



UNIVERSIDAD AUTÓNOMA DE ENCARNACIÓN
FACULTAD DE CIENCIA, ARTE Y TECNOLOGÍA
LICENCIATURA EN ANALISIS DE SISTEMAS INFORMATICOS

PROYECTO DE INVESTIGACIÓN DE GRADO

**DESARROLLO DE APLICACIÓN MÓVIL PARA LA BÚSQUEDA DE PRODUCTOS
EN NEGOCIOS DE ENCARNACIÓN MEDIANTE GEOLOCALIZACIÓN**

LINEA DE INVESTIGACIÓN:
INGENIERÍA DEL SOFTWARE Y TRANSFERENCIA DE INFORMACIÓN

ESTUDIANTE:

PATRICIA LORENA FERNÁNDEZ INSAURRALDE

TUTOR:

HUGO SENDOA

Encarnación, Paraguay

Mes - 2020

INDICE

Índice

1. INTRODUCCIÓN	7
2. MARCO TEÓRICO	8
2.1. Estudio de mercado de consumidores finales en la ciudad de Encarnación.	8
2.3. Efectos del cierre de la frontera y la pandemia a comerciantes Encarnacenos	10
2.4. Evolución de los Smartphone al alcance de los usuarios.	11
2.5. Uso de la tecnología de geolocalización para desarrollar aplicaciones móviles.....	13
2.6. Sistema operativo Android como preferencia de los usuarios	15
2.7. Desarrollo de aplicaciones para promoción de productos.....	19
2.8. Uso de la plataforma de google maps para el desarrollo de la App: Encontralo!!....	21
3. PREGUNTAS, HIPOTESIS Y OBJETIVOS	24
3.1. Preguntas:	24
3.2. Hipótesis:.....	24
3.3. Objetivos:	25
4. METODOLOGÍA DE INVESTIGACIÓN	26
4.1. Tipo y diseño de investigación.....	26
4.2. Definición del objeto de estudio.....	26
4.3. Descripción de la población y muestra	26
4.3.1. Cantidad de sujetos o muestra.....	26
4.3.2. Características de la muestra.....	27
4.3.3. Tipo de muestra.....	27
4.3.4. Justificación del tipo de la muestra	27
4.4. Procedimiento del tipo de la muestra	27
4.5. Descripción de las actividades a realizarse durante cada etapa	27
4.5.1. Instrumentos y Técnicas de Recolección de Datos.....	27
4.5.2. Procedimientos de la Aplicación de Instrumento	28
4.5.3. Delimitación del alcance del software	28
4.5.4. Metodología de gestión de calidad (técnicas a utilizar para el testeo).....	29
5. RESULTADOS Y DISCUSIÓN	29
5.1. Definición de Metodología de Desarrollo	29
5.2. Selección del Entorno Tecnológicos	32
5.3. Diseño.....	37
5.3.1. Arquitectura de la App móvil	37
5.3.2. Base de datos que fue utilizada para la interfaz de administrador:.....	38
5.3.3. Diseño de interfaz de la app móvil	42
5.3.4. Diseño de la interfaz administrador	46
.....	49
5.4. Codificación de la interfaz administrador	50
5.4.2. Codificación e implementación de librerías para la interfaz administrador	53
5.5. Codificación del API.....	63
5.5.1. Configuración de recursos para el API.....	63
5.5.1.1. Instalación de la plataforma de colaboración Postman.....	63

5.5.2.	Codificación e implementación de librerías para la API	64
5.6.	Codificación de la app móvil	66
5.6.1.	Configuración de recursos para la app móvil	66
5.6.2.	Codificación e implementación de librerías para la app móvil.....	67
	Consumo de servicios del API.....	69
	Creación del Recyclerview utilizando Cardview	71
	Funcionamiento del buscador de artículos	74
	Agregando mapas al proyecto.....	77
5.7.	Prueba de calidad y resultados	82
	Resultados obtenidos en la aplicación web.....	83
	Resultados obtenidos en la aplicación móvil	89
6.	CONCLUSIÓN	91
1.	6. REFERENCIAS BIBLIOGRÁFICAS.....	93
	Carta-pedido de aprobación de proyecto	101
	Presupuesto tentativo de la investigación	102

DICTAMEN DE EVALUACIÓN DEL TRABAJO DE FIN DE GRADO

Visto lo estipulado en el Manual de Investigación de la Universidad Autónoma de Encarnación (UNAE) referente a los procesos de evaluación, tras la evaluación de la 1ª instancia (Orientador), la validación de la 2ª instancia (decanato y CIDUNAE), la evaluación de la 3ª instancia (Evaluador) y la presentación oral, se resuelve que el Trabajo de Fin de Grado presentado el___/09/2021 en Encarnación (ciudad) y titulado “DESARROLLO DE APLICACIÓN MÓVIL PARA LA BÚSQUEDA DE PRODUCTOS EN NEGOCIOS DE ENCARNACIÓN MEDIANTE GEOLOCALIZACIÓN

” realizado por Patricia Lorena Fernández Insaurrealde resulta aprobado con calificación final _____ (número y letra).

Descargo fundamentado:

This image shows a full page of white paper with horizontal dotted lines, typical of primary school writing paper. The lines are evenly spaced and run across the entire width of the page. There are no margins, text, or other markings present.

Firmas, aclaraciones y CI miembros de la mesa

1. _____ 2. _____
3. _____ 4. _____

Sello Unidad Académica

DESARROLLO DE APLICACIÓN MÓVIL PARA LA BÚSQUEDA DE PRODUCTOS EN NEGOCIOS DE ENCARNACIÓN MEDIANTE GEOLOCALIZACIÓN

Autor: Patricia Lorena Fernández
Insaurrealde

Línea de Investigación: Ingeniería del
Software y transferencia de
información.

Resumen

La ciudad de Encarnación no cuenta con una aplicación que sea capaz de geo localizar un determinado artículo que sea de interés por parte de un usuario, la mayoría de las veces se requiere recorrer varios locales en busca de un algún producto en específico lo que conlleva a una pérdida de tiempo y otros gastos como ser combustible o en el mejor de los casos se recurren a páginas de Facebook pero con la posibilidad de buscar productos de un solo local lo que si quisiéramos buscar en varios locales estaríamos perdiendo bastante tiempo. Con este trabajo se pudo desarrollar una aplicación móvil para cubrir esas necesidades y así con solo ingresar en la sección de búsqueda algún producto deseado este sea capaz de mostrarlos con sus imágenes y características. Para el desarrollo de la app móvil, se optó por una metodología híbrida entre kanban y RUP, por poseer características ajustables a nuestro proyecto y así trabajar de una manera más rápida y eficiente. Para poder desarrollar la aplicación móvil primeramente fue creada un API mediante el cual se conectó la app móvil con la app web de manera a disponer todos los artículos ingresados a esta página web con el nombre de las empresas y su ubicación. Se diseñó una interfaz de administrador en el cual fueron introducidas todos los artículos con el nombre y ubicación de la empresa, los cuales mediante el API todos estos datos fueron extraídos en formato json y luego mostrados en la app móvil. Las pruebas del correcto funcionamiento de las funcionalidades tanto de la app móvil, la app web como la API se realizó utilizando los formularios provistos por la metodología RUP, realizando pruebas constantes aplicadas durante el proceso de desarrollo de las diferentes aplicaciones.

Palabras claves: Android, Transferencia de información, Tecnología de la información, Tecnología móvil.

1. INTRODUCCIÓN

El trabajo final de grado se realizará con el fin de realizar una aplicación móvil el cual pueda realizar una búsqueda de locales que vendan productos mostrando su geolocalización y aplicado al negocio Encarnaceno.

La ciudad de Encarnación cuenta con gran número de población, es una de las ciudades más importante del país, cuenta con una fuerte economía, sus playas y costaneras son un atractivo turístico por la cual llegan cada año miles de personas a visitarla. Esta ciudad tan importante no cuenta con una aplicación móvil que sea capaz de localizar un determinado producto ya sean vestimenta, electrodomésticos, electrónica o cualquier tipo de producto y mostrar en un mapa dichos locales con sus coordenadas y la información del producto. Por esa razón se vio la necesidad de crear una aplicación móvil para la geolocalización de negocios de la ciudad de Encarnación que vendan un determinado producto. De manera que el usuario tenga la comodidad de buscar un producto desde su aplicación móvil y facilitarle las coordenadas de los locales que tienen disponibles dichos productos.

Se pretende poner en práctica la aplicación móvil destinada a la ciudad de Encarnación para luego ver los resultados. El termino **experimento** tiene al menos dos acepciones, una general y otra particular. La general se refiere a “elegir o realizar una acción” y después observar las consecuencias (Babbie, 2009). Este uso del término es bastante coloquial; así, hablamos de “experimentar” cuando mezclamos sustancias químicas y vemos la reacción provocada, o cuando nos cambiamos de peinado y observamos el efecto que suscita en nuestras amistades dicha transformación. La particularidad de esta concepción de experimento es que requiere la manipulación intencional de una acción para analizar los posibles resultados obtenidos en cualquier experimento realizado (Sampieri, 2010).

En vista a esta situación surge el siguiente planteamiento: **¿Es posible crear una aplicación móvil para la búsqueda de productos en negocios de Encarnación mediante geolocalización?**

2. MARCO TEÓRICO

2.1. Estudio de mercado de consumidores finales en la ciudad de Encarnación.

La ciudad de Encarnación es caracterizada por tener como una de sus fuentes de ingresos al comercio minorista. Los consumidores finales en su afán de obtener la mejor relación de precio-calidad, realizan estudios de mercados con las herramientas a su disposición para así poder satisfacer sus necesidades.

Una herramienta seria la búsqueda presencial, en la cual una persona para poder realizar una compra llega a un local comercial o visita una página web y solicita las características y precios de los productos. Además solicita información para poder comparar con otras marcas y detallar sus ventajas y desventajas; sin descuidar si existen ofertas o descuentos en el precio del producto que le ayude a tomar una decisión al momento de realizar la compra (González, 2018).

Los mercados son lugares donde se concentran por una parte distintos vendedores que ofrecen sus productos mediante procesos de intercambio, los cuales constituyen la oferta; por otra parte, están los compradores que buscan algún producto para satisfacer sus necesidades aportando su dinero para conseguir dicho producto, los cuales constituyen la demanda. Ambos, la oferta y la demanda son las principales fuerzas que mueven el mercado (Thompson, s.f.).

Según Pilar (2020) los consumidores finales realizan un estudio de mercado acerca del producto que van a comprar, para ello comienzan buscando información, considerando diversos materiales de lectura, pidiendo consejos a los amigos o conocidos, buscando en páginas web y asistiendo a los establecimientos para percibir directamente las marcas o productos.

Todo estudio de mercado implica la inversión de tiempo ya que para asistir a todos los locales o buscar en todas las páginas web necesitamos invertir mucho tiempo y dinero, inversión no disponible para los consumidores finales, los cuales pretenden obtener la mejor relación de precio y calidad en el menor tiempo posible. Esta situación acarrea varios inconvenientes como la inversión necesaria para recorrer todos los locales de la ciudad de Encarnación con respecto al tiempo, la disponibilidad de los productos, la disponibilidad de tiempo del consumidor final y los horarios de atención al público de los locales comerciales.

2.2. Promoción de los productos para comerciantes Encarnacenos.

González (2018), ha definido a un producto como un conjunto de características y atributos tangibles como ser tamaño, peso, color y otros; así como también con características intangibles como marca, precio, calidad, entre otros que un consumidor final obtiene para satisfacer una necesidad. De esta manera todo producto está ligado a una necesidad o deseo.

Todos aquellos que son denominados comerciantes buscan la manera de emprender nuevos proyectos y planes de negocios los cuales involucran el esparcimiento del pueblo como un lugar de turismo y recreación más que de producción y venta de cualquier tipo de producto. Entre estos comerciantes existen aquellos que generan empleos y también los que diversifican con la difusión de innovaciones comunitarias ya sea de ecoturismo, producción orgánica, artesanía, entre otros (Guillén, Flores, Varela, Lirios, & Valdez, 2014).

Los comerciantes en busca de satisfacer las necesidades de los consumidores finales para generar ganancias por la intermediación, han implementado distintas acciones con el fin de cubrirlas, estudios de marketing, generan campañas publicitarias como ser en una radio, tv, diario, lanzan promociones, entre otros (Capasso, 2020).

Para poder satisfacer las expectativas de los diferentes tipos de clientes es necesario conocer con profundidad sus necesidades de manera que se le puedan brindar aquello que esperaban recibir a cambio de su dinero y que encuentren en el mercado aquellos productos que realmente satisfagan sus necesidades. Para esto las empresas buscan nuevos valores asociados al producto de modo que sus clientes puedan sentirse satisfechos como pueden ser el precio, la distancia que son recordadas por la competencia y adaptaciones a posibles cambios en su sector (Laza, 2019).

Tradicionalmente, Encarnación siempre fue una ciudad de perfil comercial al cual con el paso del tiempo, fueron adhiriéndose otras actividades de relevancia, conforme al crecimiento y desarrollo de la ciudad como ser el turismo que mediante la construcción del nuevo circuito comercial y gracias a su atractivo de la zona, además de tener una escasa distancia al Puente Internacional “San Roque González de Santa Cruz”, a la estación del tren binacional y a la Aduana paraguaya muchas personas cruzaban el puente para realizar compras (María Rosa, Miguel Ángel, Jacqueline, & Viviana, s.f.).

El comerciante, para que los consumidores finales puedan identificar un producto que se desea comprar, debe lograr que este localice el producto y para ello debe valerse de todas las herramientas necesarias que otorguen rapidez y sencillez en el acceso a la información (CECARM, 2013). De esta manera los productos que mejor detallen sus características tendrán más oportunidades de ser considerados y los comercios que ofrezcan estas características tendrán mejores oportunidades de venta.

Los clientes de hoy exigen respuestas rápidas y las empresas que están tomando en serio su gestión en atención al cliente buscan la manera de responder lo más eficiente, optimo y en el menor tiempo posible utilizando la tecnología a su disposición de manera que puedan mantener un buen plan de atención al cliente, con los protocolos y sistemas de escalado bien definidos, pero, sobre todo, tener bien formado al equipo para que sean capaces de trabajar de forma colaborativa (Laza, 2019).

2.3. Efectos del cierre de la frontera y la pandemia a comerciantes Encarnacenos

La actual crisis sanitaria generada por la pandemia de Covid-19 que impulso el cierre de fronteras restringiendo la movilidad de personas tuvo un gran impacto negativo en la parte socioeconómica (Bravin, Collado, & Covolo, 2020).

Aunque el principal objetivo del cierre de fronteras fue evitar la circulación del virus y avance del brote, estas medidas presumen un enorme desafío regional, de manera que fueron afectadas principalmente las ciudades de frontera ya que en estas zonas el impacto de las precauciones tomadas ralentizo los intercambios fronterizos, obstaculizo el regreso de algunas personas que se encontraban al otro lado de la frontera de manera transitoria y expuso a personas migrantes a situaciones de riesgo y vulnerabilidad (Bravin, Collado, & Covolo, 2020).

Antes de la pandemia y el cierre del puente fronterizo la ciudad de Encarnación era un lugar de gran convocatoria ya que contaba con múltiples eventos como ser festivales artísticos, competencias deportivas, carnavales, recitales además de su atractivo turístico como ser la costanera de la ciudad, lugar apropiado para la recreación y disfrute frente al río (Capasso, 2020).

Según Cantero (2013) de las personas que pasaban al Paraguay a través del puente internacional San Roque +González de Santa Cruz el 89% tienen su residencia en Argentina, el 10% en Brasil y 1% en otros países. El comercio transfronterizo tenía un lugar predominante ya que los turistas que cruzaban el puente internacional lo hacían en su mayoría para comprar en Encarnación, se estimó que un 75 % lo hacía para comprar en Encarnación, el 9% para visitar familiares, el 8% para trabajar, el 4% para estudiar y el 4% por otros motivos como diversión, gastronómicos y turismo. De las personas que venían a comprar algunos productos se pudo seleccionar que el 35% compraba productos electrónicos, 30% prendas de vestir, 20% utilitarios para automóviles, 10% juguetes, 4% telas y 1% novedades.

Con el cierre de las fronteras el sector comercial Encarnaceno fue la más afectada ya que gracias al paso fronterizo entre Encarnación-Posadas, argentinos y otros turistas tenían gran afluencia en ese sector y con el cierre de la frontera muchos locales comerciales sufrieron una escasez de ventas realizadas. Teniendo que reinventarse con otras opciones y buscar soluciones que estén a su alcance.

2.4. Evolución de los Smartphone al alcance de los usuarios.

Un Smartphone es un dispositivo electrónico que cuenta con funciones similares a un computador personal y que además de poder realizar transferencia de voz como otros celulares comunes los Smartphone han evolucionado y traen cierta cantidad de funciones que atraen la atención de los usuarios como, por ejemplo: navegar por Internet, reproducir datos multimedia, realizar transacciones bancarias, descargar juegos, conectarnos a las redes sociales, ver televisión vía streaming y muchas cosas más (Polanco & Taibo, 2011).

Aunque no existe una interpretación única de cuando se inició el ciclo de evolución de la tecnología, si existen hitos que se usan para explicar e identificar la posición que ocupan los Smartphone dentro de esta como se puede apreciar en la Ilustración 1 (Aguilar, 2018).

Ilustración 1: Línea de tiempo de la evolución de la computación personal y portátil



Fuente: (Aguilar, 2018).

Los primeros modelos se usaban como teléfonos de empresa y sus precios eran inaccesibles para la mayoría de los consumidores. Pero gracias al enorme éxito del iPhone, se ha visto la posibilidad de que las operadoras puedan dar una ayuda de compra a sus clientes de manera a tener una permanencia de los últimos modelos de Smartphone (Perenyi, 2013).

Los PC (computadoras personales) fueron sustituidos por el Smartphone como dispositivo principal de acceso a Internet, los Smartphone se convirtieron en una categoría nicho, mientras que las PC se convirtió en el principal receptor y catalizador de los avances en factores básicos como la velocidad de procesamiento, resolución de pantallas, ergonomía, duración de las baterías y desarrollo de sistemas operativos. Los servicios más populares que presta la red actualmente como ser redes sociales, buscadores, mensajería instantánea y el comercio electrónico son accedidos mediante el uso de Smartphone (Aguilar, 2018).

Según el Ministerio de Tecnologías de la Información y Comunicación (2019) de las personas que compran en internet ya sea a través de un ordenador o Smartphone se estima que el 65 por ciento compran ropas, cazados, artículos deportivos; el 31 por ciento artículos informáticos o electrónicos; el 22 por ciento artículos para el hogar y el 5 por ciento otros productos.

A nivel mundial el número de teléfonos móviles supera los 3.000 millones y se estima que siga creciendo de forma gradual durante los próximos años. China, la India y Estados Unidos son los tres países con mayor cifra de usuarios de Smartphone del planeta, superando todos ellos la barrera de los 100 millones (Fernández, 2020).

Tal como lo indica el Instituto Nacional de Estadística (2019) en Paraguay en el año 2018 el 96,7% de los hogares disponían de teléfono móvil. El 9,3 tenían teléfono fijo para poder comunicarse desde su hogar. El acceso al teléfono móvil presento una variación positiva respecto al año 2015, caso contrario ocurrió con el teléfono fijo.

El aumento de los Smartphone ha generado un mayor acceso a internet vía celular en forma diaria. El uso de los teléfonos celulares se ha vuelto indispensable en la vida cotidiana de los paraguayos de todas las edades, mediante el fácil acceso a internet para el uso de redes sociales como ser WhatsApp, Facebook, Instagram, YouTube y Twitter que son las más conocidas y utilizadas por los usuarios (Ministerio de Tecnologías de la Información y Comunicación, 2019).

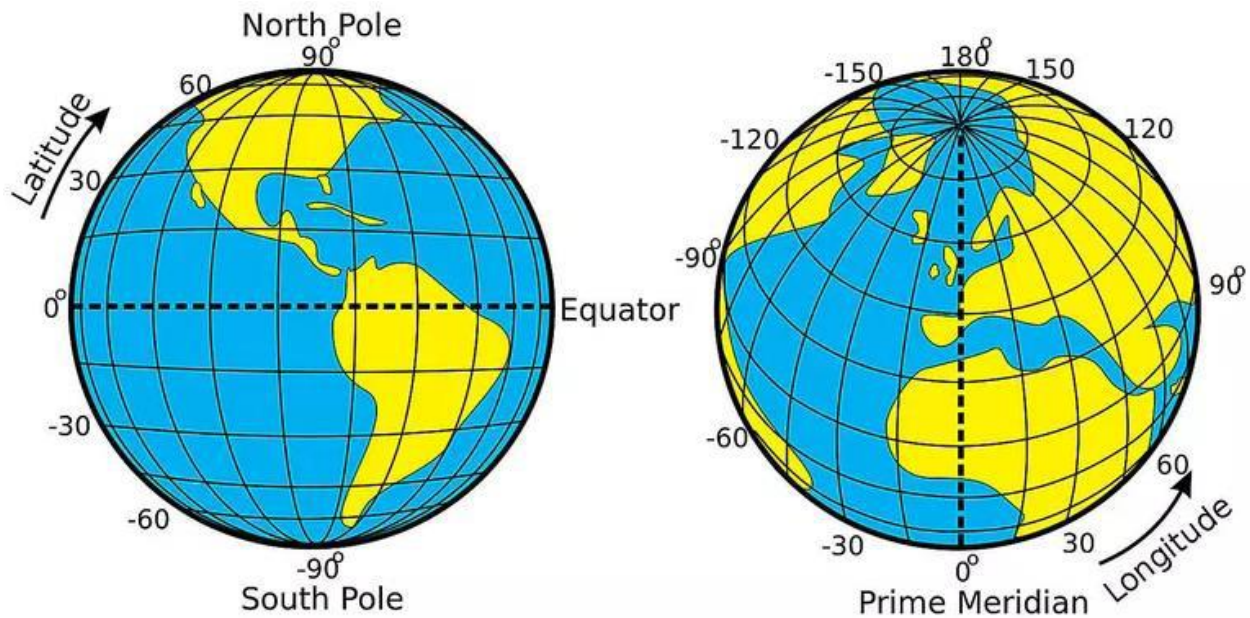
Con el paso de los años los teléfonos inteligentes conocidos como Smartphone han evolucionado de manera significativa y casi todas las personas tienen acceso al uso de estos dispositivos móviles para realizar diferentes tareas, de tal manera que se ha vuelto indispensable en la vida cotidiana de los usuarios.

2.5. Uso de la tecnología de geolocalización para desarrollar aplicaciones móviles

Para poder desarrollar aplicaciones móviles utilizamos herramientas propias de las aplicaciones móviles como la geolocalización, que es caracterizada por el desarrollo del acceso universal a coordenadas y localización geográfica en cualquier lugar y en cualquier momento. Mediante la geolocalización podemos obtener la posición de una persona, vehículo, equipamiento o algún objeto; principalmente durante un proceso de movilidad, viaje o desplazamiento. La geolocalización se basa en realizar mediciones del tiempo de propagación de las señales entre un transmisor y un receptor para obtener la distancia que los separa (Alcaraz, 2017).

Las coordenadas geográficas son aquellas que indican la posición de un punto en la superficie terrestre tomando como referencias la latitud y la longitud. La latitud es el arco contado desde su línea base el Ecuador al punto donde se encuentra el observador. Por otra parte, la longitud es la distancia que existe entre un punto cualquiera y el Meridiano de Greenwich, medida sobre el paralelo que pasa por dicho punto como se puede apreciar en la Ilustración 2 (Ibañez, Gisbert, & Moreno, s.f.).

Ilustración 2: Coordenadas geográficas

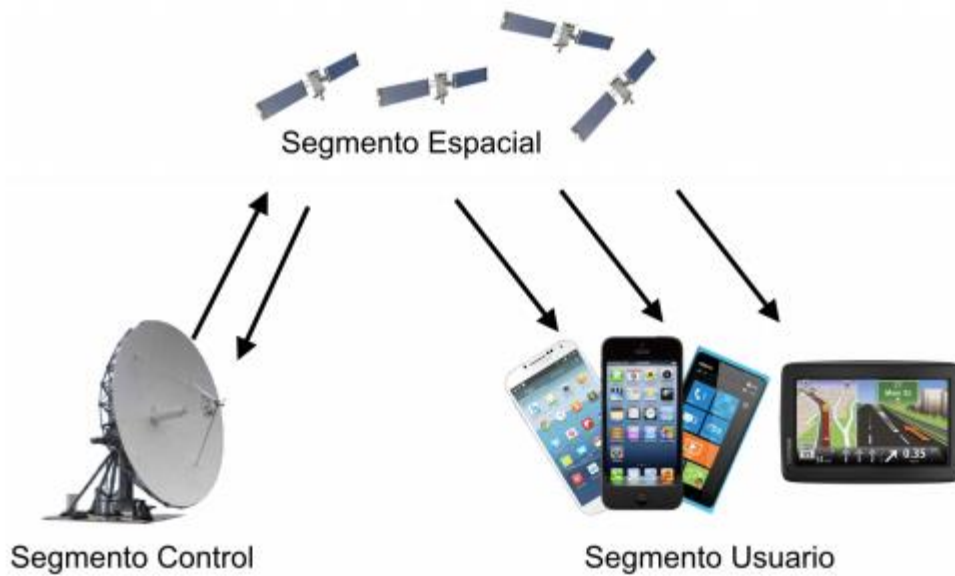


Fuente: (Aragón, 2018)

El Sistema de Posicionamiento Global (GPS) es un sistema basado en localización satelital que proporciona una posición global detallada que se representa por dos valores que son la latitud y longitud, la cual se obtiene calculando el tiempo de respuesta obtenido como resultado de la conexión a por lo menos 4 satélites para poder obtener un cálculo correcto. La latitud es la representación en grados de la posición norte-sur con respecto al punto en la superficie en la tierra, y la Longitud representa la posición este-oeste de un punto en la superficie de la tierra en grados (Padilla Roberto, 2015).

El GPS se compone de tres elementos por una parte se encuentra el segmento espacial, que se componen por los satélites que orbitan alrededor de la tierra. Por otra, el segmento de control que se compone por las estaciones terrestres, que se encargan de controlar y monitorizar la constelación de satélites de la tierra. Por último, el segmento de usuario el cual está constituido por los receptores de GPS que detectan y reciben la señal de los satélites para determinar la posición como se puede apreciar en la Ilustración 3 (Alcaraz, 2017).

Ilustración 3: Conexión GPS



Fuente: (Alcaraz, 2017)

El desarrollo de toda aplicación móvil implica poder satisfacer todas las necesidades del usuario y mediante el uso de las tecnologías ya incluidas por los Smartphone de hoy en día facilita el desarrollo de cualquier aplicación como puede ser el sistema de GPS, ya que gracias a este es más fácil obtener las coordenadas de posición de un usuario y así poder brindarles una aplicación con el cual pueda obtener un mayor beneficio.

2.6. Sistema operativo Android como preferencia de los usuarios

Lo que hace atractivo a un Smartphone es su sistema operativo ya que gracias a este se logra junto con el hardware del dispositivo desarrollar diferentes aplicaciones llamativas e importantes (Polanco y Taibo, 2011).

Un sistema operativo para dispositivos móviles es considerado el programa principal ya que este es capaz de administrar todos sus recursos de hardware y software de manera eficiente, para que el usuario pueda mantener una buena comunicación sin problemas (Polanco y Taibo, 2011).

En la actualidad existen 7 grandes compañías y sus sistemas operativos que abarcan más del 99% de los dispositivos. Estos 7 sistemas operativos son: Symbian, Research In Motion (RIM), iPhone OS, Microsoft Windows Mobile, Linux, Android y WebOs (Perenyi, 2013).

1. Symbian: Es un producto que utilizan varias compañías entre las cuales se encuentran Nokia, Sony Ericsson, Panasonic, Siemens y Samsung. Symbian es muy susceptible a un ataque de virus, considerado el más popular y destructivo de la historia, el famoso caballo de Troya, el virus utilizó exclusivamente la tecnología Bluetooth para propagarse entre los teléfonos celulares. En la actualidad, este sistema operativo está en baja expresión, ya que el mercado internacional es el más buscado por los dos grandes gigantes de la tecnología móvil, Apple y Samsung, que representan los sistemas operativos iOS y Android. Siendo entre ellos los más seguros (Quissanga, 2019).
2. Research In Motion: es una compañía de comunicación telegráfica y aparatos inalámbricos canadiense, también conocida por ser la desarrolladora de la BlackBerry. Su sistema operativo (BlackBerry OS) es propietario y proporciona multitarea. Soporta dispositivos de entrada como pantalla táctil o trackball (Perenyi, 2013).
3. iPhone OS: es el sistema operativo para dispositivos móviles de Apple, fue lanzado el 29 de junio de 2007, fue uno de los principales eventos de la historia de la telefonía móvil ya que miles de personas serían las primeras en comprar en las tiendas de Apple antes de su lanzamiento (Quissanga, 2019). Fue desarrollada originariamente para el iPhone, pero en la actualidad se ha ampliado para poder ser usado en otros dispositivos de Apple como iPod touch, iPad y Apple TV (Perenyi, 2013)
4. Microsoft Windows Mobile: es un sistema operativo desarrollado por Microsoft utilizado tanto en Smartphone como en dispositivos móviles. La última versión (Windows Phone 7) establece los requisitos mínimos para pantallas táctiles de alta resolución (Perenyi, 2013). El sistema operativo Windows Phone fue evolucionado de Windows Mobile que se distribuye en teléfonos móviles de las principales compañías del mundo; la única excepción es Nokia, que utiliza Symbian en sus teléfonos. En la actualidad encontramos el Windows Mobile 10 que son propiedad de Microsoft y están diseñados para ofrecer el mejor rendimiento posible para Smartphone y Tablets (Quissanga, 2019).
5. Linux: LiMo foundation es un consorcio de la industria cuya finalidad es la de crear un sistema operativo para dispositivos móviles, basado en Linux. Participan en el proyecto

empresas como NEC, Panasonic, Vodafone, Telefónica, Orange, Mozilla, Huawei o McAfee, entre otras. Las tecnologías clave de LiMo incluyen una 18 potente y flexible interfaz de usuario, librerías extendidas para widgets, efectos 3D para widgets, multimedia avanzado, redes sociales y localización.

6. WebOS: es un sistema operativo multitarea basado en Linux, diseñado para dispositivos de pantalla táctil. Fue desarrollado por Palm Inc, pero actualmente es propiedad de Hewlett-Packard Company (Perenyi, 2013).
7. Android: es un sistema operativo para dispositivos móviles de código abierto basado en el núcleo de Linux, lo cual permite desarrollar diferentes aplicaciones de manera ilimitada para teléfonos inteligentes; como ser, tabletas, televisores, cámaras, y otros. El SDK (kit de desarrollo de software) Android proporciona distintas herramientas necesarias como ser correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y Apis (interfaz de programación de aplicaciones) para poder empezar a desarrollar aplicaciones en la plataforma de Android utilizando el lenguaje de programación Java o Kotlin (Vanegas, 2012).

Según (Diaz, 2019) en el año 2019 cerca del 75% del mercado está dominado por dispositivos que utilizan el sistema operativo Android, un 23% de dispositivos iOS y un 2% para otros S.O. testimoniales.

Java es un lenguaje de programación lanzado por primera vez por Sun Microsystems en 1995, se puede encontrar en diferentes tipos de dispositivos desde teléfonos inteligentes hasta computadoras centrales. Java no se compila en el código del procesador nativo, sino que se basa en una "máquina virtual" que comprende un formato intermedio llamado código de bytes de Java, lo que significa que podemos desarrollar aplicaciones de Android en Windows, Linux u OS X y el compilador de Java convierte el código fuente en código de bytes (Bose, Kundu, & Mukherjee, 2018).

Bose, Kundu, & Mukherjee (2018), mencionan que el termino Kotlin (Android_and_Kotlin) es un lenguaje de programación de tipo estático que se ejecuta en la máquina virtual Java y también se puede compilar en código fuente JavaScript. Fue lanzado al público en febrero de 2016. Su

desarrollo principal es de un equipo de programadores JetBrains con sede en San Petersburgo, Rusia (el nombre proviene de la isla Kotlin, cerca de San Petersburgo).

En la Tabla 1 podemos observar las ventajas que tiene cada uno de estos dos tipos de lenguaje de programación:

Tabla 1. Ventajas de los lenguajes de programación Java y Kotlin

Java	Kotlin
<ul style="list-style-type: none">• Es el lenguaje actualmente más utilizado en el mundo y existen suficientes profesionales que conocen en este lenguaje y esto lleva implícito ciertas ventajas con respecto al lenguaje Kotlin.• En relación directa con lo anterior, la comunidad de Java posee más herramientas y documentación. Es decir, las fuentes de calidad para aprender Kotlin escasean en comparación con Java. Esto ha sido exhaustivamente comprobado realizando búsquedas en internet y preguntando a profesionales de este campo.• Hay que tener en cuenta que Kotlin no es Java, todavía existen problemas que Kotlin no es capaz de solucionar y sí Java.	<ul style="list-style-type: none">• Es Más claro y compacto, se puede hacer más trabajo en menos tiempo.• Los errores se pueden detectar en tiempo de compilación y arreglarlos antes de ejecutar la aplicación y esperar a que falle.• Es completamente interoperable con Java, puede convivir en una misma aplicación escrita en Java, lo que te puede permitir migrar tu programa poco a poco a Kotlin.• Tiene un mantenimiento sencillo y puede usarse en diversos entornos de desarrollo, permitiendo a cada programador que utilice su preferido.

Fuente: Elaboración propia con datos extraídos de Aragón (2018).

El sistema operativo Android es de código abierto y es el más utilizado por los usuarios en sus teléfonos inteligentes, además posee múltiples herramientas que facilitan el desarrollo de aplicaciones móviles y su lenguaje de programación más popular, con más soporte, comunidad, documentación y tutoriales es Java ya que existen suficientes profesionales utilizando este lenguaje como su preferido.

2.7. Desarrollo de aplicaciones para promoción de productos

En Paraguay existen varias aplicaciones que realizan estrategias de promoción de productos como puede ser:

- Monchis: es una aplicación que permite dar soluciones rápidas y efectivas a los usuarios para buscar y realizar pedidos de sus platos favoritos de una manera fácil y amigable. Cuenta con una página web súper sencilla e intuitiva, donde se pueden realizar todos los cambios de menú, precios, horarios, entre otras cosas más (Monchis, 2021).
- Turoga: es una aplicación muy útil que te ayuda a encontrar un inmueble por ciudades o barrios en todo Paraguay, ya sea para alquilar o comprar una casa, departamento, dúplex o un terreno. Esta aplicación esta disponible para las plataformas de android e iOS(Chena, 2021).

También existen aplicaciones internacionales que realizan promociones de productos como, por ejemplo:

- PedidosYa: es una aplicación móvil que te permite elegir tu comida favorita de más de 500 restaurantes con delivery online de forma fácil y rápida. Además, Te ofrece la opción de pagar tu pedido de manera online o cuando recibís tu pedido en la puerta de tu casa. Disponible en varios países en las plataformas iPhone y Android (PedidosYa, 2021).
- Alibaba.com: es uno de los mercados mayoristas más grandes del todo el mundo, tiene 20 años de experiencia ayudando a empresas de empresa a empresa (B2B) a poder vender y comprar sus productos en todo el mundo. Todo lo que necesitas lo encuentras en un solo lugar desde productos listos para enviar o personalizar hasta garantizar la calidad y enviar su pedido a su destino final (Alibaba.com, 2020).
- Amazon Movile: su misión está en centrarse en el cliente en vez de enfocarse hacia la competencia, pasión por la invención, compromiso con la excelencia operativa y pensamiento

a largo plazo. Hoy en día, un usuario puede encontrar cualquier cosa que busque en Amazon y comprarlos con un solo clic, además garantiza la entrega del producto deseado hasta la puerta de su casa. (Amazon, 2021).

Todas estas apps promocionan sus productos mediante el uso de las Tecnologías de la Información y Comunicación (TIC) que son todos aquellos elementos que se utilizan en la emisión, procesamiento y recepción de datos e información (Iñaki, 2014). Existen múltiples equipos tecnológicos que encuadran dentro del concepto de TIC, como la televisión, el teléfono, el video, el ordenador, y otros. Los medios más utilizados por la sociedad actual son los ordenadores y Smartphone que nos permiten utilizar diferentes aplicaciones informáticas y más específicamente redes de comunicación como la internet (Belloch, s.f.).

Una aplicación informática se compone de uno o varios programas informáticos que tienen por objetivo realizar una determinada operación o funciones de forma automática mediante el uso de un sistema informático. Existen varias aplicaciones hechas a medida que ofrecen un gran impacto para solucionar un problema en específico que logra facilitar ciertas tareas complejas y hacen más sencilla la experiencia del usuario (Pin, 2020).

Con el avance de la tecnología, es común que ahora escuches el término aplicaciones móviles. Estas hacen referencia a las aplicaciones diseñadas para dispositivos como el celular o la tableta. A las aplicaciones de tu computador se les conoce como aplicaciones de escritorio (Foundation, CGFGlobal, 2021).

Una aplicación móvil, también llamada app móvil, es un tipo de aplicación diseñada para ejecutarse en un dispositivo móvil, incluso si las aplicaciones suelen ser pequeñas unidades de software con funciones limitadas, se las arreglan para proporcionar a los usuarios servicios y experiencias de calidad. A diferencia de las aplicaciones diseñadas para computadoras de escritorio, las aplicaciones móviles se alejan de los sistemas de software integrados. En cambio, cada aplicación móvil proporciona una funcionalidad aislada y limitada. Por ejemplo, puede ser un juego, una calculadora o un navegador web móvil (Herazo, 2021).

Una aplicación Android puede ser diseñada a través de varios mecanismos de desarrollo como aplicaciones nativas o híbridas. Por un lado, tenemos las aplicaciones híbridas se programan

con HTML y son generadas por algún Framework de desarrollo, su ventaja es más para los desarrolladores que para los usuarios ya que se escriben en código estándar de menor complejidad y además es multiplataforma (Garita, 2013). Por el otro lado tenemos las aplicaciones nativas que están diseñadas para un único sistema operativo móvil, esto quiere decir que no se puede instalar ni usar una aplicación de Android en iPhone y viceversa, su principal beneficio es su alto rendimiento y excelente experiencia con el usuario, además de tener acceso a una gama de API que ayuda acelerar el trabajo de desarrollo y ampliar los límites del uso de la aplicación (Herazo, 2021).

Para desarrollar una aplicación móvil buscaremos satisfacer todas las necesidades del usuario y realizaremos una aplicación que pueda ser sencillo y amigable. Usaremos tecnologías propias de los móviles con el fin de usar su máximo rendimiento y tener una mejor experiencia con el usuario.

2.8. Uso de la plataforma de google maps para el desarrollo de la App: Encontralo!!

Google Maps Platform es un conjunto de API y SDK que los desarrolladores pueden usar a fin de incorporar Google Maps en apps para dispositivos móviles y páginas web o recuperar datos de Google Maps (Google Maps Platform, 2021).

Se ofrecen diferentes productos. Según tus necesidades, podrías usar una sola API o SDK o una combinación de las diferentes opciones:

Maps:

- API de Maps JavaScript
- SDK de Maps para Android
- SDK de Maps para iOS
- API de Maps Static
- API de Street View Static
- URL de Maps

- API de Maps Embed

Routes:

- API de Directions
- API de Distance Matrix
- API de Roads

Places:

- API de Places
- SDK de Places para Android
- SDK de Places para iOS
- API de Places Library, Maps JavaScript
- API de Geocoding
- API de Geolocation
- API de Time Zone

Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones (Hat, 2021).

Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones, y proporcionan oportunidades de innovación, lo cual es ideal al momento de diseñar herramientas y productos nuevos (o de gestionar los actuales) (Hat, 2021).

Debido a que simplifican la forma en que los desarrolladores integran los elementos de las aplicaciones nuevas en una arquitectura actual, las API permiten la colaboración entre el equipo comercial y el de TI. Las necesidades comerciales suelen cambiar rápidamente en respuesta a los mercados digitales en constante cambio, donde la competencia puede modificar un sector entero con una aplicación nueva. Para seguir siendo competitivos, es importante admitir la implementación y el desarrollo rápidos de servicios innovadores. El desarrollo de aplicaciones nativas de la nube es una forma identificable de aumentar la velocidad de desarrollo y se basa en la conexión de una arquitectura de aplicaciones de micro servicios a través de las API (Hat, 2021).

Las API son un medio simplificado para conectar su propia infraestructura a través del desarrollo de aplicaciones nativas de la nube, pero también le permiten compartir sus datos con clientes y otros usuarios externos. Las API públicas representan un valor comercial único porque simplifican y amplían la forma en que se conecta con sus partners y, además, pueden rentabilizar sus datos (un ejemplo conocido es la API de Google Maps) (Hat, 2021).

En Encarnación no existe una aplicación con estas características de fácil acceso y que este destinado a la compra de vestimentas, equipos electrónicos o artículos para el hogar. Atendiendo todas las necesidades del usuario se pretende realizar una aplicación móvil que se encargue de la búsqueda de productos en Encarnación mediante geolocalización. Esta aplicación debe de ser sencillo y rápido de tal manera que sea agradable al usuario.

3. PREGUNTAS, HIPOTESIS Y OBJETIVOS

3.1. Preguntas:

- ¿Cuáles son los mecanismos de marketing aplicado a los negocios de Encarnación?
- ¿Por qué desarrollar aplicación móvil para la búsqueda de productos en negocios de encarnación mediante geolocalización?
- ¿Qué metodología se adaptaría mejor para el desarrollo de la aplicación?
- ¿Cuáles son los resultados del desarrollo de la aplicación móvil?
- ¿Cuál es la aceptación de los usuarios de la ciudad de Encarnación?

3.2. Hipótesis:

Los mecanismos de marketing aplicado a los negocios de Encarnación se basan en promocionar sus productos por medio de las redes sociales, radios, tv, diario y otros dependiendo de sus posibilidades con el fin de obtener mayor cantidad de posibilidad de vender un determinado producto.

La ciudad de Encarnación no cuenta con una aplicación móvil que sea capaz de geo localizar empresas que cuenten con un determinado producto. Con el desarrollo de la aplicación móvil se pretende brindar a los negocios una estrategia de marketing para promocionar sus productos disponibles para que la ciudadanía desde la comodidad de su casa busque los productos que desee comprar y se lo pueda mostrar en un mapa los distintos locales con su información.

La metodología de desarrollo ágil sería la mejor opción para el desarrollo del software ya que este nos permite organizar el trabajo de una manera más rápida y eficiente, además se adapta mejor a los requerimientos propuestos para la elaboración de la aplicación móvil. Por otra parte, establece menor pérdida de tiempo ya que no es necesario realizar una extensa documentación al momento de plantear el desarrollo del software.

Mediante la aplicación móvil un usuario podrá saber con exactitud cuáles son los locales más conocidos que cuentan con dicho producto y se ahorra tiempo de recorrer la ciudad de Encarnación en búsqueda de locales. También será de utilidad para los turistas que no conocen bien la ciudad ya que con esta aplicación se les facilitara la búsqueda de locales cuando quieran comprar un producto.

El nivel de aceptación por parte de los usuarios de la aplicación móvil será muy bueno ya que brindará al usuario una manera eficaz de búsqueda de productos además de poder conocer cuáles son los distintos locales que ofrecen un determinado producto porque muchas veces solo conocemos unos pocos locales de venta sin embargo también existen otros.

3.3. Objetivos:

- Desarrollar una aplicación móvil para la búsqueda de productos en negocios de Encarnación mediante geolocalización.
- Recabar información sobre los mecanismos de promoción de negocios Encarnacenos.
- Determinar los requerimientos de la aplicación móvil para la geolocalización de negocios de la ciudad de encarnación.
- Diseñar una interfaz de administrador.
- Diseñar un web service para disponer los datos.
- Desarrollar la aplicación móvil.
- Obtener un margen de aceptación de usuarios.

4. METODOLOGÍA DE INVESTIGACIÓN

4.1. Tipo y diseño de investigación

El tipo de investigación es aplicada ya que se estarán aplicando una solución para la ciudad de Encarnación poniendo en práctica la aplicación móvil. La investigación aplicada es caracterizada por su interés en la aplicación, utilización y consecuencias prácticas de los conocimientos en cualquier investigación. La investigación aplicada busca el conocer para hacer, para actuar, para construir, para modificar. (Grajales., s.f.).

Se utilizará la investigación aplicada ya que este tipo de investigación está orientada a conocer las necesidades que no están siendo satisfechas. La misma hace uso de los conocimientos o teorías o de investigación básica para resolver un problema existente como así también implementar la solución. (Metodología de la investigación, 1991)

El diseño de la investigación es cuasi experimental cuantitativo ya que se pretende poner en práctica la aplicación móvil destinada a la ciudad de Encarnación para luego ver los resultados. El termino **experimento** tiene al menos dos acepciones, una general y otra particular. La general se refiere a “elegir o realizar una acción” y después observar las consecuencias (Babbie, 2009). Este uso del término es bastante coloquial; así, hablamos de “experimentar” cuando mezclamos sustancias químicas y vemos la reacción provocada, o cuando nos cambiamos de peinado y observamos el efecto que suscita en nuestras amistades dicha transformación. La esencia de esta concepción de experimento es que requiere la manipulación intencional de una acción para analizar sus posibles resultados (Sampieri, 2010).

4.2. Definición del objeto de estudio

Negocios de Encarnación que requieran de una aplicación móvil para la búsqueda de productos mediante geolocalización y mecanismos de promoción de los negocios de Encarnación.

4.3. Descripción de la población y muestra

La población a estudiar sería todos los negocios del circuito comercial de la ciudad de Encarnación que requieran una aplicación móvil para promocionar sus productos.

4.3.1. Cantidad de sujetos o muestra

De todos los negocios del circuito comercial de la ciudad de Encarnación, se realizará una prueba piloto con una determinada cantidad de negocios que deben reunir ciertas características.

4.3.2. Características de la muestra

Los sujetos de investigación serían las empresas o negocios que se dediquen a la venta de ropas, bazar, juguetes, muebles, equipos de informática o electrodomésticos que quieran adherirse a nuestra aplicación y autoricen la aplicación de los instrumentos de recolección de datos, además que sean escogidas de forma aleatoria.

4.3.3. Tipo de muestra

El tipo de muestra que implementaremos en nuestro caso es el aleatorio simple. El muestreo aleatorio quiere decir que todos los elementos que conformaran la muestra tienen la misma probabilidad de ser elegidos, a su vez existen diferentes tipos de métodos para realizar muestreos aleatorios, como pueden ser el muestreo simple, el estratificado y el muestreo por conglomerados. En el muestreo simple tenemos un conjunto de individuos de los cuales se extrae al azar algunos de ellos para obtener dicha muestra (Casal & Mateu, 2003).

4.3.4. Justificación del tipo de la muestra

La muestra corresponde a un tipo de muestreo probabilístico simple, ya que todos los encuestados serán los que reúnan ciertas características y, además serán elegidos de forma aleatoria al azar, extrayendo las empresas que salen sorteadas.

4.4. Procedimiento del tipo de la muestra

Se realizarán entrevistas a las empresas que contarán con el servicio de la aplicación móvil. Se le solicitará a la empresa una lista de productos con sus precios para poder cargar a la aplicación como primera parte del desarrollo. Una vez que funciona con datos estándares se proseguirá a utilizar el web service para traer los datos de las empresas que estas cargaran en un administrador.

4.5. Descripción de las actividades a realizarse durante cada etapa

4.5.1. Instrumentos y Técnicas de Recolección de Datos

Para la recolección de información los instrumentos utilizados consisten en seleccionar y recopilar información por medio de consultas críticas de documentos y materiales bibliográficos de bibliotecas, libros, revistas, documentos electrónicos, páginas web y otros medios digitales.

4.5.2. Procedimientos de la Aplicación de Instrumento

Se procedió a la búsqueda de información de los requerimientos para el desarrollo de aplicaciones móviles. Se elaboró un cronograma con un listado de actividades principales para poder iniciar el desarrollo de código de la aplicación móvil.

4.5.3. Delimitación del alcance del software

Alcances:

- Se podrá utilizar la aplicación en teléfonos móviles.
- Se podrá buscar locales que vendan un determinado producto ya sea por nombre del local o del producto.
- Se mostrará en un mapa de google la ubicación de los locales más cercanos al usuario según búsqueda en un radio de 1 km.
- Se mostrará información de los locales comerciales: nombre, dirección, teléfono, email.
- Los locales comerciales podrán registrarse a un administrador web para cargar sus productos.
- La aplicación se conectará mediante un web service para obtener información de los locales.
- Se podrá subir la aplicación a la App Store.

Limitaciones:

- No tendrá más características de los ya mencionado anteriormente.
- Solo la aplicación de administradores tendrá registro de usuario no así la aplicación móvil.
- La app móvil será únicamente de consulta.
- Será aplicado únicamente al casco urbano de Encarnación.
- El usuario de la aplicación móvil no podrá interactuar para brindar su recomendación, Rankin o algún otro tipo de puntajes.

4.5.4. Metodología de gestión de calidad (técnicas a utilizar para el testeo)

Se realizarán pruebas unitarias y de caja negra.

Las pruebas unitarias de software permiten evaluar por separado el correcto funcionamiento de los códigos que lo componen, lo que buscan es responder si las funciones que componen un código cumplen o no con el comportamiento esperado evaluando el código escrito para poder dar un análisis de los resultados obtenidos en dicha prueba (Villa & Giraldo, 2012).

Las pruebas de caja negra (Black-Box Testing) se centran en el correcto funcionamiento del software a partir del análisis de entradas y salidas que posee verificando que el resultado es el esperado (Villa & Giraldo, 2012).

5. RESULTADOS Y DISCUSIÓN

En este capítulo pretendemos detallar todos los pasos que se siguieron para poder desarrollar la aplicación móvil para la búsqueda de productos en negocios de Encarnación mediante geolocalización; así también el api y la interfaz de administrador.

Se desarrollará una aplicación móvil que sea capaz de geo localizar empresas dentro de la ciudad de Encarnación y mostrarlos en un mapa mediante el uso de google Maps. Primeramente, obtendremos la información de la empresa y su ubicación mediante la interfaz de administrador estos datos serán alojados en una base de datos MySql para luego ser obtenidos mediante un api que los convertirá en formato Json para poder ser consultados a través de nuestra aplicación móvil y así mostrarlos al usuario en un mapa de google Maps.

La aplicación móvil tendrá un campo con un botón buscador con la cual el usuario podrá insertar el nombre de cualquier producto que desea buscar y este será capaz de mostrar dicho producto con sus características e imágenes y luego si el usuario lo desea podrá ver la ubicación de la empresa a través de un mapa de google.

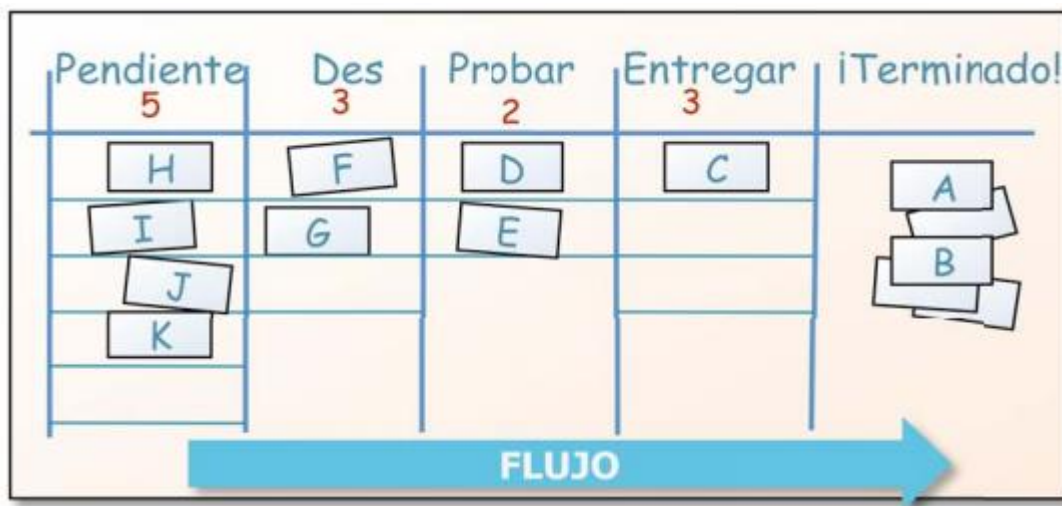
5.1. Definición de Metodología de Desarrollo

Para el desarrollo de la aplicación móvil se utilizará una metodología de desarrollo de software hibrido entre la metodología de desarrollo kanban y la metodología de desarrollo RUP de las cuales se utilizará parte de la documentación de cada una de ellas.

Mediante la metodología kanban se creará un tablero con las distintas tareas para poder dar seguimiento al proyecto.

Kanban es capaz de visualizar el flujo de trabajo o dividir el trabajo en bloques, escribe cada elemento en una tarjeta y ponlo en el muro, también utiliza columnas con nombre para ilustrar dónde está cada elemento en el flujo de trabajo como se puede observar en la Ilustración 4 (Kniberg & Skarin, 2010).

Ilustración 4: Tablero Kanban



Fuente: (Kniberg & Skarin, 2010)

Otras de sus características son que limita el WIP (Work in Progress, trabajo en curso) - asigna límites concretos a cuántos elementos pueden estar en progreso en cada estado del flujo de trabajo; además mide el lead time (tiempo medio para completar un elemento, a veces llamado "tiempo de ciclo"), optimiza el proceso para que el lead time sea tan pequeño y predecible como sea posible (Kniberg & Skarin, 2010).

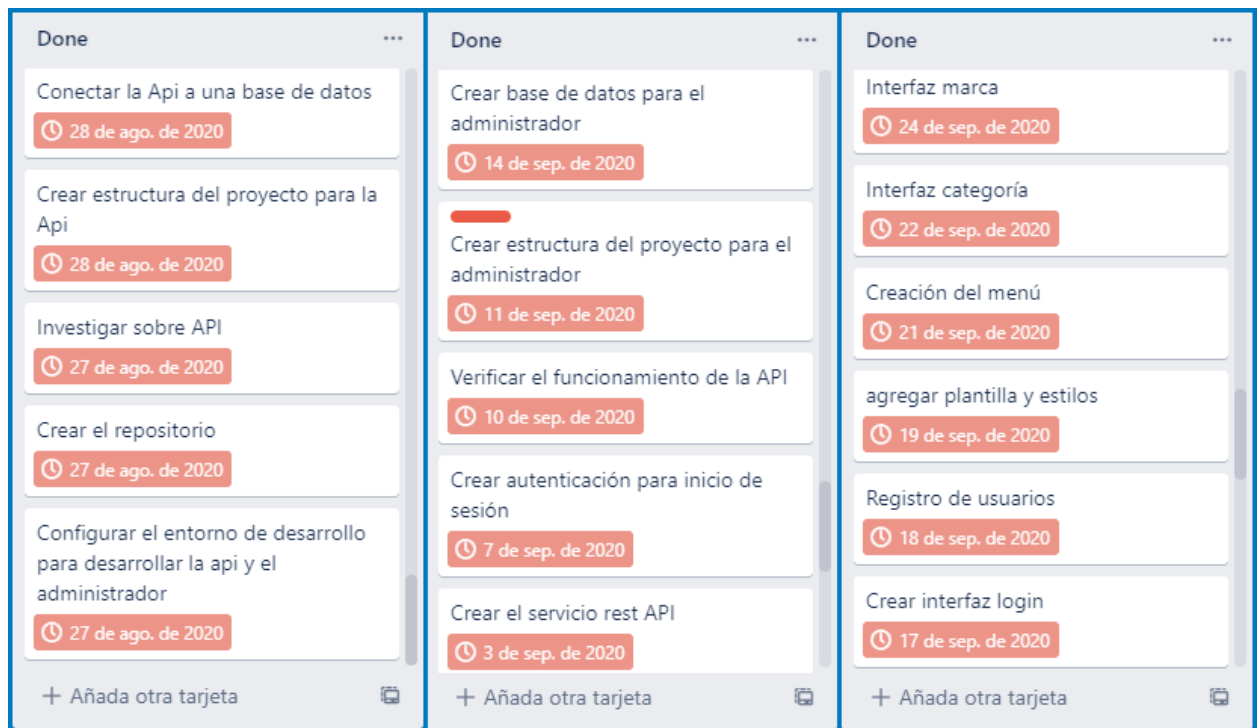
Mediante la metodología RUP se podrán diseñar el plan de configuración y el plan de test del sistema.

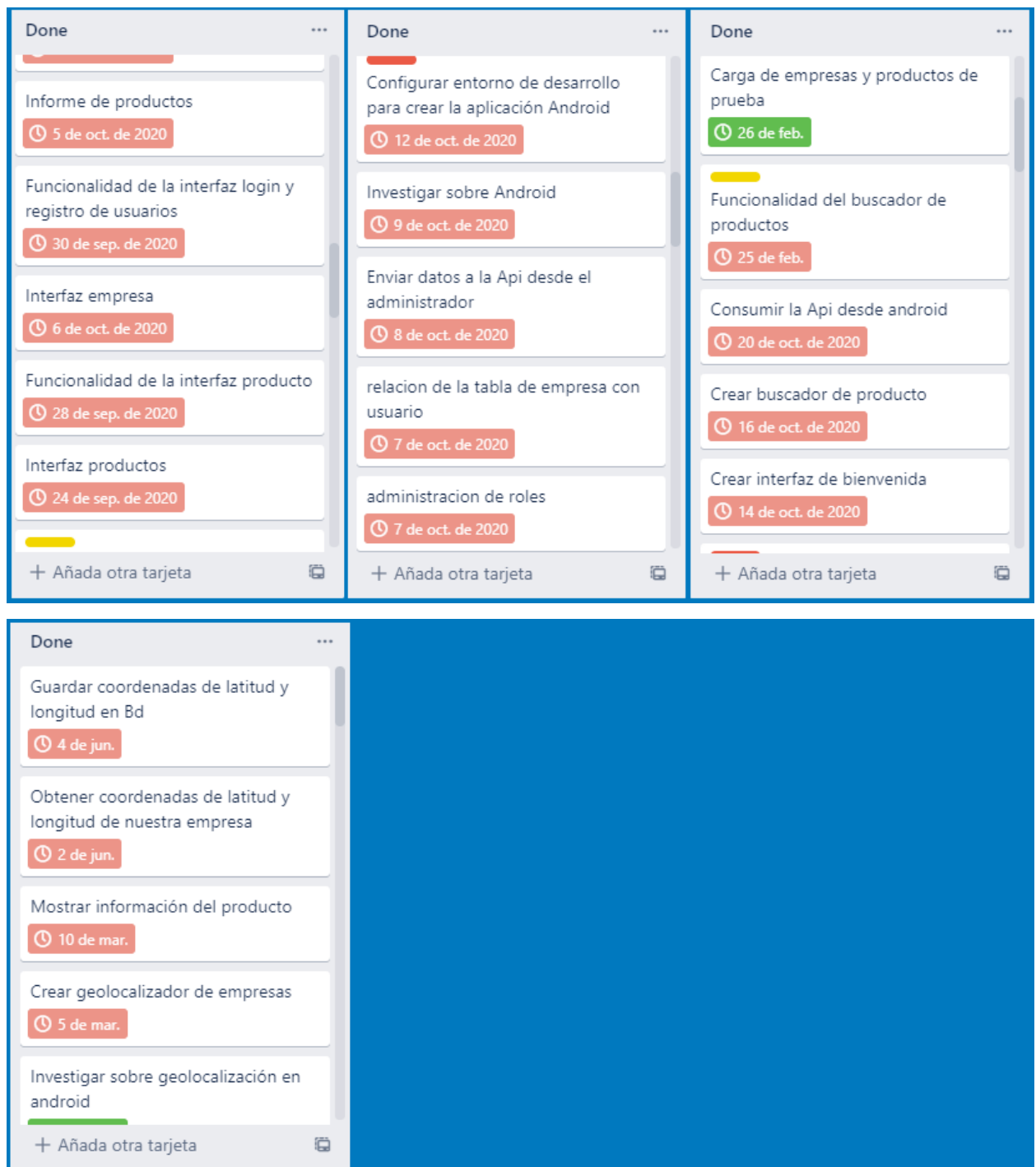
El proceso unificado Rational (RUP) es un marco de trabajo de proceso de desarrollo de software iterativo creado por Rational Software Corporation, una división de IBM desde 2003. RUP no es un proceso preceptivo concreto individual, sino un marco de trabajo de proceso adaptable, con la

idea de ser adaptado por las organizaciones de desarrollo y los equipos de proyecto de software que seleccionarán los elementos del proceso que sean apropiados para sus necesidades (Laboratorio Nacional de Calidad del Software., Marzo de 2009).

Para poder dar seguimiento al desarrollo de la aplicación móvil, el api, y la interfaz de administrador se utilizó la herramienta de administración de proyectos Trello, creando tarjetas que indican cuales son las tareas a realizar como podemos ver en la ilustración 5.

Ilustración 5: Tablero de actividades.





Fuente: Elaboración propia (2021).

5.2. Selección del Entorno Tecnológicos

Para desarrollo de la aplicación móvil, el api, y la interfaz de administrador se utilizó una diversidad de herramientas tecnológicas que conforman el entorno de desarrollo las cuales se pueden mencionar a continuación:

✓ **Para la interfaz de administrador se utilizaron las siguientes herramientas:**

- **PHP:** Es un lenguaje de secuencias de comandos del servidor, y es una herramienta de gran alcance para hacer páginas web dinámicas e interactivas rápidamente. Es una alternativa ampliamente utilizada, libre y eficaz (w3schools, 2014).
- **Laravel:** es un reciente framework de PHP que es fácil de aprender y ofrece una muy interesante propuesta, sobre todo para los desarrolladores que anden en busca de una herramienta eficiente y de rápido aprendizaje, en el desarrollo de proyectos web. Se inició el año 2011, y aprovecha las mejoras de PHP 5.3 ofreciendo una sintaxis clara y simple en la creación de código PHP. Se pueden escribir aplicaciones web con muy pocas líneas de código que además fáciles de entender, incluso para un programador recién iniciado (patricio, 2013).
- **Html:** Es el lenguaje con el que se escriben las páginas web. Las páginas web pueden ser vistas por el usuario mediante un tipo de aplicación llamada navegador. Podemos decir por lo tanto que el HTML es el lenguaje usado por los navegadores para mostrar las páginas webs al usuario, siendo hoy en día la interface más extendida en la red. Este lenguaje nos permite aglutinar textos, sonidos e imágenes y combinarlos a nuestro gusto. Además, y es aquí donde reside su ventaja con respecto a libros o revistas, el HTML nos permite la introducción de referencias a otras páginas por medio de los enlaces hipertexto (Alvarez, 2001).
- **HTML5:** Es la última versión de HTML. El término representa dos conceptos diferentes:
 - Se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos.
 - Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance. A este conjunto se le llama *HTML5* y *amigos*, a menudo reducido a *HTML5* (JosueMolina, 2015).

- **XAMPP:** Es una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl. El paquete de instalación de XAMPP ha sido diseñado para ser increíblemente fácil de instalar y usar (XAMPP, 2021).
- **Javascript:** Es un lenguaje de programación que surgió con el objetivo inicial de programar ciertos comportamientos sobre las páginas web, respondiendo a la interacción del usuario y la realización de automatismos sencillos. En ese contexto podríamos decir que nació como un "lenguaje de scripting" del lado del cliente, sin embargo, hoy Javascript es mucho más. Las necesidades de las aplicaciones web modernas y el HTML5 ha provocado que el uso de Javascript que encontramos hoy haya llegado a unos niveles de complejidad y prestaciones tan grandes como otros lenguajes de primer nivel (S.L., Guiarte Multimedia, 2014).
- **CSS:** es un lenguaje (no un lenguaje de programación) utilizado para especificar el aspecto de una página web para diferentes dispositivos. Esto entra en contraste con HTML que es un lenguaje que define la estructura de un documento para su distribución por la web. El HTML le dice al navegador como se estructura el documento mientras que el CSS le dice como debe renderizarlo (genbeta:dev, 2011).
- **CSS3:** añade muchas capacidades nuevas a la especificación anterior. Aunque aún está en proceso de estandarizado y le queda un largo camino por recorrer, ya se ha hecho un hueco al lado de HTML5 y supone uno de los mayores adelantos en el diseño web actual (genbeta:dev, 2011).
- **Bootstrap:** es un kit de herramientas de código abierto front-end más popular del mundo, que incluye variables y mixins de Sass, sistema de cuadrícula receptivo, amplios componentes prediseñados y potentes complementos de JavaScript (Bootstrap, 2021).
- **Composer:**
- **Base de datos MySQL:** Es la base de datos de código abierto de mayor aceptación mundial, ya que permite una creación asequible y fiable de aplicaciones de bases de datos integradas basadas en web de alto rendimiento y fácilmente ampliables (Oracle, 2014).

- **Atom:** es un editor de texto hackeable para el siglo XXI, de código abierto, está construida con HTML, JavaScript, CSS y la integración de Node.js. Trabaja con Git y GitHub todo su paquete ya viene incluido dentro del editor pudiendo crear nuevas ramas, establecer y confirmar, empujar y tirar así también resolver conflictos de fusión (GitHub, 2021).

✓ **Para verificar el funcionamiento de la API se utilizó la siguiente herramienta:**

- **Postman:** es una plataforma de colaboración para el desarrollo de API. Las funciones de Postman simplifican cada paso de la creación de una API y agilizan la colaboración para que pueda crear mejores API, más rápido. Envía de forma rápida y sencilla solicitudes REST, SOAP y GraphQL. Comunica el comportamiento esperado de una API mediante la simulación de puntos finales y sus respuestas sin tener que configurar un servidor backend (Postman, 2021).

✓ **Para el desarrollo de la aplicación móvil se utilizaron las siguientes herramientas:**

- **Java:** Es un lenguaje de programación orientado a objetos que fue creado por la compañía Sun Microsystems en 1995. La plataforma se puede descargar de forma gratuita y hoy en día Java se ejecuta en cientos de millones de computadoras personales de todo el mundo y en miles de millones de dispositivos, como lo son los dispositivos móviles, aparatos de televisión, aparatos especializados y consolas de juegos (Castro, 2015).
- **Android:** es un sistema operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Symbian y Blackberry OS. Lo que lo hace diferente es que está basado en Linux, **un núcleo de sistema operativo libre, gratuito y multiplataforma**. El sistema permite programar aplicaciones en una variación de Java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java (GONZALEZ, 2011).

- **Android Studio:** es el IDE (entorno de desarrollo integrado) oficial para el desarrollo de aplicaciones para Android, basado en IntelliJ IDEA. En la parte superior de potentes herramientas de edición de código y desarrolladores de IntelliJ, Android Studio ofrece aún más características que mejoran su productividad en la construcción de aplicaciones de Android (Developers, s.f.).
 - **SDK de Android:** SDK responde a las siglas *Software Development Kit*, lo que viene a ser un kit de desarrollo de software. Con él podremos desarrollar aplicaciones y ejecutar un emulador del sistema Android de la versión que sea. Todas las aplicaciones Android se desarrollan en lenguaje Java con este kit (Subirats, 2014).
- ✓ **Pata el sistema de control de versiones se utilizaron las siguientes herramientas:**
- **Git:** es un sistema de control de versiones distribuido de código abierto y gratuito diseñado para manejar todo, desde proyectos pequeños a muy grandes, con velocidad y eficiencia. Es fácil de aprender y ocupa poco espacio con un rendimiento increíblemente rápido. Supera a las herramientas SCM como Subversion, CVS, Perforce y ClearCase con características como bifurcaciones locales económicas, áreas de preparación convenientes y múltiples flujos de trabajo (Git, 2021).
 - **GitHub:** es una plataforma de alojamiento de código para el control de versiones y la colaboración. Le permite a usted y a otras personas trabajar juntos en proyectos desde cualquier lugar. Es la mejor forma de crear y enviar software. Potente colaboración, revisión de código y gestión de código para proyectos privados y de código abierto. (GitHub Guides, 2020).

Para el desarrollo y administración de todo el proyecto fueron utilizadas las herramientas mencionadas anteriormente, ya que se posee experiencia en el uso de las mismas como así también en algunas herramientas se invirtió tiempo para poder adaptarse a su uso a través de largas horas de capacitaciones, mediante el cual se logró obtener los mejores resultados para el uso de las mismas.

5.3. Diseño

Con el objetivo de diseñar una interfaz de administrador, diseñar un api y desarrollar la aplicación móvil se diseñaron esquemas con los cuales podemos observar cómo se plantearon todas las pantallas ya sean de inicio de sesión, abm, menús, o en el caso de la app móvil como serían sus pantallas para buscar un producto y como seria visto en el mapa de google.

En el caso de la Interfaz de administrador se utilizaron plantillas de adminlte, ya que ofrece unas pantallas amigables y sencilla para el usuario, además de ser muy fácil de integrar a cualquier proyecto hecho en Laravel.

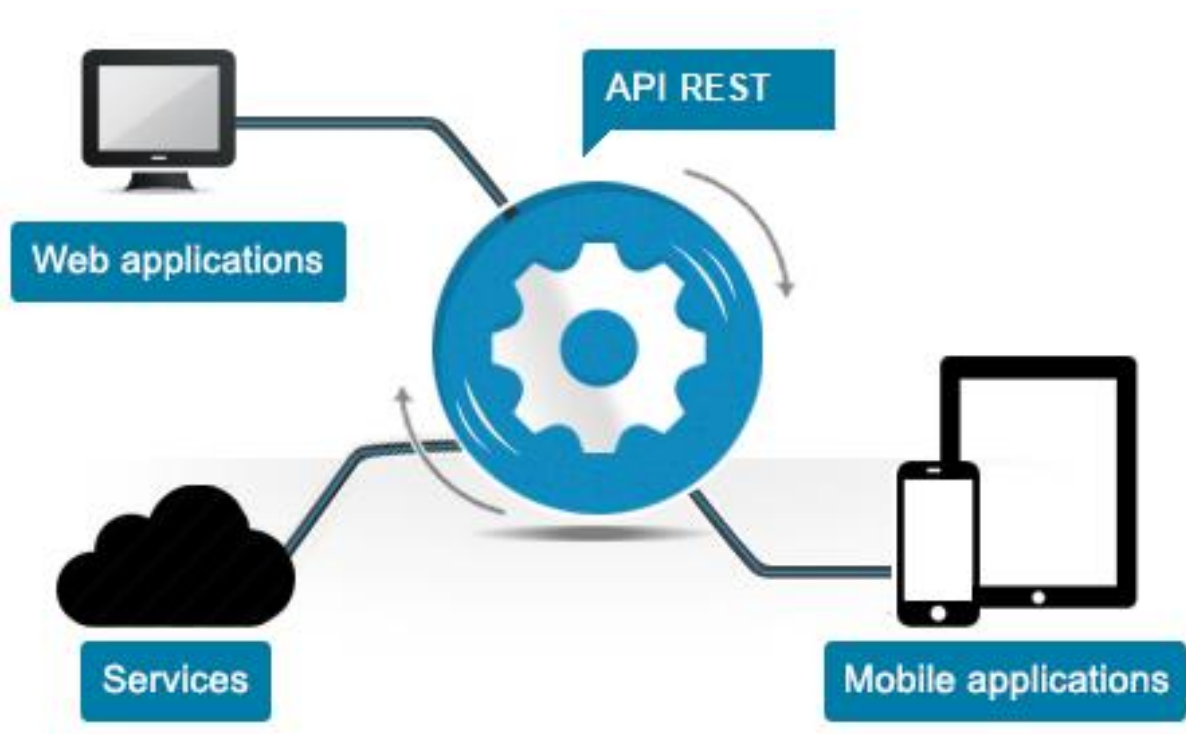
Con lo que se refiere a la aplicación móvil utilizamos paleta de colores para definir los estilos de las pantallas ya sea de buscador, así como también de los botones; además de ello se utilizó el mapa de google maps para integrarlo a nuestro proyecto.

5.3.1. Arquitectura de la App móvil

La arquitectura de la app móvil está formada por tres partes fundamentales que serían la interfaz de administrador, el api y la aplicación móvil.

La interfaz de administrador será el encargado de guardar todos los productos con sus respectivos detalles además de agregar una imagen de una determinada empresa, de la empresa se guardarán todos sus datos y su ubicación geográfica y estos serán alojados en una base de datos mysql para que posteriormente mediante un api sean consultados por nuestra app móvil como lo podemos observar en la ilustración 6.

Ilustración 6: Arquitectura de la app móvil.



Fuente: Elaboración propia (2021).

5.3.2. Base de datos que fue utilizada para la interfaz de administrador:

Base de datos MySQL: Es un sistema de gestión de bases de datos relacionales eficaz, sólido y flexible. Se utiliza con frecuencia para la creación de base de datos con acceso a páginas web dinámicas, así también para la creación de sistemas de transacciones on-line o ara cualquier otra solución profesional que implique almacenar datos teniendo la posibilidad de realizar múltiples y rápidas consultas (Cobo, Gómez, Pérez, & Rocha, 2005).

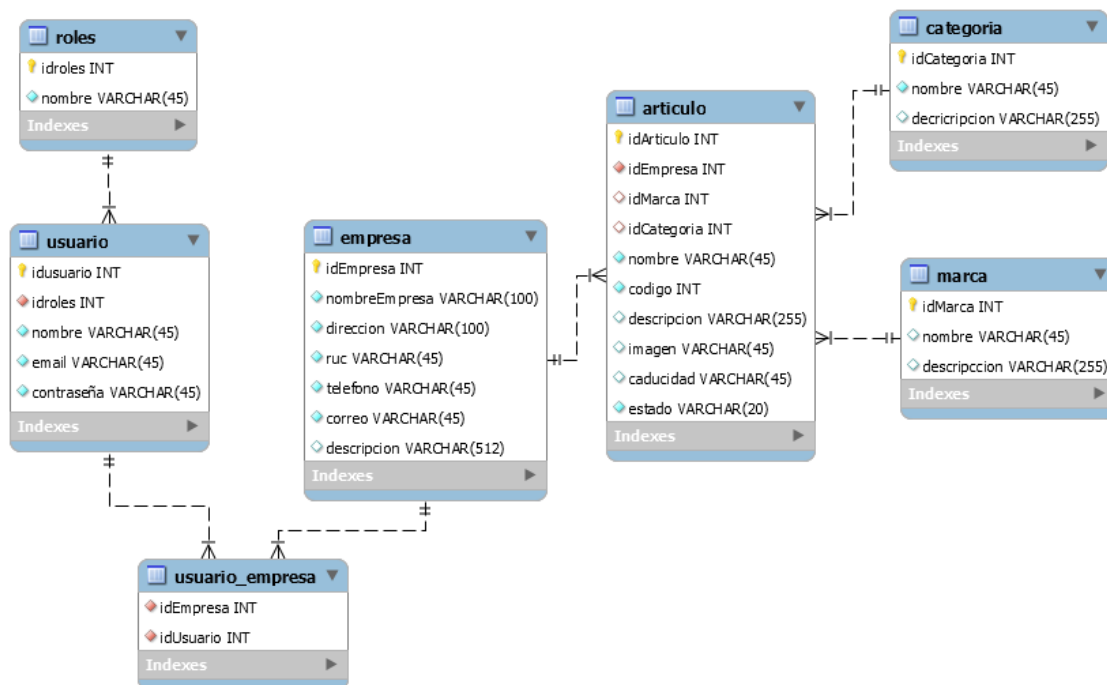
MySQL brinda varias ventajas respecto a otros sistemas gestores de base de datos:

- Es fácil de adaptar y personalizar a nuestras necesidades, ya que tiene licencia pública.
- Haciendo uso de la licencia GPL puede ser descargado de forma gratuita mediante Internet.
- Su integración con otras aplicaciones se puede hacer fácilmente gracias a que está desarrollado en el lenguaje de programación C y C++.

- También se puede solicitar una licencia comercial para aquellos proyectos basados en MySQL que desean no ser de “código abierto”.
- Utiliza el lenguaje SQL (Lenguaje de Consulta Estructurado) que es el lenguaje de consulta más utilizado y estandarizado para acceder a las bases de datos relacionales.
- Controla el acceso simultáneo de un gran número de usuarios a los datos y asegura el acceso solo a usuarios autorizados gracias a que su sistema se basa en el modelo cliente-servidor, ya que este permite trabajar como servidor multiusuario y de subprocesamiento múltiple.
- Es portable, lo que quiere decir, puede ser llevado a cualquier plataforma informática, además está disponible en más de veinte plataformas diferentes incluyendo las distribuciones más usadas de Linux., sistema operativo Mac X, Unix y Microsoft Windows (Cobo, Gómez, Pérez, & Rocha, 2005).

Para la interfaz de administrador se diseñó una base de datos denominada “admempresa” con las tablas relacionales con sus respectivos atributos y métodos como lo podemos observar en la ilustración 7.

Ilustración 7: Base de Datos “admempresa”



Fuente: Elaboración propia (2020)

Para poder conectar nuestro sistema a una base de datos nos dirigimos a nuestro proyecto Empresa ubicado en nuestro editor de texto Atom y abrimos el archivo **.env** y modificamos el **DB_DATABASE** con el nombre de nuestra base de datos y en **DB_USERNAME** con el nombre de root y en el **DB_PASSWORD** dejar vacío, como podemos observar en la Ilustración 8.

Ilustración 8: Conexión de la base de datos

```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:n7mqYqmQ/T8pGiI6G8E1IhL/ld2TL79pZutV1vbkeDU=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 GOOGLE_MAPS_API_KEY=AIzaSyCo4LRswMXoWZhaq-7YmGBhggYihH4ZzZE
8
9 LOG_CHANNEL=stack
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=admEmpresa
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 QUEUE_CONNECTION=sync
21 SESSION_DRIVER=file
22 SESSION_LIFETIME=120
23
24 REDIS_HOST=127.0.0.1
25 REDIS_PASSWORD=null
26 REDIS_PORT=6379
27
28 MAIL_HOST=smtp.mailtrap.io
29 MAIL_PORT=2525
30 MAIL_USERNAME=null
31 MAIL_PASSWORD=null
32 MAIL_ENCRYPTION=null
33 MAIL_FROM_ADDRESS=null
34 MAIL_FROM_NAME="${APP_NAME}"
35
36 AWS_ACCESS_KEY_ID=
37 AWS_SECRET_ACCESS_KEY=
38 AWS_DEFAULT_REGION=us-east-1
39 AWS_BUCKET=
40
41 PUSHER_APP_ID=
42 PUSHER_APP_KEY=
43 PUSHER_APP_SECRET=
44 PUSHER_APP_CLUSTER=mt1
45
46 MIX_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
47 MIX_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"
48
49
```

Fuente: Elaboración propia (2021)

Script de la tabla empresa donde se almacenan el nombre de la empresa con su información como se muestra en la ilustración 9, además de su latitud y longitud que podrá ser utilizado por la app móvil mediante un api.

```
CREATE TABLE `empresas` (
  `idEmpresa` int(10) UNSIGNED NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  `nombreEmpresa` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `direccion` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `ruc` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `telefono` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
```



```

`correo` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
`descripcion` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
`latitud` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
`longitud` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
INSERT INTO `empresas` (`idEmpresa`, `created_at`, `updated_at`, `nombreEmpresa`,
`direccion`, `ruc`, `telefono`, `correo`, `descripcion`, `latitud`, `longitud`) VALUES
(2, NULL, NULL, 'Roperia Patricia', 'Circuito comercial', '564263-2', '0981546897',
'paty@gmail.com', 'Venta de ropas en general', '-27.35953006077587', '-55.848282006977094'),
(3, NULL, NULL, 'Ropería Patricia2', 'Circuito Comercial', '5246597-4', '0985654687',
'paty2@gmail.com', 'Venta de ropas para niños', '-27.359625347842872', '-55.84830346464921');

```

Ilustración 9: Tabla empresa

	idEmpresa	created_at	updated_at	nombreEmpresa	direccion
<input type="checkbox"/>  Editar  Copiar  Borrar	2	NULL	NULL	Roperia Patricia	Circuito comercial
<input type="checkbox"/>  Editar  Copiar  Borrar	3	NULL	NULL	Ropería Patricia2	Circuito Comercial

ruc	telefono	correo	descripcion	latitud	longitud
564263-2	0981546897	paty@gmail.com	Venta de ropas en general	-27.35953006077587	-55.848282006977094
5246597-4	0985654687	paty2@gmail.com	Venta de ropas para niños	-27.359625347842872	-55.84830346464921

Fuente: Elaboración propia (2021)

Script de la tabla artículos donde se almacena toda la información del producto además de una imagen de esta como se muestra en la ilustración 10, para poder ser obtenidos mediante un api por la aplicación móvil y así poder mostrar al usuario.

```

CREATE TABLE `articulos` (
  `idArticulo` int(10) UNSIGNED NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  `idEmpresa` int(10) UNSIGNED NOT NULL,
  `idCategoria` int(10) UNSIGNED NOT NULL,
  `idMarca` int(10) UNSIGNED NOT NULL,
  `nombre` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,

```

```

`codigo` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
`descripcion` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
`imagen` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
`estado` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
INSERT INTO `articulos` (`idArticulo`, `created_at`, `updated_at`, `idEmpresa`,
`idCategoria`, `idMarca`, `nombre`, `codigo`, `descripcion`, `imagen`, `estado`) VALUES
(1, NULL, NULL, 3, 1, 1, 'Pantalon', '26546587', 'Pantalon para caballero', 'th (1).jpg',
'Activo'),
(2, NULL, NULL, 2, 1, 1, 'Remera', '5465987', 'Remera para caballero color negro',
'ga050004696_2-073ec9f2bc0c0bd3bf705615cf3904c9-1024-1024.jpg', 'Activo');

```

Ilustración 10: Tabla de productos

idArticulo	updated_at	created_at	idEmpresa	idCategoria	idMarca	nombre	codigo	descripcion	imagen	estado
1	NULL	NULL	3	1	1	Pantalon	26546587	Pantalon para caballero	pantalon.jpg	Activo
2	NULL	NULL	2	1	1	Remera	5465987	Remera para caballero color negro	remera.jpg	Activo
3	NULL	NULL	2	1	1	pantalon	5462136	pantalon para criatura	pantalon1.jpg	Activo

Fuente: Elaboración propia (2021)

5.3.3. Diseño de interfaz de la app móvil

Para el diseño de interfaz de la app móvil utilizamos la herramienta balsamiq en el cual se definieron los botones de búsqueda y los estilos para listar los productos encontrados con sus respectivos detalles; así como también la cantidad de interfaces que tendrá y como se verá el mapa de google en nuestra aplicación.

Primeramente, se desarrolló un api con el fin de comunicar nuestra interfaz de administrador que es una aplicación web con nuestra aplicación móvil. Con el objetivo de hacer peticiones a nuestro api utilizamos Retrofit.

Para poder listar los productos en nuestra aplicación utilizamos RecyclerView y CardView que permite crear listas dinámicas en Android. Así también para poder obtener las imágenes de un producto de nuestra página web se utilizó la librería Volley.

Para poder observar en el mapa de google la ubicación de la empresa que cuenta con el producto obtenido en la búsqueda se utilizó el api de google maps, así también mostrar mediante un puntero el nombre de la empresa y sus detalles.

En la app móvil nuestra pantalla de buscador cuenta con una lista de todos los productos traídos de nuestra página web con sus nombres, detalles y una imagen. Así también cuenta con campo de texto en el cual se puede ingresar cualquier nombre de un producto para poder realizar la búsqueda tal como se muestra en la Ilustración 11.

Ilustración 11: Pantalla buscador



Fuente: Elaboración propia (2020)

Podemos observar en la Ilustración 12 como nuestra pantalla nos muestra una imagen del producto obtenido en la búsqueda, así también sus detalles y la opción de poder ver la dirección exacta de la empresa en un mapa de google si así lo desea el usuario.

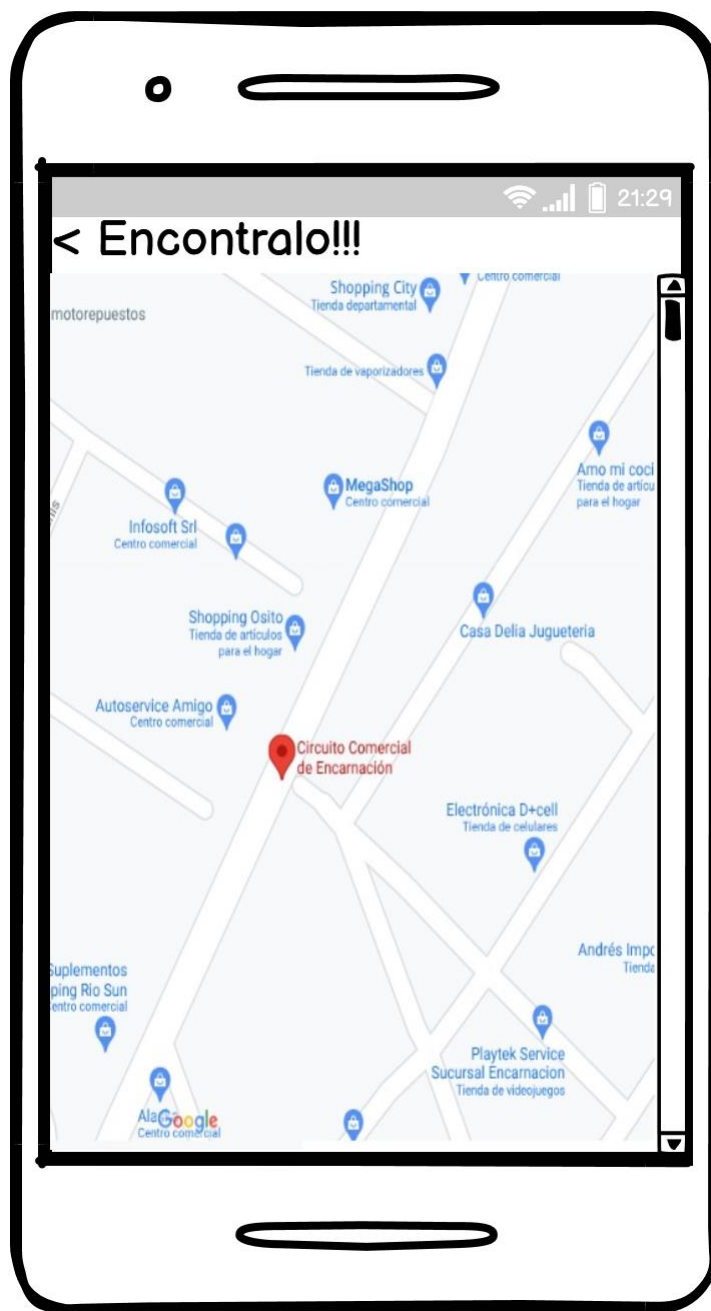
Ilustración 12: Pantalla de descripción del producto



Fuente: Elaboración propia (2020)

En la última pantalla de nuestra aplicación móvil nos muestra el mapa de google y un puntero con el nombre de la empresa y un número de contacto tal como podemos observar en la Ilustración 13.

Ilustración13: Pantalla de google maps



Fuente: Elaboración propia (2020)

5.3.4. Diseño de la interfaz administrador

Para la aplicación web se utilizó las plantillas de **AdminLTE 3** desarrollada con Bootstrap 3, el cual cuenta con vistas agradables para el usuario y ya trae predefinidas los botones de menús, estilos y galerías que pueden ser utilizadas adaptando a nuestro proyecto y dependiendo de lo que queremos mostrar se puede sacar o agregar distintas opciones como estilos, colores o diseños. Además, es una plantilla totalmente responsiva lo que quiere decir

puede ser utilizado en teléfonos móviles o Tablet, así como también en computadoras personales o de escritorio

En la pantalla de login podemos encontrar el logo de la página Encontralo y más abajo un cuadro de texto que solicita el email y la contraseña para poder ingresar a la página web, así también cuenta con la opción de recordar la contraseña y email si así el usuario lo desea. Como también tiene la opción de dirigirnos a otra pantalla en caso de que el usuario olvido la contraseña o desee registrarse creando una cuenta nueva como se puede apreciar en la Ilustración 14.

Ilustración 14: Diseño de Login

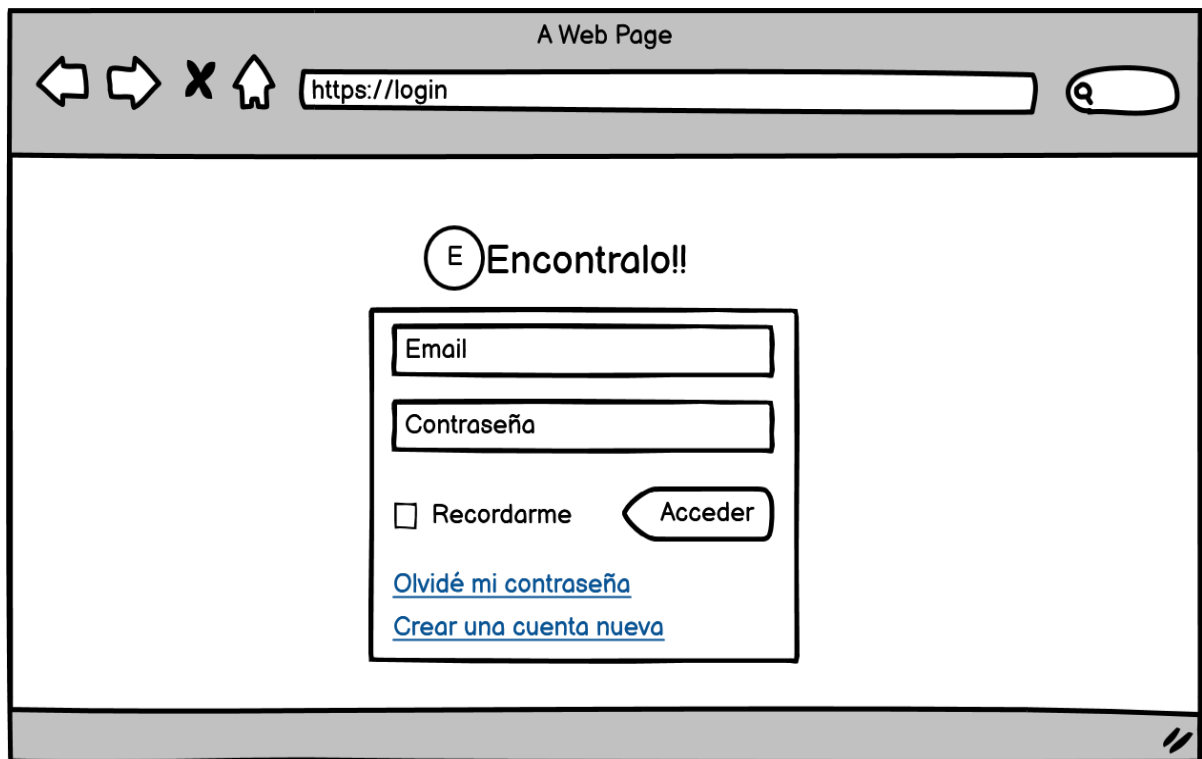


Diagrama de la interfaz de login de la página Encontralo!!

La interfaz se muestra dentro de un navegador web con el título "A Web Page". La barra de direcciones contiene "https://login" y un icono de lupa.

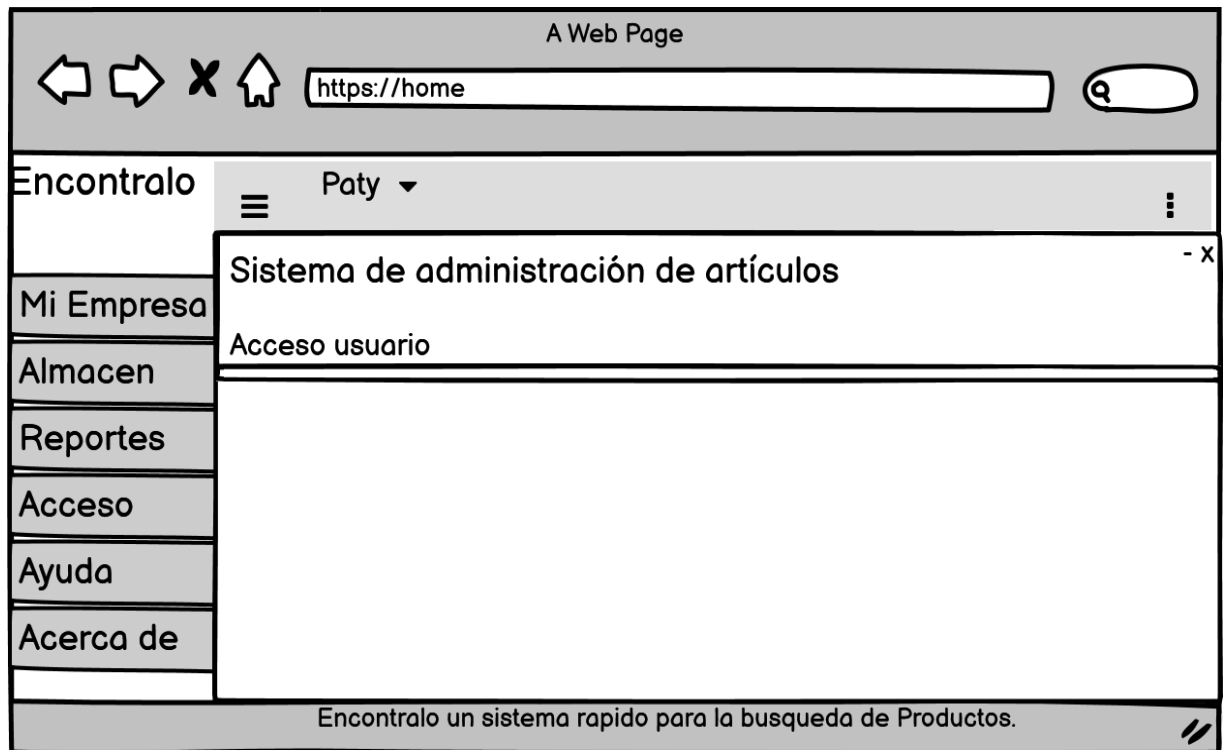
El contenido principal de la página es:

- Logo: Un círculo con la letra "E" y el texto "Encontralo!!".
- Formulario de login:
 - Campo de texto "Email".
 - Campo de texto "Contraseña".
 - Caja de verificación "Recordarme".
 - Botón "Acceder".
 - Enlaces de texto: [Olvidé mi contraseña](#) y [Crear una cuenta nueva](#).

Fuente: Elaboración propia (2020)

La pantalla principal de nuestro sistema cuenta con los botones de menú, así también nos muestra el usuario logueado en la parte superior con la opción de cerrar sesión. En la parte central nos indicara el acceso correspondiente ya sea de usuario o de administrador, además de una pequeña descripción del sistema, así como se puede ver en la Ilustración 15.

Ilustración 15: Pantalla principal

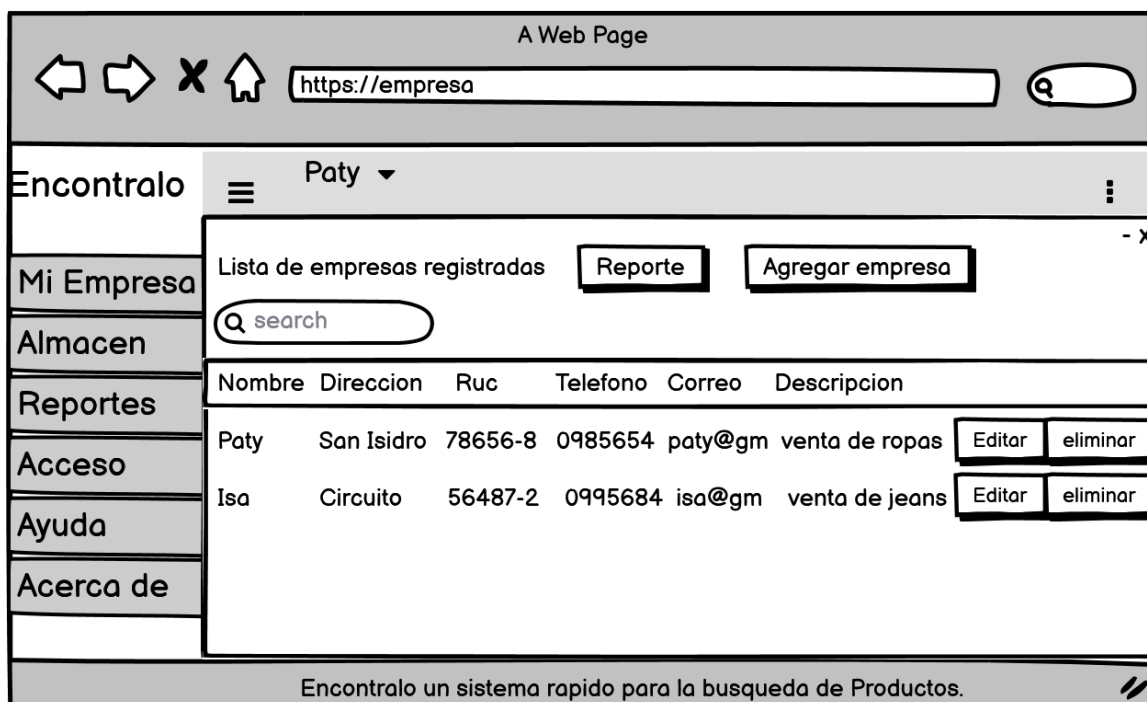


Fuente: Elaboración propia (2020)

La pantalla de index nos muestra todas las opciones que tiene nuestra tabla de empresas como ser la de agregar, eliminar y editar, así también nos muestra la opción de mostrar un reporte y buscar las empresas por su nombre como lo podemos observar en la ilustración 16.

Todas estas características son aplicables a las demás tablas como ser la de productos, marcas y categorías.

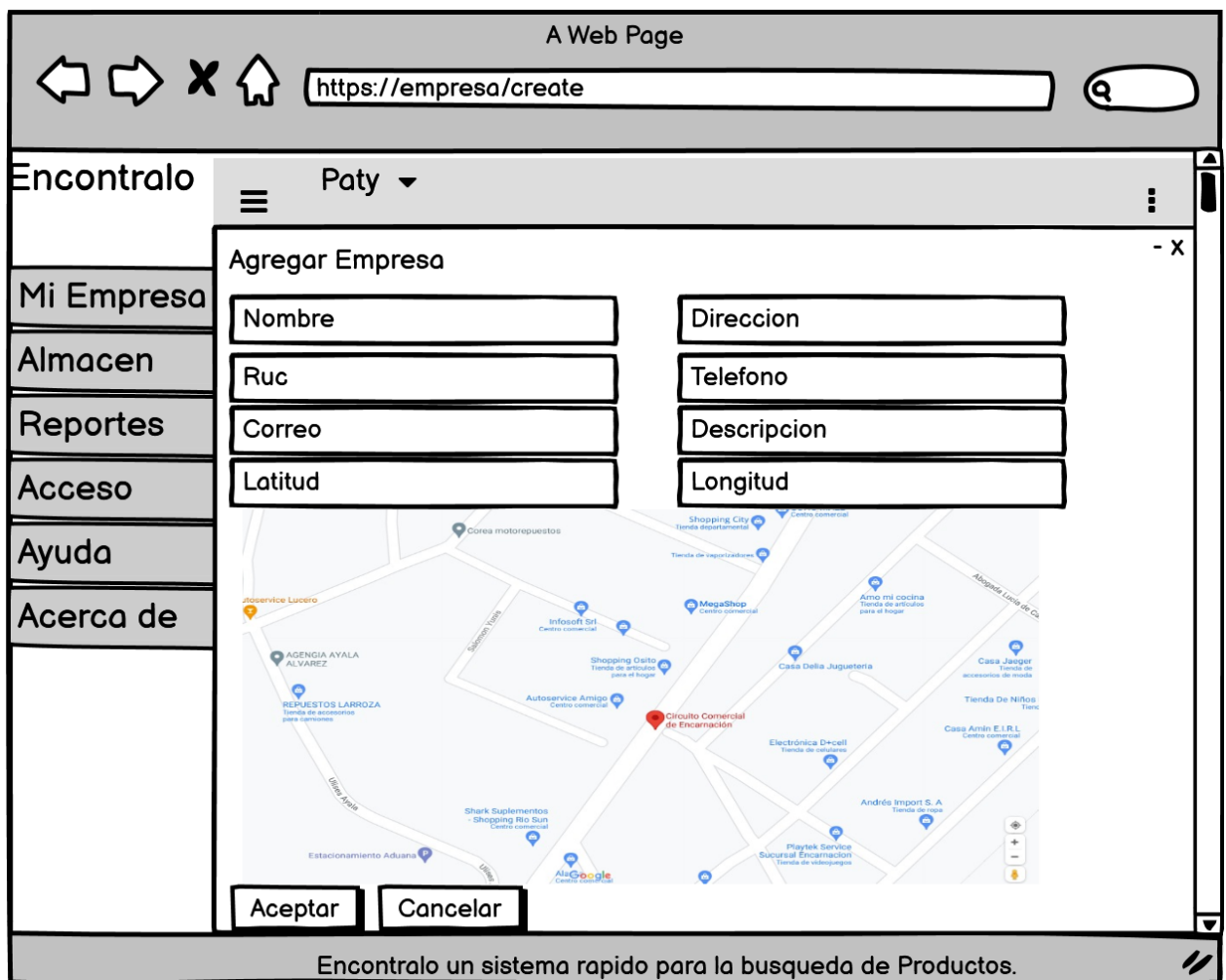
Ilustración 16: Pantalla de index



Fuente: Elaboración propia (2020)

Contamos con una pantalla en la que podemos agregar una empresa para ello debemos de ingresar todos los datos requeridos, además de marcar en un mapa de google la latitud y longitud de la ubicación de la empresa, luego seleccionar el botón de aceptar para enviar todos los datos o cancelar en caso que no se desee enviar nada. Todas estas características son aplicables a las demás tablas como ser la de productos, marcas y categorías, así como lo podemos apreciar en la ilustración 17.

Ilustración 17: Agregar empresa



Fuente: Elaboración propia (2020)

5.4. Codificación de la interfaz administrador

Para poder dar inicio al proceso de codificación primeramente fueron analizados todos los requerimientos del sistema para luego seguir usados la metodología de desarrollo escogida para un mejor seguimiento de los avances obtenidos, además se utilizarán como guía las interfaces realizadas.

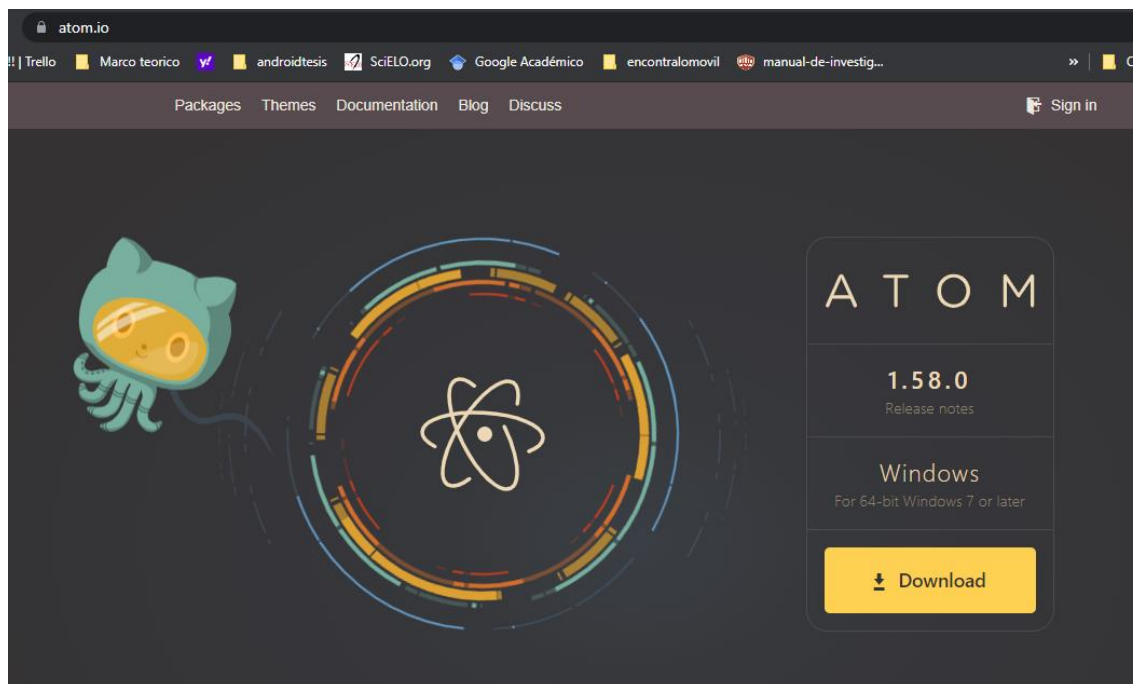
5.4.1. Configuración de recursos para la interfaz de administrador

Para poder desarrollar la aplicación web que sería utilizado como nuestra interfaz de administrador encargado de almacenar todas las empresas y sus productos se utilizaron diferentes herramientas como ser atom, xampp, php, composer y el gestor de base de datos mysql.

5.4.1.1. Instalación del editor de texto atom:

Para instalar el editor de texto ATOM nos dirigimos a su página oficial <https://atom.io/>. Y descargamos el ejecutable compatible con nuestro sistema operativo que por defecto dicha página web ya lo reconoce tal como lo podemos observar en la Ilustración 18.

Ilustración 18: Descarga de editor de texto ATOM



Fuente: Elaboración propia (2021).

Una vez que tengamos el ejecutable procedemos a instalar abriendo el instalador y siguiendo todas las instrucciones.

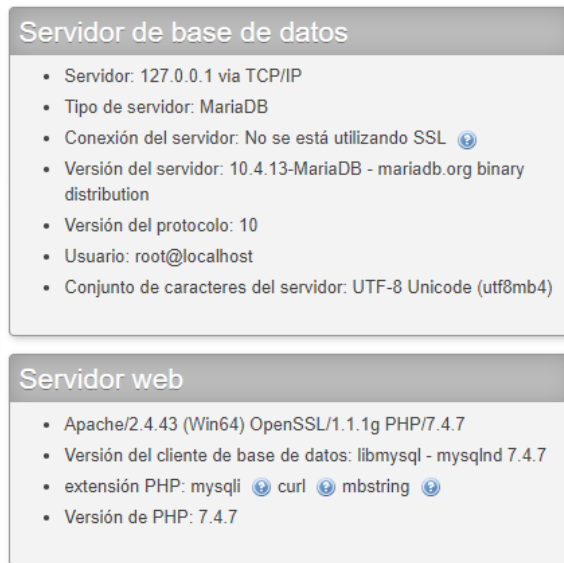
5.4.1.2. Instalación de las librerías del lenguaje PHP y gestor de base de datos MySQL mediante el paquete de instalación XAMPP

Para poder instalar todas las dependencias del lenguaje de programación PHP se utilizó el paquete de instalación XAMPP el cual además trae incluido el gestor de base de datos MySQL.

Xampp es una herramienta de software libre el cual consiste en el sistema de gestión de base de datos mysql, el servidor Web Apache y los expertos para lenguajes de script: php y Perl. El nombre Xampp proviene del acrónimo X (el cual significa que es utilizado para cualquiera de los diferentes sistemas operativos, esto quiere decir que es una herramienta multiplataforma), Apache, mysql, php, Perl. Desde la versión 5.6.15, xampp cambió la base de datos de mysql a MariaDB, esta última es un fork de mysql con licencia gpl.

Para la realización de este proyecto se utilizó el servidor de MariaDB en su versión 10.4.13, con el conjunto de caracteres UTF-8 Unicode (utf8mb4), la versión del cliente de base de datos fue de libmysql - mysqlnd 7.4.7, la extensión PHP fue mysqli curl mbstring y la versión de PHP fue 7.4.7 tal como se muestran en la Ilustración 19.

Ilustración 19: Gestor de Base de datos

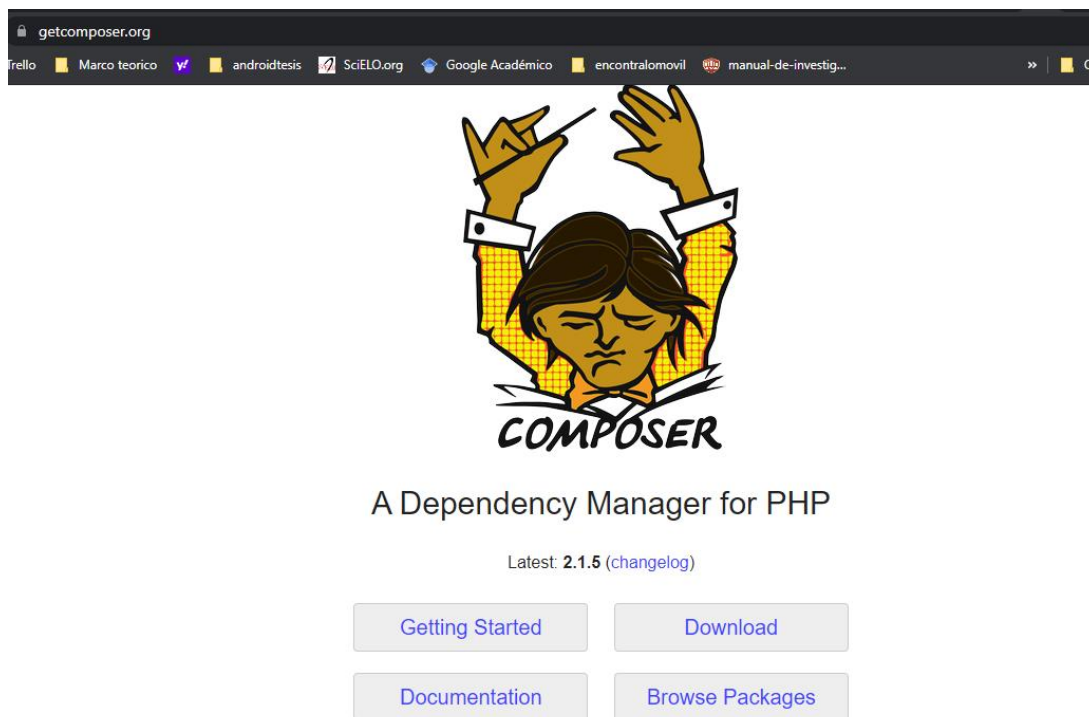


Fuente: Elaboración propia (2021)

5.4.1.3. Instalación de composer

Como requisito para la ejecución de un proyecto desarrollado en Laravel, necesitamos utilizar Composer ya que es una aplicación que nos sirve para la gestión de paquetes y dependencias del lenguaje PHP. Para poder instalar nos dirigimos a su sitio oficial y descargamos el archivo ejecutable como podemos ver en la Ilustración 20.

Ilustración 20: Descarga de composer



Fuente: Elaboración propia (2021)

5.4.2. Codificación e implementación de librerías para la interfaz administrador

Para la codificación de la aplicación web de nuestra interfaz de administrador se utilizó el lenguaje de programación PHP con el framework de desarrollo Laravel, el cual cuenta con una arquitectura Modelo-vista-controlador (MVC). El modelo es el que realiza las peticiones a la base de datos, a un repositorio de datos o un api; la vista es lo que se muestra al usuario mediante el navegador; y el controlador es el que se encarga de resolver a quién enviar la información.

La estructura de un proyecto de Laravel consiste en que un usuario realiza peticiones a una página web mediante el uso de rutas y estas envían las notificaciones al controlador, esta procesa dicha información y decide si muestra una vista o son enviadas al modelo quien se encarga de hacer consultas a la base de datos y retorno la información al controlador, este último procesa dicha información para enviarla a la vista y mostrar lo obtenido al usuario final tal como podemos observar en la ilustración 21.

Ilustración 21: Estructura de un proyecto laravel.

De esta manera se crea el proyecto y así también se instalan todas las librerías y dependencias en su última versión para poder ser utilizadas y seguir con el desarrollo.

Laravel cuenta con una librería con el cual fácilmente podemos integrar el login y registro de usuario a nuestra página web, para ello usaremos el siguiente comando: **composer require laravel/ui** y seguidamente el siguiente comando para las autenticaciones: **php artisan ui:auth** como se puede observar en la ilustración 23.

Ilustración 23: Creación del login

```
C:\Empresa>composer require laravel/ui
Using version ^3.3 for laravel/ui
./composer.json has been updated
Running composer update laravel/ui
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
  - Locking laravel/ui (v3.3.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
  - Downloading laravel/ui (v3.3.0)
  - Installing laravel/ui (v3.3.0): Extracting archive
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/sail
Discovered Package: laravel/tinker
Discovered Package: laravel/ui
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
74 packages you are using are looking for funding.
Use the `composer fund` command to find out more!

C:\Empresa>php artisan ui:auth
Authentication scaffolding generated successfully.
```

Fuente: Elaboración propia (2021)

Para poder desarrollar un ABM (Altas, Bajas y Modificaciones) necesitamos crear el modelo, la vista y el controlador; empezamos creando el modelo para ello usamos el comando **php**

artisan make:model, luego agregamos la ruta, seguidamente añadimos el controlador con el comando `php artisan make:controller EmpresaController` y por ultimo definimos la vista utilizando las plantillas blade.

En nuestro ejemplo creamos un ABM para poder listar, agregar, eliminar y modificar empresas con el siguiente comando **php artisan make:model Empresa** con el cual se crea el modelo satisfactoriamente para las empresas como se puede ver en la Ilustración 24.

Ilustración 24: Modelo de empresa

```
C:\Empresa>php artisan make:model Empresa
Model created successfully.
```

Fuente: Elaboración propia (2021)

El código que obtuvimos a partir de este comando fue un modelo vacío el cual agregamos algunas funciones y métodos; primeramente, colocamos a que tabla de nuestra base de datos va hacer referencia, luego indicamos los atributos como lo podemos observar en la Ilustración 25.

Ilustración 25: Código del modelo

```
7  class Empresa extends Model
8  {
9      protected $table = 'empresas';
10
11     protected $primaryKey = 'idEmpresa';
12
13     public $timestamps = false;
14
15     protected $fillable = [
16         'nombreEmpresa',
17         'direccion',
18         'ruc',
19         'telefono',
20         'correo',
21         'descripcion',
22         'latitud',
23         'longitud'
24     ];
25     public function users(){
26         return $this->belongsToMany('User::class')->withTimesTamps();
27     }
28     protected $guarded = [];
29 }
30
```


Fuente: Elaboración propia (2021)

Definimos una ruta del tipo resource la cual va dirigida a un controlador con el fin de crear un grupo de rutas que se encargue de realizar las diferentes peticiones como ser index, delete, edit, store, update y destroy; como lo podemos observar en la Ilustración 26.

Ilustración 26: Rutas

```
16 < Route::get('/', function () {  
17     return view('welcome');  
18 });  
19  
20 Auth::routes();  
21  
22 < Route::group(['middleware' => 'auth'], function () {  
23  
24     // Las rutas que incluyas aquí pasarán por el middleware 'auth'  
25  
26     Route::resource('empresa', 'EmpresaController');  
27
```

Fuente: Elaboración propia (2021)

Creamos un controlador con el comando **php artisan make:controller Empresa**, con el fin de agregar funciones que están relacionados a la tabla de empresas así como lo podemos observar en la Ilustración 27.

Ilustración 27: Creación de controlador

```
C:\Empresa>php artisan make:controller EmpresaController  
Controller created successfully.
```

Fuente: Elaboración propia (2021)

Con este comando se obtuvo un archivo de controlador en el cual agregamos todas las funciones necesarias para el funcionamiento del abm empresas los cuales son index que nos muestra la pantalla inicial de nuestro abm, create para crear un objeto empresa como lo podemos observar en la Ilustración 28, store para almacenar objetos en la base de datos, show para mostrar, delete que nos permite borrar un elemento de la lista, edit para editar elementos, update para actualizar y destroy para destruir un objeto y eliminarlo de la tabla de la base de datos como lo podemos observar en la Ilustración 29.

Ilustración 28: Creación del index

```

public function index(Request $request){
    if ($request) {
        $query = trim($request->get('searchText'));
        $usuarioEmpresa= DB::table('usuario_empresas')->get();
        $user= DB::table('users')->get();
        $id = Auth::user()->id;
        $empresa = DB::table('empresas as e')
        ->join('usuario_empresas as ue','ue.idEmpresa','=','e.idEmpresa')
        ->join('users as u','ue.id','=','u.id')
        ->select('e.idEmpresa','e.nombreEmpresa','e.direccion','e.ruc', 'e.telefono','e.correo',
        'e.descripcion','ue.id')
        ->where('ue.id','=',$id)
        ->where('nombreEmpresa', 'LIKE', '% ' . $query . '%')
        ->groupBy('e.idEmpresa','e.nombreEmpresa','e.direccion','e.ruc', 'e.telefono','e.correo',
        'e.descripcion','ue.id')
        ->paginate(5);
        return view('empresa.index', ['empresa' => $empresa, 'usuario_empresas'=>$usuarioEmpresa,
        'users'=>$user, 'searchText'=> $query]);
    }
}

public function create(){
    $empresas=DB::table('empresas')->get();
    $usuarioEmpresa = DB::table('usuario_empresas')->get();
    return view("empresa.create",["empresas"=>$empresas,"usuario_empresas"=>$usuarioEmpresa]);
}

```

Fuente: Elaboración propia (2021)

Ilustración 29: Función edit y destroy

```

~
public function edit($idEmpresa)
{
    $usuarioEmpresa = DB::table('usuario_empresas')->get();
    return view('empresa.edit', ['empresa' => Empresa::findOrFail($idEmpresa),
    "usuario_empresas"=>$usuarioEmpresa]);
}
public function update(Request $request, $idEmpresa)
{
    $empresa = Empresa::findOrFail($idEmpresa);
    $empresa-> nombreEmpresa = $request->get('nombreEmpresa');
    $empresa-> direccion = $request->get('direccion');
    $empresa-> ruc = $request->get('ruc');
    $empresa-> telefono = $request->get('telefono');
    $empresa-> correo = $request->get('correo');
    $empresa-> descripcion = $request->get('descripcion');
    $empresa-> latitud = $request->get('latitud');
    $empresa-> longitud = $request->get('longitud');
    $empresa->update();
    return redirect( '/empresa');
}
public function destroy($idEmpresa)
{
    $empresa = Empresa::findOrFail($idEmpresa);
    $empresa->delete();
    return redirect( '/empresa');
}
}

```

Fuente: Elaboración propia (2021)

Seguidamente necesitamos un formulario para la validación de los campos de nuestra empresa el cual obtenemos mediante el comando **php artisan make:request EmpresaFormRequest** como se puede apreciar en la Ilustración 30.

Ilustración 30: Creación de Formulario Request

```

C:\Empresa>php artisan make:request EmpresaFormRequest
Request created successfully.

```

Fuente: Elaboración propia (2021)

Con este archivo que obtuvimos a partir de este comando primeramente damos autorización al usuario para realizar modificaciones en la función `authorize` cambiando su valor por `true`. Luego agregamos las reglas de validación como ser campos requeridos y cantidad máxima de caracteres permitido como lo podemos observar en la Ilustración 31.

Ilustración 31: Validación de formulario

```

14     public function authorize()
15     {
16         return true;
17     }
18     /**
19      * Get the validation rules that apply to the request.
20      *
21      * @return array
22      */
23     public function rules()
24     {
25         return [
26             'nombreEmpresa'=>'required|max:255',
27             'direccion'=>'required|max:100',
28             'ruc'=>'max:45',
29             'telefono'=>'required|max:45',
30             'correo'=>'required|max:100',
31             'descripcion'=>'max:512',
32             'latitud'=>'max:512',
33             'longitud'=>'max:512',
34         ];
35     }

```

Fuente: Elaboración propia (2021)

Para poder crear las vistas de nuestro abm utilizamos el motor de plantillas blade ya que los trozos de código se repiten en más de una vista como por ejemplo los menús desplegables y no tiene sentido estar reescribiendo código repetido en todas las vistas.

Para que la creación de las vistas sea aún más sencilla utilizamos adminLTE 3 utilizando el comando **composer require jeroennoten/laravel-adminlte** y luego instalamos todos los paquetes necesarios con el comando **php artisan adminlte:install** así como podemos ver en la Ilustración 32.

Ilustración 32: Instalación de adminLTE 3

```

C:\Empresa>composer require jeroennoten/laravel-adminlte
Using version ^3.6 for jeroennoten/laravel-adminlte
./composer.json has been updated
Running composer update jeroennoten/laravel-adminlte
Loading composer repositories with package information
Updating dependencies
Lock file operations: 2 installs, 0 updates, 0 removals
  - Locking almasaeed2010/adminlte (v3.1.0)
  - Locking jeroennoten/laravel-adminlte (v3.6.1)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 2 installs, 0 updates, 0 removals
  - Downloading almasaeed2010/adminlte (v3.1.0)
  - Downloading jeroennoten/laravel-adminlte (v3.6.1)
  - Installing almasaeed2010/adminlte (v3.1.0): Extracting archive
  - Installing jeroennoten/laravel-adminlte (v3.6.1): Extracting archive
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: jeroennoten/laravel-adminlte
Discovered Package: laravel/sail
Discovered Package: laravel/tinker
Discovered Package: laravel/ui
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
74 packages you are using are looking for funding.
Use the `composer fund` command to find out more!

C:\Empresa>php artisan adminlte:install
Basic assets installed successfully.
Configuration file installed successfully.
Translation files installed successfully.
The installation is complete.

```

Fuente: Elaboración propia (2021)

Para poder utilizar las plantillas instaladas y sus componentes en nuestro archivo de vista de nuestras empresas registradas y poder mostrarlos al usuario a través de una pantalla debemos de llamar a los layouts como podemos observar en la Ilustración 33.

Ilustración 33: Vista de empresas registradas

```

@extends ('layouts.admin')
@section ('contenido')
<div class="container">
  <h2>Lista de empresas registradas
  <a href="{{ route('reportePDFEmpresa')}}" target="_blank" type="button" class="btn btn-primary">Reporte Empresa</a>
  <a href="empresa/create"><button type="button" class="btn btn-success float-right">Agregar Empresa</button></a>
</h2>
<nav class="navbar navbar-light float-right">
  <form class="form-inline">
    <input name="searchText" class="form-control mr-sm-2" type="search" placeholder="Buscar por nombre" aria-label="Search">
    <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Buscar</button>
  </form>
</nav>
<table class="table table-hover table-dark">
  <thead>
    <tr>
      <th scope="col">Nombre</th>
      <th scope="col">Direccion</th>
      <th scope="col">Ruc</th>
      <th scope="col">Telefono</th>
      <th scope="col">Correo</th>
      <th scope="col">Descripcion</th>
    </tr>
  </thead>
  <tbody>

```

Fuente: Elaboración propia (2021)

Mediante el uso de las plantillas adminLTE fueron creadas todas las vistas de todos los elementos de nuestra aplicación como ser ABM, menús, index, pantallas para crear y modificar un elemento, simplemente se creó una vista principal el cual fue llamado en otras vistas de manera a minimizar código y evitar código repetido.

Para poder crear el mapa el cual indicará la dirección exacta del usuario mediante su latitud y longitud se utilizó la clase **GoogleMaps\ServiceProvider\GoogleMapsServiceProvider::class**, el cual fue introducido en nuestro formulario de empresas mediante un **div** el cual llama a una función **iniciarMapa** como se muestra en la Ilustración 34 para poder indicar la posición y que este sea guardada en la base de datos al enviar dicho formulario con todos sus datos.

Ilustración 34: Crear Mapa

```

<div class="col-md-12">
  <div class="" id="mapa" style="width: 90%; height:500px"></div>
</div>
<script type="text/javascript">
  function iniciarMapa(){
    var latitud = -27.359391894383084;
    var longitud = -55.84904911875535;
    coordenadas = {
      lng: longitud,
      lat: latitud
    }
    generarMapa(coordenadas);}
  function generarMapa(coordenadas){
    var mapa = new google.maps.Map(document.getElementById('mapa'),
    {
      zoom:18,
      center: new google.maps.LatLng(coordenadas.lat,coordenadas.lng)
    });
    marcador = new google.maps.Marker({
      map: mapa,
      draggable: true,
      position: new google.maps.LatLng(coordenadas.lat, coordenadas.lng)
    });
    marcador.addListener('dragend', function(event){
      document.getElementById("latitud").value = this.getPosition().lat();
      document.getElementById("longitud").value = this.getPosition().lng();
    })
  }
</script>
<script src="https://maps.googleapis.com/maps/api/js?Key=AIzaSyCo4LR5wMXowZhaq-7YmGBhggYih"
</script>

```

Fuente: Elaboración propia (2021)

5.5. Codificación del API.

Con el fin de que nuestra aplicación móvil tenga comunicación con nuestro administrador web fue creado un API.

5.5.1. Configuración de recursos para el API

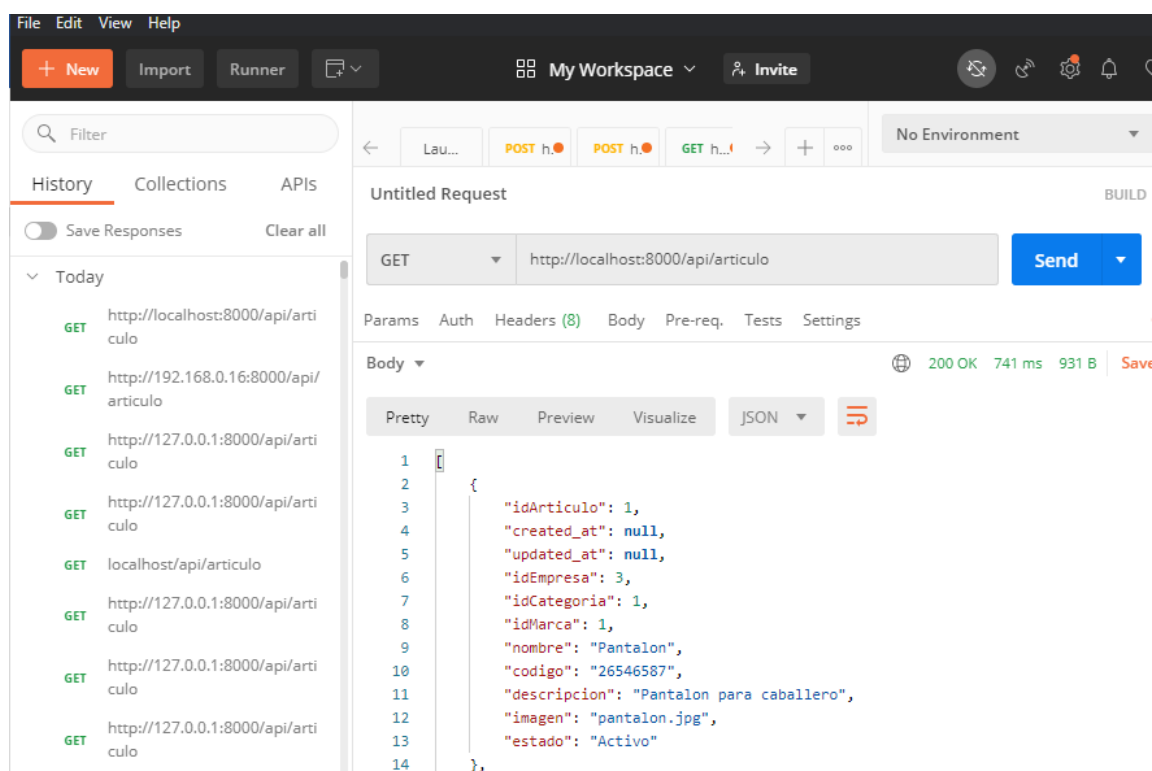
Para poder desarrollar nuestra API utilizamos las mismas herramientas que fueron utilizadas para crear nuestro administrador web como ser xampp, php, atom, y además de todo eso utilizamos postman con el objetivo de poder testear el funcionamiento de nuestra API.

5.5.1.1. Instalación de la plataforma de colaboración Postman

Para poder testear el funcionamiento de nuestra API instalamos la herramienta de testing Postman, para ello nos dirigimos a su sitio oficial y nos registramos para poder descargar su ejecutable y así instalar en nuestra maquina siguiendo todas las instrucciones.

Postman ya que nos permite crear peticiones en APIs, enviar solicitudes REST, SOAP y GraphQL de manera rápida y sencilla además de poseer una interfaz de usuario simple y fácil de manejar como podemos ver en la Ilustración 35.

Ilustración 35: Herramienta de testing Postman



Fuente: Elaboración propia (2021)

5.5.2. Codificación e implementación de librerías para la API

Como en este caso nuestro API tiene como objetivo comunicarse solo con una aplicación móvil y otra web decidimos crear un API sencillo con el fin de obtener todos los artículos con sus imágenes almacenadas en la base de datos de nuestra aplicación web y también información de las empresas con sus coordenadas.

Para poder crear el API de nuestra aplicación primeramente creamos los controladores luego llamamos a ese controlador en el archivo de api tal como podemos ver en la Ilustración 36.

Ilustración 36: Creación del API


```

namespace App\Http\Controllers\Api;

use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use App\Articulo;

class articuloController extends Controller
{
    function all(){
        $articulo = Articulo::all();
        return $articulo;
    }
}

Route::get('/articulo', 'Api\articuloController@all');

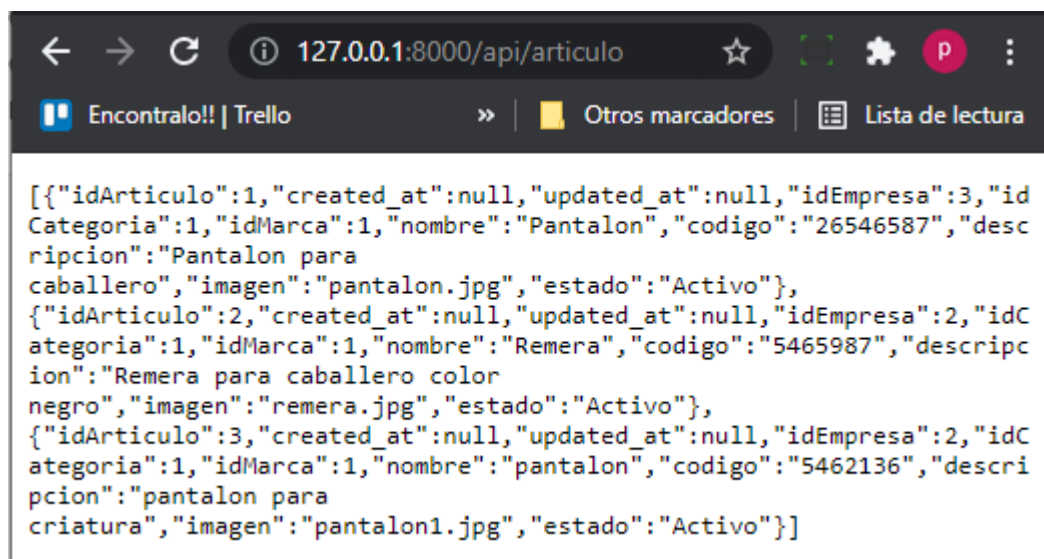
Route::middleware('auth:api')->get('/user', function (Request $request) {
    return $request->user();
});

```

Fuente: Elaboración propia (2021)

Mediante esta clase articuloController se puede obtener los datos de los artículos en formato Json como lo podemos observar en la Ilustración 37, para poder ser utilizados dentro de nuestra aplicación móvil.

Ilustración 37: Artículos en formato Json



```

[{"idArticulo":1,"created_at":null,"updated_at":null,"idEmpresa":3,"idCategoría":1,"idMarca":1,"nombre":"Pantalon","codigo":"26546587","descripcion":"Pantalon para caballero","imagen":"pantalon.jpg","estado":"Activo"},
{"idArticulo":2,"created_at":null,"updated_at":null,"idEmpresa":2,"idCategoría":1,"idMarca":1,"nombre":"Remera","codigo":"5465987","descripcion":"Remera para caballero color negro","imagen":"remera.jpg","estado":"Activo"},
{"idArticulo":3,"created_at":null,"updated_at":null,"idEmpresa":2,"idCategoría":1,"idMarca":1,"nombre":"pantalon","codigo":"5462136","descripcion":"pantalon para criatura","imagen":"pantalon1.jpg","estado":"Activo"}]

```

Fuente: Elaboración propia (2021)

5.6. Codificación de la app móvil

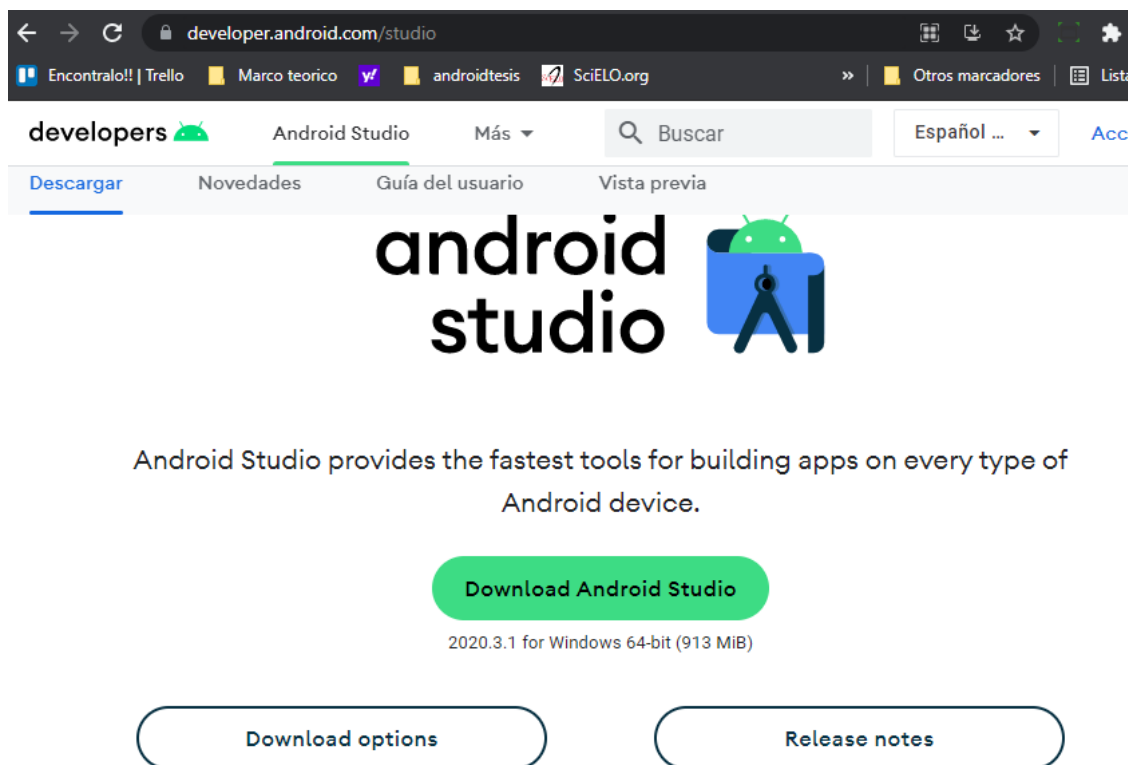
Para poder desarrollar nuestra aplicación móvil se utilizó el lenguaje de programación Android con su interfaz de usuario oficial Android Studio y para poder obtener los datos que mostraremos en nuestra aplicación utilizamos nuestra API.

5.6.1. Configuración de recursos para la app móvil

5.6.1.1. Instalación del IDE Android studio

Para poder instalar el IDE oficial de Android Studio nos dirigimos a su página oficial <https://developer.android.com/studio> y descargamos su ejecutable como se puede ver en la Ilustración 38.

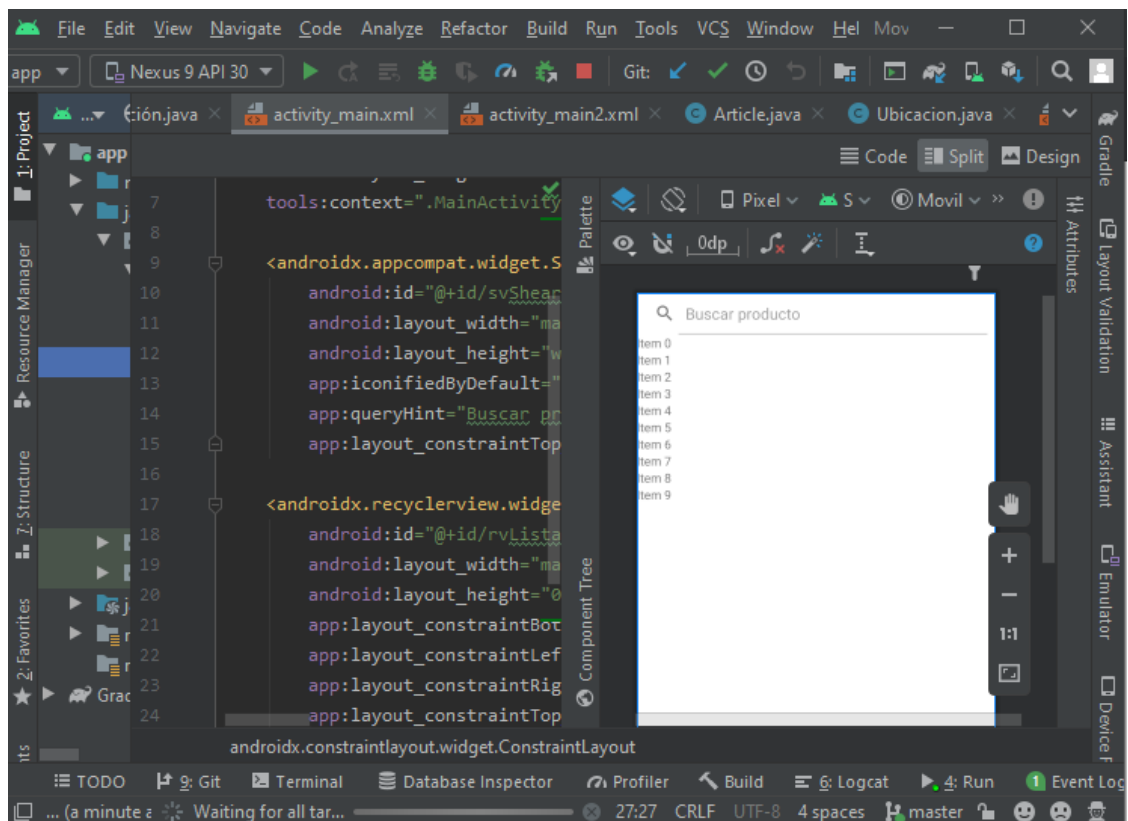
Ilustración 38: Descarga de IDE Android studio



Fuente: Elaboración propia (2021)

Una vez tengamos el ejecutable procedemos a instalar en nuestra maquina siguiendo todas las instrucciones al final obtendremos el IDE de Android Studio que nos proporciona herramientas para poder crear aplicaciones en todo tipo de dispositivos Android como se puede apreciar en la Ilustración 39.

Ilustración 39: IDE Android studio

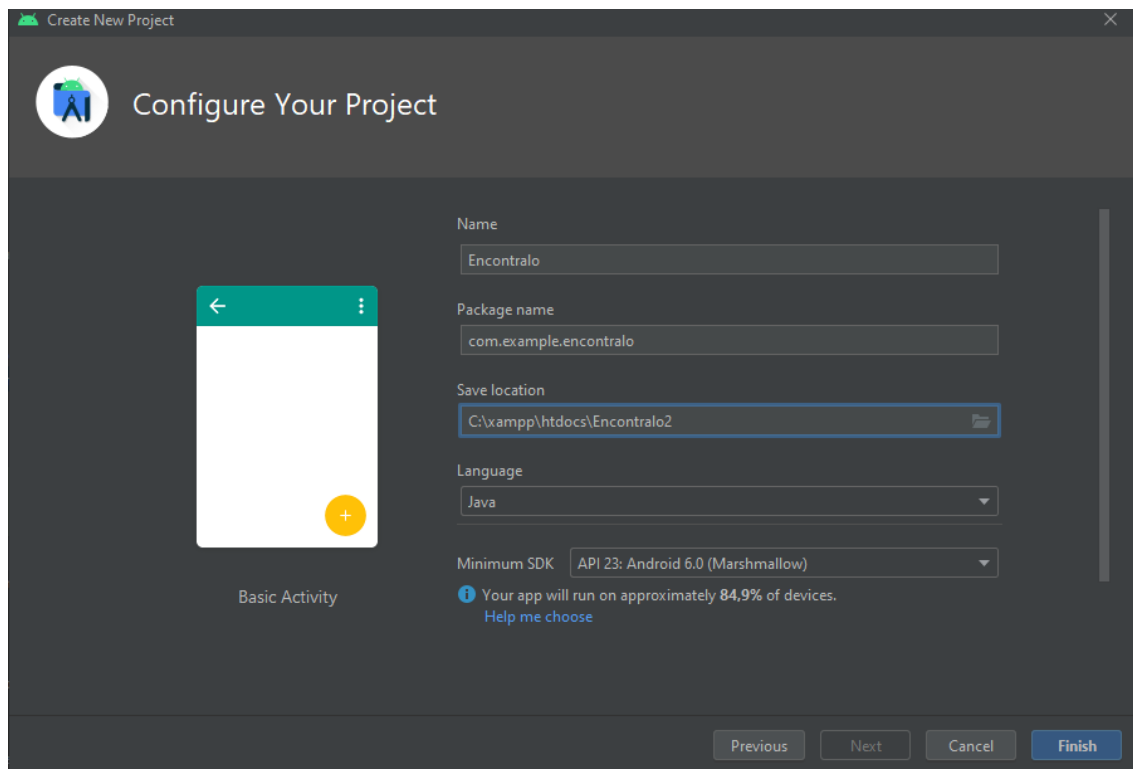


Fuente: Elaboración propia (2021)

5.6.2. Codificación e implementación de librerías para la app móvil

Para poder desarrollar nuestra aplicación móvil primeramente creamos un nuevo proyecto en Android Studio agregando un nombre y seleccionando desde que versión de Android queremos dar soporte o sea desde que versión va ser compatible nuestra aplicación, también indicamos en que carpeta serán guardadas nuestros archivos creados como lo podemos ver en la Ilustración 40.

Ilustración 40: Configuración de proyecto Android

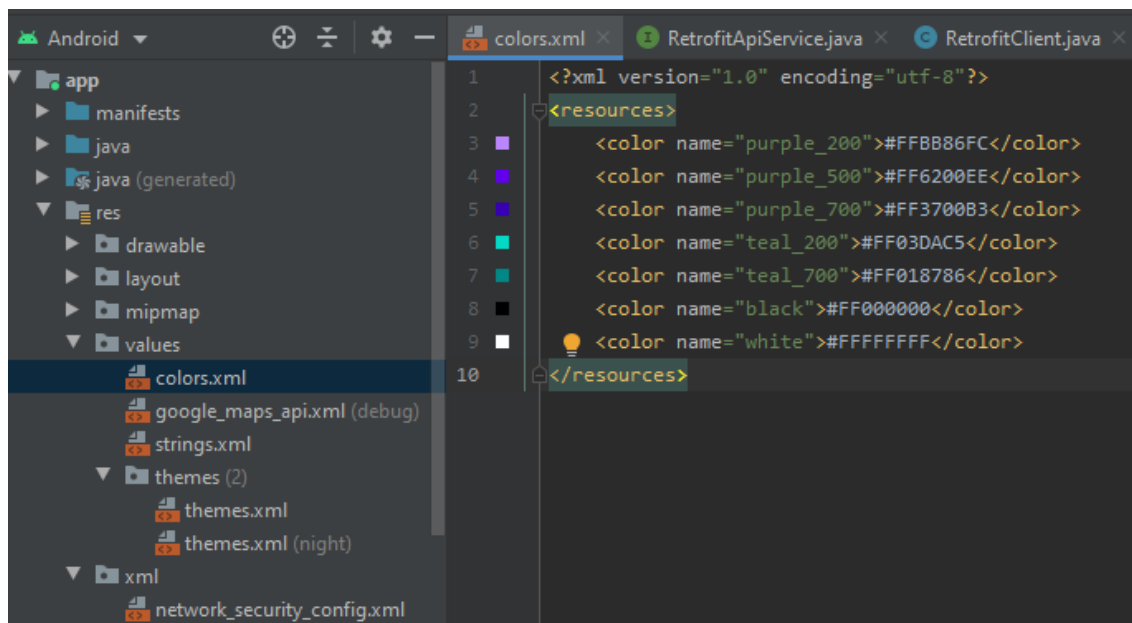


Fuente: Elaboración propia (2021)

Al finalizar la creación del proyecto obtendremos una lista de carpetas y archivos los cuales debemos de configurar para obtener el resultado que queremos.

Primeramente, empezamos configurando los colores de nuestra aplicación para ello dos dirigimos a la carpeta res, seguido a la carpeta values y finalmente al archivo color.xml, haciendo uso de la página www.materialpalette.com seleccionamos dos colores y el material designe nos muestra una visualización de cómo se vería en nuestra aplicación con una paleta de colores, tomamos los tres colores principales que nos muestra y lo cambiamos en nuestro archivo tal como se muestra en la Ilustración 41.

Ilustración 41: Diseño de colores



Fuente: Elaboración propia (2021)

Consumo de servicios del API

Para poder hacer peticiones al API utilizamos retrofit para ello necesitamos agregar dentro de la carpeta de Gradle Scripts en el archivo build.gradle dos librerías las cuales se muestran en la ilustración xx, la primera sería la librería de retrofit en su versión 2.9 y la segunda el convertidor a gson como se muestra en la ilustración 42.

Ilustración 42: Librerías retrofit

```
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.8.0'
```

Fuente: Elaboración propia (2021)

Una vez que tengamos implementado las librerías creamos una carpeta model en cual agregaremos un archivo Article.java el cual tendrá todos los atributos de nuestros artículos con sus getters y setters como podemos observar en la Ilustración xx, esto nos facilitará comunicarnos con el Api ya que utilizando gson se realizará el parse de los objetos como se puede ver en la ilustración 43.

Ilustración 43: Clase Artículo

```

package com.example.movil.model;

import java.io.Serializable;

public class Article implements Serializable {
    private int idArticulo;
    private int idEmpresa;
    private int idCategoria;
    private int idMarca;
    private String nombre;
    private String codigo;
    private String descripcion;
    private String imagen;
    private String estado;

    public Article(int idArticulo, String nombre, String codigo, String descripcion, String imagen) {
        this.idArticulo = idArticulo;
        this.nombre = nombre;
        this.codigo = codigo;
        this.descripcion = descripcion;
        this.imagen = imagen;
    }

    public int getIdArticulo() { return idArticulo; }

    public void setIdArticulo(int idArticulo) { this.idArticulo = idArticulo; }

    public int getIdEmpresa() { return idEmpresa; }

    public void setIdEmpresa(int idEmpresa) { this.idEmpresa = idEmpresa; }
}

```

Android Studio Arct
Update...

Fuente: Elaboración propia (2021)

Luego creamos una carpeta retrofit_data en el cual agregaremos dos archivos uno con el nombre de RetrofitApiService.java y el otro RetrofitClient.java, con estos archivos logramos una conexión al Api tal como podemos observar en las ilustraciones 44 y 45.

Ilustración 44: Clase RetrofitApiService

```

1   package com.example.movil.retrofit_data;
2
3   import com.example.movil.model.Article;
4
5   import java.util.List;
6
7   import retrofit2.Call;
8   import retrofit2.http.GET;
9   import retrofit2.http.POST;
10
11  public interface RetrofitApiService {
12
13      @GET("articulo")
14      Call<List<Article>> getItemsDB();
15

```

Fuente: Elaboración propia (2021)

Ilustración 45: Clase RetrofitClient

```

1   package com.example.movil.retrofit_data;
2
3   import retrofit2.Retrofit;
4   import retrofit2.converter.gson.GsonConverterFactory;
5
6   public class RetrofitClient {
7       private static Retrofit retrofit = null;
8       public static final String URL_BASE = "http://192.168.0.16:8000/api/";
9
10      @
11      public static RetrofitApiService getApiService() {
12          if (retrofit == null) {
13              retrofit = new Retrofit.Builder()
14                  .baseUrl(URL_BASE)
15                  .addConverterFactory(GsonConverterFactory.create())
16                  .build();
17          }
18          return retrofit.create(RetrofitApiService.class);
19      }
20  }

```

Fuente: Elaboración propia (2021)

Creación del RecyclerView utilizando Cardview

Para poder crear un recyclerview con cardview primeramente debemos de descargar sus dependencias como se muestra en le Ilustración 46.

Ilustración 46: Dependencias RecyclerView

```
implementation 'androidx.recyclerview:recyclerview:1.1.0'  
implementation 'androidx.cardview:cardview:1.0.0'
```

Fuente: Elaboración propia (2021)

Luego en nuestro archivo `activity_main.xml` agregamos el `recyclerview` tal como podemos ver en la ilustración 47.

Ilustración 47: Activity

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/rvLista"  
    android:layout_width="match_parent"  
    android:layout_height="0dp"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintLeft_toLeftOf="parent"  
    app:layout_constraintRight_toRightOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/svSearch"  
/>
```

Fuente: Elaboración propia (2021)

También debemos de crear nuestra vista para cada artículo que será mostrado en la pantalla de nuestra app móvil el cual contara con una imagen, un título con el nombre del artículo y su descripción que se verán como tarjetas individuales ya que tendrán una elevación, un borde, un margen, y separación de cada tarjeta, de modo a tener una vista más atractiva, como podemos observar en la Ilustración 48.

Ilustración 48: Código para la vista de los artículos


```

<?xml version="1.0" encoding="UTF-8"?>
<androidx.cardview.widget.CardView
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_height="wrap_content"
    android:layout_width="match_parent"
    android:layout_margin="10dp"
    app:cardCornerRadius="5dp"
    app:cardElevation="10dp"
    android:orientation="vertical">
    <LinearLayout
        android:layout_height="match_parent"
        android:layout_width="match_parent"
        android:orientation="horizontal"
        android:padding="10dp">
        <ImageView
            android:id="@+id/imgArticulo"
            android:layout_height="100dp"
            android:layout_width="100dp"
            android:layout_marginRight="10dp"
            android:scaleType="centerCrop" />
        <LinearLayout
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:orientation="vertical">
            <TextView
                android:id="@+id/tvNombre"
                android:layout_height="wrap_content"
                android:layout_width="match_parent"
                android:textSize="16sp"
                android:textStyle="bold"/>
            <TextView
                android:id="@+id/tvDescripcion"
                android:layout_height="wrap_content"
                android:layout_width="match_parent"
                android:textSize="12sp"/>
            </LinearLayout>
        </LinearLayout>
    </androidx.cardview.widget.CardView>

```

Fuente: Elaboración propia (2021)

Seguidamente creamos un adaptador para ello en una carpeta denominada adapter creamos un archivo con el nombre de RecyclerViewAdapter, en el que agregaremos las clases onCreateViewHolder, este crea un nuevo recyclerview e inicializa los campos para poder ser utilizados por dicho recyclerview y onBindViewHolder sirve para actualizar su contenido, como lo podemos observar en la ilustración 49.

Ilustración 49: RecyclerViewAdapter

```

public RecyclerViewAdapter(List<Article> items, RecyclerViewItemClickListener itemClick){
    this.items = items;
    this.itemClick = itemClick;
    this.originalItems = new ArrayList<>();
    originalItems.addAll(items);
}

@NonNull
@Override
public RecyclerViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.articulo_view, parent,
    return new RecyclerViewHolder(view);
}

@Override
public void onBindViewHolder(@NonNull RecyclerViewHolder holder, int position) {
    Article item = items.get(position);
    // holder.imgArticulo.setImageResource(item.getImagen());

    Picasso.get()
        .load( path: domain_image+item.getImagen())
        .into(holder.imgArticulo);
    holder.tvNombre.setText(item.getNombre());
    holder.tvDescripcion.setText(item.getDescripcion());

    holder.itemView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { itemClick.itemClick(item); }
    });
}

```

Fuente: Elaboración propia (2021)

Para poder mostrar las imágenes de los artículos necesitar descargar la librería de Picasso agregando en el archivo de build.gradle el comando que se muestra en la ilustración x, luego como se muestra en la ilustración anterior agregamos la función Picasso.get con el fin de exponer las imágenes como se muestra en la ilustración 50.

Ilustración 50: Librería Picasso

```

38 implementation 'com.squareup.picasso:picasso:2.71828'

```

Fuente: Elaboración propia (2021)

Funcionamiento del buscador de artículos

Para poder implementar nuestro buscador dentro del recyclerview que ya hemos creado primeramente en nuestro archivo activity_main.xml agregamos un SearchView con su posición y dimensiones como lo podemos observar en la ilustración 51.

Ilustración 51: Cuadro de buscador

```
<androidx.appcompat.widget.SearchView
    android:id="@+id/svSearch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    app:iconifiedByDefault="false"
    app:queryHint="Buscar producto"
    app:layout_constraintTop_toTopOf="parent"/>
```

Fuente: Elaboración propia (2021)

Luego en el archivo MainActivity.java definimos las variables a ser utilizadas por nuestro buscador e inicializamos nuestro buscador, también implementamos los métodos initListener, onQueryTextSubmit, onQueryTextChange que llama a la función de filtrar como lo podemos observar en la ilustración 52.

Ilustración 52: Definición de variables

```

public class MainActivity extends AppCompatActivity implements RecyclerView.Adapter.OnItemClickListener,
    SearchView.OnQueryTextListener {
    private RecyclerView rvLista;
    private SearchView svSearch;
    private RecyclerView.Adapter adapter;
    private List<Article> items;
    private RetrofitApiService retrofitApiService;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initView();
        initValues();
        initListener(); }
    private void initView(){
        rvLista = findViewById(R.id.rvLista);
        svSearch = findViewById(R.id.svSearch);
    }
    private void initValues(){
        retrofitApiService = RetrofitClient.getApiService();
        LinearLayoutManager manager = new LinearLayoutManager( context: this);
        rvLista.setLayoutManager(manager);
       .getItemsSQL();
    }
    private void initListener(){
        svSearch.setOnQueryTextListener(this);
    }
    @Override
    public void onItemClick(Article item) {
        Intent intent = new Intent( packageContext: this, Información.class);
        intent.putExtra( name: "itemDetail", item);
        startActivity(intent);
    }
    @Override
    public boolean onQueryTextSubmit(String query) { return false; }
    @Override
    public boolean onQueryTextChange(String newText) {
        adapter.filter(newText);
    }
}

```

Fuente: Elaboración propia (2021)

Finalmente, en el archivo recyclerViewAdapter.java creamos el método filter el cual será el encargado de filtrar los artículos a partir de una letra ingresada en el campo del buscador e ira limpiando los datos obtenidos a medida se borren letras del buscador el código lo podemos observar en la ilustración 53.

Ilustración 53: Método filtrar artículos

```

public void filter(String strSearch){
    if(strSearch.length() == 0){
        items.clear();
        items.addAll(originalItems);
    }
    else {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.N) {
            List<Article> collect = originalItems.stream()
                .filter(i -> i.getNombre().toLowerCase().contains(strSearch))
                .collect(Collectors.toList());

            items.clear();
            items.addAll(collect);
        }
        else{
            for(Article i:originalItems){
                items.clear();
                if (i.getNombre().toLowerCase().contains(strSearch)){
                    items.add(i);
                }
            }
        }
    }
    notifyDataSetChanged();
}
}

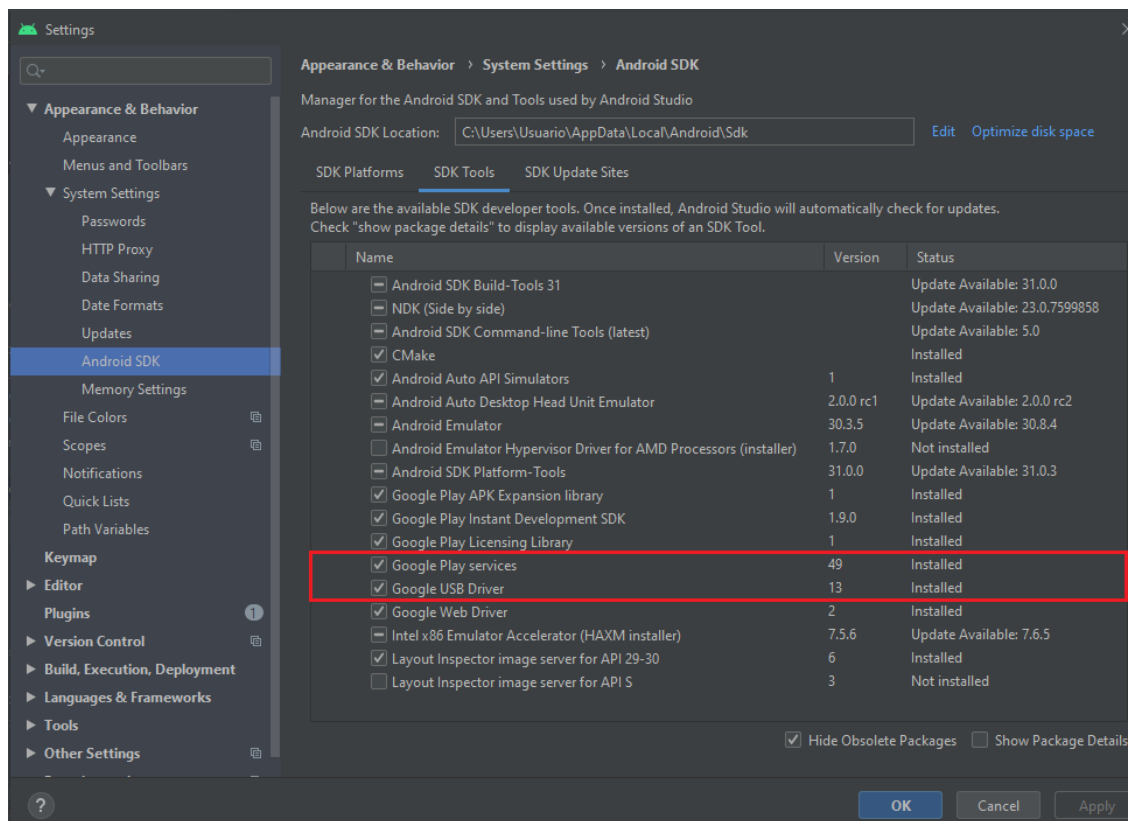
```

Fuente: Elaboración propia (2021)

Agregando mapas al proyecto

Para poder integrar mapas a nuestro proyecto de Android primeramente debemos de instalar los paquetes de Google Play services y como estamos trabajando con Windows también es necesario instalar el paquete de Google USB Driver como lo podemos apreciar en la ilustración 54.

Ilustración 54: Paquetes para integrar mapas

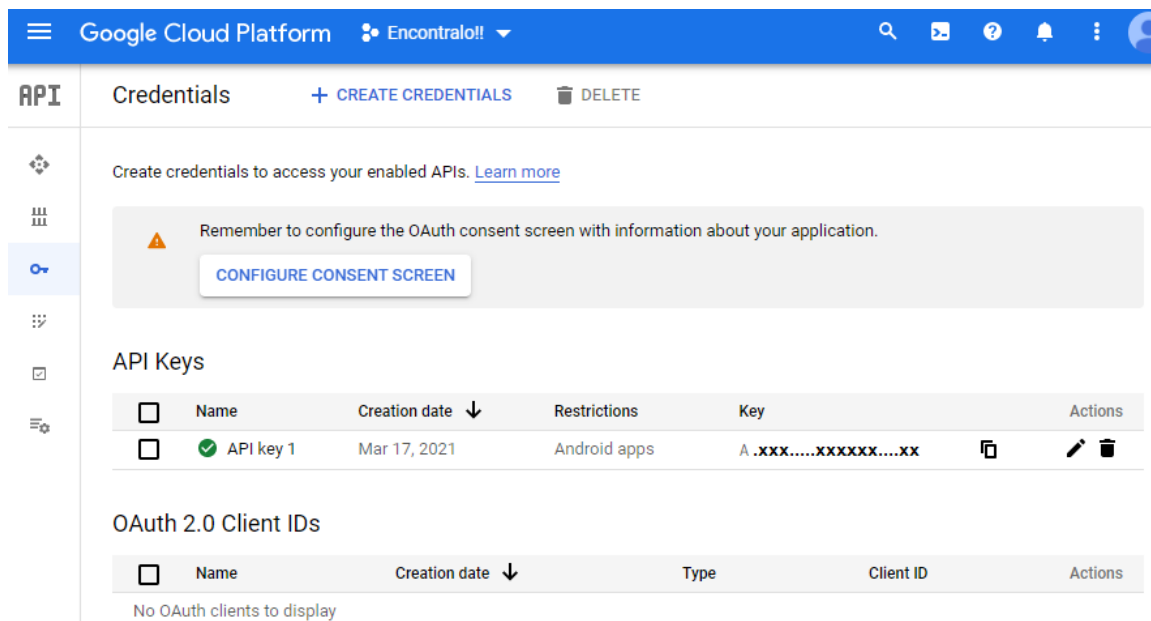


Fuente: elaboración propia (2021)

Para poder agregar a nuestro proyecto estos paquetes simplemente seleccionamos las casillas necesarias y luego presionamos el botón apply de esta manera se instalarán dichos paquetes.

Una vez tengamos los paquetes instalados necesitamos obtener las credenciales de Google Maps para ello nos dirigimos a la página <https://console.cloud.google.com/>, donde creamos un proyecto y nos dirigimos a la parte de credenciales donde nos permitirá crear una nueva credencial seleccionando la clave de API seguidamente debemos de seleccionar las características de nuestra aplicación y ya tendremos nuestra credencial de API de Google Maps como lo podemos apreciar en la ilustración 55, para ser utilizada en nuestro proyecto.

Ilustración 55: Credencial API



Fuente: Elaboración propia (2021)

Para tener un buen funcionamiento de los mapas de google debemos de habilitar el SDK de Maps para Android con el cual agregaremos los mapas a nuestra aplicación Android, el API de Places indicado para obtener información de ciertos lugares, API de codificación geográfica que nos servirá para convertir direcciones a coordenadas geográficas, API de geolocalización en caso de no poder utilizar el GPS nos mostrara información de algunas de las torres de wifi, todas estas API habilitadas lo podemos ver en la ilustración 56.

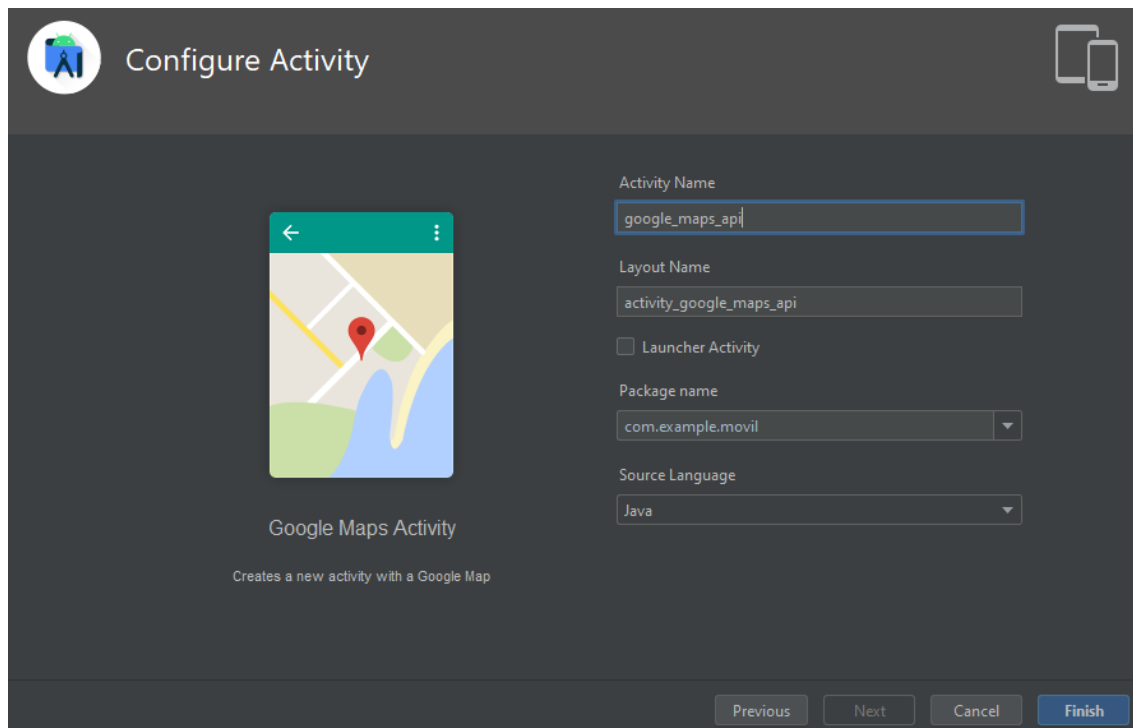
Ilustración 56: API habilitadas

Google Cloud Platform				
API				
API habilitadas				
Seleccione una API para ver los detalles. Las cifras corresponden a los últimos 30 días.				
API ↑	Peticiones	Errores	Latencia promedio (ms)	
API de codificación geográfica	0	0	-	Detalles
API de geolocalización	0	0	-	Detalles
API de JavaScript de Maps	0	0	-	Detalles
SDK de Maps para Android	2	0	-	Detalles
API de Places	0	0	-	Detalles

Fuente: Elaboración propia (2021)

Seguidamente creamos un activity para google maps con el nombre de google_maps_api.xml como se muestra en la ilustración 57.

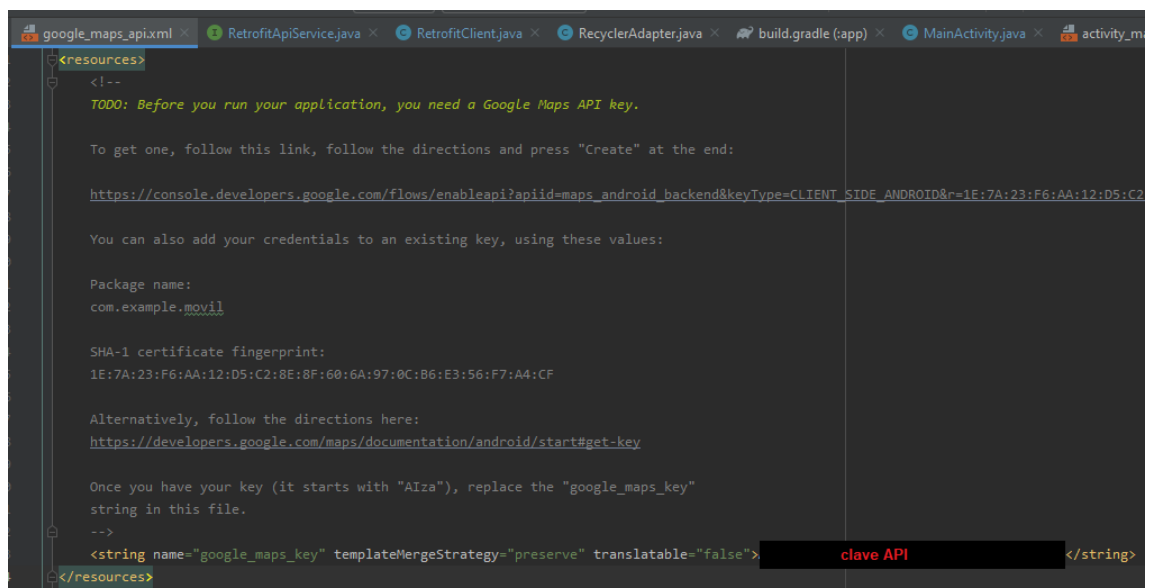
Ilustración 57: Crear Activity para google maps



Fuente: Elaboración propia (2021)

Al presionar en el botón finish nos creara el siguiente código donde ingresaremos la clave API obtenida de la página de <https://console.cloud.google.com/>, como se muestra en la ilustración 58.

Ilustración 58: Código Activity Google Maps



Fuente: Elaboración propia (2021)

Seguidamente en el archivo activity_main2.xml creamos un botón ir al mapa como se muestra en la ilustración 59 y este nos mostrara el mapa con la ubicación de la empresa que posea un artículo buscado.

Ilustración 59: Botón Ir al mapa

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Ir al mapa"
    app:layout_constraintTop_toBottomOf="@+id/tvDescripcionDetail"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    android:onClick="irMapa"
/>
```

Fuente: Elaboración propia (2021)

Luego en nuestro archivo de activity denominado informacion.java creamos un método irMapa como se muestra en la ilustración 60, el cual se encargará de mostrar el mapa en una nueva pantalla.

Ilustración 60: Método ir mapa

```
public void irMapa(View v){
    Intent i = new Intent( packageContext: this, Ubicacion.class);
    startActivity(i);
}
```

Fuente: Elaboración propia (2021)

El botón creado anteriormente llamará al archivo ubicacion.java el cual será el encargado de mostrar el mapa como lo vemos en la Ilustración 61.

Ilustración 61: Método ubicación mapa

```

public class Ubicacion extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_ubicacion);
        // Obtain the SupportMapFragment and get notified when the map is ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
            .findFragmentById(R.id.map);
        mapFragment.getMapAsync( onMapReadyCallback: this);
    }

    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;

        // Add a marker in Sydney and move the camera
        //LatLng sydney = new LatLng(-34, 151);
        LatLng empresa = new LatLng( v: -27.366474560237002, v1: -55.858196318149574);
        mMap.addMarker(new MarkerOptions().position(empresa).title("Marker in Sydney"));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(empresa));
    }
}

```

Fuente: Elaboración propia (2021)

5.7. Prueba de calidad y resultados

Para realizar las pruebas de las funcionalidades de la app móvil, la API y la aplicación web se ejecutaron pruebas de caja negra mediante formularios como se observa en la ilustración 62.

Ilustración 62: Formulario de testeo

FORMULARIO DE TESTEO DE FUNCIONALIDAD		
Responsable:		
Fecha:		
Formulario N°:		
Modulo:	Modulo Empresas (Formulario de registro de datos)	
Funcionalidad:	Registro de empresas de la ciudad de Encarnación	
	Procesos	
	SI	NO
Crea registro		
Modifica registro		
Elimina registro		
Valida correctamente todos los campos del registro		
Genera el registro de auditoria		
Genera la respuesta de confirmación de cada proceso		
Observaciones:		

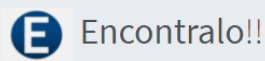
Fuente: Elaboración propia (2021)

Resultados obtenidos en la aplicación web

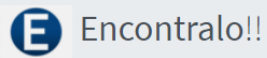
Ilustración 63: Vistas de inicio, iniciar sesión y crear usuario nuevo

[LOGIN](#)[REGISTER](#)

Encontralo!!

[QUIENES SOMOS](#)[EMPRESAS](#)[PRODUCTOS](#)[CONTACTO](#)

Autenticarse para iniciar sesión

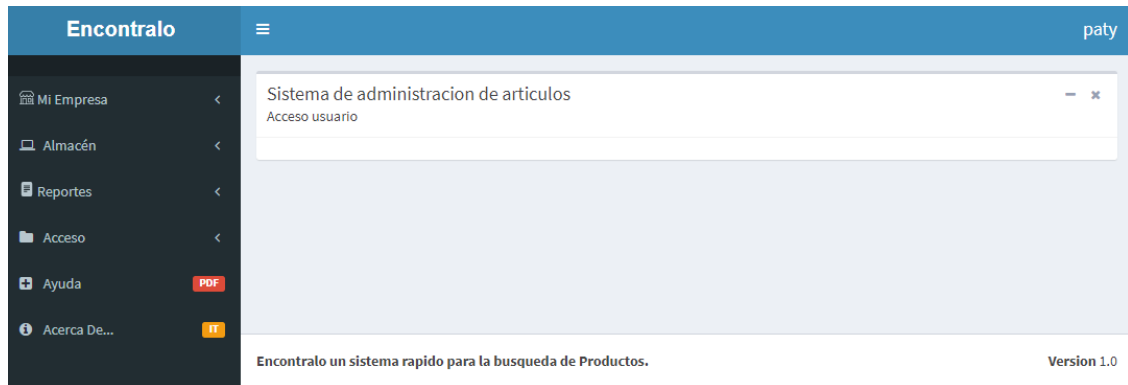
☐ Recordarme[Acceder](#)[Olvidé mi contraseña](#)[Crear una nueva cuenta](#)

Crear una nueva cuenta

[Registrarse](#)[Ya tengo una cuenta](#)

Fuente: Elaboración propia (2021)

Ilustración 64: Vista de acceso de usuario







Fuente: Elaboración propia (2021)

Ilustración 65: Vistas de listas de artículos, crear y modificar artículos

Lista de Articulos registrados [Reporte Articulo](#) [Agregar Articulo](#)


Buscar por nombre

Nombre	Categoria	Marca	Empresa	Codido	Descripcion	Imagen	Estado	Opciones
chort para caballero	Ropas	Adidas	Ropería Patricia2	54261789			Activo	Editar Eliminar
pantalon	Ropas	Adidas	Ropería Patricia2	55226812	pantalon para criatura		Activo	Editar Eliminar
Remera	Ropas	Adidas	Ropería Patricia2	5465987	Remera para caballero color negro		Activo	Editar Eliminar
Pantalon	Ropas	Adidas	Ropería Patricia2	26546587	Pantalon para caballero		Activo	Editar Eliminar

Agregar Articulo

Empresa <input type="text" value="Ropería Patricia"/>	Nombre <input type="text" value="Ingrese el nombre del articulo"/>
Categoria <input type="text" value="Ropas"/>	Marca <input type="text" value="Adidas"/>
Codigo <input type="text" value="Ingrese su codigo"/>	Descripcion <input type="text" value="Ingrese su descripcion"/>
Imagen <input type="button" value="Seleccionar archivo"/> Ningún archivo seleccionado	
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>	

Editar artículo

Nombre	<input type="text" value="pantalon"/>	Categoría	<input type="text" value="Ropas"/>
Marca	<input type="text" value="Adidas"/>	Empresa	<input type="text" value="Ropería Patricia2"/>
Código	<input type="text" value="55226812"/>	Descripción	<input type="text" value="pantalon para criatura"/>
Imagen			
<input type="button" value="Seleccionar archivo"/> Ningún archivo seleccionado			
			
<input type="button" value="Guardar"/> <input type="button" value="Cancelar"/>			

Fuente: Elaboración propia (2021)

Ilustración 66: Reporte de artículos

Reporte Artículo

Nombre	Categoría	Marca	Empresa	Código	Descripción	Estado
Pantalon	Ropas	Adidas	Ropería Patricia2	26546587	Pantalon para caballero	Activo
Remera	Ropas	Adidas	Ropería Patricia2	5465987	Remera para caballero color negro	Activo
pantalon	Ropas	Adidas	Ropería Patricia2	55226812	pantalon para criatura	Activo
chort para caballero	Ropas	Adidas	Ropería Patricia2	54261789		Activo

Fuente: Elaboración propia (2021)

Ilustración 67: Vistas de listas de empresas registradas, crear y editar empresas

Lista de empresas registradas

Nombre	Dirección	Ruc	Teléfono	Correo	Descripción	
Ropería Patricia	Circuito comercial	564263-2	0981546897	paty@gmail.com	Venta de ropas en general	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
Ropería Patricia2	Circuito Comercial	5246597-4	0985654687	paty2@gmail.com	Venta de ropas para niños	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Agregar Empresa

Nombre

Ingrese el nombre de la empresa

Direccion

Ingrese la direccion de la empresa

Ruc

Ingrese el ruc de la empresa

Telefono

Ingrese el telefono de la empresa

Correo

Ingrese el correo de la empresa

Descripcion

Ingrese su descripcion

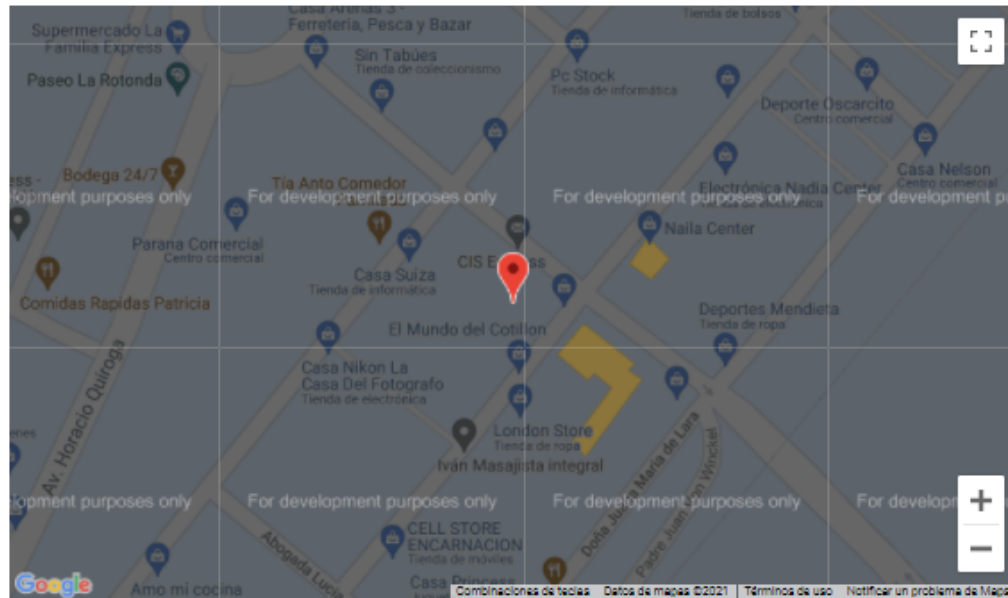
Latitud

Ingrese su latitud

Longitud

Ingrese su longitud

Mover el marcador para guardar la ubicación de su local



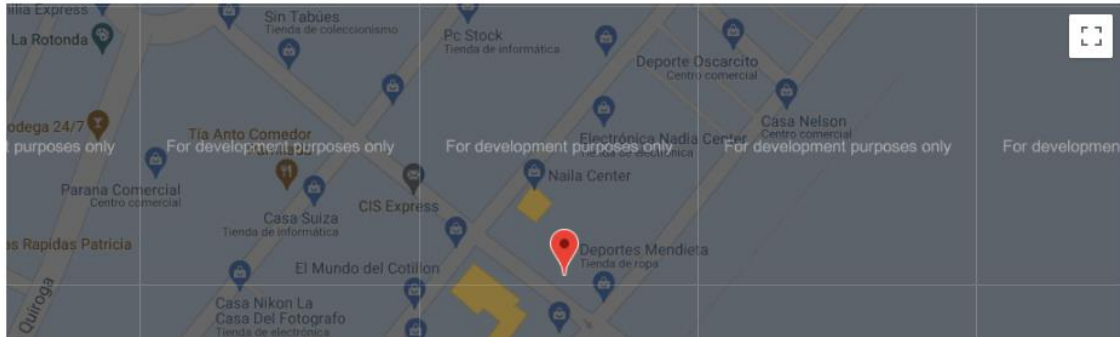
Aceptar

Cancelar

Editar Empresa

Nombre	Direccion
<input type="text" value="Roperia Patricia"/>	<input type="text" value="Circuito comercial"/>
Ruc	Telefono
<input type="text" value="564263-2"/>	<input type="text" value="0981546897"/>
Correo	Descripcion
<input type="text" value="paty@gmail.com"/>	<input type="text" value="Venta de ropas en general"/>
Latitud	Longitud
<input type="text" value="-27.35953006077587"/>	<input type="text" value="-55.848282006977094"/>

Mover el marcador para cambiar la ubicación de su local



Fuente: Elaboración propia (2021)

Ilustración 68: Vista de reporte de empresas registradas

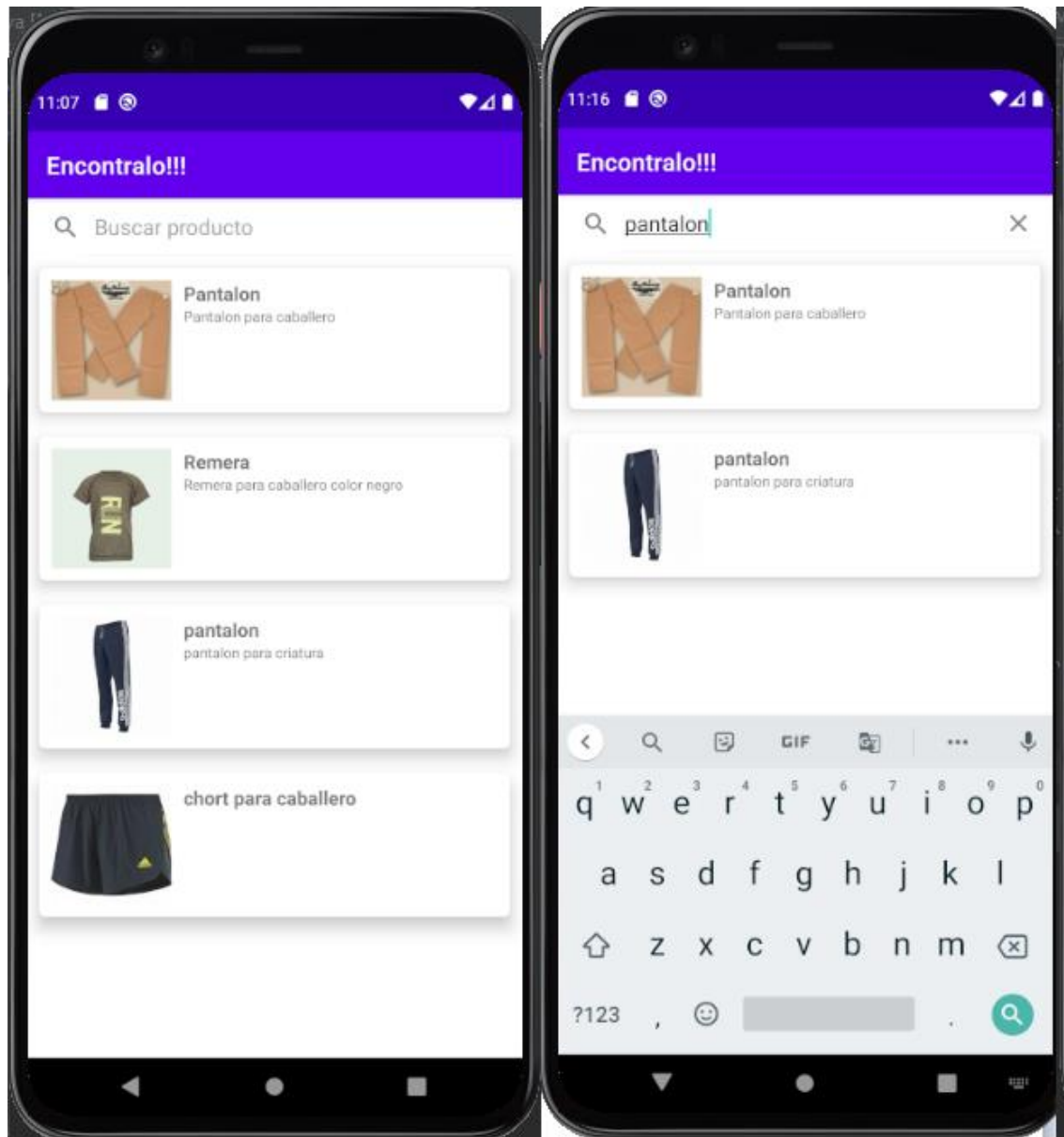
Reporte Empresa

Nombre	Direccion	Ruc	Telefono	Correo	Descripcion
Roperia Patricia	Circuito comercial	564263-2	0981546897	paty@gmail.com	Venta de ropas en general
Ropería Patricia2	Circuito Comercial	5246597-4	0985654687	paty2@gmail.com	Venta de ropas para niños

Fuente: Elaboración propia (2021)

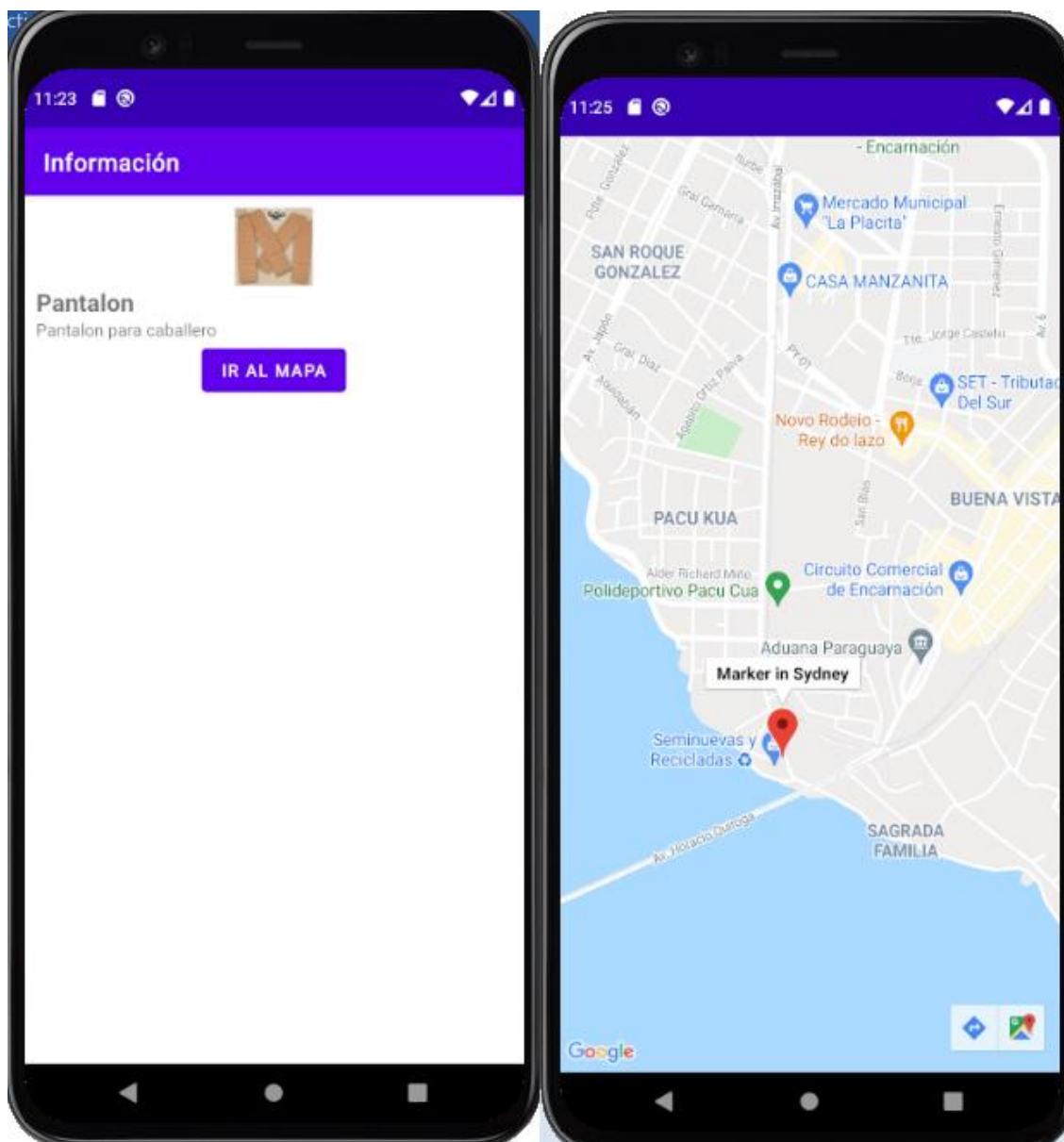
Resultados obtenidos en la aplicación móvil

Ilustración 69: Vistas del buscador de productos



Fuente: Elaboración propia (2021)

Ilustración 70: Vistas ir al mapa



Fuente: Elaboración propia (2021)

Estos fueron los resultados obtenidos durante el proceso de elaboración tanto de la aplicación web como la aplicación móvil.

6. CONCLUSIÓN

Durante el proceso de recolección de datos, se describió que la ciudad de Encarnación es caracterizada por tener como una de sus fuentes de ingresos al comercio minorista. Los consumidores finales en su afán de obtener la mejor relación de precio-calidad, realizan estudios de mercados con las herramientas a su disposición para así poder satisfacer sus necesidades. Con el cierre de las fronteras el sector comercial Encarnaceno fue la más afectada ya que gracias el paso fronterizo entre Encarnación-Posadas, argentinos y otros turistas tenían gran afluencia en ese sector y con el cierre de la frontera muchos locales comerciales sufrieron una escasez de ventas realizadas. Teniendo que reinventarse con otras opciones y buscar soluciones que estén a su alcance. Por esta razón se vio la necesidad de crear una aplicación móvil para la geolocalización de negocios de la ciudad de Encarnación que vendan un determinado producto. De manera que el usuario tenga la comodidad de buscar un producto desde su aplicación móvil y facilitarle las coordenadas de los locales que tienen disponibles dichos productos.

Para poder desarrollar tanto la app móvil, la app web como la API se utilizó la metodología de desarrollo híbrida basado en Kanban y RUP, mediante Kanban se creó un tablero con las distintas tareas para poder dar seguimiento al proyecto y así trabajar de una manera más rápida y eficiente.

Para poder desarrollar la aplicación móvil primeramente fue creada una API mediante la cual se conectó la app móvil con la app web de manera a disponer todos los artículos ingresados a esta página web con el nombre de las empresas y su ubicación.

Se diseñó una interfaz de administrador en la cual fueron introducidas todos los artículos con el nombre y ubicación de la empresa, los cuales mediante la API todos estos datos fueron extraídos en formato JSON y luego mostrados en la app móvil.

Se desarrolló una app móvil la cual cuenta con un botón buscador con la cual el usuario puede insertar el nombre de cualquier producto que desea buscar y este es capaz de mostrar dicho producto con sus características e imágenes.

Las pruebas del correcto funcionamiento de las funcionalidades tanto de la app móvil, la app web como la API se realizó utilizando los formularios provistos por la metodología RUP, realizando pruebas constantes aplicadas durante el proceso de desarrollo de las diferentes aplicaciones.

Como recomendaciones y líneas futura de investigación se propone:

- Extender el trabajo a otras ciudades del departamento de Itapúa ya que este proyecto está destinado únicamente a la zona del circuito comercial de Encarnación, pero podría ser aplicado a otras ciudades con algunos ajustes necesarios.
- Agregar un sistema de chat para poder comunicarse directamente con la empresa.
- Crear un registro de usuarios para la app móvil, ya que en este caso solo se creó el registro de usuarios para la app web.
- Permitir a los usuarios realizar pequeñas encuestas o recomendaciones de manera que las empresas de la ciudad de Encarnación conozcan sus necesidades e inquietudes acerca de los productos buscados.

6. REFERENCIAS BIBLIOGRÁFICAS.

(s.f.). Obtenido de <https://www.monchis.com.py/>

ADN Paraguay . (1 de Agosto de 2016). *adn POLITICO*. Obtenido de Hay más celulares que habitantes en Paraguay: <https://www.adndigital.com.py/hay-mas-celulares-que-habitantes-en-paraguay/#:~:text=Seg%C3%BAn%20empresa%20que%20realiz%C3%B3,el%20117%25%20de%20la%20poblaci%C3%B3n.>

Alcaraz, D. (Junio de 2017). *APORTACIONES A LA GEOLOCALIZACIÓN EFICIENTE*. Obtenido de <https://accedacris.ulpgc.es/handle/10553/41793?mode=full>

Alibaba.com. (2020). Obtenido de https://buyer.alibaba.com/?spm=a2700.8293689-es_ES.scGlobalHomeHeader.696.573f1061MQcie1&tracelog=hd_serv_rs

Alvarez, R. (24 de 09 de 2001). *desarrolloweb.com*. Obtenido de Introducción al HTML: <http://www.desarrolloweb.com/articulos/534.php>

Amazon. (2021). Obtenido de <https://www.aboutamazon.com/about-us>

Aragón, S. R. (2018). *IOT – Posicionamiento en mapas de Google empleando bases de datos y coordenadas GPS*. Obtenido de <https://idus.us.es/handle/11441/80872>

Belloch, C. (s.f.). *RECURSOS TECNOLÓGICOS (TIC)*. Obtenido de <http://eduteka.icesi.edu.co/gp/upload/NRTLogo1.pdf>

Bootstrap, E. d. (30 de Marzo de 2021). *Cree sitios rápidos y receptivos con Bootstrap*. Obtenido de <https://getbootstrap.com/>

Bose, S., Kundu, A., & Mukherjee, M. (2018). *A COMPARATIVE STUDY: JAVA VS KOTLIN PROGRAMMING IN ANDROID APPLICATION DEVELOPMENT*. Obtenido de <https://pdfs.semanticscholar.org/c0ee/43434064520cdde7222318bf6c4d2db69177.pdf>

Cantero, A. A. (26 de agosto de 2013). *DINÁMICA FRONTERIZA ENTRE POSADAS (ARGENTINA) Y ENCARNACIÓN (PARAGUAY)*. Obtenido de http://www.gaea.org.ar/contribuciones/Contribuciones2013/Cantero_2013.pdf

Capasso, A. C. (2020). *YO TRABAJO EN LA FRONTERA. Dossier Experiencias, tensiones y disputas sobre las condiciones laborales de los trabajadores fronterizos*. Obtenido de [DISPUTAS Y RIVALIDADES DE FRONTERA. COMPARACIÓN DE LAS FUERZAS COMPETITIVAS ENTRE EL SECTOR COMERCIAL DE LAS CIUDADES DE POSADAS \(ARGENTINA\) Y ENCARNACIÓN \(PARAGUAY\):](http://edicionesfhyics.fhyics.unam.edu.ar/index.php/dytelf/article/view/317/252)
<http://edicionesfhyics.fhyics.unam.edu.ar/index.php/dytelf/article/view/317/252>

Castro, L. (2015). *About en Español*. Obtenido de ¿Qué es Java?:
<http://aprenderinternet.about.com/od/Glosario/g/Que-Es-Java.htm>

CECARM. (2013). Obtenido de [https://www.cecarm.com/Guia_La_busqueda_del_producto_y_el_proceso_de_compra_e
n_un_negocio_virtual.pdf-6538](https://www.cecarm.com/Guia_La_busqueda_del_producto_y_el_proceso_de_compra_en_un_negocio_virtual.pdf-6538)

Christian Chena SA. (2021). *Turoga*. Obtenido de <https://apprecs.com/android/com.turoga.turoga/turoga>

Developers. (s.f.). Recuperado el 07 de 05 de 2016, de Android Estudio general:
<http://developer.android.com/tools/studio/index.html>

Díaz, L. G. (2019). *BIG DATA APLICADO AL ANÁLISIS DE APLICACIONES*. Obtenido de <http://calderon.cud.uvigo.es/bitstream/handle/123456789/320/20191108%20Resumen%20BIG%20DATA%20APLICADO%20AL%20ANALISIS%20DE%20APLICACIONES%20ANDROID%20GUTIERREZ%20nueva%20plantilla.pdf?sequence=1&isAllowed=y>

Fernández, R. (16 de septiembre de 2020). *statista*. Obtenido de <https://es.statista.com/estadisticas/636569/usuarios-de-telefonos-inteligentes-a-nivel-mundial/>

Foundation, G. C. (31 de Marzo de 2021). *CGFGlobal*. Obtenido de ¿Qué son las aplicaciones?:
<https://edu.gcfglobal.org/es/cultura-tecnologica/que-son-las-aplicaciones-o-programas/1/>

Foundation, G. C. (02 de Abril de 2021). *GCFGGlobal*. Obtenido de ¿Qué son las aplicaciones web?: <https://edu.gcfglobal.org/es/informatica-basica/que-son-las-aplicaciones-web/1/>

G., T. G. (s.f.). *Tipos de Investigación*. Obtenido de Tipos de Investigación: <http://tgrajales.net/investipos.pdf>

GanttProject.org. (04 de 08 de 2014). *uptodown*. Obtenido de Gantt Project: <http://gantt-project.uptodown.com/>

Garita Araya, R. A. (2013). *Tecnología Móvil: desarrollo de sistemas y aplicaciones para las Unidades de Información*. Obtenido de <https://www.redalyc.org/articulo.oa?id=476848738003>

genbeta:dev. (25 de junio de 2011). *Breve introducción a CSS3*. Obtenido de Breve introducción a CSS3: <http://www.genbetadev.com/desarrollo-web/breve-introduccion-a-css3>

genbeta:dev. (09 de 01 de 2014). *NetBeans*. Obtenido de NetBeans: <http://www.genbetadev.com/herramientas/netbeans-1>

Gilibets, L. (07 de 13 de 2013). *Blog de IEBSchool*. Obtenido de Qué es la metodología Kanban y cómo utilizarla: <http://comunidad.iebschool.com/iebs/general/metodologia-kanban/>

Git. (30 de Marzo de 2021). Obtenido de <https://git-scm.com/>

GitHub. (03 de Marzo de 2021). *ATOM*. Obtenido de <https://atom.io/>

GitHub Guides. (24 de Julio de 2020). Obtenido de Hola Mundo: <https://guides.github.com/activities/hello-world/>

GONZALEZ, A. N. (08 de 02 de 2011). *xatakandroid*. Obtenido de ¿Qué es Android?: <http://www.xatakandroid.com/sistema-operativo/que-es-android>

González, R. M. (11 de 12 de 2018). *Marketing XXI*. Obtenido de Concepto de producto: <https://www.marketing-xxi.com/concepto-de-producto-34.htm>

Google Maps Platform. (02 de Abril de 2021). Obtenido de Documentación:

https://developers.google.com/maps/faq/?hl=es&_ga=2.183069222.1487143879.1617152052-1285008333.1616020731#whatis

Hat, R. (05 de Abril de 2021). ¿QUÉ ES UNA API? Obtenido de

<https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces#:~:text=Una%20API%20es%20un%20conjunto,de%20saber%20c%C3%B3mo%20est%C3%A1n%20implementados.>

Herazo, L. (2021). TECNOLOGÍA. Obtenido de ¿QUÉ ES UNA APLICACIÓN MÓVIL?:

<https://anincubator.com/que-es-una-aplicacion-movil/>

Ibañez Asensio, S., Gisbert Blanquer, J. M., & Moreno Ramón, H. (s.f.). SISTEMA DE

[COORDENADAS GEOGRÁFICAS. Obtenido de https://riunet.upv.es/bitstream/handle/10251/8931/Coordenadas%20geogr%C3%A1ficas.pdf](https://riunet.upv.es/bitstream/handle/10251/8931/Coordenadas%20geogr%C3%A1ficas.pdf)

Instituto Nacional de Estadística . (2019). Obtenido de Tecnología de Información y

[Comunicación en el Paraguay\(TIC\). EPHC 2015-2018: https://www.ine.gov.py/Publicaciones/Biblioteca/Tics/documento%20TICS.%20final.pdf](https://www.ine.gov.py/Publicaciones/Biblioteca/Tics/documento%20TICS.%20final.pdf)

Iñaki Chaves, J. I. (2014). Universidad, ciudadanía y TIC. Obtenido de

<https://www.redalyc.org/articulo.oa?id=515551536009>

JosueMolina. (01 de 09 de 2015). mozilla. Obtenido de HTML5:

<https://developer.mozilla.org/es/docs/HTML/HTML5>

Kekanto. (s.f.). Kekanto. Recuperado el 18 de 05 de 2016, de sobre Kekanto:

<https://kekanto.com.br/>

Kniberg, H., & Skarin, M. (2010). Kanban y Scrum – obteniendo lo mejor de ambos. Obtenido de

https://d1wqtxts1xzle7.cloudfront.net/38261265/KanbanVsScrum_Castellano_FINAL-printed.pdf?1437578501=&response-content-

[disposition=inline%3B+filename%3DKanban_vs_Scrum_Castellano_DRAFT_3_PRINT.pdf&Expires=1619929583&Signature=NtA7llrn8x2lMg6os9C3Z6RKTzOj~](#)

[Kristel Malave Polanco, J. L. \(2011\). “Android” el sistema operativo de Google para dispositivos móviles. Obtenido de https://dialnet.unirioja.es/servlet/articulo?codigo=7165367](#)

[Laboratorio Nacional de Calidad del Software. \(Marzo de 2009\). INGENIERÍA DEL SOFTWARE METODOLOGÍAS Y CICLOS DE VIDA. España: INTECO.](#)

[Londoño, J. H. \(06 de abril de 2005\). Ingeniería de Software. Obtenido de Ingeniería de Software: http://ing-sw.blogspot.com/2005/04/tipos-de-pruebas-de-software.html](#)

[Luna, J. M. \(03 de junio de 2009\). Ingeniero de Gestion. Obtenido de Ingeniero de Gestion: http://ingenierogestion.blogspot.com/2009/06/pruebas-de-caja-negra-y-caja-blanca.html](#)

[María Clara Stavile Bravin, A. P. \(2020\). Fronteras amenazadas. Potencialidades y desafíos del MERCOSUR para responder a la pandemia.](#)

[María Rosa, S. N., Miguel Ángel, S. S., Jacqueline, V. H., & Viviana, P. F. \(s.f.\). El proceso de ocupación del Nuevo Circuito Comercial de Encarnación.](#)

[Martinig & Associates. \(2014\). Methods & Tools. Obtenido de ArgoUML: http://www.methodsandtools.com/tools/tools.php?argouml](#)

[Memmel, R. \(14 de 05 de 2015\). Memmel emprendimientos. Obtenido de Cinco aplicaciones paraguayas para dispositivos móviles muy útiles y recomendadas: http://www.rafamemmel.com/2015/cinco-apps-paraguayas-muy-utiles-y-recomendadas/](#)

[Mercadé, A. \(28 de Abril de 2015\). DEUSTOFORMACION. Obtenido de https://www.deustoformacion.com/blog/marketing-digital/herramientas-geolocalizacion-mas-utilizadas#:~:text=El%20concepto%20de%20geolocalizaci%C3%B3n%20se,la%20utilizaci%C3%B3n%20de%20dispositivos%20m%C3%B3viles.](#)

Metodología de la investigación. (1991). En R. Hernández Sampieri, C. Fernández Collado, & M. d. Baptista Lucio, *Metodología de la investigación 5ta edición* (pág. 656). México.

Ministerio de Tecnologías de la Información y Comunicación. (09 de mayo de 2019). *Acceso y Uso de TIC en Paraguay 2018*. Obtenido de <https://www.mitic.gov.py/materiales/publicaciones/acceso-y-uso-de-tic-en-paraguay-2018>

Monchis. (2021). Obtenido de <https://www.monchis.com.py/>

Mora, S. L. (s.f.). Programación de aplicaciones web: Historia, principios básicos y clientes web. C/.Cottolengo, 25 - San Vicente (Alicante): Club universitario.

Oficina de Coordinación Nacional de Posicionamiento, N. y. (03 de Abril de 2021). *Gps.gov*. Obtenido de Sistema de Posicionamiento Global: [https://www.gps.gov/spanish.php#:~:text=El%20Sistema%20de%20Posicionamiento%20Global%20\(GPS\)%20es%20un%20sistema%20de,civiles%20en%20todo%20el%20mundo.](https://www.gps.gov/spanish.php#:~:text=El%20Sistema%20de%20Posicionamiento%20Global%20(GPS)%20es%20un%20sistema%20de,civiles%20en%20todo%20el%20mundo.)

Oracle. (2014). *Oracle*. Obtenido de MySQL: <http://www.oracle.com/es/products/mysql/overview/index.html>

Padilla Roberto, Q. V. (12 de Marzo de 2015). *Monitoreo y localización de personas extraviadas utilizando Arduino y GSM/GPS*. Obtenido de <https://www.redalyc.org/articulo.oa?id=81642256015>

patricio. (15 de 04 de 2013). *Desarrollando Webs Dinámicas*. Obtenido de Laravel un framework PHP fácil de usar: <http://desarrollandowebdinamicas.blogspot.com.ar/2013/04/laravel-un-framework-php-facil-de-usar.html#more>

PedidosYa. (2021). Obtenido de https://www.pedidosya.com.py/restaurantes/encarnacion?pycat=googlesearch&cmp=sem_bra_PY_NAT_01&ag=pedidosya%20%28e%29&kwd=pedidosya&mt=e&net=g&dev=

[c&sitelink=&br=1&gclid=CjwKCAjwjuqDBhAGEiwAdX2cj7u3scUIZ-lzVak2Y182CXUaWberegF4uEh9YswQztIT_AJo_FlRaxoCAn0QA](https://repositorio.usp.br/handle/2454/44444)

Pilar, H. S. (2020). *ANÁLISIS DEL COMPORTAMIENTO DEL CONSUMIDOR Y DE LOS FACTORES QUE INFLUYEN EN LA DECISIÓN DE COMPRA EN EL SUPERMERCADO EL SÚPER DEL DISTRITO DE CHICLAYO – 2018*. Obtenido de <https://repositorio.uss.edu.pe/bitstream/handle/20.500.12802/7275/Herrera%20Saavedra%20Fiorella%20del%20Pilar.pdf?sequence=1&isAllowed=y>

Pin, J. J. (2020). “*APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE PROCESOS EN LA UNIDAD EDUCATIVA FISCAL MIXTA “DR. JOSÉ VILIULFO CEDEÑO SÁNCHEZ”*”. Obtenido de http://repositorio.unesum.edu.ec/bitstream/53000/2299/1/TESIS_ARGUELLO%20PIN%20JASON%20JONATHAN.pdf

Portal Educativo. (05 de Abril de 2021). Obtenido de Latitud y longitud: <https://www.portaleducativo.net/quinto-basico/686/Latitud-y-longitud>

Postman. (30 de Marzo de 2021). *Postman*. Obtenido de La plataforma de colaboración para el desarrollo de API: <https://www.postman.com/>

S.L., Guiarte Multimedia. (2014). *DesarrolloWeb.com*. Obtenido de Javascript a fondo: <http://www.desarrolloweb.com/javascript/>

Sampieri, D. R. (2010). *METODOLOGÍA DE LA INVESTIGACIÓN Quinta edición*. México: McGRAW-HILL / INTERAMERICANA EDITORES, S.A. DE C.V.

Software, E. (02 de Abril de 2021). *Qué es la geolocalización y cómo funciona*. Obtenido de <https://www.evaluandosoftware.com/la-geolocalizacion-funciona/>

Subirats, J. (03 de 04 de 2014). *fandroides*. Obtenido de QUÉ ES Y PARA QUE SIRVE EL SDK: <https://plus.google.com/+JoanSubiratsLlaveria/posts>

THE TORTOISESVN TEAM. (2015). *TortoiseSVN*. Obtenido de Acerca de TortoiseSVN: <http://tortoisesvn.net/about.html>

Thompson, I. (s.f.). *Definición de Mercado*. Obtenido de

https://d1wqtxts1xzle7.cloudfront.net/55012638/definicion_de_mercado.pdf?1510737639=&response-content-disposition=inline%3B+filename%3DDefinicion_de_Mercado.pdf&Expires=1619282406&Signature=V55AMrgT1wW5dECAIYBZJMrAlNCOp0FUdf8cHB3DwzavGfgkT701t-j7AYIOUB3bg

Vanegas, C. A. (14 de setiembre de 2012). *Desarrollo de aplicaciones sobre Android*. Obtenido de Application Development for Android:

<https://revistas.udistrital.edu.co/index.php/vinculos/article/view/4275/5967>

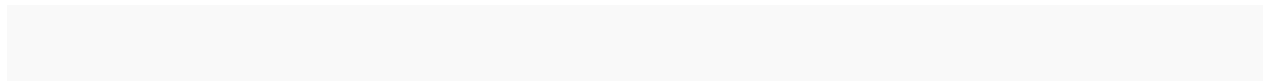
Villa Betancur, A., & Giraldo Plaza, J. E. (abril de 2012). *Automatización de pruebas unitarias de códigos PHP*. Obtenido de <https://www.redalyc.org/pdf/849/84923878021.pdf>

w3schools. (2014). *w3schools.com*. Obtenido de PHP 5 Tutorial:

<http://www.w3schools.com/php/>

XAMPP. (30 de Marzo de 2021). Obtenido de Apache + MariaDB + PHP + Perl:

<https://www.apachefriends.org/es/index.html>



Carta-pedido de aprobación de proyecto

Encarnación....de.....de 2020.

Sr.

Lic. Gabriel Sotelo

Director de la carrera Análisis de Sistemas Informáticos

Universidad Autónoma de Encarnación

Presente

Me dirijo a Usted con el objeto de solicitar la aprobación del título tentativo de la Tesis
**“DESARROLLO DE APLICACIÓN MÓVIL PARA LA BUSQUEDA DE PRODUCTOS
EN NEGOCIOS DE ENCARNACIÓN MEDIANTE GEOLOCALIZACIÓN.”**, para obtener
el título de Licenciatura en Análisis de Sistemas Informáticos, proponiendo como tutor al Ing.
Hugo Sendoa.

La investigación que se desarrollará consistirá en desarrollar una aplicación móvil en la cual se podrá buscar un determinado producto para la geolocalización de las empresas de la ciudad de Encarnación que cuentan con dicho producto y luego mostrará en un mapa estos locales con su información.

La elección del tema es debido a que Encarnación es una ciudad en constante crecimiento y con una visita turística cada vez mayor y no cuenta con una aplicación de búsqueda de locales que cuentan con un determinado producto.

En la espera de una respuesta favorable a la petición, me despido de Usted con mi consideración más distinguida.

Patricia Lorena Fernández Insaurrealde
(0995369092 – patrilorenfer@gmail.com)

Alumno

Hugo Sendoa

Tutor

Presupuesto tentativo de la investigación

Materiales	Monto
Internet	1200000
Notebook	6000000
Papeles	600000
Tinta	600000
Impresora Hp C3180	800000
Transporte	2500000
Gastos de investigación	2500000
Total:	14200000