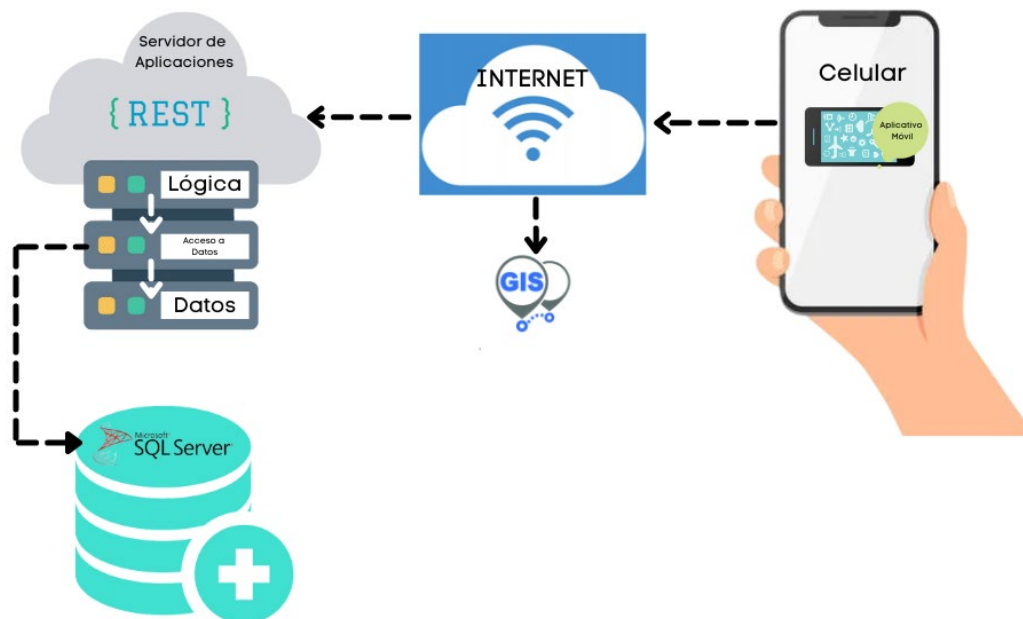


Figura 2-14 *Arquitectura prototipo aplicativo móvil*

2.2.3. Historias de usuario

En las tablas de la 2-1 a la 2-4 se representa cada escenario correspondiente a una iteración de desarrollo, para la codificación de este prototipo se han definido cuatro iteraciones cada una con una funcionalidad específica.

La Tabla 2-1 corresponde a la primera iteración cuya funcionalidad es el login en la aplicación.

Tabla 2-1 Historia de Usuario # 1

Historia de Usuario No. 1

<i>Persona</i>	Como Claudine
<i>Acción</i>	Necesito validar mis credenciales
<i>Valor para el usuario</i>	Para poder ingresar a la aplicación

La Tabla 2-2 corresponde a la segunda iteración cuya funcionalidad es el registro de las mascotas.

Tabla 2-2 Historia de Usuario # 2

Historia de Usuario No. 2

<i>Persona</i>	Como Claudine
<i>Acción</i>	Necesito ingresar la información de mi mascota
<i>Valor para el usuario</i>	Para poder tenerla registrada

La Tabla 2-3 corresponde a la tercera iteración cuya funcionalidad es obtener la ubicación en tiempo real de la mascota.

Tabla 2-3 Historia de Usuario # 3

Historia de Usuario No. 3

<i>Persona</i>	Como Claudine
<i>Acción</i>	Necesito obtener la ubicación de mi mascota
<i>Valor para el usuario</i>	Para poder localizarla en el mapa

La Tabla 2-4 corresponde a la cuarta iteración cuya funcionalidad es añadir la geovalla para la mascota.

Tabla 2-4 Historia de Usuario # 4

Historia de Usuario No. 4

<i>Persona</i>	Como Claudine
<i>Acción</i>	Necesito definir los límites de la valla virtual
<i>Valor para el usuario</i>	Para poder marcar el perímetro de ubicación de mi mascota

2.3. Diagramas de casos de uso

2.3.1. Caso de uso General

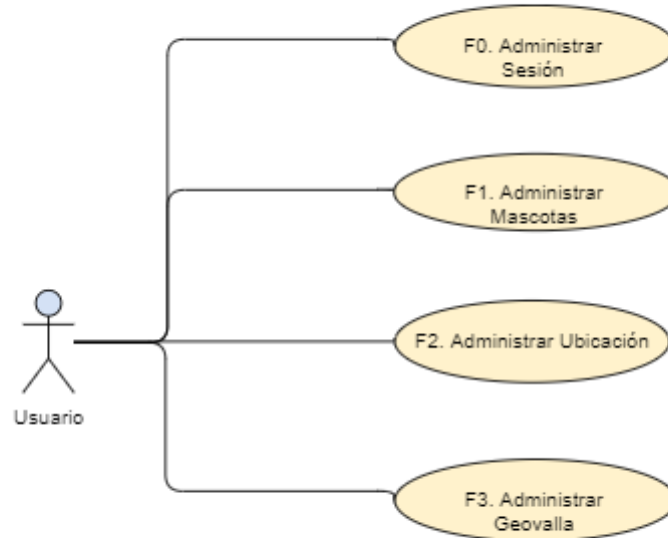


Diagrama 2-1 Caso de Uso General

2.3.2. Caso de uso F0 – Administrar Sesión

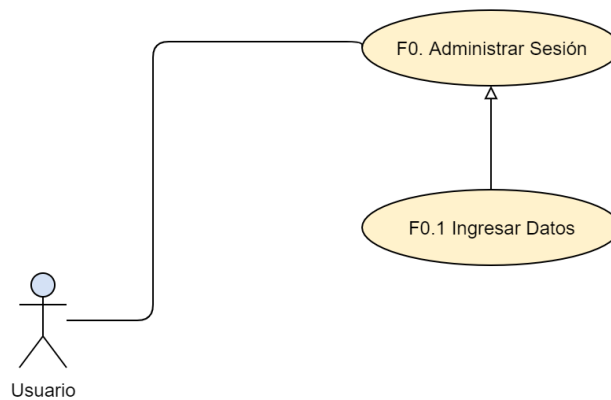


Diagrama 2-2 Caso de Uso - F0. Adm. Sesión

En la Tabla 2-5 se detallan los flujos básicos y alternativos del Caso de Uso F0, correspondiente a la funcionalidad de Administrar Sesión de la app.

Tabla 2-5 Flujos básicos y alternativos F0 Adm. Sesión

F0. ADMINISTRAR SESIÓN		
Descripción	Los usuarios necesitan iniciar sesión para poder utilizar las funcionalidades de la aplicación	
Flujos Básicos	ACTOR	SISTEMA
	F 0.1 Ingresar Datos	
	<ul style="list-style-type: none"> • Ingresar a la aplicación 	Muestra la pantalla de inicio de sesión
	<ul style="list-style-type: none"> • Ingresar usuario y password 	
Flujos Alternativos	<ul style="list-style-type: none"> • Da clic en el botón INGRESAR 	Valida los datos y permite el ingreso al sistema de la app móvil
	<ul style="list-style-type: none"> • F 0.1 Ingresar datos erróneos 	Envía un mensaje de error, los datos ingresados son incorrectos

2.3.3. Caso de uso F1 – Administrar Mascotas

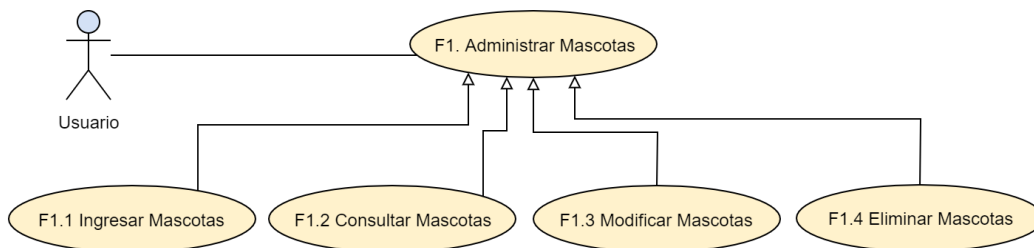


Diagrama 2-3 Caso de Uso - F1. Adm. Mascotas

En la Tabla 2-6 se detallan los flujos básicos y alternativos del Caso de Uso F1, correspondiente a la funcionalidad de Administrar Mascotas de la app.

Tabla 2-6 Flujos básicos y alternativos F1 Adm. Mascotas

F1. ADMINISTRAR MASCOTAS			
Descripción	Los usuarios necesitan registrar los datos básicos de sus mascotas previamente para poder obtener su ubicación		
Flujos Básicos	ACTOR	SISTEMA	
	F 1.1 Ingresar Mascotas		
	<ul style="list-style-type: none"> • Selecciona la opción Nueva Mascota 	Muestra la pantalla de ingreso de datos de la mascota	
	<ul style="list-style-type: none"> • Ingresa los datos de la mascota (nombre y raza) 		
	<ul style="list-style-type: none"> • Selecciona la opción GUARDAR 	Valida y guarda los datos de la mascota	
	F 1.2 Consultar Mascotas		
	<ul style="list-style-type: none"> • Va a la pantalla principal de mascotas 	Muestra las mascotas en una lista	
	F 1.3 Modificar Mascotas		
	<ul style="list-style-type: none"> • Selecciona la mascota que quiere modificar 	Muestra los datos correspondientes a la selección	
	<ul style="list-style-type: none"> • Ingresa los datos modificados 		
	<ul style="list-style-type: none"> • Selecciona la opción GUARDAR 	Valida y guarda los datos modificados de la mascota	

		F 1.4 Eliminar Mascotas
Flujos Alternativos	<ul style="list-style-type: none"> • Selecciona la mascota que quiere modificar 	
	<ul style="list-style-type: none"> • Da clic en el botón ELIMINAR 	Elimina la mascota del sistema
	<ul style="list-style-type: none"> • F 1.1 No ingresa todos los datos 	Envía un mensaje de error, faltan datos por ingresar
	<ul style="list-style-type: none"> • F 1.3 Presiona el botón GUARDAR 	Envía un mensaje de error, no se pudo modificar la mascota
	<ul style="list-style-type: none"> • F 1.4 Presiona el botón ELIMINAR 	Envía un mensaje de error, no se puede eliminar la mascota

2.3.4. Caso de uso F2 – Administrar Ubicación

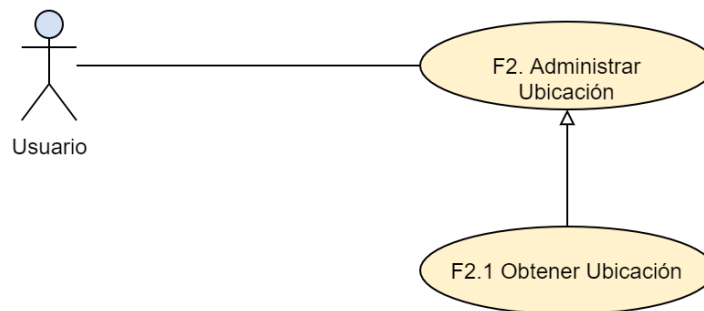


Diagrama 2-4 Caso de Uso - F2. Adm. Ubicación

En la Tabla 2-7 se detallan los flujos básicos y alternativos del Caso de Uso F2, correspondiente a la funcionalidad de Administrar Ubicación de la app.

Tabla 2-7 Flujos básicos y alternativos F2 Adm. Ubicación

F2. ADMINISTRAR UBICACIÓN		
Descripción	Los usuarios podrán obtener la ubicación en tiempo real de sus mascotas	
Flujos Básicos	ACTOR	SISTEMA
	F 2.1 Obtener Ubicación	
	<ul style="list-style-type: none"> • Selecciona la mascota que quiere localizar 	Muestra el mapa con la ubicación en tiempo real de la mascota
Flujos Alternativos	<ul style="list-style-type: none"> • F 2.1 Selecciona la mascota que quiere localizar 	Envía un mensaje de error, mascota no localizada

2.3.5. Caso de uso F3 – Administrar Geovalla

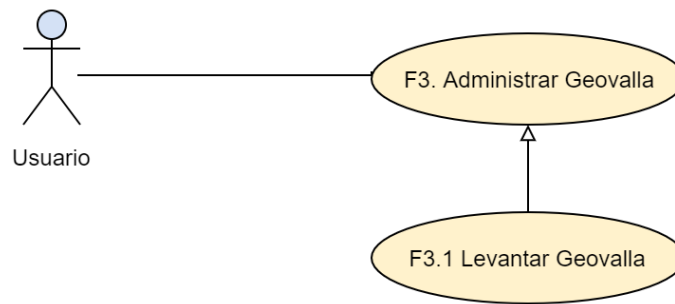


Diagrama 2-5 Caso de Uso - F3. Adm. Geovalla

En la Tabla 2-8 se detallan los flujos básicos y alternativos del Caso de Uso F3, correspondiente a la funcionalidad de Administrar Geovalla de la app.

Tabla 2-8 Flujos básicos y alternativos F3 Adm. Geovalla

F3. ADMINISTRAR GEOVALLA		
Descripción	Los usuarios podrán hacer uso de una geovalla para su mascota	
Flujos Básicos	ACTOR	SISTEMA
	F 3.1 Levantar Geovalla	
	<ul style="list-style-type: none"> Selecciona la opción GEOVALLA 	Muestra un perímetro virtual alrededor de la ubicación de la mascota
Flujos Alternativos	<ul style="list-style-type: none"> F 3.1 Selecciona la opción GEOVALLA 	Envía un mensaje de error, no se puede delimitar la geovalla

3. CAPÍTULO 3 - DESARROLLO

Este capítulo comprende todo el desarrollo de las iteraciones del proyecto, incluye el Plan de Iteraciones XP, los criterios de aceptación en la fase de Análisis, el diseño de las interfaces gráficas, las tarjetas CRC en la codificación y las pruebas de aceptación.

3.1. Planificación de las iteraciones

En la siguiente Tabla 3-1 se visualiza el cronograma de cada una de las iteraciones de desarrollo.

Tabla 3-9 Plan de Iteraciones

ACTIVIDADES	Noviembre				Diciembre				Enero				Febrero			
	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
3. DESARROLLO																
3.2 Primera Iteración																
3.3 Segunda Iteración																
3.4 Tercera Iteración																
3.5 Cuarta Iteración																

3.2. Primera iteración

3.2.1. Análisis

En la siguiente Tabla 3-2 se visualiza el criterio de aceptación para la Historia de Usuario 1.

Tabla 3-10 Criterio de Aceptación # 1

Criterio de Aceptación No. 1

<i>Given</i>	Dadas las credenciales válidas que se ingresan en el formulario
<i>When</i>	Cuando se hace clic en el botón INGRESAR
<i>Then</i>	Entonces el usuario es redirigido a su página de perfil

3.2.2. Diseño

3.2.2.1. Interfaz gráfica

Para el diseño del prototipo de la interfaz gráfica se hizo uso de Justinmind, herramienta de software todo en 1 para prototipado de aplicaciones móviles y web.

La Figura 3-1 muestra el diseño del prototipo de la pantalla de ingreso a la aplicación, mientras que la Figura 3-2 muestra cómo se vería la misma ya en el móvil .



Figura 3-15 Prototipo Interfaz Gráfica - Ingreso

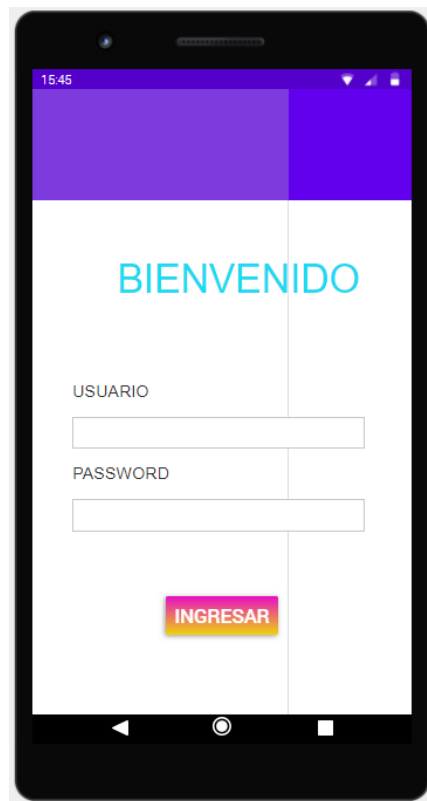


Figura 3-16 Prototipo Vista en Móvil - Ingreso

3.2.3. Codificación y pruebas

- Tarjeta CRC

En la Tabla 3-3 se representa la Tarjeta de Clase - Responsabilidad – Colaboración para la primera funcionalidad de inicio de sesión.

Tabla 3-11 Tarjeta CRC # 1

<i>Atributos</i>		USUARIO	
		Ingresar credenciales	<i>Responsabilidades</i>
- User_id			
- User_name			
- User_password			
		Sistema	<i>Colaboraciones</i>

- Prueba de aceptación

En la Tabla 3-4 se visualiza la prueba de aceptación para la Historia de Usuario 1.

Tabla 3-12 Prueba de Aceptación # 1

Prueba de Aceptación No. 1

<i>Acción</i>	Validación de credenciales
<i>Resultado Esperado</i>	Éxito: el usuario es redirigido a su página de perfil
<i>Estado</i>	Aceptado

3.3. Segunda iteración

3.3.1. Análisis

En la siguiente Tabla 3-5 se visualiza el criterio de aceptación para la Historia de Usuario 2.

Tabla 3-13 Criterio de Aceptación # 2

Criterio de Aceptación No. 2

<i>Given</i>	Dados los datos válidos de mi mascota ingresados en el formulario
<i>When</i>	Cuando se hace clic en el botón AGREGAR
<i>Then</i>	Entonces el usuario es redirigido a la página donde se encuentran enlistadas sus mascotas

3.3.2. Diseño

3.3.2.1. Interfaz gráfica

La Figura 3-3 muestra el diseño del prototipo de la pantalla con la lista de mascotas correspondientes al usuario, mientras que la Figura 3-4 muestra cómo se vería la misma ya en el móvil .

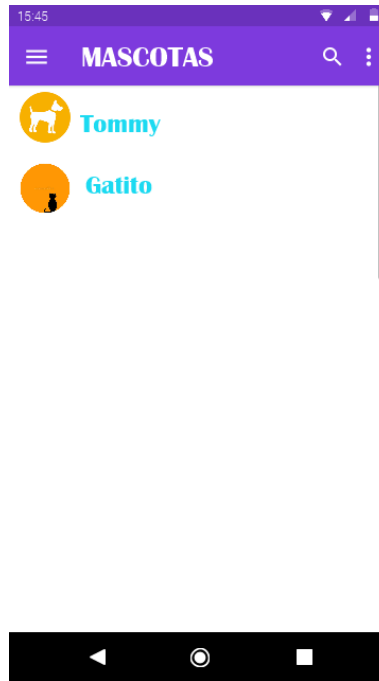


Figura 3-17 *Prototipo Interfaz Gráfica - Lista Mascotas*

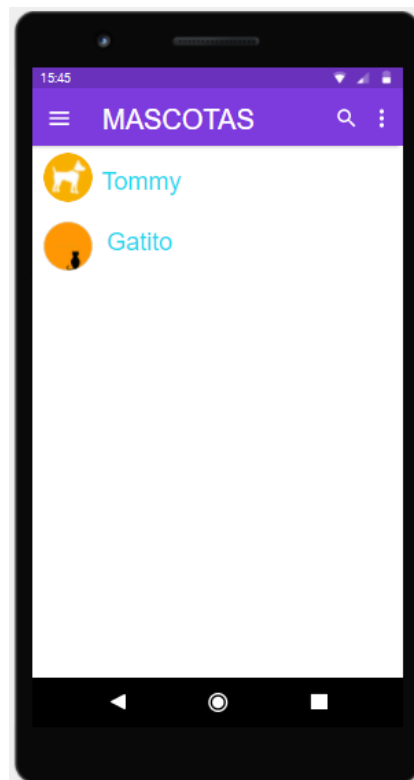


Figura 3-18 *Prototipo Vista en Móvil - Lista Mascotas*

3.3.3. Codificación y pruebas

- Tarjeta CRC

En la Tabla 3-6 se representa la Tarjeta de Clase - Responsabilidad – Colaboración para la segunda funcionalidad de administración de mascotas.

Tabla 3-14 Tarjeta CRC # 2

<i>Atributos</i>	USUARIO	
	Ingresar mascota	<i>Responsabilidades</i>
	Consultar macota	
	Actualizar	
	Eliminar mascota	
- Pet_id - Pet_name - Pet_raza	Sistema Pet	<i>Colaboraciones</i>

- Prueba de aceptación

En la Tabla 3-7 se visualiza la prueba de aceptación para la Historia de Usuario 2.

Tabla 3-15 Prueba de Aceptación # 2

Prueba de aceptación No. 2

<i>Acción</i>	Ingreso de mascotas
<i>Resultado Esperado</i>	Éxito: el usuario es redirigido a la página donde se encuentran enlistadas sus mascotas
<i>Estado</i>	Aceptado

3.4. Tercera iteración

3.4.1. Análisis

En la siguiente Tabla 3-8 se visualiza el criterio de aceptación para la Historia de Usuario 3.

Tabla 3-16 Criterio de Aceptación # 3

Criterio de Aceptación No. 3

<i>Given</i>	Dadas las mascotas enlistadas en la página de mascotas
<i>When</i>	Cuando se hace clic en el nombre de una mascota
<i>Then</i>	Entonces el usuario es redirigido a la pantalla donde se obtiene la ubicación en tiempo real de su mascota

3.4.2. Diseño

3.4.2.1. Interfaz gráfica

La Figura 3-5 muestra el diseño del prototipo de la pantalla cuando se logra obtener la posición en tiempo real de una mascota, mientras que la Figura 3-6 muestra cómo se vería la misma ya en el móvil.

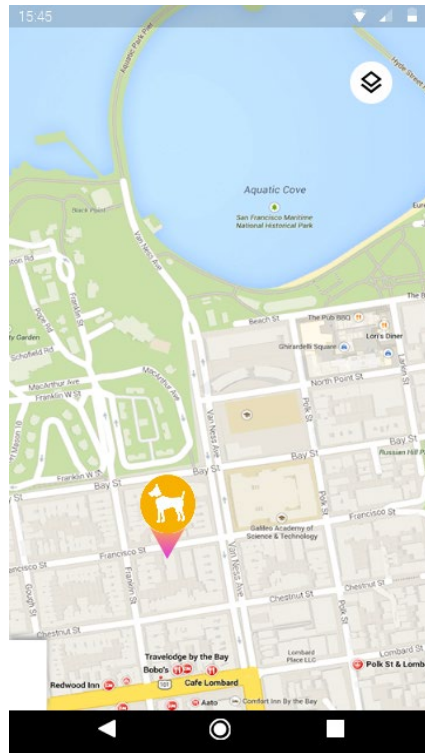


Figura 3-19 Prototipo Interfaz Gráfica – Geolocalización

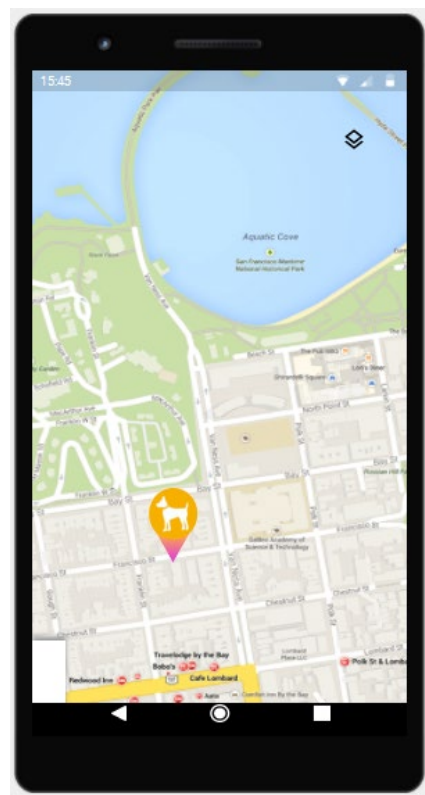


Figura 3-20 Prototipo Vista en Móvil - Geolocalización

3.4.3. Codificación y pruebas

- **LIMITACIONES**

Como se ha venido planteando, este prototipo de app móvil haría uso de un GPS tracker para el rastreo de mascotas, el cual pertenece a la compañía estadounidense Ablegrid.

Por motivos ajenos a la codificación y ya que se trata de un chip no homologado en el país y de fabricación extranjera, el desarrollo del prototipo se ha visto afectado en la parte técnica debido a que el dispositivo de rastreo GPS no está diseñado para ser usado con aplicaciones externas, es decir se debe hacer uso de la app de rastreo proporcionada por la compañía.

En la Figura 3-7 se visualiza la tarjeta SIM proporcionada por la compañía que fabrica el rastreador GPS.



Figura 3-21 Tarjeta GSM SIM Ablegrid

Existen dos maneras de utilizar las funcionalidades de rastreo del TK-102:

- **1. Modo de Texto SMS**

Se envía un mensaje SMS desde el móvil al número celular de la sim card del GPS tracker, después de unos segundos el rastreador responderá con un link de Google Maps con la ubicación del dispositivo.

- **2. Modo de Datos en Vivo**

Se puede configurar el rastreador para que suba los datos de la geolocalización a los servidores de la compañía y se puede hacer uso de su aplicativo móvil o página web para visualizar el GPS Tracker moviéndose en tiempo real.

Existe un limitante técnico de gran importancia para el desarrollo de la geovalla; en la guía de configuración e instalación del dispositivo GPS mostrada en la Figura 3-8, se menciona que “debido al complejo setup de la geovalla en el modo text SMS, descontinuaron esa funcionalidad para esta modalidad. La geovalla solo puede ser usada en Live Data Mode con la aplicación móvil del fabricante” (AbleGrid, 2018).

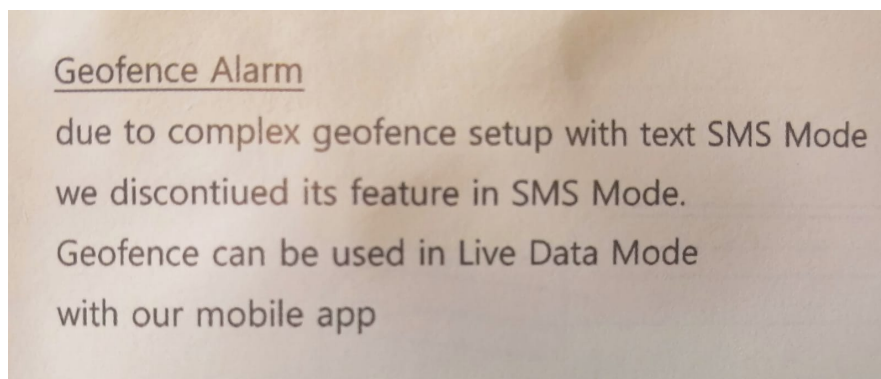


Figura 3-22 Limitaciones configuración geovalla en Modo de Texto SMS

En la siguiente tabla 3-9 se detalla la comparación de las funcionalidades habilitadas del rastreador GPS TK-102 de acuerdo con el país, en este caso Estados Unidos donde se lo adquirió y Ecuador.

Tabla 3-17 Cuadro comparativo funcionalidades TK-102

Funcionalidades del dispositivo de rastreo TK-102

Funcionalidad	Estados Unidos	Ecuador	Observaciones
Rastreo por Modo Texto SMS	SI	SI	El TK-102 viene con una tarjeta SIM GSM de Ablegrid, por lo que las tarjetas SIM ecuatorianas cuando se recupera la ubicación muchas veces la latitud y longitud sale en 0 o no devuelve la posición.
Aplicación de Rastreo	SI	NO	Solo puede usarse la aplicación de rastreo de Ablegrid, no funciona con apps ajenas
Geovalla	SI	NO	Funcionalidad inhabilitada en modo texto, solo funciona en Modo de Datos en Vivo con app

Con el fin de mitigar la falla técnica por la que no se puede hacer uso del dispositivo GPS de rastreo para el prototipo, se hará uso de celulares para simular el GPS Tracker.

- Tarjeta CRC

En la Tabla 3-10 se representa la Tarjeta de Clase - Responsabilidad – Colaboración para la tercera funcionalidad de administración de ubicaciones.

Tabla 3-18 Tarjeta CRC # 3

CRC No. 3

<i>Atributos</i>	USUARIO	
	Localizar mascota	<i>Responsabilidades</i>
- Latitud - Longitud		
	Sistema Ubicación	<i>Colaboraciones</i>

- Prueba de aceptación

En la Tabla 3-11 se visualiza la prueba de aceptación para la Historia de Usuario 3.

Tabla 3-19 Prueba de Aceptación # 3

Prueba de Aceptación No. 3

<i>Acción</i>	Localización de mascotas
<i>Resultado Esperado</i>	Éxito: el usuario es redirigido a la pantalla donde se obtiene la ubicación en tiempo real de su mascota
<i>Estado</i>	Aceptado

3.5. Cuarta iteración

3.5.1. Análisis

En la siguiente Tabla 3-12 se visualiza el criterio de aceptación para la Historia de Usuario 4.

Tabla 3-20 Criterio de Aceptación # 4

Criterio de Aceptación No. 4

<i>Given</i>	Dada la mascota ubicada en tiempo real en el mapa
<i>When</i>	Cuando se hace clic en la opción GEOVALLA
<i>Then</i>	Entonces el usuario puede establecer el límite de su geovalla

3.5.2. Diseño

3.5.2.1. Interfaz gráfica

La Figura 3-9 muestra el diseño del prototipo de la pantalla cuando la geovalla es creada para la mascota, mientras que la Figura 3-10 muestra cómo se vería la misma ya en el móvil.

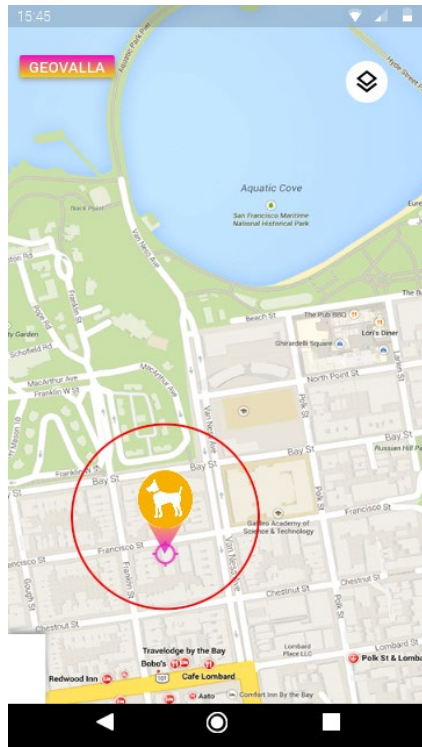


Figura 3-23 Prototipo Interfaz Gráfica – Geofencing

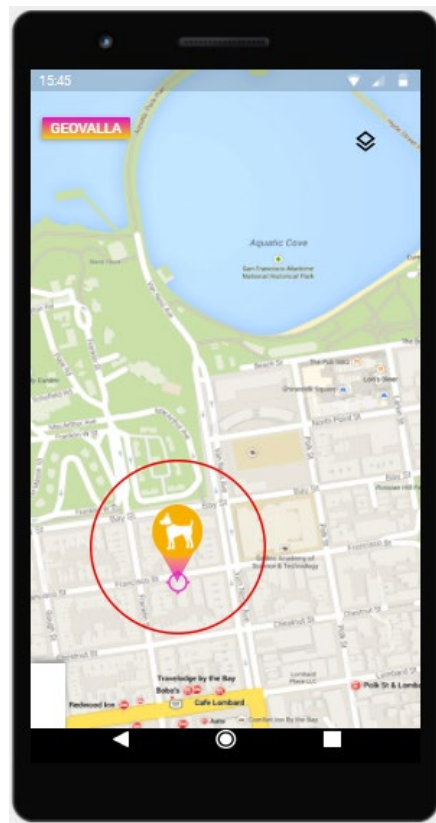


Figura 3-24 Prototipo Vista en Móvil - Geofencing

3.5.3. Codificación y pruebas

- Tarjeta CRC

En la Tabla 3-13 se representa la Tarjeta de Clase - Responsabilidad – Colaboración para la cuarta funcionalidad de administración de geovalla.

Tabla 3-21 Tarjeta CRC # 4

CRC No. 4

<i>Atributos</i>	USUARIO	
	Levantar geovalla	<i>Responsabilidades</i>
- Last_location - Radio_geovalla	Sistema Ubicación	<i>Colaboraciones</i>

- Prueba de aceptación

En la Tabla 3-14 se visualiza la prueba de aceptación para la Historia de Usuario 4.

Tabla 3-22 Prueba de Aceptación # 4

Prueba de Aceptación No. 4

<i>Acción</i>	Levantamiento de geovalla
<i>Resultado Esperado</i>	Éxito: el usuario puede establecer el límite de su geovalla
<i>Estado</i>	Aceptado

4. CAPÍTULO 4 – EJECUCIÓN

Este capítulo cubre en un inicio la configuración del dispositivo de rastreo TK-102 que por motivos técnicos fue reemplazado por móviles para efectos de codificación del prototipo, y la puesta en marcha del aplicativo en celulares a manera de guía de inicialización para su correcto uso.

4.1. Configuración del dispositivo de rastreo GPS

El dispositivo GPS empleado es el rastreador TK-102 de Ablegrid adquirido en Estados Unidos lo que hace que su configuración necesite unos pasos adicionales a los mencionados en el manual de configuración provisto por la empresa.

Por primera vez es imprescindible cargar el dispositivo por 1 a 2 horas antes de iniciar la configuración de red.

Debe utilizar un chip registrado en cualquier operadora, en este caso Claro, con saldo ya que para su correcto funcionamiento necesita comunicarse por SMS con el celular que lo está rastreando y para verificar que el chip ingresado en nuestro dispositivo esté registrado en la red es necesario revisar el foco LED en la parte lateral derecha del mismo donde se pueden presentar los siguientes escenarios:

- Foco LED azul intermitente: buscando red.
- Foco LED azul sólido: la tarjeta sim no se encuentra registrada con la operadora.
- No LED: Listo para usar.

La configuración se realiza mediante SMS; el formato de los comandos a seguir constituye de dos partes: el nombre del comando seguido de la clave del dispositivo que por defecto es **123456**, los pasos a seguir se resumen a continuación:

- Para iniciar se debe textear: **begin123456**, si todo está correcto la respuesta debe ser **begin ok!**, como en la Figura 4-1.

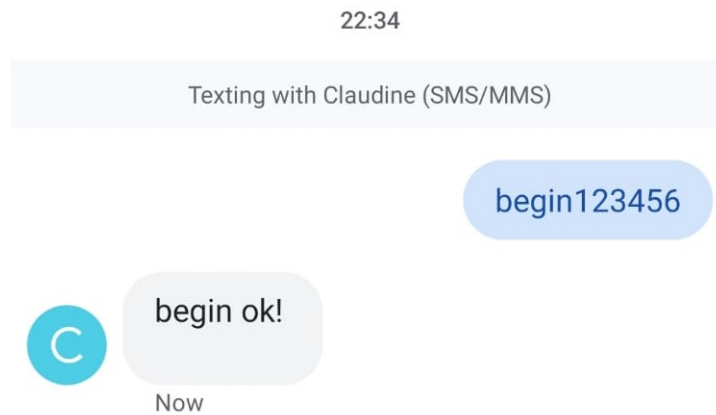


Figura 4-25 Configuración - Inicio

En caso de que el APN no se encuentre configurado, no se recibirá el mensaje **begin ok!** como respuesta, adicionalmente se deberá repetir el proceso una vez más después de enviar el mensaje **APN123456 internet.claro.com.ec** cuya respuesta sería: **apn ok!**, como en la Figura 4-2.

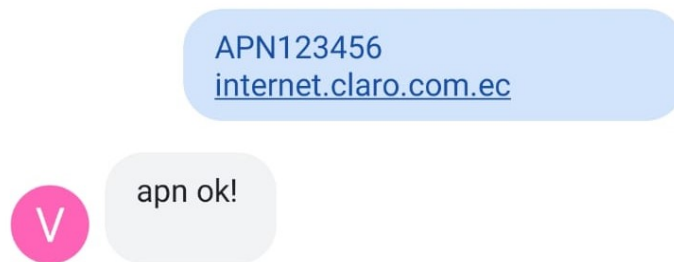


Figura 4-26 Configuración - APN

- Para configurar las respuestas por SMS se debe textear: **SMS123456**, respuesta: **sms ok!**, como en la Figura 4-3.

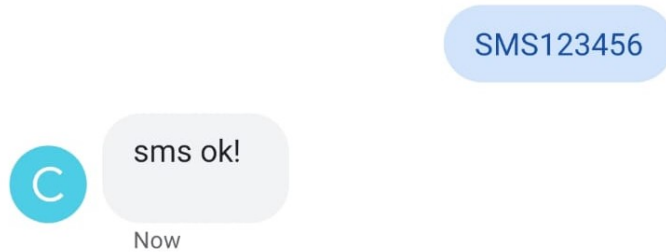


Figura 4-27 Configuración - SMS

- Para empezar a usar el rastreador y obtener la ubicación en tiempo real se debe textear: [smslink123456](#), como se muestra en la Figura 4-4 y cuya respuesta es un link a Google Maps con la ubicación del dispositivo, mostrado en la Figura 4-5.

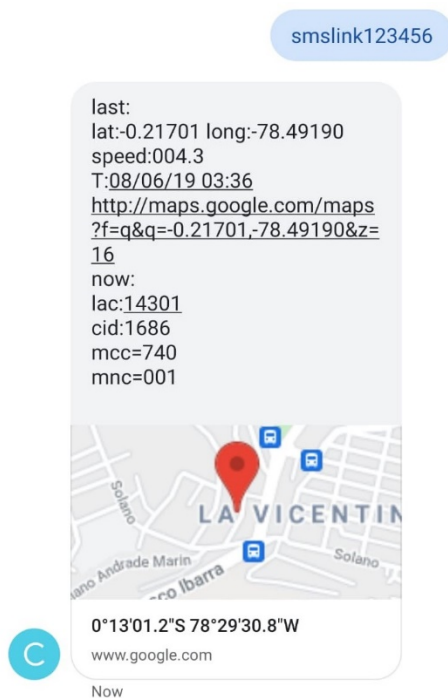


Figura 4-28 Configuración - GPS



Figura 4-29 Configuración - Ubicación GPS mapa

En caso de que el link a la ubicación GPS tenga valores en 0 para las coordenadas, se debe textear el siguiente comando `t020s***n12345` y la respuesta sería: `t020s***n ok!`, como se muestra en la Figura 4-6.

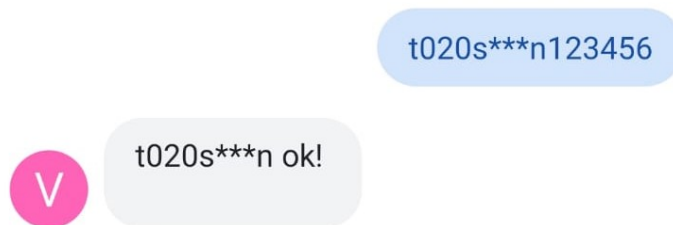


Figura 4-30 Configuración - Valores coordenadas

4.2. Puesta en marcha – Funcionamiento en móviles

Antes de proceder a utilizar la aplicación en nuestro móvil es necesario instalar el APK.

Nota: En algunos dispositivos, por motivos de seguridad, es necesario darle permisos a la aplicación para que pueda instalarse desde un origen desconocido ya que Google últimamente ha incrementado los niveles de seguridad para la instalación de aplicaciones externas.

Los permisos pueden configurarse en la opción **Ajustes/Privacidad** de cualquier celular.

- Como primer paso, en la pantalla de inicio de nuestra app vamos a ingresar el usuario y la contraseña, si son verificados correctamente, se redirigirá a la siguiente pantalla donde se puede ingresar las mascotas.
- En la pantalla de mascotas, se debe ingresar el nombre y la raza, si se registra correctamente, se mostrará la mascota enlistada en la pantalla.
- Una vez registrada la mascota en caso de errores, se puede modificar o eliminar la misma.
- La pantalla principal de la funcionalidad de mascotas muestra una lista de todas las mascotas ingresadas por usuario.
- Para localizar a una mascota en tiempo real, basta con presionar sobre el nombre y escoger la opción “Localizar”, esto nos llevará a ingresar el radio de la geovalla y al dar clic en “Aceptar”, se obtendrá la ubicación de la mascota.
- Finalmente, para hacer uso de la geovalla, se debe hacer clic unos segundos en cualquier lugar del mapa donde se encuentra la mascota, esto añadirá la geovalla con el radio ingresado respectivo, se enviará una notificación cuando la mascota ingrese y salga de la geovalla.

Se debe recordar, que se esperaba utilizar un dispositivo de rastreo GPS colocado en la correa de cualquier mascota, pero ya que ahora se va a simular el uso del TK-102, los dispositivos móviles a ser usados en su reemplazo deberán aparentar ser la mascota en movimiento.

I. CONCLUSIONES

a. Conclusiones del desarrollo de la disertación

- En los fundamentos teóricos se evidencia que la utilización de las API de Google tiene un gran potencial para el desarrollo de nuevas aplicaciones que incluyan tecnología GPS.
- En el levantamiento de requerimientos se concluye que es necesario como primer punto antes de realizar cualquier plan de disertación, realizar un análisis del dispositivo de hardware que se quiera usar y presentar alternativas en caso de que este no cumpla con los requisitos necesarios para la implementación de cualquier software, con el fin de evitar errores técnicos que comprometan la culminación desarrollo del producto.
- En el desarrollo, la metodología ágil Extreme Programming sin duda alguna provee un enfoque rápido y claro de lo que se necesita para desarrollar un sistema o aplicación móvil, es necesario contar con una infraestructura que nos permita una retroalimentación constante con el cliente, facilitando la comunicación y acortando tiempos de codificación.
- En la ejecución, se evidencia la importancia de la correcta configuración del dispositivo de rastreo o del celular a ser monitoreado, haciendo énfasis en la optimización del uso de las funcionalidades de la aplicación.

b. Conclusiones técnicas

- El hardware en uso: Ablegrid 2G GPS Tracker, no funciona con aplicaciones externas ya que la compañía como parte de sus servicios ofertados incluye una app móvil exclusiva para el uso conjunto con el chip.
- Para hacer uso de las API de Google es necesario crear una cuenta en la Plataforma de Google Cloud, adicionalmente se necesita crear una credencial o API Key para vincular las API con la aplicación en desarrollo, siendo necesario restringirla ingresando la clave SHA de la app en la configuración de la API Key.
- En Android Studio, para el desarrollo de este prototipo se utilizó la librería play-services-maps y play-services-location para acceder a las API de Google, y para consumir los servicios creados en .NET se utilizó okhttp3.
- En cuanto a los permisos que deben ser otorgados en Android Studio para el correcto funcionamiento de la aplicación se encuentran: INTERNET, ACCESS_NETWORK_STATE, ACCESS_WIFI_STATE, ACCESS_FINE_LOCATION Y ACCESS_COARSE_LOCATION.

II. RECOMENDACIONES

- Para un trabajo posterior, se sugiere que se añada al presente estudio un módulo que permita una comunicación con los dueños, en caso de que la mascota se extravíe.
- Es importante verificar que los rastreadores GPS ya sean extranjeros o nacionales, cuenten con todas sus funcionalidades desbloqueadas para hacer uso en cualquier parte del mundo.
- Es conveniente tener clara la documentación necesaria a presentarse correspondiente a la metodología Programación Extrema.
- Se sugiere que este proyecto se aproveche como motivación para el desarrollo de una geovalla que podría ser de producción masiva (modelo de negocio).
- Se recomienda restringir la API Key de Google para evitar el uso malintencionado por parte de otras aplicaciones, estableciendo un esquema de seguridad.
- Se aconseja hacer uso de la plataforma Firebase para mejorar en un futuro el desarrollo de aplicaciones con tecnología GPS, ya que provee más herramientas y soporta múltiples plataformas.
- Para que se añada correctamente la geovalla es importante que se autoricen los permisos de funcionamiento de la app en background o `ACCESS_BACKGROUND_LOCATION`, ya que desde Android 10 (nivel API 29) es necesario según las funciones que lo requieran, en este caso crear un cliente de geovallas.

III. BIBLIOGRAFÍA

- A.L.EPH. (2021, Abril 13). *¿Qué es una red de comunicación y sus tipos?* Obtenido de <https://aleph.org.mx/que-es-una-red-de-comunicacion-y-sus-tipos>
- Abdishakur, H., & Jayakrishnan, V. (2019). *Geospatial Data Science Quick Start Guide*. Birmingham: Packt Publishing Ltd.
- AbleGrid. (2018). *Setup and Installation Guide TK-102 Gps Tracker*.
- Ablegrid Corp. (2019). *Ablegrid TK Series RealTime GPS Tracker GSM GPRS System Vehicle Tracking Device TK102 Mini Spy*. Obtenido de <https://www.ablegrid.com/products/ablegrid-realtime-gps-tracker-gsm-gprs-system-vehicle-tracking-device-tk102-mini-spy?variant=31050163719>
- Aeroterra. (2022). *¿Qué es SIG?* Obtenido de <https://www.aeroterra.com/es-ar/que-es-gis/introduccion>
- Agarwal, P. (2019). *.Net Framework and Programming in ASP.NET*. MeetCoogole.
- Agile Alliance. (2021). *Agile Alliance Extreme Programming (XP)*. Obtenido de [https://www.agilealliance.org/glossary/xp/#q=~\(infinite~false~filters~\(postType~\('~post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video\)~tags~\('~xp\) \)~searchTerm~'~sort~false~sortDirection~'asc~page~1](https://www.agilealliance.org/glossary/xp/#q=~(infinite~false~filters~(postType~('~post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~('~xp))~searchTerm~'~sort~false~sortDirection~'asc~page~1)
- Álvarez García, A., de las Heras del Dedo, R., & Lasa Gómez, C. (2012). *Manual Imprescindible de Métodos Ágiles y Scrum*. Anaya Multimedia.
- Anderson, R., Latham, L., Addie, S., Montemagno, J., Dykstra, T., Myers, A., & Dresko, A. (2022, Enero 31). *Docs Microsoft*. Obtenido de *Create web APIs with ASP.NET Core*: <https://docs.microsoft.com/en-us/aspnet/core/web-api/?view=aspnetcore-6.0>
- Android Example. (2022). *Incomming SMS Broadcast Receiver - Android Example*. Obtenido de <https://androidexample.com/incomming-sms-broadcast-receiver>
- Arias, F. (2017, Noviembre 29). *Que es un enlace satelital*. Obtenido de <https://www.loomio.org/d/2bByrCzK/que-es-un-enlace-satelital>
- Arroyo Díaz, C. (2019a). *Programación en JAVA: entorno de programación: sintaxis, elementos, estructuras de control*. Six Ediciones.
- Arroyo Díaz, C. (2019b). *Programación en JAVA II: Clases – Construcción de objetos – Encapsulamiento - Herencia*. Six Ediciones.
- Avast. (2021). *Avast Academy*. Obtenido de *¿Qué es TCP/IP y cómo funciona?*: <https://www.avast.com/es-es/c-what-is-tcp-ip>
- BBVA API Market. (2016, Marzo 23). *API REST: qué es y cuáles son sus ventajas en el desarrollo de proyectos*. Obtenido de <https://www.bbvaapimarket.com/es/mundo-api/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos/>
- BE Software. (2020, Mayo 5). *¿Qué es C# y para qué sirve?* Obtenido de <https://bsw.es/que-es-c/>
- Beas Arco, J., & Gallego Cano, J. (2019). *Diseño de redes de datos y telecomunicaciones (FPB IMRTD)*. Editorial Editex S. A.
- Bello, E. (2021, Abril 28). *IEBS Descubre qué es el Extreme Programming y sus características*. Obtenido de <https://www.iebschool.com/blog/que-es-el-xp-programming-agile-scrum/>
- Benítez, M. Á., & Arias, Á. (2017). *Curso de Introducción a la Administración de Bases de Datos* (Segunda ed.). IT Campus Academy.

- Díaz Alcolea, C. (2020, Noviembre 09). *OpenWebinars Qué es .NET Core*. Obtenido de <https://openwebinars.net/blog/que-es-net-core/>
- Entity Framework Core. (2021). *What is the equivalent of .Configuration.ProxyCreationEnabled in EF Core?* Obtenido de <https://entityframeworkcore.com/knowledge-base/55234943/what-is-the-equivalent-of-configuration-proxycreationenabled-in-ef-core->
- Entity Framework Tutorial. (2020). *Creating a Model for an Existing Database in Entity Framework Core*. Obtenido de <https://www.entityframeworktutorial.net/efcore/create-model-for-existing-database-in-ef-core.aspx>
- Explore Group. (2019, Abril 21). *Explore Group The Most Popular Databases 2019*. Obtenido de <https://www.explore-group.com/blog/the-most-popular-databases-2019/bp46/>
- Fawad, S. (2021, Agosto 15). *Programming Digest*. Obtenido de Receiving and Sending Application in android Studio: <https://programmingdigest.com/receiving-and-sending-application-in-android-studio/>
- Fernández, Y. (2018, Febrero 3). *Xataka Basics - Instalar APK en Android: Cómo hacerlo y cuales son los riesgos*. Obtenido de <https://www.xataka.com/basics/instalar-apk-en-android-como-hacerlo-y-cuales-son-los-riesgos>
- Fernández, Y. (2021, Septiembre 15). *Xataka Basics - APN o nombre del punto de acceso: qué es, para qué sirve y cómo configurarlo*. Obtenido de <https://www.xataka.com/basics/apn-nombre-punto-acceso-que-sirve-como-configurarlo>
- Franceschi, H. (2017). *Android App Development*. Massachusetts: Jones & Bartlett Learning.
- Gabillaud, J. (2015). *SQL Server 2014: Administración de una base de datos transaccional*. Ediciones ENI.
- García, P. (2021, Agosto 11). *Geoinnova ¿Qué es un SIG, GIS o Sistema de Información Geográfica?* Obtenido de <https://geoinnova.org/blog-territorio/que-es-un-sig-gis-o-sistema-de-informacion-geografica/>
- Garmin Ltd. (2017). *Garmin ABOUT GPS*. Obtenido de <https://www8.garmin.com/aboutGPS/>
- Geeks for Geeks. (2021, Enero 18). *How to get user location in Android*. Obtenido de <https://www.geeksforgeeks.org/how-to-get-user-location-in-android/>
- *GitBook*. (s.f.). Obtenido de Creación de una API: <https://juanda.gitbooks.io/webapps/content/api/arquitectura-api-rest.html>
- Gómez Palomo, S., & Morada Gil, E. (2020). *Aproximación a la Ingeniería de Software* (Segunda ed.). Editorial Universitaria Ramón Areces.
- González, A. (2011). *Gestión de bases de datos*. Ediciones de la U.
- Google. (2022). *Encuentre los productos adecuados para su mapa personalizado*. Obtenido de <https://mapsplatform.google.com/maps-products/?hl=es-419#maps>
- Google Developers. (s.f.). *2.1 - Part 1: Sending and Receiving SMS Messages*. Obtenido de https://google-developer-training.github.io/android-developer-phone-sms-course/Lesson%202/2_p_sending_sms_messages.html
- Google Developers. (2017, 10 31). *Verificación automática de SMS con la API de SMS Retriever*. Obtenido de <https://developers.google.com/identity/sms-retriever/overview>
- Google Developers. (s.f.). *Developers Introducción a Android Studio*. Obtenido de <https://developer.android.com/studio/intro>
- Guérin, B.-A. (2018). *ASP.NET con C# en Visual Studio 2017*. Ediciones ENI.
- Haldar, S. (2016). *SQLite Database System Design and Implementation*. Auto Publicado.
- Haro, E., Guarda, T., Zambrano Peñaherrera, A., & Ninahualpa Quiña, G. (2019). Desarrollo backend para aplicaciones web, Servicios. *Revista Ibérica de Sistemas y Tecnologías de Información*, 309-321.

- Hueso Ibáñez, L. (2015). *Administración de Sistemas Gestores de Bases de Datos*. RA-MA. Obtenido de <https://www.geeksforgeeks.org/advantages-of-database-management-system/>
- Huidobro Moya, J. (2012). *Comunicaciones Móviles. Sistemas GSM, UMTS y LTE*. Ra-Ma.
- Huidobro, J. (2014). *Telecomunicaciones: tecnologías, redes y servicios* (Segunda ed.). Ra-Ma Editorial.
- IBM Corporation. (2020). *Protocolos TCP/IP*. Obtenido de <https://www.ibm.com/docs/es/aix/7.1?topic=protocol-tpip-protocols>
- Ichi.Pro. (2021). *¿Qué es un archivo APK? Diferencia entre crear un APK de Android y generar un archivo APK firmado*. Obtenido de <https://ichi.pro/es/que-es-un-archivo-apk-diferencia-entre-crear-un-apk-de-android-y-generar-un-archivo-apk-firmado-64097223182400>
- Icy Science. (2022). *¿Qué es una red celular?*. Obtenido de <https://es.theastrologypage.com/cellular-network>
- Instituto de Ciencias Básicas e Ingeniería de la UAEH. (2018, Diciembre 4). *Definición de Red de Comunicaciones y su Importancia*. Obtenido de http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro27/135_definicion_de_red_de_comunicaciones_y_su_importancia.html
- Javapapers. (2014, Octubre 24). *Get Current Location in Android*. Obtenido de <https://javapapers.com/android/get-current-location-in-android/>
- Kappagantula, S. (2019, Octubre 29). *Edureka! What is DBMS? – A Comprehensive Guide to Database Management Systems*. Obtenido de <https://www.edureka.co/blog/what-is-dbms>
- KIO. (2021). *¿Qué son y para qué sirven los protocolos de comunicación de redes?* Obtenido de <https://www.kionetworks.com/blog/data-center/protocolos-de-comunicacion-de-redes>
- KIO. (2021). *Entérate cuáles son las nuevas redes de comunicación*. Obtenido de <https://www.kionetworks.com/blog/data-center/nuevas-redes-de-comunicacion>
- Konidena, R., Xu, S., Yang, C., Spradling, G., Yu, H., Iqbal, S., . . . Assaf, W. (2022, Enero 27). *Docs Microsoft Determine the version, edition, and update level of SQL Server and its components*. Obtenido de <https://docs.microsoft.com/en-us/troubleshoot/sql/general/determine-version-edition-update-level>
- Kukushkin, A. (2018). *Introduction to Mobile Network Engineering: GSM, 3G-WCDMA, LTE and the Road to 5G*. John Wiley & Sons Ltd. Obtenido de *Introduction to Mobile Network Engineering: GSM, 3G-WCDMA, LTE and the Road to 5G*
- Leader Redes y Comunicaciones. (2021, Julio 30). *Una infraestructura de redes y comunicaciones con futuro*. Obtenido de <https://www.leader-network.com/mantenimiento-redes/una-infraestructura-de-redes-y-comunicaciones-con-futuro/>
- Lee, T. (2021, Septiembre 02). *Docs Microsoft*. Obtenido de Procedimiento Agregar o quitar referencias con el Administrador de referencias: <https://docs.microsoft.com/es-es/visualstudio/ide/how-to-add-or-remove-references-by-using-the-reference-manager?view=vs-2022>
- Lee, T., Petersen, T., Anand, M., Martens, J., Hogenson, G., Parente, J., . . . Robertson, C. (2022, Febrero 10). *Docs Microsoft Bienvenido al IDE de Visual Studio*. Obtenido de <https://docs.microsoft.com/en-us/visualstudio/get-started/visual-studio-ide?view=vs-2022>
- Letelier, P., & Penadés, M. (2006). *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. *Técnica Administrativa*, 05(26).
- Limones, E. (2021, Septiembre 17). *OpenWebinars Protocolo de red: Qué es, tipos y características*. Obtenido de <https://openwebinars.net/blog/protocolo-de-red-que-es-tipos-y-caracteristicas/>
- López Mendoza, M. (2020, Septiembre 18). *OpenWebinars Extreme Programming: Qué es y cómo aplicarlo*. Obtenido de <https://openwebinars.net/blog/extreme-programming-que-es-y-como-aplicarlo/>
- Luján Castillo, J. D. (2015a). *ANDROID. Aprende desde cero a crear aplicaciones*. RC Libros.

- Luján Castillo, J. D. (2019b). *Desarrollo de Aplicaciones Android con Android Studio: Conoce Android Studio*. Editorial Académica Española.
- Maplink. (2021). *Conecta tu empresa con las APIs de Google Maps Platform*. Obtenido de <https://maplink.global/es/google-maps-platform/>
- Máster Móviles UA. (2019). *Programación Hipermedia para Dispositivos Móviles*. Obtenido de <https://mastermoviles.gitbook.io/programacion-hipermedia-para-dispositivos-moviles/servicios-rest>
- Máster Móviles UA. (2019). *Protocolos de comunicación en red*. Obtenido de <https://mastermoviles.gitbook.io/tecnologias2/protocolos-de-comunicacion-en-red>
- Matus, D. (2021, Julio 14). *¿Qué son CDMA y GSM? Te explicamos las principales diferencias*. Obtenido de <https://es.digitaltrends.com/celular/redes-cdma-y-gsm/>
- Mozilla. (2022, Febrero 16). *MDN Web Docs*. Obtenido de Generalidades del protocolo HTTP: <https://developer.mozilla.org/es/docs/Web/HTTP/Overview>
- Muñoz Ordenes, A. (2020, Noviembre 19). *Arquitecturas y Aplicaciones Móviles: ¿Cuál es la Mejor Decisión?* Obtenido de <https://www.gestionenti.com/post/arquitecturas-y-aplicaciones-m%C3%B3viles-cu%C3%A1-es-la-mejor-decisi%C3%B3n>
- Nader, Y. (2020, Enero 14). *Hackr.io What is Programming language?* Obtenido de <https://hackr.io/blog/what-is-programming-language>
- Nait Sidi Moh, A., Ait-Cheik-Bihi, W., Bakhouya, M., Gaber, J., & Wack, M. (2013, Agosto 26). On the Use of Location-Based Services and Geofencing Concepts for Safety and Road Transport Efficiency. *Communications in Computer and Information Science*, 183, 135-144. doi:10.1007/978-3-319-03737-0_14
- National Coordination Office for Space-Based Positioning, Navigation, and Timing. (2022, Febrero 2). *GPS.gov Space Segment*. Obtenido de <https://www.gps.gov/systems/gps/space/#generations>
- Nexus Integra. (s.f.). *API REST: por qué lo necesitas*. Obtenido de <https://nexusintegra.io/es/rest-api-por-que-lo-necesitas/>
- Nolasco Valenzuela, J. (2016). *Desarrollo de Aplicaciones Móviles con Android*. Ra-Ma.
- Noriega Martinez, R. (2017). *El Proceso de Desarrollo de Software* (Segunda ed.). IT Campus Academy.
- Orbegozo Arana, B. (2014). *Desarrollo de Aplicaciones C# con Visual Studio .NET*. Publicaciones Altaria S.L.
- Ordoñez, J. (2021, Septiembre 17). *LaserAnts*. Obtenido de ¿Por qué deberías de considerar las REST API's para la arquitectura de tu aplicación?: <https://laserants.com/2021/09/17/por-que-deberias-de-considerar-las-rest-apis-para-la-arquitectura-de-tu-aplicacion/>
- Orellana, R. (2022, Febrero 3). *Qué es la red 5G, qué ventajas ofrece, cómo funciona (y todo lo demás)*. Obtenido de <https://es.digitaltrends.com/celular/que-es-la-red-5g/>
- Ortega, D., Guevara, M., & Benavides, J. (2016). ELEMENTARY: UN FRAMEWORK DE PROGRAMACIÓN WEB. *Télématique*, 144-171.
- Parada, M. (2019, Noviembre 23). *OpenWebinars Qué es SQL Server*. Obtenido de <https://openwebinars.net/blog/que-es-sql-server/>
- Pascual, J. R. (2019, Diciembre 01). *Disrupción Tecnológica Arquitectura de Servicios Web*. Obtenido de <https://www.disrupciontecnologica.com/arquitectura-de-servicios-web/>
- Penalva, J. (2017, Enero 15). *LG G5, análisis: el smartphone más atrevido del mercado es también muy bueno*. Obtenido de <https://www.xataka.com/analisis/lg-g5-analisis-el-smartphone-mas-atrevido-del-mercado-es-tambien-muy-bueno>

- Pérez Arbesú, L. (2021, Junio 22). *Computer Weekly*. Obtenido de Networking, redes, cableado: Similitudes y diferencias: <https://www.computerweekly.com/es/consejo/Networking-redes-cableado-Similitudes-y-diferencias>
- Pérez Esteso, M. (2019, Julio 18). *Geeky Theory*. Obtenido de Qué es una API REST y para qué se utiliza: <https://geekytheory.com/que-es-una-api-rest-y-para-que-se-utiliza/>
- Pérez, E. (2021, Enero 25). *Huawei P30 Lite, análisis: la triple cámara llega a uno de los grandes aspirantes a superventas del año*. Obtenido de <https://www.xataka.com/analisis/huawei-p30-lite-analisis-caracteristicas-precio-especificaciones>
- Pintado Blanco, T., & Sánchez Herrera, J. (2017). *Nuevas Tendencias en Comunicación Estratégica*. ESIC.
- Pressman, R. (2021). *Ingeniería de Software: un enfoque práctico* (Novena ed.). McGraw-Hill Interamericana de España S.L.
- Programacion.net. (s.f.). *Enviar, recibir y ver el historial de SMS con Android*. Obtenido de https://programacion.net/articulo/Enviar_recibir_y_ver_el_historial_de_sms_con_android_1089
- ProgrammableWeb. (2022). *Search the Largest API Directory on the Web*. Obtenido de <https://www.programmableweb.com/apis/directory>
- Programmer World. (2020, Febrero). *How to track your location using GPS and send it over sms in your Android App? – Complete Source Code*. Obtenido de <https://programmerworld.co/android/how-to-track-your-location-using-gps-and-send-it-over-sms-in-your-android-app-complete-source-code/>
- Programmer World. (2020, Diciembre 20). *YouTube*. Obtenido de How to read and forward certain SMSes programmatically in your Android App? - complete source code: <https://www.youtube.com/watch?v=GhKaoAu-23A>
- Programmer World. (2021, Julio). *Design a location tracking App using GPS in Android Studio*. Obtenido de <https://programmerworld.co/android/design-a-location-tracking-app-using-gps-in-android-studio/>
- QA Stack. (s.f.). *¿Cómo puedo leer mensajes SMS desde el dispositivo mediante programación en Android?* Obtenido de <https://qastack.mx/programming/848728/how-can-i-read-sms-messages-from-the-device-programmatically-in-android>
- Radio Waves Orange. (2022). *¿cómo funciona una red móvil?* Obtenido de <https://radio-waves.orange.com/es/como-funciona-una-red-movil/>
- Raffino, M. E. (2018, Noviembre 27). *Concepto.de Lenguaje de Programación*. Obtenido de <https://concepto.de/lenguaje-de-programacion/>
- Red Hat Inc. (s.f.). *Red Hat ¿Qué es la metodología ágil?* Obtenido de <https://www.redhat.com/es/devops/what-is-agile-methodology>
- Red Hat Inc. (2017, Octubre 31). *¿Qué es una API?* Obtenido de <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>
- Red Hat Inc. (2020, Mayo 8). *¿Qué es una API de REST?* Obtenido de <https://www.redhat.com/es/topics/api/what-is-a-rest-api>
- Red Hat Inc. (2020, Marzo 09). *¿Qué es una arquitectura de aplicaciones?* Obtenido de <https://www.redhat.com/es/topics/cloud-native-apps/what-is-an-application-architecture#:~:text=Una%20arquitectura%20de%20aplicaciones%20describe,dise%C3%B1ar%20una%20aplicaci%C3%B3n%20bien%20estructurada.>
- Red Hat Inc. (2020, Junio 21). *Red Hat Diferencias entre REST y SOAP*. Obtenido de <https://www.redhat.com/es/topics/integration/whats-the-difference-between-soap-rest>
- Red Hat Inc. (s.f.). *Red Hat El concepto de IDE*. Obtenido de <https://www.redhat.com/es/topics/middleware/what-is-ide>

- Robledano, Á. (2019, Julio 22). *OpenWebinars Qué es NET Framework*. Obtenido de <https://openwebinars.net/blog/que-es-net-framework/>
- Robledo, M. F. (2020, Enero 14). *Marketing and Web 20 Lenguajes de Programación más usados en 2021*. Obtenido de <https://www.marketingandweb.es/marketing/lenguajes-de-programacion-mas-usados/>
- Rodriguez, A. (2015, Febrero 09). *IBM Developer Servicios Web de RESTful: Los aspectos básicos*. Obtenido de <https://developer.ibm.com/es/technologies/web-development/articles/ws-restful/>
- Román, M. (2021). *Paradigma Digital*. Obtenido de Leyendo SMS en tu aplicación Android de forma segura.: <https://www.paradigmadigital.com/techbiz/leyendo-sms-app-android/>
- Rosselló Villán, V. (2019, Marzo 15). *IEBS Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa*. Obtenido de <https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>
- Rouse, M. (2015, Enero). *TechTarget SQL Server*. Obtenido de <https://searchdatacenter.techtarget.com/es/definicion/SQL-Server>
- Rungta, K. (2019). *Learn SQL Server in 1 Day*. Publicación Independiente.
- Salas-Zárate, M. d., Alor-Hernández, G., Valencia-García, R., Rodríguez-Mazahua, L., Rodríguez-González, A., & López Cuadrado, J. L. (2015). Analyzing best practices on Web development frameworks: The lift approach. *Science of Computer Programming, 102*, 1-19.
- Samsung. (2021, Diciembre 14). *¿Qué es GSM?* Obtenido de <https://www.samsung.com/latin/support/mobile-devices/what-is-gsm/>
- Sarka, D., Radivojevic, M., & Durkin, W. (2018). *SQL Server 2017 Developer's Guide: A professional guide to designing and developing enterprise database applications*. Birmingham: Packt Publishing Ltd.
- Sayago Heredia, J., Flores, E., & Recalde, A. (2019). Análisis Comparativo entre los Estándares Orientados a Servicios Web SOAP, REST y GRAPHQL. *Revista Antioqueña de las Ciencias Computacionales y la Ingeniería de Software*, 10-22.
- Schults, C. (2020). *Make me a programmer What Is a Programming Framework?* Obtenido de <https://makemeaprogrammer.com/what-is-a-programming-framework/>
- Servicio Web del Gobierno Vasco. (2021). *Servidores GIS*. Obtenido de <https://www.euskadi.eus/documentacion/servidores-gis-p-class-migasestandar-grupo-a-href-informacionservicios-plataformaweb01-a4ogainfes-servicios-de-plataforma-a-br-area-a-href-informacionservicios-plataformaweb01-a4ogainfes-1966-uso-de-productos-y-arquit>
- Si Aguanta Sitio Web. (2019). *Red celular: ¿Qué es?, función y frecuencias*. Obtenido de <https://siaguanta.com/c-tecnologia/red-celular/>
- SICMA21. (2021, Abril 22). *Redes de comunicación industrial: todo lo que necesitas saber*. Obtenido de <https://www.sicma21.com/que-son-las-redes-de-comunicacion-industrial/#:~:text=Las%20comunicaciones%20entre%20dispositivos%20se,entre%20los%20dispositivos%20a%20comunicar.>
- Singh, V. (2020, Abril 09). *Hackr.io What is Frameworks?* Obtenido de <https://hackr.io/blog/what-is-frameworks>
- Small, C. (2018). *What are Hardware and Software?* Britannica Educational Publishing y Rosen Publishing.
- SolarWinds. (2019, Diciembre 13). *DNSstuff What to Know Before Choosing a Database Management System*. Obtenido de <https://www.dnsstuff.com/database-management-system>
- SQL Server Tutorial. (2020). *SQL SERVER TUTORIAL.NET SQL Server Tutorial*. Obtenido de <https://www.sqlservertutorial.net/>
- SQLite Consortium. (2020). *SQLite What Is SQLite?* Obtenido de <https://www.sqlite.org/index.html>

- SQLite Tutorial. (2020). *SQLITE Tutorial What Is SQLite*. Obtenido de <https://www.sqlitetutorial.net/what-is-sqlite/>
- SQLSERVERTUTORIAL.NET. (2021). *SQLSERVERTUTORIAL.NET SQL Server Tutorial*. Obtenido de <https://www.sqlservertutorial.net/>
- Stack Overflow. (2016). *How to get GPS coordinates when get/send SMS? [duplicate]*. Obtenido de <https://stackoverflow.com/questions/33238725/how-to-get-gps-coordinates-when-get-send-sms>
- Stack Overflow. (2017, Marzo). *Can we locate a user via user's phone number in Android?* Obtenido de <https://stackoverflow.com/questions/21372685/can-we-locate-a-user-via-users-phone-number-in-android/21714011>
- Stack Overflow. (2017, Septiembre). *Recibir y leer contenido de SMS en Android*. Obtenido de <https://es.stackoverflow.com/questions/90495/recibir-y-leer-contenido-de-sms-en-android>
- Stack Overflow. (2018, Agosto). *how to get current location as geocode and send as SMS Android?* Obtenido de <https://stackoverflow.com/questions/29157528/how-to-get-current-location-as-geocode-and-send-as-sms-android>
- Stack Overflow. (2021, Marzo). *How can I read SMS messages from the device programmatically in Android?* Obtenido de <https://stackoverflow.com/questions/848728/how-can-i-read-sms-messages-from-the-device-programmatically-in-android>
- Stack Overflow. (2021, Octubre). *Programmatically obtain the phone number of the Android phone*. Obtenido de <https://stackoverflow.com/questions/2480288/programmatically-obtain-the-phone-number-of-the-android-phone>
- Stack Overflow. (2021, Diciembre). *What is the simplest and most robust way to get the user's current location on Android?* Obtenido de <https://stackoverflow.com/questions/3145089/what-is-the-simplest-and-most-robust-way-to-get-the-users-current-location-on-a/3145655#3145655>
- Stack Overflow. (2022, Enero). *Adding ADO.Net Entity Framework gives "The project's target framework does not contain Entity Framework runtime assemblies"*. Obtenido de <https://stackoverflow.com/questions/70580916/adding-ado-net-entity-framework-gives-the-projects-target-framework-does-not-c>
- Systems Group. (2019, Septiembre 4). *La Ingeniería de Software ¿Qué es y qué utilidad tiene?* Obtenido de <https://systemsgroup.es/tecnologias-de-la-informacion/la-ingenieria-de-software-que-es-y-que-utilidad-tiene/32363/>
- Tehreem-naem, I. A. (2021, Enero 28). *Astera Definición de la API REST: ¿Qué es una API REST (API RESTful)?* Obtenido de <https://www.astera.com/es/type/blog/rest-api-definition/>
- TIOBE Software BV. (2021, Junio). *TIOBE Index for June 2021*. Obtenido de <https://www.tiobe.com/tiobe-index/>
- Trafaniuc, V. (2022, Enero 10). *¿Sabes qué es Google APIs? Descúbrelo ahora mismo*. Obtenido de <https://maplink.global/blog/es/que-son-google-apis/>
- Trejo Medina, D. (2017). *Introducción a la ingeniería de software, planeación y gestión de proyectos informáticos*. lulu.com.
- Tuenti España. (2022). *¿Qué es GPRS, 2G, 3G, 3,5G, HSPA, 4G, LTE, etc.?* Obtenido de <https://www.tuenti.es/ayuda/que-es-gprs-2g-3g-hspa-4g-lte/>
- Tutorials Point. (2020). *Tutorials Point Concept of Hardware and Software*. Obtenido de https://www.tutorialspoint.com/computer_concepts/computer_concepts_hardware_software_concept.htm
- Tutorials Point. (2021). *Android - Sending SMS*. Obtenido de https://www.tutorialspoint.com/android/android_sending_sms.htm

- UNAM. (s.f.). *Repositorio Digital de la Facultad de Ingeniería UNAM - A7*. Obtenido de Capítulo 4 Enlaces Satelitales: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/162/7/A7.pdf>
- Verizon. (2022). *Verizon Connect ¿Qué es el GPRS?* Obtenido de <https://www.verizonconnect.com/mx/glosario/que-es-gprs/#:~:text=GPRS%20significa%20General%20Packet%20Radio,los%20usuarios%20de%20dispositivos%20m%C3%B3viles>.
- Vetrivel, J. (2022). *VetBosSel Blog*. Obtenido de Android Location Tracker Mobile Number Android Studio: <https://www.vetbossel.in/android-location-tracker-mobile-number-android-studio/>
- Villalobos, J. (2020, Noviembre 26). *Certia El laberinto de las versiones de .NET*. Obtenido de <https://www.certia.net/el-laberinto-de-las-versiones-de-net/>
- Villegas, J. (2020, Diciembre). Evolución de SQL Server en múltiples plataformas. *CompartiMOSS*, 41-42. Obtenido de <https://www.compartimoss.com/revistas/numero-46/sql-server-everywhere/>
- VIU. (2017, Febrero 10). *Sistemas GPRS, concepto general y usos principales*. Obtenido de <https://www.universidadviu.com/es/actualidad/nuestros-expertos/sistemas-gprs-concepto-general-y-usos-principales>
- VIU. (2018, Octubre 10). *¿Qué es GSM y cómo funciona?* Obtenido de <https://www.universidadviu.com/es/actualidad/nuestros-expertos/que-es-gsm-y-como-funciona>
- Wagner, B., Costa Santos, R., Sharkey, K., Coulter, D., Pine, D., Schonning, N., . . . Onderka, P. (2021, Noviembre 30). *Docs Microsoft Un recorrido por el lenguaje C#*. Obtenido de <https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>
- Wagner, B., Holvoet, S., McCallum, B., Victor, Y., Pine, D., Marusyk, R., . . . Wenzel, M. (2022, Febrero 4). *Docs Microsoft*. Obtenido de C# language versioning: <https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/configure-language-version>
- Warren, G., Coulter, D., Victor, Y., Martinez, J., & Rump, A. (2020, Enero 17). *Docs Microsoft .NET Framework versions and dependencies*. Obtenido de <https://docs.microsoft.com/en-us/dotnet/framework/migration-guide/versions-and-dependencies>
- Wells, D. (1999). *Extreme Programming The Rules of Extreme Programming*. Obtenido de <http://www.extremeprogramming.org/rules.html>
- Wells, D. (2000). *Extreme Programming Iteration*. Obtenido de <http://www.extremeprogramming.org/map/iteration.html>
- Williams, E. (2019, Septiembre 6). *Medium Android Studio Latest Version 3.5*. Obtenido de <https://medium.com/acquaint-softtech/android-studio-latest-version-3-5-explained-in-less-than-5-minutes-e80daf9eaeceb>
- Williams, E. (2019, Septiembre 10). *Quora What is the latest version of Android Studio?* Obtenido de <https://www.quora.com/What-is-the-latest-version-of-Android-Studio>
- *Xataka Basics - Cómo instalar aplicaciones en APK en un móvil Android*. (2020, Octubre 13). Obtenido de <https://www.xatakandroid.com/tutoriales/como-instalar-aplicaciones-en-apk-en-un-movil-android>
- Yogi Hosting. (2021, Junio 7). *Entity Framework – How to Create an EDMX file in Visual Studio*. Obtenido de <https://www.yogihosting.com/entity-framework-create-edmx-file/>