



UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIAS DE LA ADMINISTRACION

ESCUELA DE INGENIERIA DE SISTEMAS

INGENIERIA DE SISTEMAS Y TELEMATICA

**Título de la Tesis**

GeoVentas

**Elaborado por**

Christian Alberto Merchán Astudillo

**Tutor**

Mba. Fernando Balarezo R.

2015-2016

**Cuenca – Ecuador**

## **Dedicatoria**

Dedico este trabajo principalmente a Dios, por haberme dado la vida y permitirme culminar con esta etapa tan importante de mi formación profesional. A mis padres ya que siempre me brindaron su ayuda, comprensión y cariño en los momentos más difíciles de mi vida llevándome siempre por el camino correcto. A toda mi familia que siempre estuvieron pendientes de mi avance

## **Agradecimiento**

Quiero agradecer de una manera muy sincera a todos mis profesores, compañeros y familiares que de una manera u otra me han brindado su apoyo para culminar con otra etapa de mi vida.

De manera muy especial al Mba. Fernando Balarezo quien no solo me dio la oportunidad de realizar el trabajo de Titulación mediante sus consejos y seguimientos, sino que también de manera desinteresada me enseñó muchas cosas con el transcurso del tiempo, no solo a lo que el trabajo se refiere, sino también a lo de la vida diaria, le agradezco por brindarme su amistad y por la confianza que puso en mí.

De igual manera agradezco al Phd. Francisco Salgado que siempre me brindó su apoyo y seguimiento en los momentos que lo necesitaba, sin su ayuda hubiese sido difícil conseguir este logro en mi vida.

## Índice de Contenido

Capítulo 1.....	3
1.1    Introducción .....	3
1.2    Antecedentes .....	6
1.3    Descripción del Problema .....	7
1.3    Alcance.....	8
1.4    Objetivos .....	8
1.4.1    Objetivo General.....	8
1.4.2    Objetivos Específicos.....	8
Capítulo 2.....	10
2.    Estudio de las Plataformas a Desarrollar.....	10
2.1    Estudio de la tecnología Android .....	10
2.2    Estructura del Sistema Operativo Android.....	12
2.3    Android Studio .....	15
2.4    JSON (JavaScript Object Notation).....	17
2.5    Especificaciones de la tecnología a desarrollar .....	18
2.5.1    Instalación y Configuración de JDK.....	18
2.5.2    Instalación y Configuración de Android Studio .....	23
2.5.3    Instalación de Eclipse. ....	28
2.5.4    Instalación de SDK para Eclipse.....	30
2.5.5    Vincular SDK con Eclipse .....	32
2.6    Elección de Herramienta de Software a Utilizar.....	36
2.6.1    Justificación .....	37
2.6.2    Decisión .....	40
Capítulo 3.....	41

3. Diseño del Software .....	41
3.1 Requerimientos del Software .....	41
3.1.1 Requerimientos Funcionales.....	41
3.1.2 Requerimientos no Funcionales.....	42
3.2 Restricciones .....	43
3.3 Evolución futura de la aplicación.....	44
3.4 Diseño de la Aplicación .....	45
3.4.1 Casos de Uso .....	45
3.4.2 Descripción del Caso de Uso.....	45
3.4.3 Diagrama de Clases .....	47
3.5 Análisis de herramientas de Desarrollo.....	48
3.5.1 Uso de GPS en Android Studio .....	48
3.5.2 Uso de Bases de datos en Android Studio.....	49
3.5.3 Uso de cámaras en dispositivos móviles con Android Studio.....	50
3.5.4 Desarrollo HTML.....	51
Capítulo 4.....	53
4. Implementación, Documentación, Pruebas .....	53
4.1 Implementación de la aplicación.....	53
4.2 Desarrollo de pruebas con dispositivos móviles .....	59
Conclusiones.....	63
Bibliografía .....	66
Anexos .....	68

**Índice de Tablas**

Tabla 1 Características de Android Studio y Eclipse (AcademiaAndroid, 2014) .....	16
Tabla 2 Elección de Software .....	37
Tabla 3 Caso de uso ingresar al sistema .....	45
Tabla 4 Caso de uso Selección de ruta.....	46
Tabla 5 Caso de uso Realizar pedido.....	46
Tabla 6 Caso de uso Ver estadísticas.....	46
Tabla 7 Caso de uso Ayuda .....	47

## Índice de Imágenes

Imagen 1 Versiones de Android (IformAndroid, 2014). .....	11
<b>Imagen 2</b> Estructura Android.....	14
Imagen 3 Descarga de JDK.....	19
Imagen 4 Selección de JDK.....	19
Imagen 5 Selección de destino.....	20
Imagen 6 Comienzo de Instalación.....	20
Imagen 7 Componentes a Instalar.....	21
Imagen 8 Comienzo de Instalación JDK .....	21
Imagen 9 Destino de Instalación JDK .....	22
Imagen 10 Instalación de JDK.....	22
Imagen 11 Fin de Instalación de JDK.....	23
Imagen 12 Descarga Android Studio .....	23
Imagen 13 Inicio de Instalación.....	24
Imagen 14 Selección de Componentes .....	24
Imagen 15 Términos y Licencias .....	25
Imagen 16 Selección de Destino.....	25
Imagen 17 Asignación de RAM .....	26
Imagen 18 Instalación del Producto.....	26
Imagen 19 Descarga de SDK.....	27
Imagen 20 Descarga de SDK fin .....	27
Imagen 21 Descarga de IDE Eclipse .....	28

Imagen 22 Selección de Destino.....	28
Imagen 23 Ventana principal de Eclipse .....	29
Imagen 24 Entorno de desarrollo.....	29
Imagen 25 Página de descarga de SDK .....	30
Imagen 26 Instalación de SDK .....	30
Imagen 27 Descarga de complementos para SDK.....	31
Imagen 28 Selección de complementos a instalar .....	31
Imagen 29 Instalar nuevo Software .....	32
Imagen 30 Selección de nombre y ubicación .....	32
Imagen 31 Selección de complementos .....	33
Imagen 32 Detalles de la instalación .....	33
Imagen 33 Términos y condiciones .....	34
Imagen 34 Confirmar instalación.....	34
Imagen 35 Reiniciar Eclipse .....	35
Imagen 36 Selección de SDK instalado.....	35
Imagen 37 Confirmar vinculación .....	36
Imagen 38 <b>Casos de Uso</b> .....	45
Imagen 39 Diagrama de Clases.....	47
Imagen 40 Login.....	53
Imagen 41 Verificación de usuario .....	53
Imagen 42 Menú principal .....	54
Imagen 43 Menú de locales .....	54
Imagen 44 Selección de proceso.....	55

Imagen 45 Selección de Proceso 2.....	55
Imagen 46 Selección de productos .....	56
Imagen 47 Catálogo .....	56
Imagen 48 Cantidad de producto .....	57
Imagen 49 Forma de pago.....	57
Imagen 50 Confirmar pedido .....	58
Imagen 51 Firma .....	58
Imagen 52 Pedido realizado.....	58
Imagen 53 Datos sincronizados .....	59
Imagen 54 Galaxy Tab 3.....	60
Imagen 55 Galaxy Tab 4.....	60
Imagen 56 Galaxy S4.....	61
Imagen 57 Galaxy S5.....	61
Imagen 58 Galaxy S6.....	62

## RESUMEN

En el presente trabajo se estudiaron los aspectos de mejora y la posterior implementación en una empresa comercial de alcance nacional, en el proceso pedido de productos por parte de la fuerza de ventas: considerando la dispersión de los clientes alrededor del país, basados en la incorporación de dispositivos móviles y los recursos tecnológicos que poseen los sistemas operativos Android.

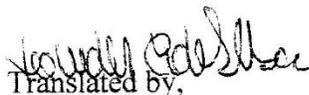
El presente trabajo contempla los aspectos de gestión de campo mediante la implementación de una solución para los dispositivos Android. Para el desarrollo previamente se realizó una evaluación entre varias herramientas de desarrollo, optándose por “Android Studio”. El segundo aspecto contempla el control centralizado para lo cual se desarrolló una página web con información de la gestión de campo, incluyendo información de geolocalización.

### ABSTRACT

This paper studies the aspects related to the improvement and subsequent implementation of a product ordering process for the sales department of a nationwide trading company, taking into account the dispersion of customers around the country. This was done through the incorporation of mobile devices and technological resources that the Android operating systems have.

This work includes field management aspects by implementing a solution for Android devices. Previous to its application, an assessment to several development tools was conducted; deciding on "Android Studio". The second aspect deals with centralized control, for which a web site with field data management, including geolocation information was developed.



  
Translated by,  
Lic. Lourdes Crespo

## Capítulo 1

### 1.1 Introducción

Las tecnologías móviles han causado un significativo cambio en el mundo permitiendo la comunicación inmediata entre los lugares más lejanos del mundo, entre estas tecnologías móviles tenemos a los celulares, tabletas, computadores laptops.

Todos estos dispositivos tecnológicos han experimentado una evolución rápida con el paso del tiempo, ya que como sabemos, hace tan solo algo más de veinte años era imposible imaginar el manejo de una tableta o un celular en el cual podamos acceder a la mayoría de servicios que solo podíamos encontrar en un computador. (Balaguero, 2008)

La tecnología en el ámbito de las comunicaciones para móviles ha evolucionado pasando de tecnologías 1G (Primera Generación), que no llegaban a velocidades de transferencia de 10 kbps, en la década de los 90s hasta la 4G (Cuarta Generación) en la actualidad con velocidades de transferencia de hasta 100 Mbps, toda esta evolución se ha dado en muy poco tiempo. Estos adelantos nos permiten navegar a velocidades muy grandes en la internet, hecho que ayuda a que las personas no solo utilicen sus celulares y tabletas para comunicarse sino también como una herramienta de trabajo. (Ocaña)

El incremento de la funcionalidad de los dispositivos, sumado al incremento de servicios en las redes, ha generado a su vez el desarrollo de sistemas operativos completos, que se orientan a cubrir las características funcionales y de gestión de hardware; entre las principales dominantes

del mercado contamos a IOS, Android, marginalmente Windows Mobile, y otros de poca difusión.

El desarrollo de aplicaciones móviles con sistemas operativos Android ha crecido notablemente debido a la creación de herramientas de fácil acceso que ayudan notablemente a los desarrolladores. Android es un sistema operativo basado en el software libre Linux mismo que cuenta con herramientas gratuitas que nos permiten acceder fácilmente al desarrollo de aplicaciones de forma rápida y fácil.

Los dispositivos en la actualidad cuentan con geo-localizadores, accesos inalámbricos, cámara y otros elementos que fácilmente pueden ser utilizados para realizar aplicaciones que apoyan a la gestión o a una solución de un problema. (Durán, Peinado, & Alosnso, 2015)

El uso de las tecnologías móviles demuestra que las aplicaciones creadas apoyan tanto al entretenimiento como al trabajo, abriendo así un mundo de posibilidades tanto para los desarrolladores como para los dueños de la marca de los dispositivos, al desarrollar novedosas soluciones que tienden a ser sencillas de manejar y productivas al mismo tiempo. (Sabater, 2015)

Se ha demostrado que la mayoría de las personas descargan aplicaciones a sus celulares ya sean estas gratuitas o de pago, lo que demuestra que las personas se apoyan en las nuevas tecnologías tanto en su trabajo como en actividades no laborales, desde aplicaciones comerciales que ayudan a la gestión de una empresa hasta un simple juego o una alarma. (Goggin, 2006; Purcell, 2011)

Los avances tecnológicos también provocan un avance en nuevos tipos de soluciones y como consecuencia de esto tecnologías van quedando obsoletas y dejan de ser utilizadas.

Como podemos observar los dispositivos tecnológicos también cumplen un ciclo de vida ya que en una época fueron lo mejor que existía, pero con el paso del tiempo aparecen nuevas que provocan desactualización en los modelos y tecnologías anteriores. (Ruano)

Es indudable el desarrollo que han sufrido los dispositivos móviles y esto lo podemos observar en dispositivos celulares o tablets con pantallas táctiles y con la incorporación de tecnologías que pocos años atrás no existían, dando esto una gran apertura para la creación de nuevas y novedosas aplicaciones que simplifican tareas cotidianas tanto en nuestros trabajos como en nuestras casas. (Vacas Aguilar, 2007)

El uso del Global Position System (GPS) en los dispositivos móviles nos sirve para determinar una ubicación en cualquier parte del mundo. Este proceso, tecnológicamente se apoya en la detección de señales de dos o más satélites y por medio de una triangulación se obtiene las coordenadas, las cuales luego pueden ser trasladadas e incorporadas por una aplicación y ubicarlas dentro de un mapa para visualizarlas fácilmente. (Masumoto, 1992)

Gracias al avance de las tecnologías GPS, se logra recoger gran cantidad de datos en un solo día con robots que pueden cubrir grandes áreas en poco tiempo. También contamos con la ayuda de los satélites los cuales están actualizando la información en tiempo real. Hoy en día no se puede comparar la cantidad de información que se puede recoger con la ayuda del GPS con los métodos antiguos de cartografía, todos estos datos ayudaran a la aplicación a obtener los datos del lugar exacto de donde se realizó una transacción y mostrar de una forma gráfica con un índice de error bajo (Buzay & Ruiz, 2012).

## 1.2 Antecedentes

La tienda digital de Android es la conocida como Play Store, en donde se encuentran alojadas la mayoría de aplicaciones desarrolladas por los programadores. Existen allí varias Apps enfocadas a dar servicios empresariales y particularmente en lo que se refiere a la fuerza de ventas, pero pocas de ellas cumplen o satisfacen los requerimientos y necesidades empresariales específicas y particulares ya que se enfocan tan solo en la solución de problemas puntuales y dejan otros sin ser atendidos. Es por esta razón que se abre un campo muy amplio para los desarrolladores, quienes pueden aprovechar los avances tecnológicos que se han dado hasta el momento y dar soluciones a empresas reales. Un ejemplo de lo dicho es el poco desarrollo que existe en las aplicaciones que permiten vigilar el desempeño laboral y el correcto uso de las horas de trabajo, como en el caso del agente vendedor también conocido como fuerza de ventas, de quien se requiere un control específico de tareas cada vez que finaliza un proceso de pedido.

Es entonces que gracias al GPS se podrán enviar, no solo los datos de venta sino también la geolocalización de donde fue realizado el pedido lo que apoyará a la empresa a la mejora de varios procesos, como son el área de ventas, facturación, bodegas, y otros.

De igual manera aprovechando las cámaras frontales con las que cuentan los dispositivos se logrará captar la imagen de la persona que está realizando la compra, o de encontrarse cerrado el local, permite confirmar ese hecho por medio la captura digital, lo que permite que el trabajo de la fuerza de ventas pueda ser documentado y posteriormente validado.

Con esta información enviada, las empresas podrán tomar decisiones oportunas mejorando de esta manera su productividad.

### 1.3 Descripción del Problema

Las funcionalidades de los sistemas móviles presentan características que apoyan a las empresas en muchos ámbitos, siendo una de ellas el aporte a aquellas que cuentan con clientes dispersos, cambiando la forma de gestión de ventas de un proceso que era en su totalidad manual a uno sistematizado y eficiente.

En la actualidad existen empresas que utilizan métodos manuales para realizar sus pedidos, siguen un proceso en el cual la fuerza de ventas toma las órdenes de los clientes y estas son acumuladas diariamente. Luego, al terminar la jornada, son entregadas a otro proceso en el cual se tiene que digitalizarlas para que inicien procesos de facturación, logística, contabilización, inventarios, etc.

Todos estos procesos llevan demasiado tiempo, lo que provoca que, desde el pedido hasta la entrega, el cliente tenga que esperar varios días para poder recibir su orden provocando eventuales desabastecimientos del producto. Esta es la razón por la cual se ha visto conveniente sistematizar estos procesos mediante la ayuda de los dispositivos móviles. La mejora será evidente ya que con la APP el mismo momento en el que se realiza el pedido la información será enviada y registrada en una base de datos centralizada, lo que permitirá al resto de procesos tomar decisiones en el mismo momento y no tener que esperar a que la fuerza de ventas entregue sus pedidos de forma personal o manual.

Con la aplicación garantizaremos que la fuerza de ventas está realizando las visitas personalmente a los clientes, ya que en el momento que se realice el pedido se incluirá información georeferenciada, disponiendo de la ubicación de la gestión utilizando el GPS del dispositivo móvil.

### **1.3 Alcance**

Se considerará dentro de los alcances del proyecto:

- Desarrollo de la aplicación “GeoVentas” para sistemas operativos Android y que esta funcione de manera correcta en los distintos dispositivos con esta tecnología.
- La aplicación realizará los pedidos a una base de datos centralizada y enviará los datos del pedido, geolocalización y fotografías correspondientes.
- Desarrollo de una página Web en la que se podrá observar los datos enviados por la aplicación para que el personal realice la toma de decisiones correspondientes a cerca de la fuerza de ventas.

### **1.4 Objetivos**

#### **1.4.1 Objetivo General**

- Diseñar e implementar una aplicación para sistema operativo Android que permita realizar la gestión de ventas y geolocalización.

#### **1.4.2 Objetivos Específicos**

- Sistematizar información sobre el proyecto.
- Desarrollar la aplicación para ventas con auto localización luego de evaluar opciones de software para desarrollo de la aplicación.

- Desarrollar la página web para control de la información enviada por la aplicación.
- Realizar la documentación del proyecto.

## Capítulo 2

### 2. Estudio de las Plataformas a Desarrollar

#### 2.1 Estudio de la tecnología Android

La idea nace de cuatro jóvenes Andy Rubin, Rich Miner, Nick Sears y Chris White que trabajaban en una empresa con el nombre de Android Inc, en el año 2003 en Palo Alto, California. Se disponen a desarrollar un sistema operativo libre basado en Linux para dispositivos con pantalla táctil, su objetivo fue generar una nueva experiencia a los usuarios que venían ya acostumbrados a los dispositivos que contaban con un sistema operativo básico. Google fue el que respaldó económicamente a esta empresa para finalmente en el 2005 comprarla en su totalidad. . (Torres, 2014)

El nombre de Android nace en base a la novela de Philip K. Dick ¿Sueñan los androides con ovejas eléctricas? Que luego paso al cine con el nombre de “*Blade Runner*” en los cuales se hace alusión a un grupo de androides replicantes de modelo Nexus-6.

Android en su mayoría es de código abierto y se encuentra bajo licencias Apache aunque todavía existen algunas partes que son propiedad de Google. En este sistema operativo contamos con una máquina virtual llamada *Dalvik* la cual es encargada de compilar todas las APPs basadas en programación orientada a objetos JAVA.

Hoy existen millones de desarrolladores que se encargan de dar diferentes utilidades a los dispositivos, ya sean estos con fines de lucro o de una manera desinteresada. Es así que podemos

contar con aproximadamente 1.000.000 apps, en sus principios esta tienda de APPs tenía el nombre de Android Market pero luego de varios cambios en el 2008 toma su nombre definitivo de Play Store.

Play Store al momento de subir aplicaciones por los desarrolladores es muy permisivo ya que solo realiza comprobaciones básicas como si no tiene virus, no es software ilegal, o tenga malas intenciones con los usuarios. Lo que no realiza es una validación de su correcto funcionamiento lo cual para muchas personas es una desventaja ya que descargarse una APP y que esta no sirva causa muchos conflictos y con el paso del tiempo así esta haya sido eliminada deja creados muchos ficheros en el dispositivo lo cual causa lentitud y consume memoria. Por otra parte, para los desarrolladores es una ventaja ya que les da la oportunidad de publicar sus creaciones y de esta manera muchas apps se han hecho famosas. (Tecnología y Actualidad, 2014)

Un punto muy curioso de Android es que todas sus versiones toman nombres de postres en inglés y llevan un orden alfabético como podemos observar en la Imagen 1 (López Michelone, 2013), incluyendo en cada versión mejoras y utilidades para los diferentes dispositivos.

**Imagen 1 Versiones de Android (IformAndroid, 2014).**



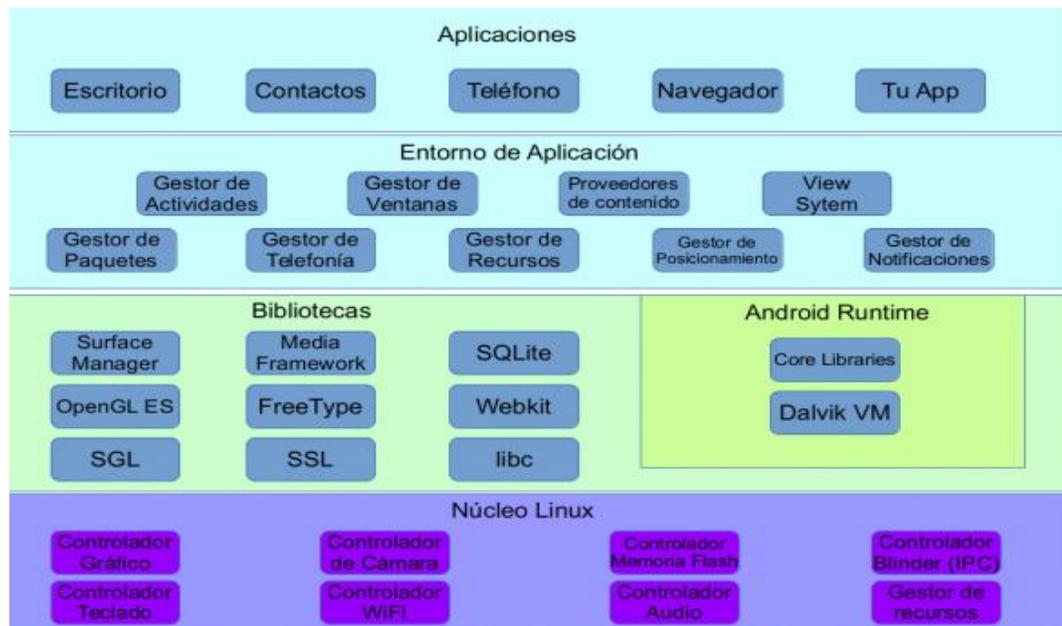
## 2.2 Estructura del Sistema Operativo Android

El sistema operativo Android está formado por un esquema en forma de pila en el cual encontramos el sistema operativo, middleware y las aplicaciones como se puede observar en la **Imagen 2**.

A continuación, se describirá cada uno de sus componentes

- **Aplicaciones:** En este nivel podemos encontrar las aplicaciones que vienen por defecto en el dispositivo, las creadas por los fabricantes y las desarrolladas por terceros.
- **Framework de Aplicaciones:** Encontramos las herramientas necesarias para el desarrollo de todas las aplicaciones que se encuentran en el dispositivo dentro de estas tenemos las siguientes
- **Gestor de Actividades:** Gestiona el ciclo de vida de cada aplicación.
- **Gestor de Ventanas:** Gestiona las ventanas de cada aplicación.
- **Gestor de Telefonía:** Todas las funciones propias del dispositivo, como pueden ser llamadas, mensajes, contactos.
- **Proveedor de Contenido:** Es la encargada de compartir los datos con todas las aplicaciones.
- **View System:** Proporciona los elementos para la construcción de interfaces para los usuarios.
- **Gestor de Posicionamiento:** Proporciona información de geolocalización

- **Gestor de Notificaciones:** Formato para indicar al usuario los eventos que ocurre en las aplicaciones
- **Librerías:** Son el corazón del Sistema operativo ya que dan a Android sus funciones más características, estas librerías fueron escritas en código C/C++.
  - **libc:** Son las cabeceras y sus funciones.
  - **Surface Manager:** Es la que realiza y gestiona todas las ventanas de las aplicaciones a cada momento.
  - **OpenGL/SL y SGL:** Encargada de las librerías graficas 3D y 2D.
  - **Media:** Contiene los códec necesarios para la reproducción de archivos multimedia.
  - **Free Type:** Permite utilizar de una manera sencilla las distintas fuentes.
  - **SSL:** Protocolo para realizar comunicaciones seguras.
  - **SQLite:** Gestión de las bases de datos.
  - **WebKit:** Núcleo principal para el navegador incluido en el dispositivo.
- **Android Runtime:** Aquí encontramos las librerías JAVA y la máquina virtual Dalvik que es en donde se ejecutan todas las APPs.
- **Núcleo Linux:** Contiene los drivers necesarios para que los componentes de hardware puedan ser utilizados, cuando se incluye un nuevo componente de hardware se tienen que crear las librerías o drivers para que este pueda ser reconocido, está basado en un núcleo Linux 2.6. (Ableson, Collins, & Sen, 2009).

**Imagen 2** Estructura Android

### Ventajas

- Es un sistema operativo de código abierto por lo que existen muchas aplicaciones desarrolladas y la mayoría de estas son gratuitas.
- Es multitarea por lo que se pueden tener abiertas varias tareas, se encarga de gestionarlas, ponerlas en modo de suspensión y cerrarlas si se detecta que estas llevan demasiado tiempo inactivas.
- Al ser un sistema operativo perteneciente a Google trae muchos beneficios ya que también posee acceso a sus distintas funciones como es Gmail, Drive, Hangouts y Google+.

## Desventajas

- Al ser multitarea esto implica un alto consumo de batería en los dispositivos a pesar de que las aplicaciones se encuentren suspendidas.
- Existe incompatibilidad de las aplicaciones debido a las apariciones de nuevas versiones del sistema operativo.
- Es necesario descargar aplicaciones de Google Play para optimizar el funcionamiento del sistema operativo.

## 2.3 Android Studio

Android Studio es una herramienta creada por Google y se dio a conocer por primera vez el 16 de mayo del 2013 en el congreso de desarrolladores de Google I/O, posee un entorno de desarrollo Integrado (IDE) basado en IntelliJ IDEA perteneciente a la compañía JetBrains razón por la cual es mucho más rápido y ligero que la herramienta de desarrollo Eclipse ya que indexa todo el proyecto y luego puede tomar decisiones sobre él, está desarrollado en JAVA por lo que posee una licencia Apache 2.0.

Esta herramienta es una alternativa al uso de ECLIPSE proporcionando algunas ventajas, entre ellas tenemos una mayor velocidad al desarrollar versiones que cubren las carencias que Eclipse no lograba solucionar con sus plugins.

El 8 de diciembre del 2014 sale a la luz su primera versión estable con el nombre de Android Studio 1.0 dejando así de dar soporte a los *plugins* necesarios para Eclipse, es por esta razón que Google recomienda usar Android Studio al desarrollar nuevas aplicaciones para dispositivos que cuenten con un sistema operativo Android.

En la Tabla 1 podemos observar la comparativa de algunas de las características principales de Android Studio y Eclipse ADT. (FedeProEx, 2013)

**Tabla 1 Características de Android Studio y Eclipse (AcademiaAndroid, 2014)**

Características	Android Studio	Eclipse
Sistema de construcción	Gradle	ANT
Construir variantes y generación de múltiples APK	Si	No
Refactorización y completado avanzado de código Android	Si	No
Diseño del editor gráfico	Si	Si
Firma APK y gestión de almacén de claves	Si	Si
Soporte para Google Cloud Platform	Si	No
Vista en tiempo real de renderizado de layouts	Si	No
Nuevos módulos en proyecto	Si	No
Editor de navegación	Si	No
Generador de assets	Si	No
Datos de ejemplo en diseño de layout	Si	No
Visualización de recursos desde editor de código	Si	No

#### Ventajas

- Al ser desarrollado por Google, tendrá un constante mantenimiento y actualizaciones.
- Interfaz amigable e intuitiva.

- Facilita el desarrollo gracias a su autocompletado de código Android.
- Fácil creación de empaquetado.
- Más ligero y rápido que otras herramientas de desarrollo.
- Al tener Android Studio ya tenemos todos los elementos necesarios para el desarrollo en Android.

#### Desventajas

- No es una herramienta madura en su totalidad por lo que sigue en desarrollo.
- Nivel de aprendizaje lento para nuevos desarrolladores.
- Puede ser complicado encontrar ayuda con los errores encontrados al momento del desarrollo.

Android Studio al ser una herramienta desarrollada por Google, presenta un futuro prometededor ya que fue pensado en ayudar a los desarrolladores de esta plataforma.

## 2.4 JSON (JavaScript Object Notation)

JSON es un formato ligero utilizado para el intercambio de datos entre distintos lenguajes de programación compatibles como ActionScript, C, C++, C#, ColdFusion, Common Lisp, Del-

phi, E, Eiffel, Java, JavaScript, ML, Objective-C, Objective CAML, Perl, PHP, Python, Rebol, Ruby, Lua y Visual FoxPro, por su simplicidad de escritura y lectura a causado que sea muy utilizado por desarrolladores y sea una alternativa al uso de XML.

Un JSON puede guardar varios valores como son:

- Número (entero o float)
- String (entre comillas simples)
- Booleano (true o false)
- Array (entre corchetes [ ])
- Objeto (entre llaves { })
- Null

## **2.5 Especificaciones de la tecnología a desarrollar**

Para el desarrollo de la aplicación “GeoVentas” se utilizará la herramienta de desarrollo Android Studio, y para la creación de la página Web que servirá para la visualización de la información, PHP y HTML.

A continuación, se explicará la instalación de cada una de las herramientas necesarias para el desarrollo de la aplicación “GeoVentas”.

### ***2.5.1 Instalación y Configuración de JDK***

Para instalar el entorno JDK 8 en el sistema operativo Windows debemos seguir los siguientes pasos:

1. Ingresar a la Web de Oracle <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html> en el cual encontraremos las últimas versiones disponibles del JDK.

### Imagen 3 Descarga de JDK

Java SE Development Kit 8u65		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
<input type="radio"/> Accept License Agreement <input checked="" type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux ARM v6/v7 Hard Float ABI	77.69 MB	jdk-8u65-linux-arm32-vfp-hflt.tar.gz
Linux ARM v8 Hard Float ABI	74.66 MB	jdk-8u65-linux-arm64-vfp-hflt.tar.gz
Linux x86	154.67 MB	jdk-8u65-linux-i586.rpm
Linux x86	174.84 MB	jdk-8u65-linux-i586.tar.gz
Linux x64	152.69 MB	jdk-8u65-linux-x64.rpm
Linux x64	172.86 MB	jdk-8u65-linux-x64.tar.gz
Mac OS X x64	227.14 MB	jdk-8u65-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.71 MB	jdk-8u65-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.01 MB	jdk-8u65-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	140.22 MB	jdk-8u65-solaris-x64.tar.Z
Solaris x64	96.74 MB	jdk-8u65-solaris-x64.tar.gz
Windows x86	181.24 MB	jdk-8u65-windows-i586.exe
Windows x64	186.57 MB	jdk-8u65-windows-x64.exe

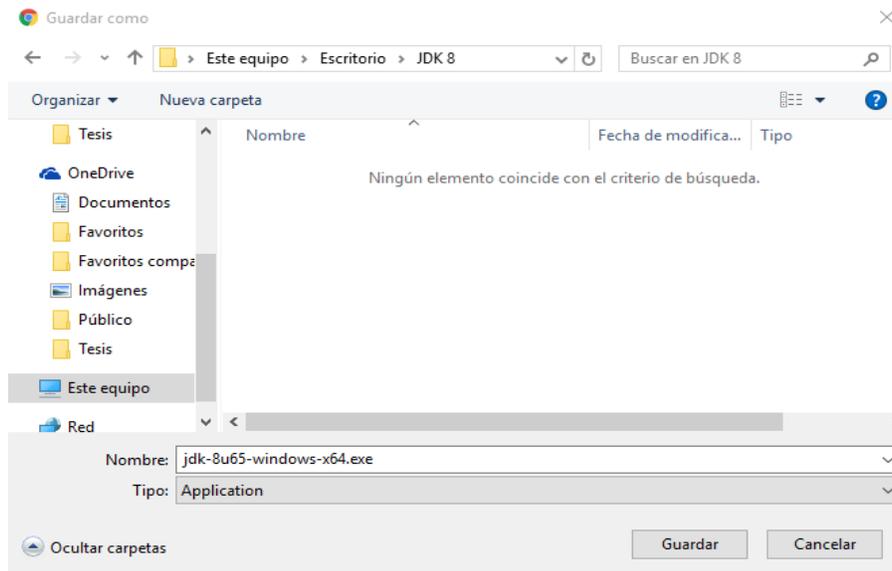
2. Aceptamos los términos de la licencia y seleccionamos la versión necesaria para nuestro sistema operativo.

### Imagen 4 Selección de JDK

Java SE Development Kit 8u65		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux ARM v6/v7 Hard Float ABI	77.69 MB	jdk-8u65-linux-arm32-vfp-hflt.tar.gz
Linux ARM v8 Hard Float ABI	74.66 MB	jdk-8u65-linux-arm64-vfp-hflt.tar.gz
Linux x86	154.67 MB	jdk-8u65-linux-i586.rpm
Linux x86	174.84 MB	jdk-8u65-linux-i586.tar.gz
Linux x64	152.69 MB	jdk-8u65-linux-x64.rpm
Linux x64	172.86 MB	jdk-8u65-linux-x64.tar.gz
Mac OS X x64	227.14 MB	jdk-8u65-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.71 MB	jdk-8u65-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.01 MB	jdk-8u65-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	140.22 MB	jdk-8u65-solaris-x64.tar.Z
Solaris x64	96.74 MB	jdk-8u65-solaris-x64.tar.gz
Windows x86	181.24 MB	jdk-8u65-windows-i586.exe
Windows x64	186.57 MB	jdk-8u65-windows-x64.exe

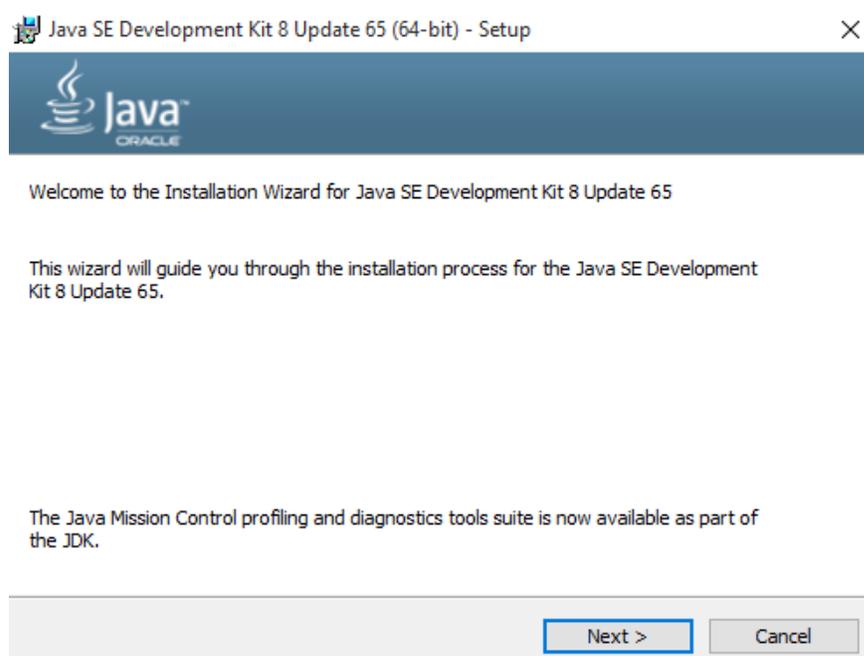
3. Seleccionamos el destino donde queremos descargar el archivo y presionamos el botón guardar.

### Imagen 5 Selección de destino



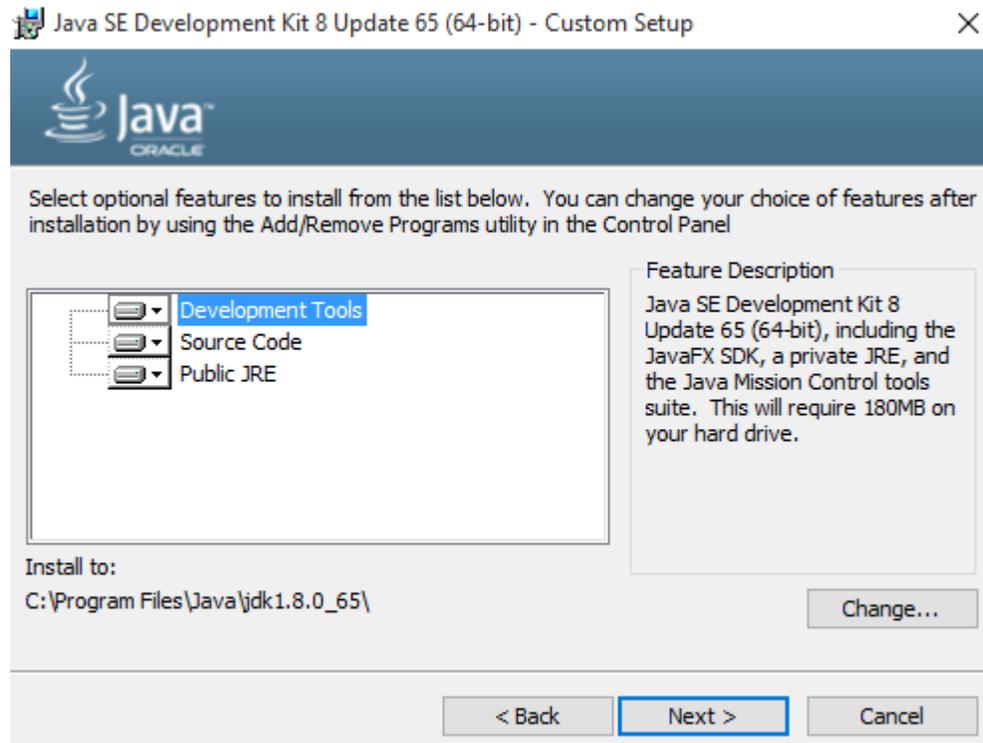
4. Damos doble click en el ejecutable descargado y a continuación presionamos el botón Next.

### Imagen 6 Comienzo de Instalación



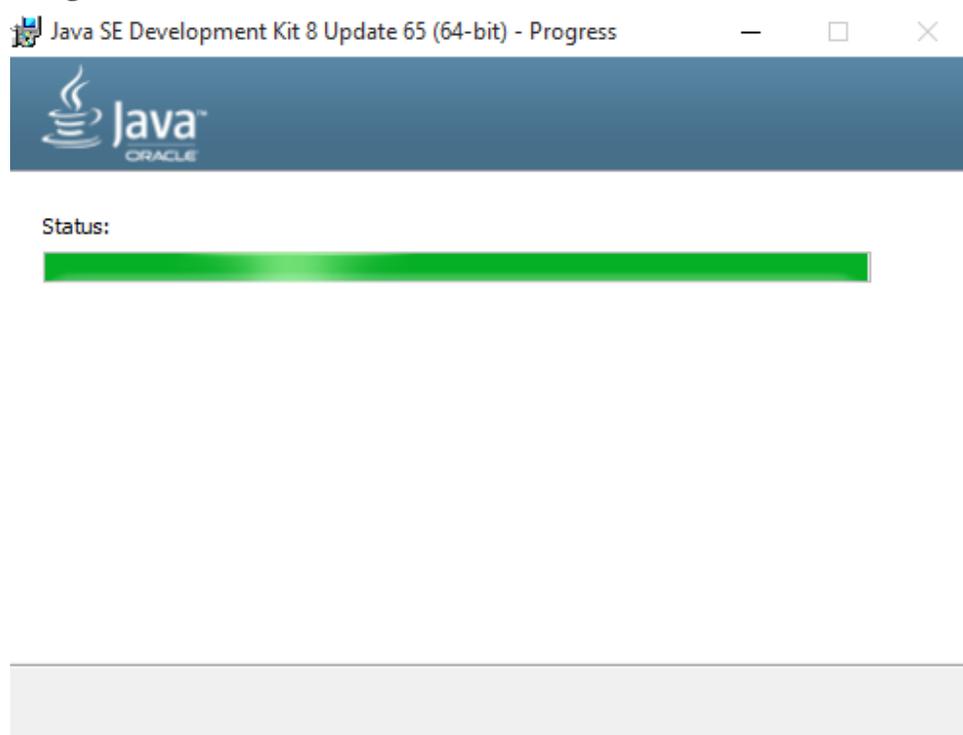
5. Seleccionamos los componentes que deseamos instalar y pulsamos en Next.

### Imagen 7 Componentes a Instalar

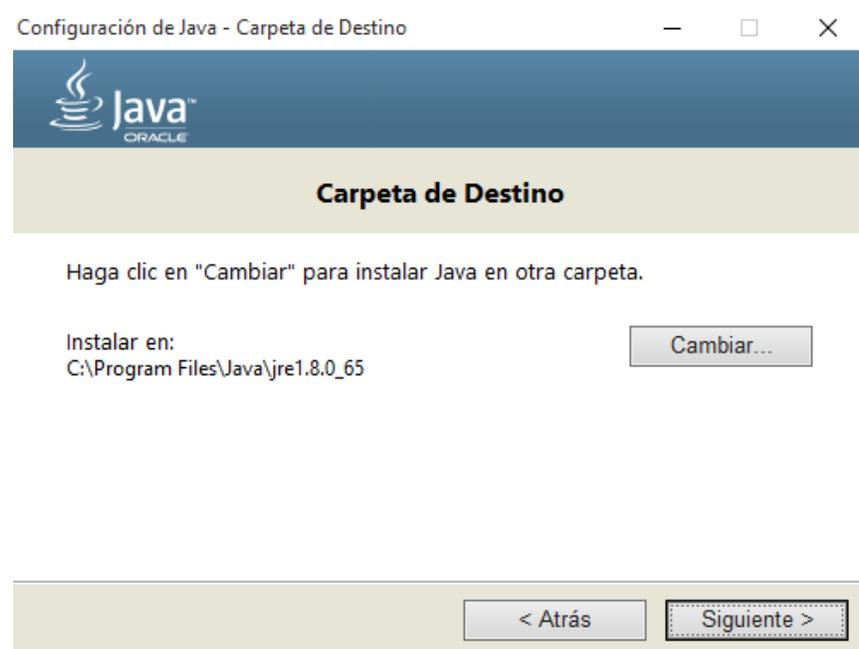


6. Comenzará la instalación del producto.

### Imagen 8 Comienzo de Instalación JDK



7. Seleccionamos el destino en donde se instalará el producto y pulsamos en Next.

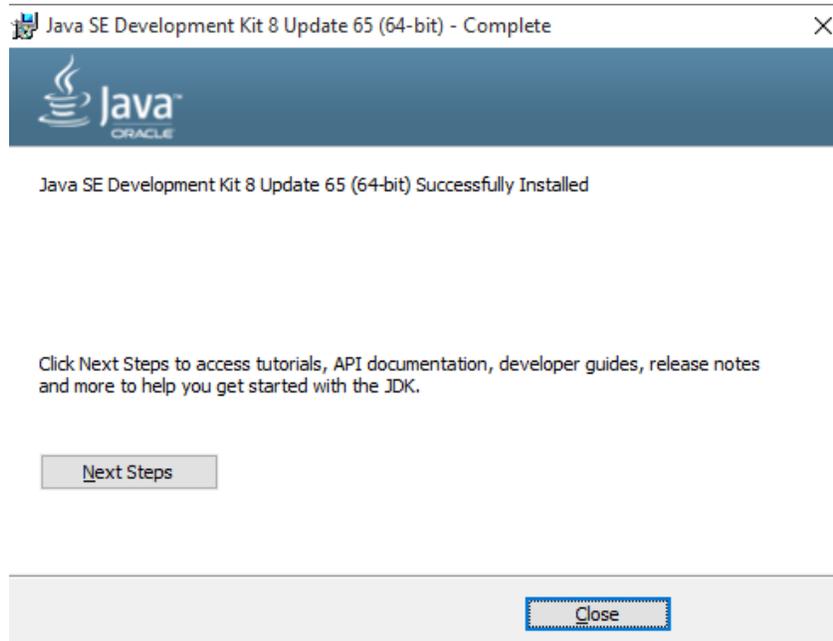
**Imagen 9 Destino de Instalación JDK**

8. Se instalará el producto.

**Imagen 10 Instalación de JDK**

- Para finalizar nos mostrará un mensaje que el producto instalado correctamente y seleccionamos el botón Close.

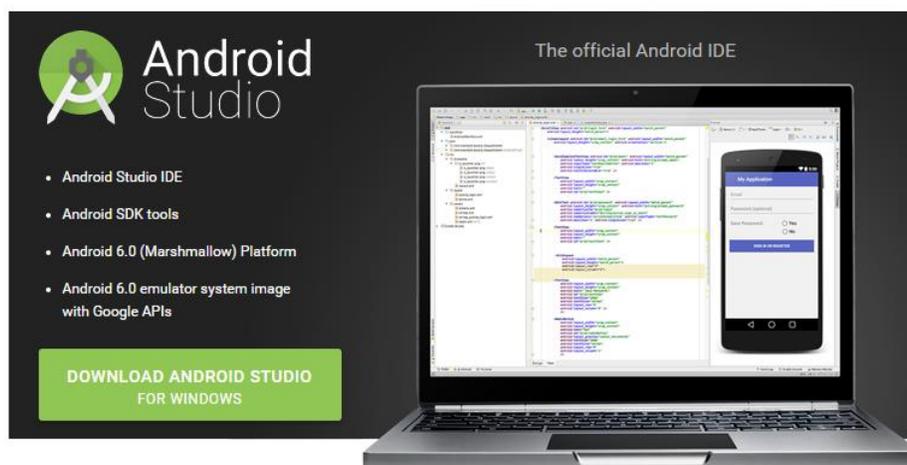
### Imagen 11 Fin de Instalación de JDK



### 2.5.2 Instalación y Configuración de Android Studio

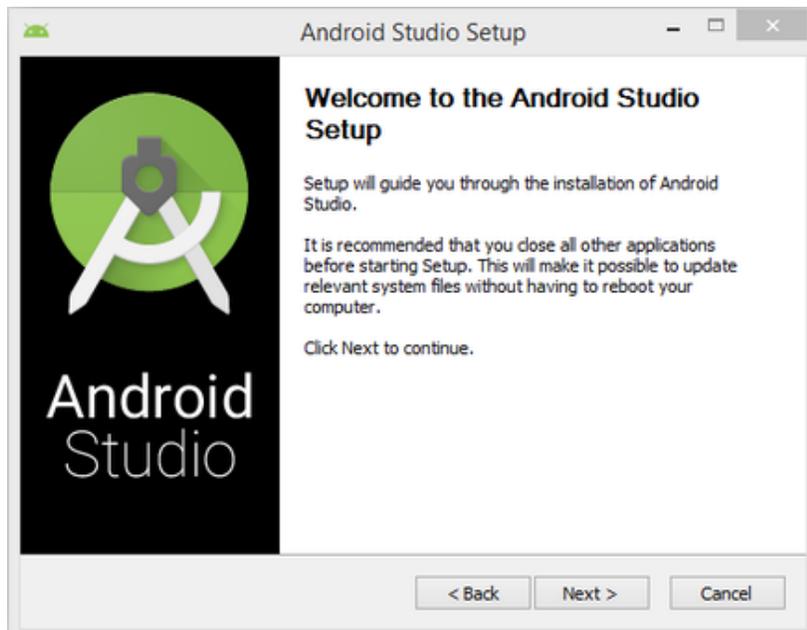
- Para descargar Android Studio en nuestro computador, nos dirigimos a la página de desarrolladores de Google <http://developer.android.com/intl/es/sdk/index.html> en donde seleccionaremos la descarga para nuestro sistema operativo en este caso es Windows.

### Imagen 12 Descarga Android Studio



- Una vez descargado damos doble click en el instalador, esperamos un momento a que se abra y en la ventana de instalación oprimir el botón Next.

**Imagen 13 Inicio de Instalación**



- Seleccionamos los componentes que deseamos instalar y pulsamos Next.

**Imagen 14 Selección de Componentes**



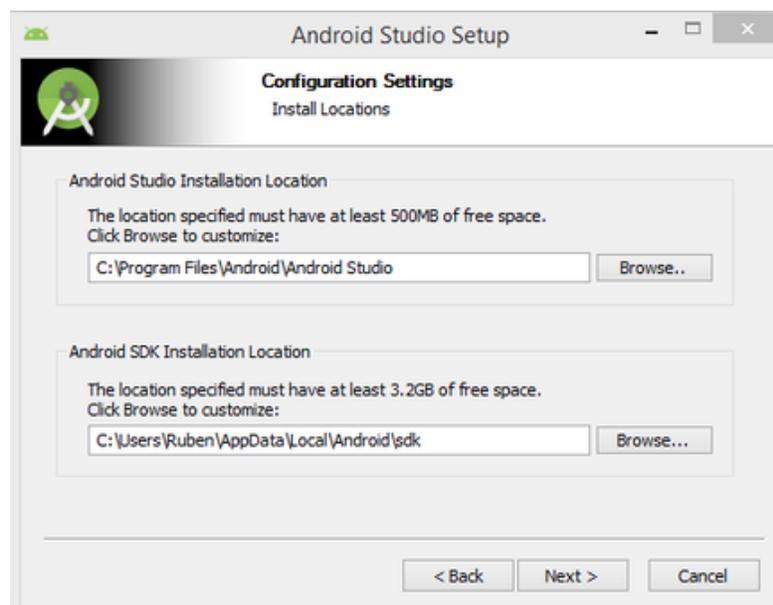
4. Aceptar términos y licencias click en Next.

### Imagen 15 Términos y Licencias



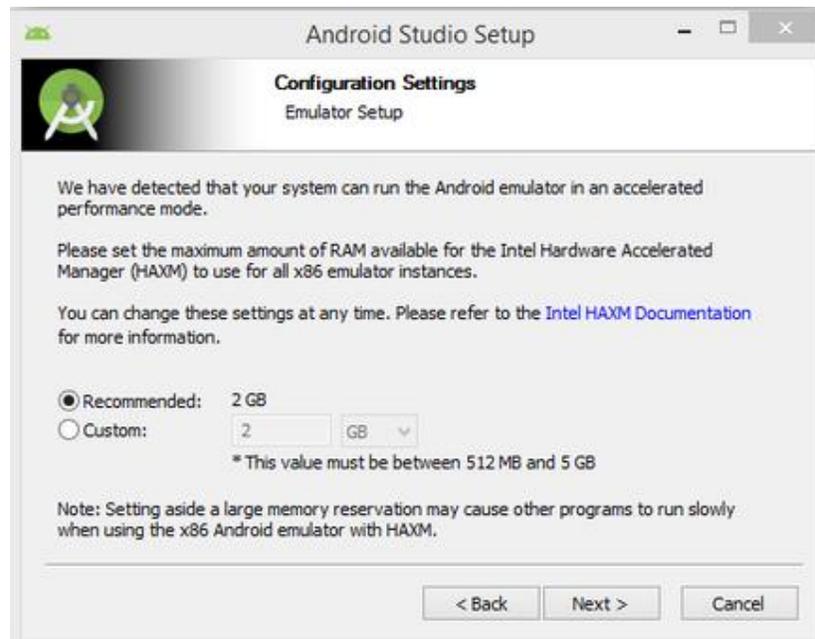
5. Seleccionamos la ruta de instalación y la ruta en donde se encuentra el JDK previamente instalado y pulsar en Next.

### Imagen 16 Selección de Destino



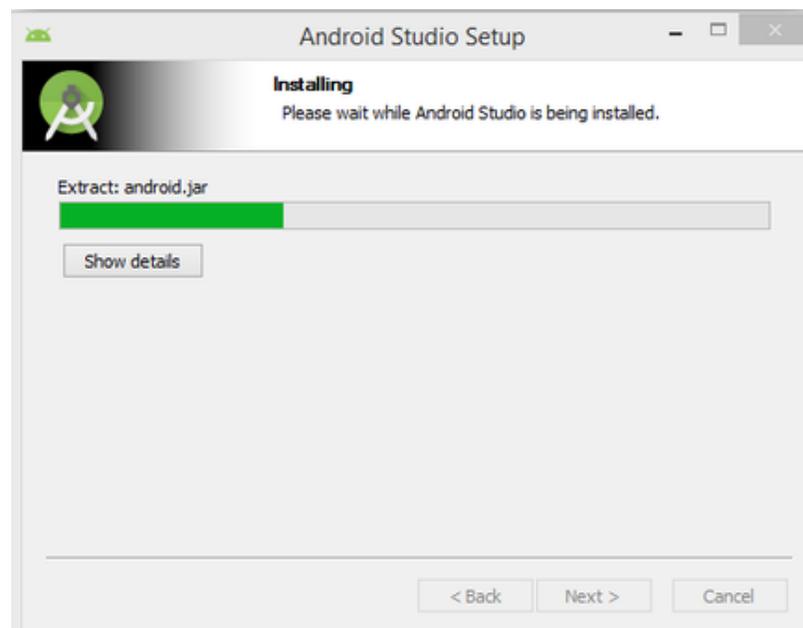
6. Seleccionamos la memoria RAM que deseamos asignar a la aplicación, mientras más memoria se le asigne será más rápida la emulación de los proyectos, click en Next.

### Imagen 17 Asignación de RAM



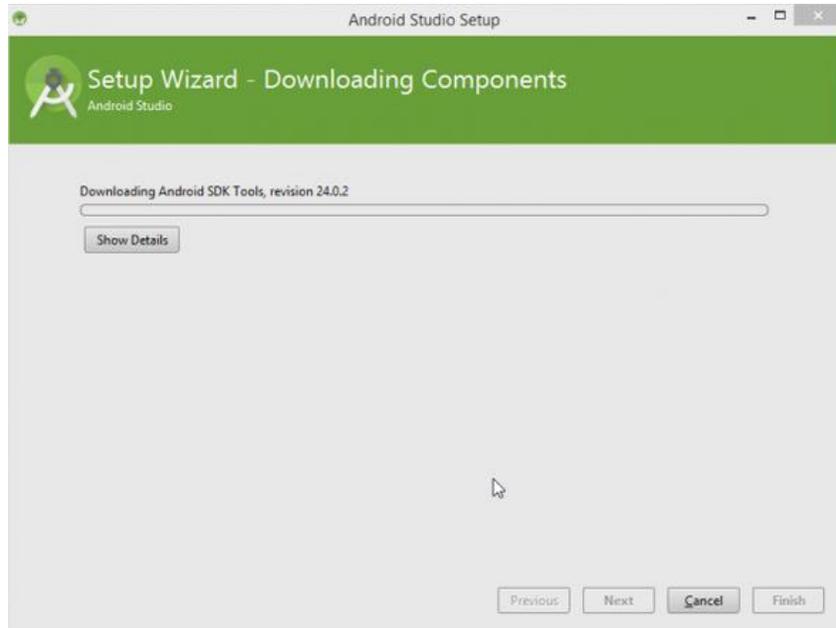
7. Comenzará la instalación de Android Studio al finalizar pulsamos en Next.

### Imagen 18 Instalación del Producto

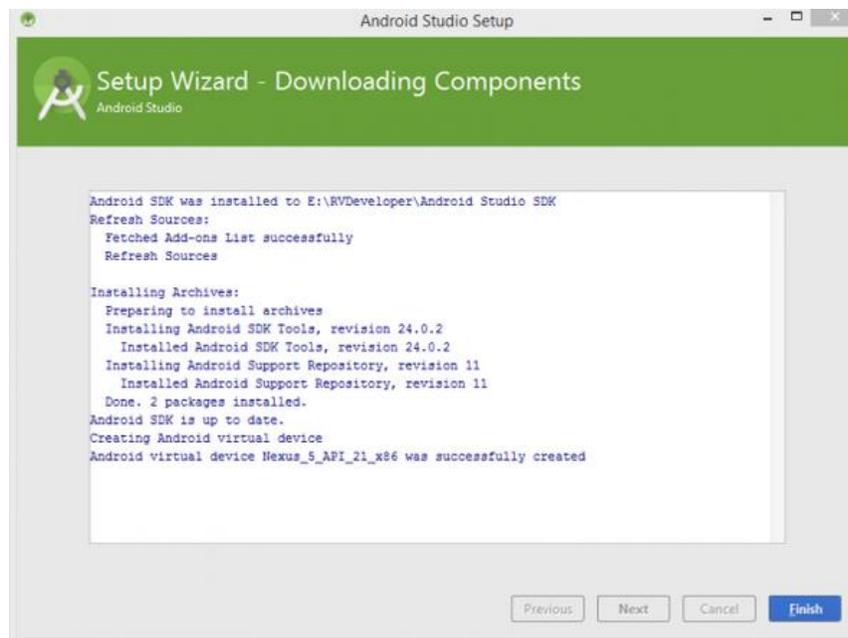


- Terminada la instalación Android Studio automáticamente descargará las librerías necesarias y seleccionamos el botón finish.

### Imagen 19 Descarga de SDK



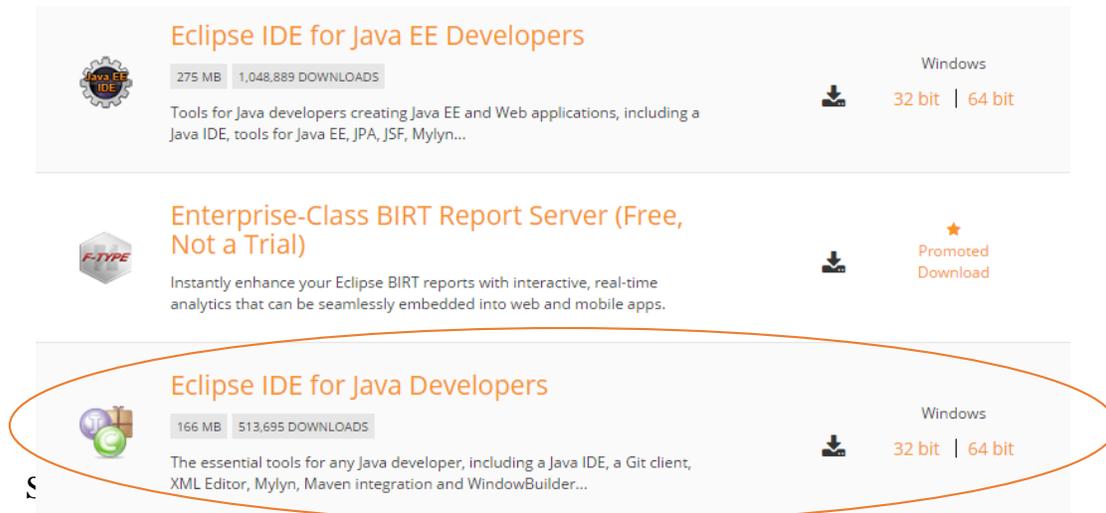
### Imagen 20 Descarga de SDK fin



### 2.5.3 Instalación de Eclipse.

1. Nos dirigimos a la página de eclipse <https://eclipse.org/downloads/> y descargamos Eclipse IDE for Java Developers.

#### Imagen 21 Descarga de IDE Eclipse



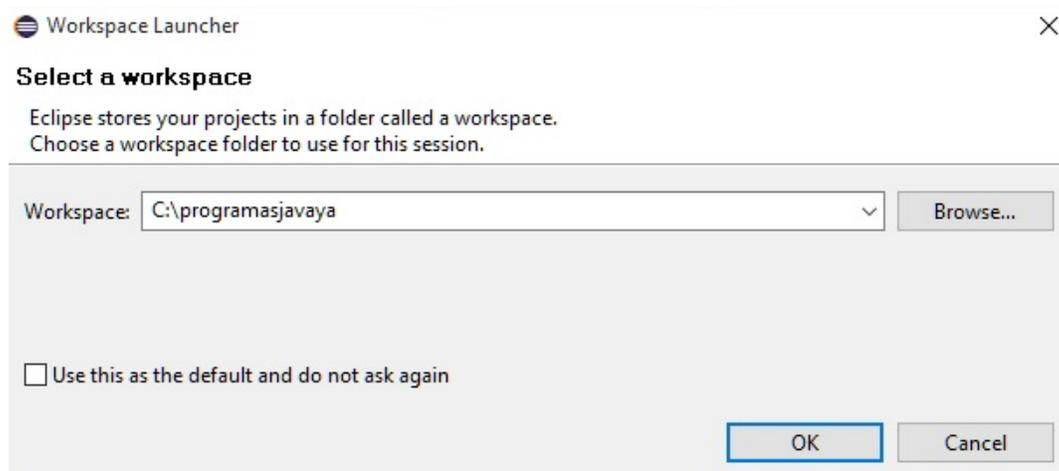
The screenshot shows the Eclipse website's download page. It features three download options:

- Eclipse IDE for Java EE Developers**: 275 MB, 1,048,889 DOWNLOADS. Windows, 32 bit | 64 bit.
- Enterprise-Class BIRT Report Server (Free, Not a Trial)**: Instantly enhance your Eclipse BIRT reports with interactive, real-time analytics that can be seamlessly embedded into web and mobile apps. Promoted Download.
- Eclipse IDE for Java Developers**: 166 MB, 513,695 DOWNLOADS. Windows, 32 bit | 64 bit. This option is circled in orange.

The description for the circled option reads: "The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven integration and WindowBuilder..."

2.  $\xi$

#### Imagen 22 Selección de Destino



The screenshot shows the Eclipse Workspace Launcher dialog box. It has a title bar "Workspace Launcher" and a close button. The main heading is "Select a workspace". Below this, it says "Eclipse stores your projects in a folder called a workspace. Choose a workspace folder to use for this session." There is a text field labeled "Workspace:" containing the path "C:\programasjavaya" and a "Browse..." button. At the bottom, there is a checkbox labeled "Use this as the default and do not ask again" which is currently unchecked. There are "OK" and "Cancel" buttons at the bottom right.

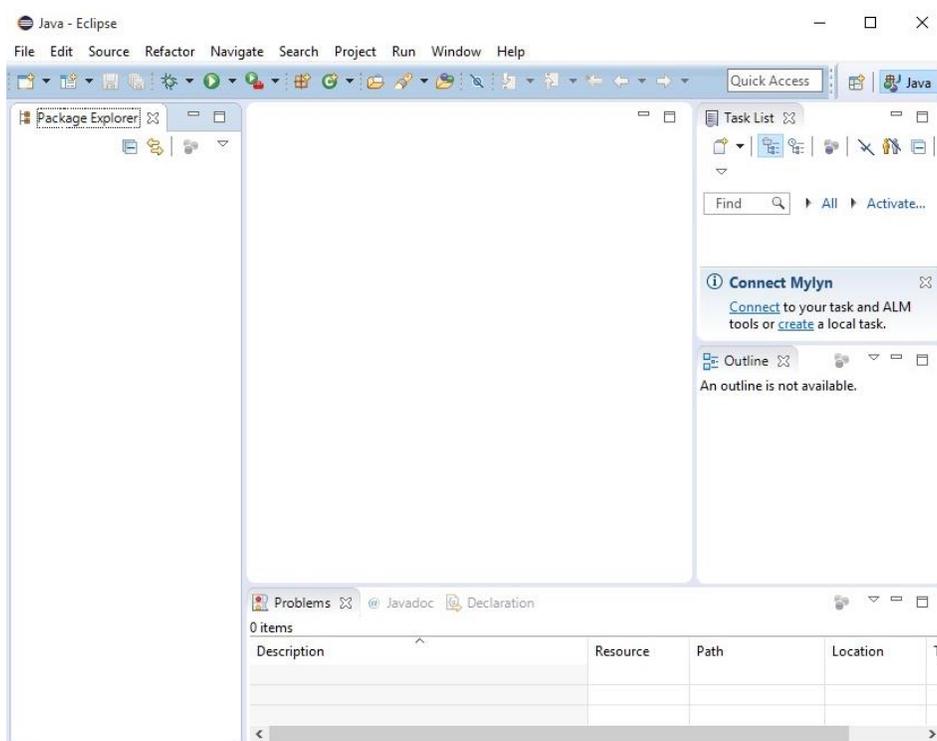
- Una vez termine la instalación se abrirá la ventana de Eclipse.

### Imagen 23 Ventana principal de Eclipse



- Pulsamos en el ícono Workbench y nos mostrará el entorno de desarrollo.

### Imagen 24 Entorno de desarrollo



### 2.5.4 Instalación de SDK para Eclipse

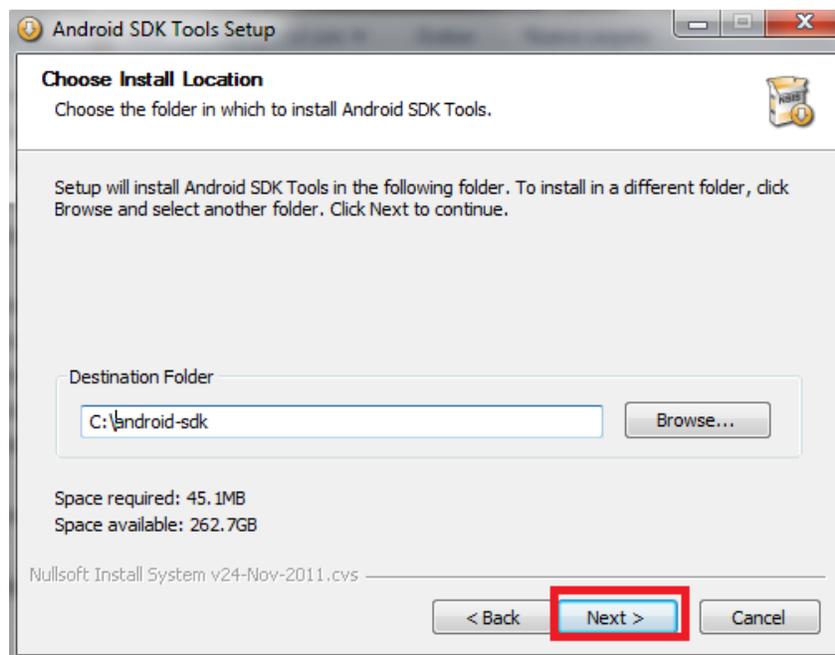
1. Nos dirigimos a la página <http://developer.android.com/intl/es/sdk/index.html#Other> y nos descargamos el SDK.

#### Imagen 25 Página de descarga de SDK

Platform	Package	Size	SHA-1 Checksum
Windows	<a href="#">installer_r24.4.1-windows.exe</a> (Recommended)	151659917 bytes	f9b59d72413649d31e633207e31f456443e7ea0b
	<a href="#">android-sdk_r24.4.1-windows.zip</a>	199701062 bytes	66b6a6433053c152b22bf8cab19c0f3fef4eba49
Mac OS X	<a href="#">android-sdk_r24.4.1-macosx.zip</a>	102781947 bytes	85a9cccb0b1f9e6f1f616335c5f07107553840cd
Linux	<a href="#">android-sdk_r24.4.1-linux.tgz</a>	326412652 bytes	725bb360f0f7d04eaccff5a2d57abdd49061326d

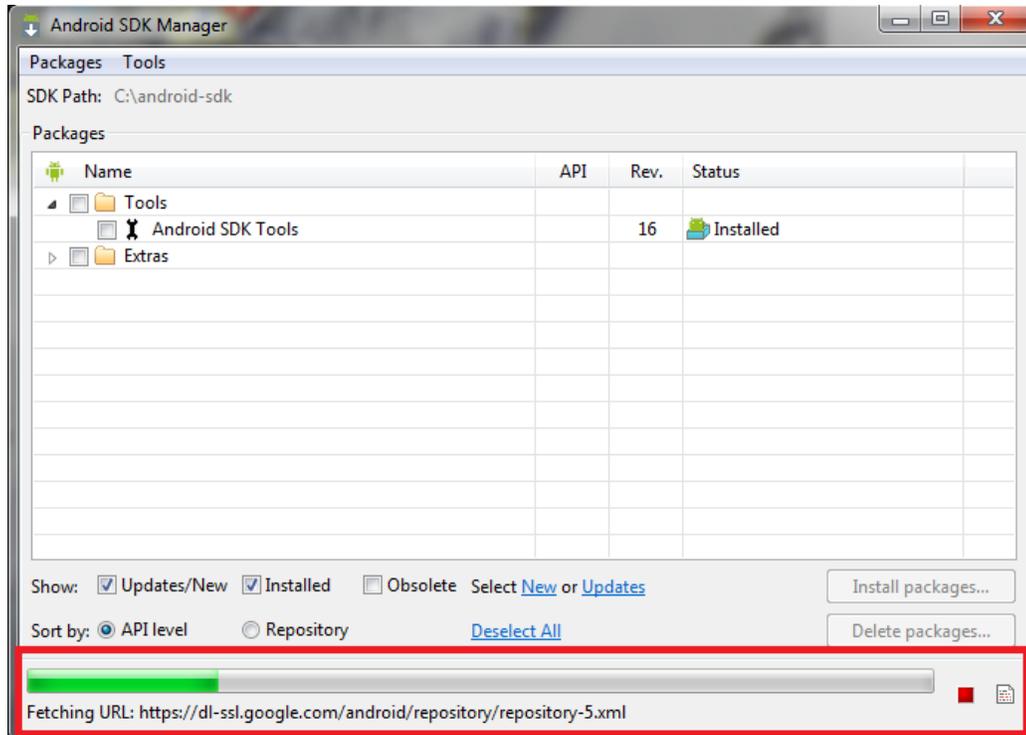
2. Ejecutamos el instalador de SDK seleccionamos la ruta de instalación y presionamos Next.

#### Imagen 26 Instalación de SDK



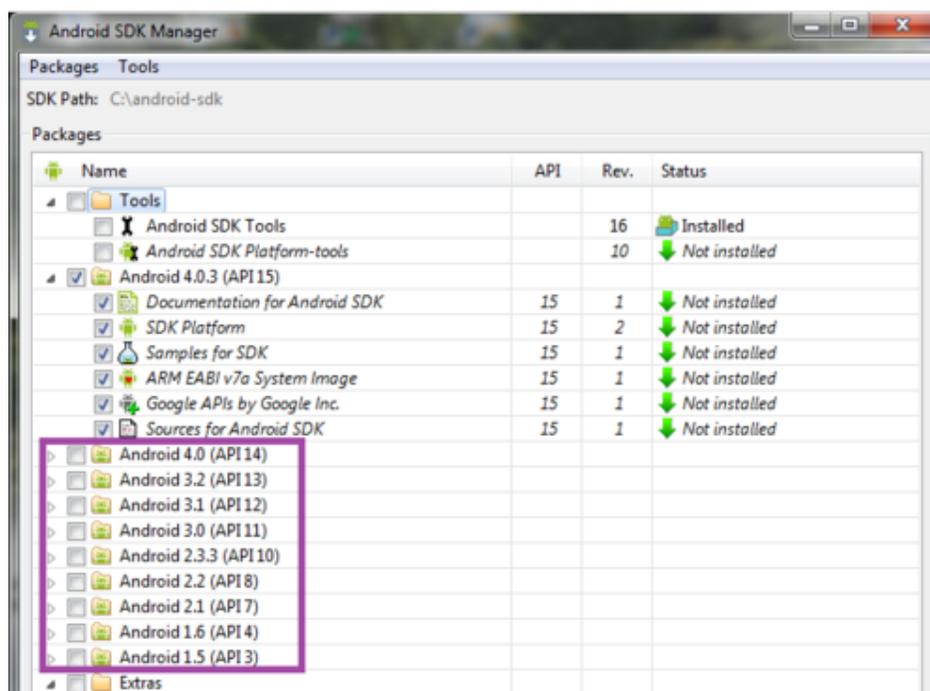
- Terminada la instalación se abrirá la ventana de SDK donde esperamos un momento a que busque en internet los nuevos complementos.

### Imagen 27 Descarga de complementos para SDK



- Seleccionamos los complementos que deseamos instalar y pulsamos en Install.

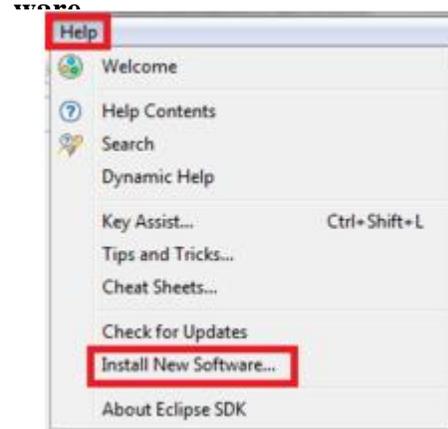
### Imagen 28 Selección de complementos a instalar



### 2.5.5 Vincular SDK con Eclipse

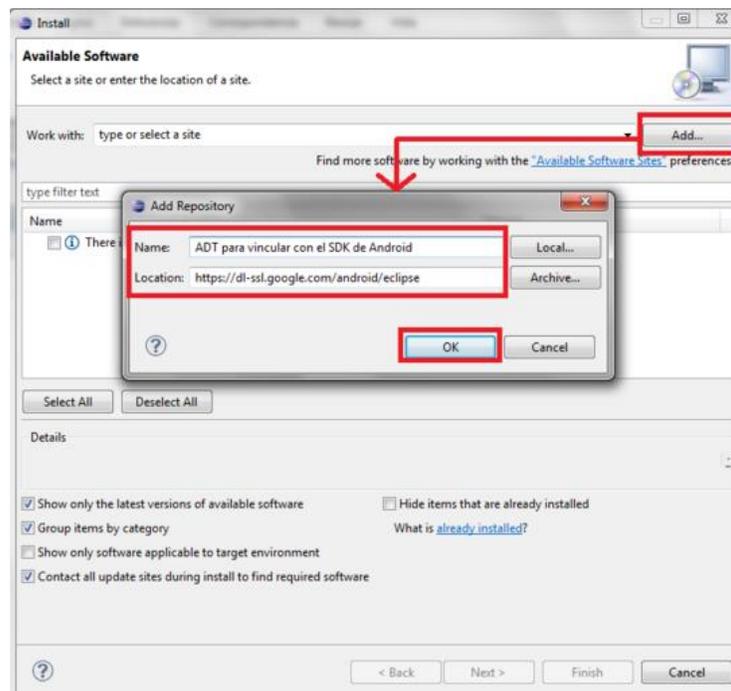
1. Abrimos Eclipse y nos dirigimos al menú Help, luego seleccionamos Install New Software.

**Imagen 29 Instalar nuevo Soft-**



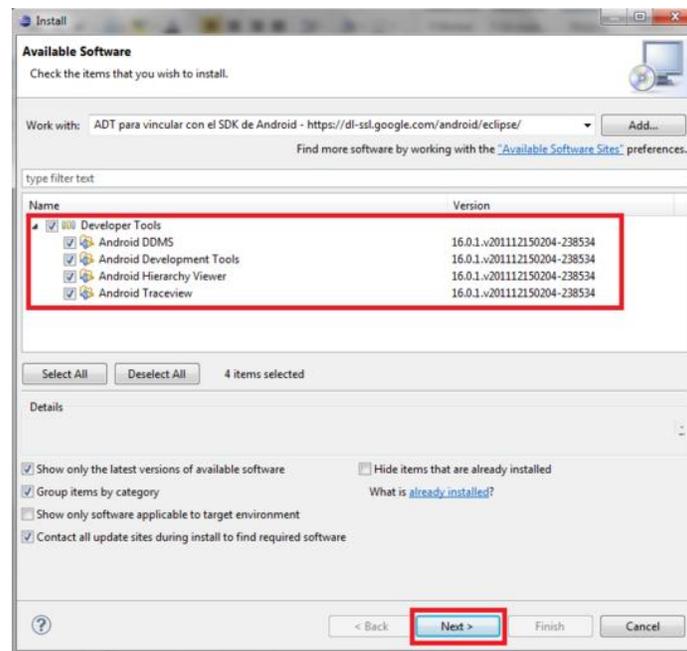
2. Se nos abre la ventana en donde pulsamos el botón Add y llenamos los campos Name con el nombre que deseamos para identificar la aplicación y en el campo URL ponemos la siguiente dirección <https://dl-ssl.google.com/android/eclipse/> presionamos OK.

**Imagen 30 Selección de nombre y ubicación**



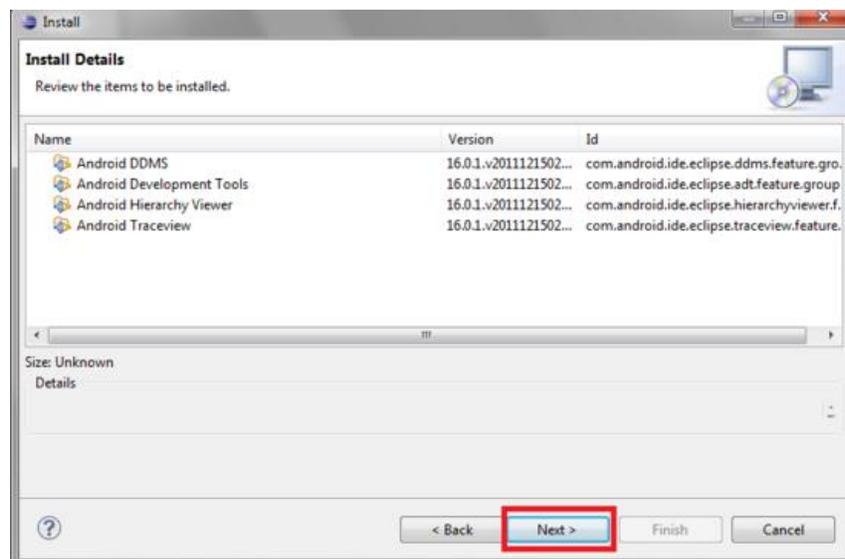
3. Seleccionamos la casilla Developer Tools y presionamos Next.

### Imagen 31 Selección de complementos



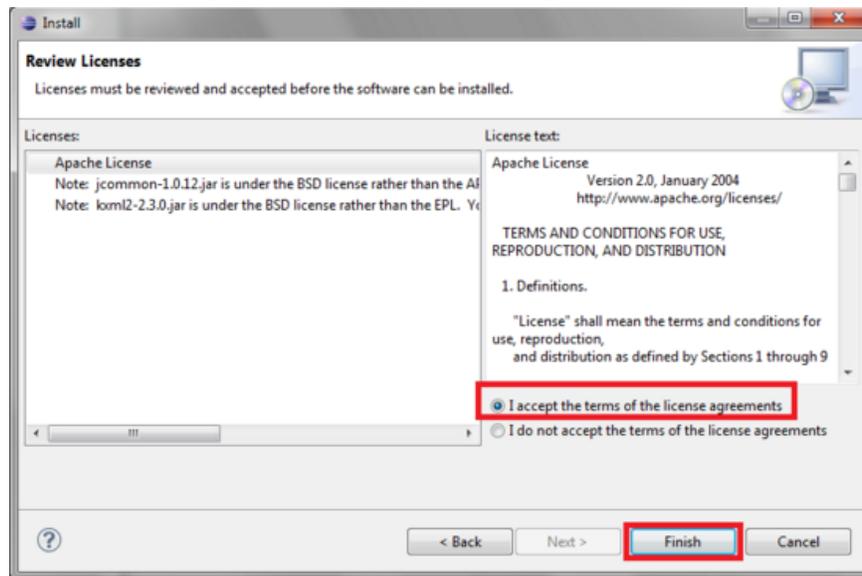
4. En la siguiente ventana nos muestra los detalles de la instalación presionamos Next.

### Imagen 32 Detalles de la instalación



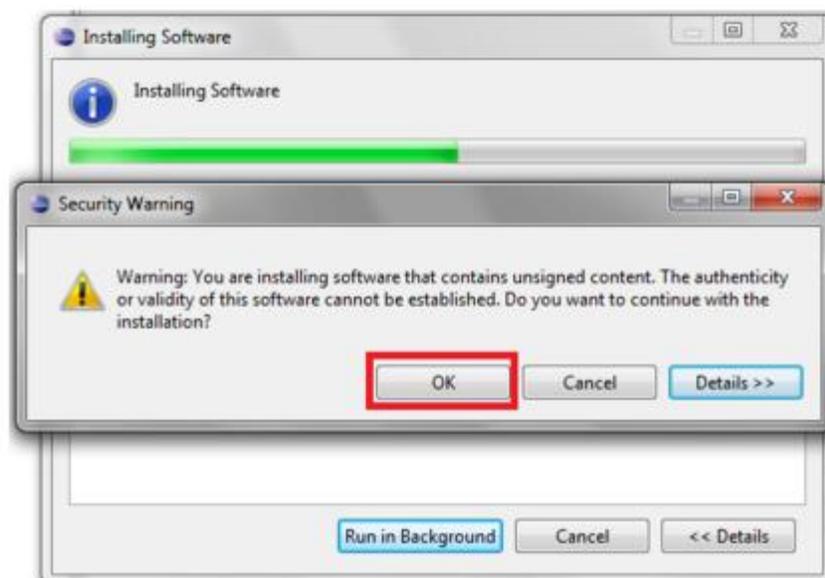
5. Aceptamos los términos y licencias y oprimimos el botón Next.

### Imagen 33 Términos y condiciones



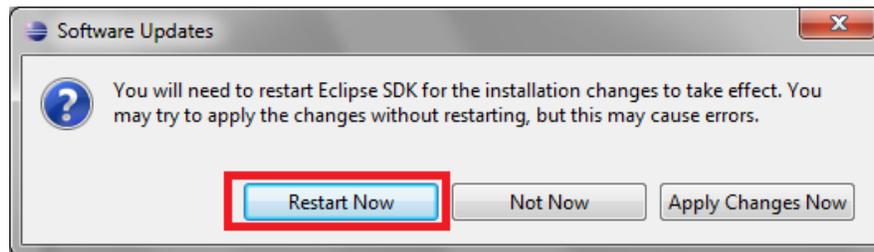
6. Comienza la instalación y nos dará una advertencia que existe contenido no firmado, pulsamos en OK.

### Imagen 34 Confirmar instalación



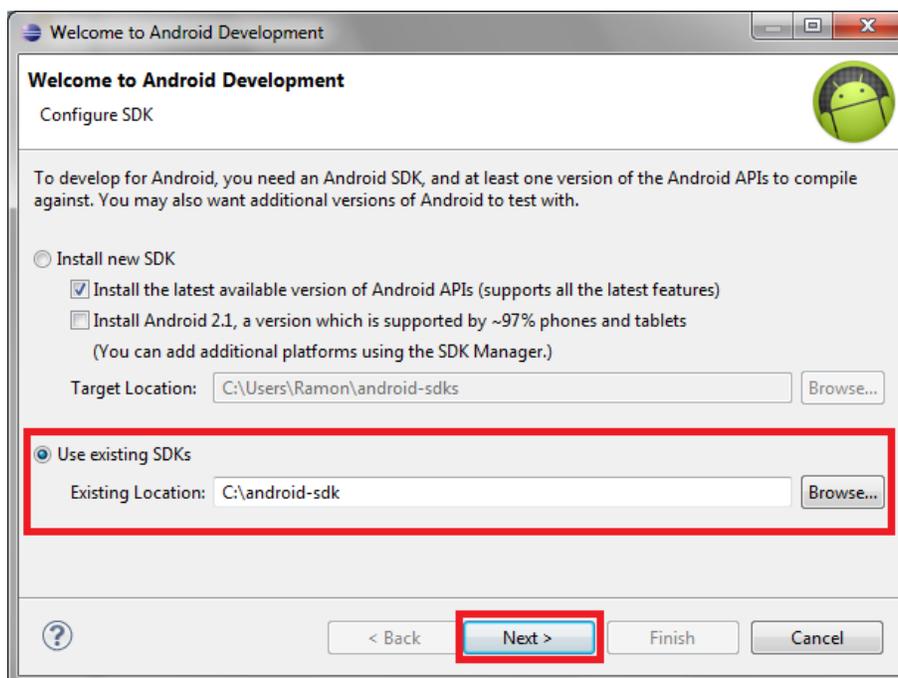
7. Terminada la instalación reiniciamos Eclipse con el botón Restart Now.

### Imagen 35 Reiniciar Eclipse



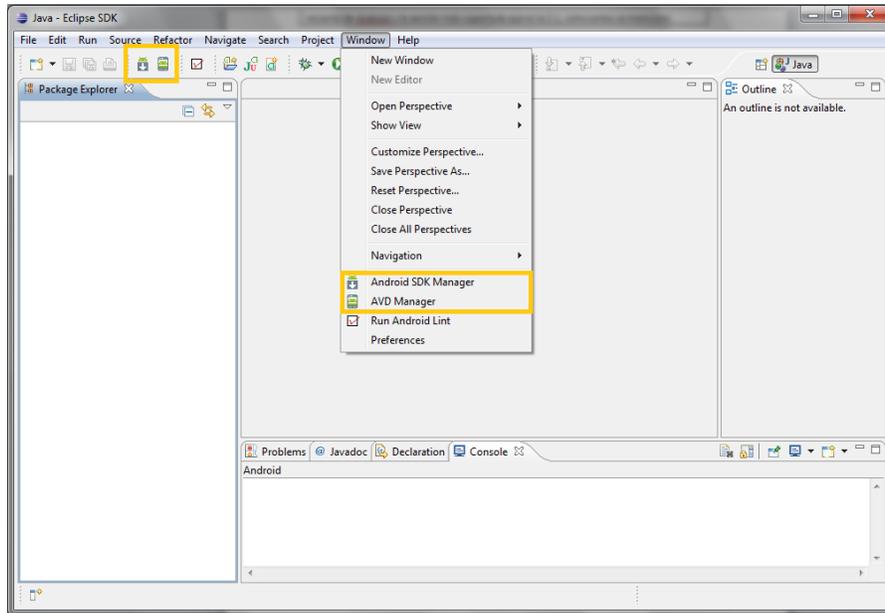
8. Al reiniciar Eclipse nos abrirá la siguiente ventana donde tenemos que seleccionar Use existing SDKs.

### Imagen 36 Selección de SDK instalado



9. Comprobamos que las Eclipse y SDK se encuentran vinculadas. (Invarato, 2012)

**Imagen 37 Confirmar vinculación**



## 2.6 Elección de Herramienta de Software a Utilizar

Para la elección de la herramienta de Software a utilizar para el desarrollo de la aplicación, se creó una tabla con parámetros y puntajes de las herramientas Eclipse SDK y Android Studio.

Los puntajes serán de 1- 10 en donde 1 sería la puntuación más baja y 10 la más alta.

**Tabla 2 Elección de Software**

<b>Parámetros</b>	<b>Eclipse SDK</b>	<b>Android Studio</b>
Fácil búsqueda y descarga del Software	8	10
Fácil Instalación	4	10
Actualización automática de plugins	1	8
Soporte para varias versiones de Android	5	10
Soporte y Actualizaciones de Software	5	10
Soporte para distintos dispositivos	3	10
Interfaz gráfica amigable	5	8
Manuales y tutoriales	10	8
Refactorización y completado avanzado de código	1	10
Vista en tiempo real de layouts	4	10
Fácil empaquetado de proyecto	4	9
<b>Total</b>	<b>50</b>	<b>103</b>

### 2.6.1 Justificación

**Fácil búsqueda y descarga de Software:** Debido a que Eclipse dejó de obtener soporte constante por Google se hace más complicado encontrar los instaladores en la red debido a que no solo es necesario el Software de eclipse sino también las librerías para programación en Android, por otra parte, Android Studio es la herramienta desarrollada por Google y la podemos encontrar de forma rápida dentro de su propia página.

**Fácil Instalación:** Al instalar eclipse es necesario instalar también las librerías para programación en Android por lo que se necesita realizar configuraciones adicionales dentro de este software y lo vuelve complicado, por esta razón recibe una calificación de 4/10, Android Studio es mucho más fácil de instalar, ya que de manera automática una vez instalada busca si existen actualizaciones o nuevas librerías y las instala de forma silenciosa mientras trabajamos en la herramienta.

**Actualización automática de plugins:** El software Eclipse no cuenta con actualizaciones automáticas por lo que se tiene que buscar los plugins necesarios en la Internet e instalarlos, al contrario, Android Studio cada vez que es utilizado realiza una búsqueda automática de actualizaciones y de encontrarlas las instala sin ningún problema.

**Soporte para varias versiones de Android:** Al programar en eclipse estamos limitados a que nuestro Software funcione solo en algunas plataformas de Android mientras que con eclipse podemos seleccionar desde que versión y para qué tipo de dispositivo estamos desarrollando el software pudiendo ser para relojes, celulares, tablets, Smart Tv entre otras.

**Soporte y Actualizaciones de Software:** Como se mencionó anteriormente Eclipse poco a poco está dejando de obtener soporte de Google por lo que con el paso del tiempo será una herramienta obsoleta para la programación en Android, mientras que Android Studio al ser una herramienta desarrollada por el propio Google está teniendo una gran acogida en la actualidad y seguirá creciendo con el paso del tiempo.

**Soporte para distintos dispositivos:** En la herramienta de desarrollo Eclipse estamos limitados a programar para un dispositivo específico, mientras que en Android Studio podemos elegir para que dispositivo queremos desarrollar y desde que versión de Android funcionará, según lo elegido nos brindara las facilidades correspondientes.

**Interfaz gráfica amigable:** Si comparamos la interfaz gráfica entre Eclipse y Android Studio nos daremos cuenta de inmediato que la de Android Studio es superior, permitiéndonos encontrar lo que deseamos en poco tiempo y se encuentra todo debidamente organizado.

**Refactorización y completado avanzado de código:** Eclipse no cuenta para nada con esta ventaja lo que es un perjuicio para los desarrolladores debido a que toma mucho tiempo escribir largas cadenas de código, al contrario, Android Studio puede predecir los códigos que vamos a emplear y nos dará sugerencias pudiendo corregir los códigos mal escritos o incompletos con una simple combinación de teclas lo que ayuda a la velocidad con la que se desarrolla la aplicación.

**Vista en tiempo real de Layouts:** Tanto Eclipse como Android Studio nos permiten ver los Layouts de cómo quedará nuestra aplicación, los lugares en donde se encontrarán los botones, cajas de texto, etc. Pero Android Studio nos muestra de manera mucho más precisa y en tiempo real lo que estamos programando, al igual que nos permite escoger el dispositivo en el que queremos tener una vista previa de cómo quedará pudiendo ser una tablet, un celular, un reloj entre otros.

**Fácil empaquetado de proyecto:** Para lograr empaquetar un proyecto en Eclipse se tiene que seguir una serie de pasos lo que lo hace un poco complejo, pero por otra parte la herramienta Android Studio facilita mucho más el empaquetado permitiéndonos con un par de clicks obtener nuestro proyecto empaquetado y listo para subirlo a Google Play.

### 2.6.2 Decisión

Como se puede observar en la Tabla 2, Android Studio cuenta con más ventajas que Eclipse es por esta razón que se vio conveniente realizar el desarrollo en esta herramienta, posee una interfaz de usuario amigable y ofrece ayuda al usuario, razón por la cual el desarrollo es mucho más fácil y rápido, posee soporte por grandes comunidades de desarrolladores en la red y por su creador Google.

Es por esta razón que se ha optado por elegir Android Studio como software para el desarrollo de la aplicación “GeoVentas” debido a que la aplicación en un futuro seguirá creciendo y necesitará contar con herramientas actualizadas que se acoplen a las tecnologías actuales.

## Capítulo 3

### 3. Diseño del Software

#### 3.1 Requerimientos del Software

##### 3.1.1 Requerimientos Funcionales

A continuación, se numerarán los requerimientos funcionales con los que contara la aplicación “GeoVentas”

- El usuario podrá acceder a su cuenta ingresando sus datos que serán el usuario y contraseña.
- En el momento que un usuario ingresa el sistema descargará los datos asignados a este usuario, entre ellos tenemos las rutas, clientes, direcciones, productos.
- El usuario podrá iniciar un proceso de pedido para cada cliente, una vez iniciado el proceso tendrá que ser terminado caso contrario los datos ingresados serán descartados.
- El usuario podrá seleccionar varias opciones al momento de iniciar un proceso de ventas entre ellas encontraremos local cerrado, el cliente no desea realizar el pedido, guardar pedido.
- El usuario podrá realizar una fotografía en el caso de ser necesario antes de terminar el proceso de ventas.
- El sistema mostrará los productos que se encuentran en la base de datos los cuales podrán ser seleccionados por el usuario en el proceso de ventas, pueden ser seleccionados uno o varios productos de ser necesarios.
- Una vez terminado el proceso de ventas, el sistema guardará los datos.
- El sistema mostrara información de las estadísticas de ventas.

### 3.1.2 Requerimientos no Funcionales

La aplicación GeoVentas contará con los siguientes requerimientos no funcionales los cuales se mostrarán por categorías.

Rendimiento.

- El sistema ofrecerá respuesta al usuario en tiempo real.
- El tiempo de respuesta de la aplicación no podrá demorar más de 5 segundos.

Seguridad.

- Se realizará la verificación de que el usuario pertenece a la empresa.
- El sistema no tendrá acceso a internet.

Fiabilidad.

- El sistema ofrecerá una fiabilidad del 90% en su funcionamiento

Disponibilidad.

- La aplicación contará con un 100 % de disponibilidad una vez haya sido instalada en el dispositivo como aplicación nativa.

Mantenibilidad.

- Se contará con mantenimiento permanente, de igual manera se podrá agregar funciones con el paso del tiempo de verse conveniente.

Portabilidad.

- La aplicación ofrecerá compatibilidad con varias plataformas Android desde la versión 4.4.2 o superiores.
- Podrá ser descargada en cualquier momento y cualquier lugar mediante la tienda de Android Google Play como versión de prueba.

Operatividad.

- La aplicación podrá ser utilizada por personal de ventas el cual previamente haya tenido una pequeña tutoría, de igual manera contará con manuales de funcionamiento.

Interoperabilidad.

- Contará con interoperabilidad ente tecnología web y tecnología Android.

### **3.2 Restricciones**

El dispositivo en el que se ejecute la aplicación deberá cumplir con los siguientes requisitos mínimos:

Procesador: ARM de 500 MHz

Memoria RAM: 1Gb

Disco Duro: 200 Mb

Pantalla: Multitouch

GPS: Si

WIFI: Si

Datos: Si

Cámara: Frontal y posterior

Versión de Sistema Operativo: 4.4.2 y superiores

### **3.3 Evolución futura de la aplicación**

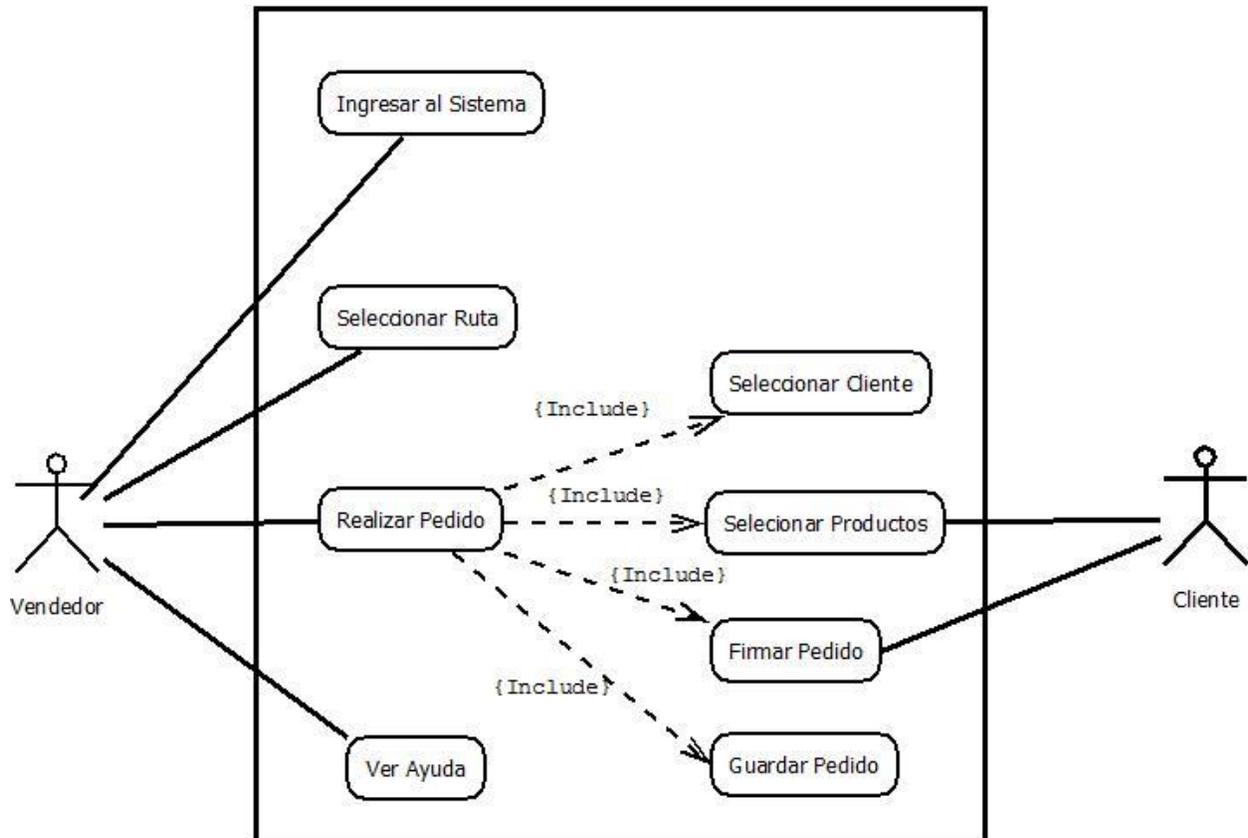
La aplicación GeoVentas deja abiertas muchas opciones para desarrollo entre las cuales se planea desarrollar módulos de rutas, despachos, inventarios, abastecimiento.

Con las cuales poco a poco se irá formando una aplicación completa para cumplir con las necesidades de empresas reales.

### 3.4 Diseño de la Aplicación

#### 3.4.1 Casos de Uso

Imagen 38 Casos de Uso



#### 3.4.2 Descripción del Caso de Uso

Tabla 3 Caso de uso ingresar al sistema

Ingresar al Sistema	
<b>Descripción</b>	El sistema permitirá el Login del usuario, al ingresar al sistema se descargarán los datos que se encuentran en la base de datos centralizada a la base de datos del dispositivo móvil.
<b>Estabilidad</b>	Alta
<b>Comentario</b>	En el caso de no contar con internet estos datos se descargan al dispositivo para que este pueda trabajar sin conexión hasta reconectarse a una red

**Tabla 4 Caso de uso Selección de ruta**

	<b>Selección de Ruta</b>
<b>Descripción</b>	Una vez que el usuario inició sesión tendrá acceso a las rutas asignadas para su visita.
<b>Estabilidad</b>	Alta
<b>Comentario</b>	Se selecciona el lugar en el cual se realizará un nuevo pedido para luego enviar los datos al servidor.

**Tabla 5 Caso de uso Realizar pedido**

	<b>Realizar Pedido</b>
<b>Descripción</b>	El agente vendedor tendrá acceso a todos los productos disponibles los cuales podrán ser seleccionados, si el pedido se guardó y se desea realizar una modificación se podrá realizar posteriormente desde la ventana modificaciones.
<b>Estabilidad</b>	Alta
<b>Comentario</b>	Una vez que se comienza el proceso de pedidos no se podrá salir de este hasta guardar los datos.

**Tabla 6 Caso de uso Ver estadísticas**

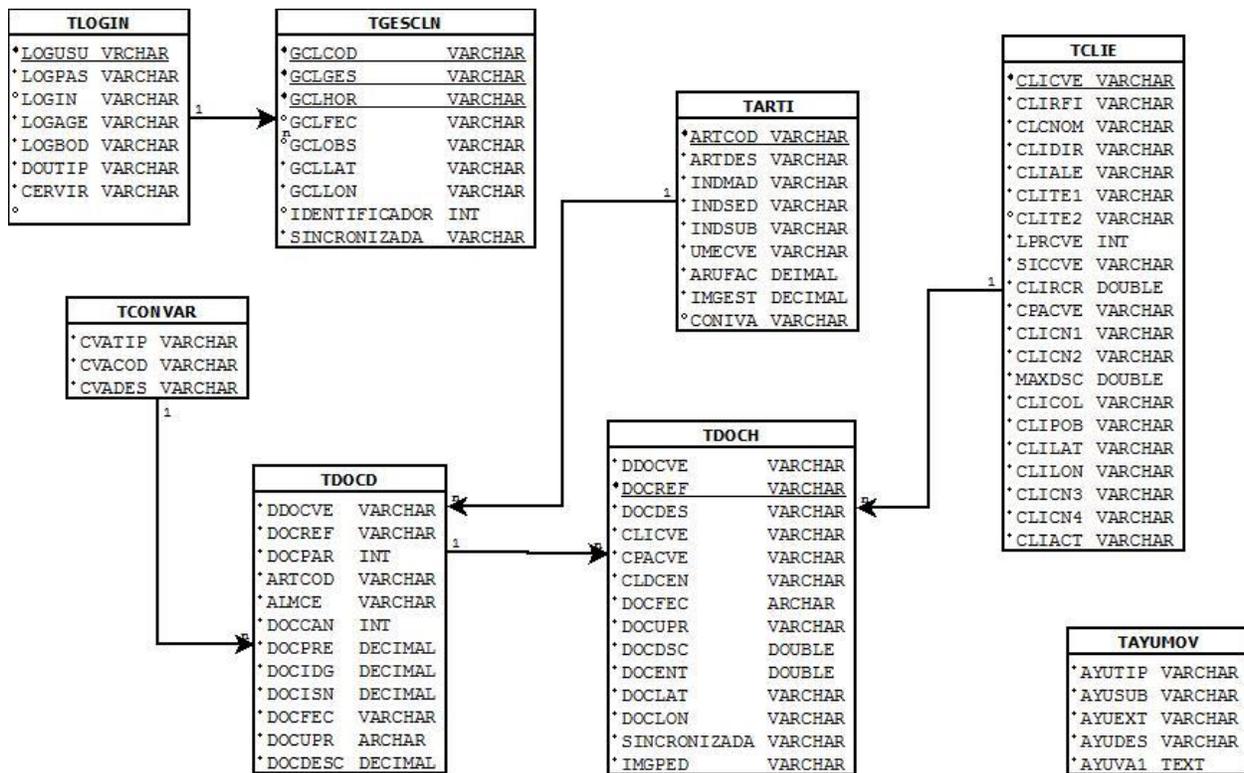
	<b>Ver Estadísticas</b>
<b>Descripción</b>	El sistema realizara un gráfico con las estadísticas de ventas de la persona logeada en el sistema, se podrá ver por clientes, por día, por semana y por mes según desee el usuario.
<b>Estabilidad</b>	Alta
<b>Comentario</b>	Si el usuario no realizó ninguna venta la tabla se mostrará vacía.

Tabla 7 Caso de uso Ayuda

Ayuda	
<b>Descripción</b>	El sistema mostrará un menú con temas de ayuda para su uso entre ellas podremos encontrar como realizar el Login, el proceso para realizar un pedido, realizar modificaciones o sincronizar la base de datos.
<b>Estabilidad</b>	Alta
<b>Comentario</b>	

### 3.4.3 Diagrama de Clases

Imagen 39 Diagrama de Clases



### 3.5 Análisis de herramientas de Desarrollo

#### 3.5.1 Uso de GPS en Android Studio

Para el uso del GPS en los dispositivos con Android primero se tiene que importar las librerías de localización como se muestra a continuación.

```
import android.content.Context;
import android.location.GpsSatellite;
import android.location.GpsStatus;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.util.Log;
```

Luego se crea la clase en la cual implementaremos las librerías de localización y se crean las variables principales con las cuales se tratarán los datos de la forma necesaria.

```
public class GPSLocalizacion extends Activity implements LocationListener,
    GpsStatus.Listener {

    // Variables principales
    private String txtEstadoLocalizacion;

    private LocationManager locManager;
    private LocationListener locListener;
    private Bundle b;
    private String id;
    private GpsStatus mStatus;
    private String Latitud = "";
    private String Longitud = "";
```

Por ultimo utilizamos las funciones necesarias para extraer la información de latitud y longitud cada cierto tiempo, en este caso cada segundo, mediante el GPS de nuestro dispositivo.

```
// función principal para realizar la localización
public ArrayList<String> Localizar() {
```

```

try{
    // Obtenemos una referencia al LocationManager
    locationManager = (LocationManager) getSystemService(Context.LOCA-
LOCATION_SERVICE);
    locationManager.addGpsStatusListener(this);
    // Obtenemos la última posición conocida
    Location loc = locationManager
        .getLastKnownLocation(LocationManager.GPS_PROVIDER);

    // Mostramos la última posición conocida
    ArrayList<String> coorde = mostrarPos(loc);

    // Nos registramos para recibir actualizaciones de la posición
    locListener = this;
    // solicitamos actualizaciones cada 1 segundos
    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,
1000,
        10, locListener);
    return coorde;
}

```

### 3.5.2 Uso de Bases de datos en Android Studio

Para usar la base de datos que posee Android no se necesita descargar ninguna aplicación extra a nuestro dispositivo solamente importamos las librerías de SQLite y con esto ya podemos crear la base de datos que necesitemos dentro de nuestro dispositivo con tecnología Android.

```

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteDatabase.CursorFactory;
import android.database.sqlite.SQLiteOpenHelper;

```

Importadas las librerías creamos la clase DataBase en la cual procedemos a la creación de la Base de Datos y las tablas necesarias de la siguiente forma.

```

public class DataBase extends SQLiteOpenHelper {

    public static class TablaCliente{
        public static String IdCliente = "CLICVE";
        public static String CICliente = "CLIRFI";
        public static String NomEmpresa = "CLCNOM";
    }

    String tablaDireccionEntrega = "CREATE TABLE TCLDE (" +
        "CLICVE VARCHAR(10)," +
        "CLDCEN VARCHAR(10)," +
        "CLDDIR VARCHAR(40)," +
        "CLDTE1 DECIMAL(12,0)," +
        "CLDTE2 DECIMAL(12,0)," +
        "CLDCOL VARCHAR(30)," +
        "CLDPOB VARCHAR(30))";

```

### 3.5.3 Uso de cámaras en dispositivos móviles con Android Studio

Para el manejo de las cámaras del dispositivo implementamos el siguiente código en el cual primero obtenemos los parámetros necesarios, luego inicializamos la cámara y por último se cargan los datos en un bitmap y los guardamos en la memoria de nuestro dispositivo.

```

// Obtengo los parametros
parameters = mCamera.getParameters();
mCamera.setParameters(parameters);
mCamera.startPreview();

// realizo la inicializacion de la camara
Camera.PictureCallback mCall = new Camera.PictureCallback() {
    @Override
    public void onPictureTaken(byte[] data, Camera camera) {
        // cargo la imagen en un bitmap
        bmp = BitmapFactory.decodeByteArray(data, 0, data.length);
        // cargo la imagen en un imagen view
        // iv_image.setImageBitmap(bmp);

        String fotoIncognito = cabeceraClienteDeterminidad.get(0)
            .getIMGPED();

```

### 3.5.4 Desarrollo HTML

Para el desarrollo de la página WEB se utilizará archivos JSON los cuales usan un formato sencillo para un rápido intercambio de datos entre la aplicación, la página WEB y la base de datos.

A continuación, se mostrará un ejemplo de cómo se usará este formato de programación.

```
<?php
    $usuario=$_GET["id"]; // obtenemos el parámetro necesario para realizar la consulta en la BD
    require('conexion.php'); // llamamos a la clase que permite la conexión con la BD.
    $query="SELECT * FROM `tgescln` // consulta para obtener datos necesarios de la BD.
INNER JOIN tclie
on GCLCOD=CLICVE
INNER JOIN tconvar
ON CVATIP='TGC'
AND GCLGES=`CVACOD`
WHERE SINCRONIZADA =UPPER('".$usuario."')";
    $resultado=$mysqli->query($query); // guarda la información en una variable
?>
```

Luego utilizamos formato html normal para crear las tablas e introducir la información que deseamos que se muestre a la persona encargada de realizar el seguimiento a la fuerza de ventas, entre esta información tendremos los nombres de los vendedores, el lugar donde se realizó el pedido y la geolocalización de cada uno de ellos.

La Geolocalización se realizó utilizando el API de Google Maps al cual le pasamos las coordenadas capturadas por la aplicación y nos mostrará de forma gráfica el lugar aproximado de donde se realizó el pedido.

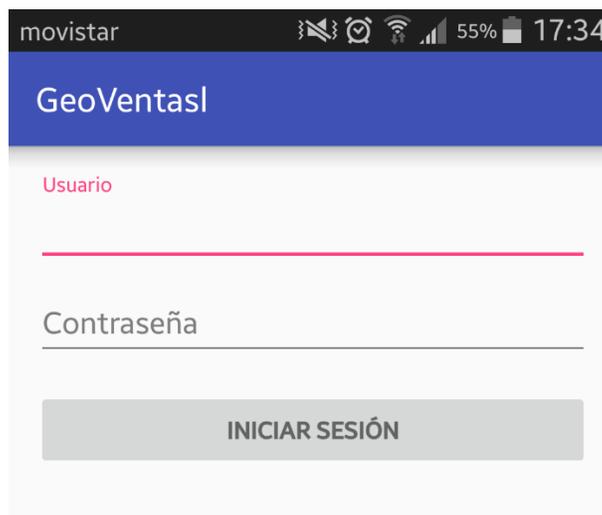
## Capítulo 4

### 4. Implementación, Documentación, Pruebas

#### 4.1 Implementación de la aplicación

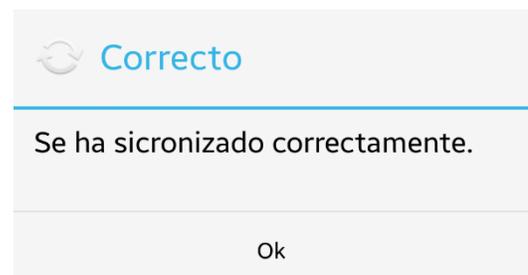
Ingresamos el usuario y la contraseña con la que cuenta cada trabajador.

**Imagen 40 Login**



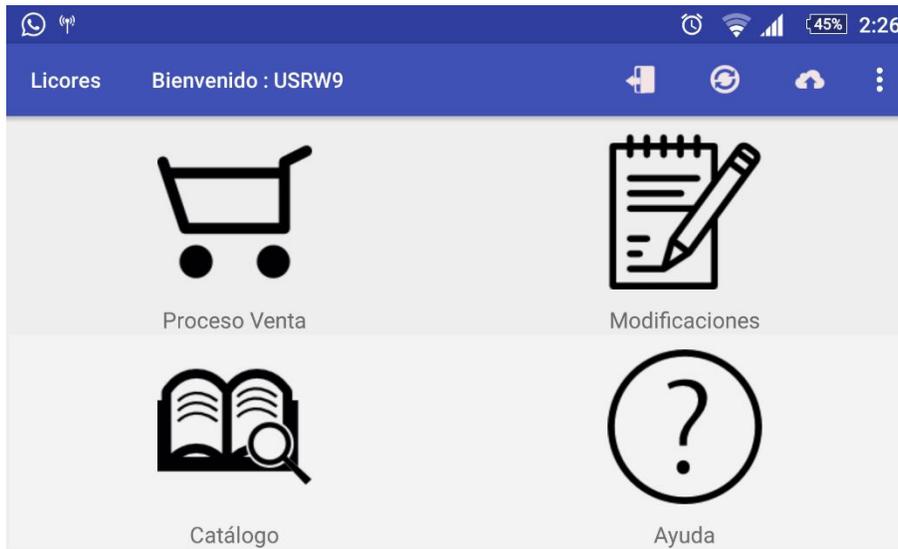
Realizado el login, la aplicación realizará la descarga de los datos necesarios para su funcionamiento.

**Imagen 41 Verificación de usuario**



Se mostrará el menú principal

**Imagen 42 Menú principal**



Ingresamos en Proceso de Venta en donde nos mostrará todos los locales asignados al vendedor

**Imagen 43 Menú de locales**



Seleccionamos el local al que se desea realizar el proceso de venta

### Imagen 44 Selección de proceso

Datos Cliente ★

**010000348**

RUC/CI :0103946711  
 Crédito  
 Cliente tiene Crédito a 30 días  
 Cupo \$ 5.23  
 Canal  
 Canal 1.1 OFF TR  
 SubCanal 2.2 CONVEN  
 Max Descuento : 5.23  
 Ciudad : 5.23

**MARIA2**

ZAVALA TORRES MARIA  
 AUGUSTA  
 TURUHUAYCO 3-76 Y  
 CABOGAN  
 4088989

INICIO PROCESO

CERRAR PROCESO

Seleccionamos el proceso que deseamos realizar pulsando en el botón inicio de proceso.

### Imagen 45 Selección de Proceso 2

**Desea continuar?**

Una vez inicio el proceso, debera de terminar

Cancel      Aceptar

Datos Cliente ★

**010000348**

**Gestión**

PRESENTACION DE LA MARCA

PRESENTACION DEL PRODUCTO

REVISION DE LOS OBJETIVOS

VERIFICACION DE SALES DRIVERS

PEDIDO

**MARIA2**

ZAVALA TORRES MARIA  
 AUGUSTA  
 TURUHUAYCO 3-76 Y  
 CABOGAN  
 4088989

Para el proceso de pedido se nos desplegará la siguiente pantalla en donde podemos escoger los artículos para el pedido pulsando el botón **AÑADIR ARTÍCULOS** o a su vez si se conoce el código del artículo podemos ingresarlo automáticamente.

**Imagen 46** Selección de productos

Cantidad	Pre. Uni	Precio

Cupo	Descuento	IVA	Subtotal	Total
5.23	0.0	0.0	0.0	0.0

Seleccionamos el artículo solicitado

**Imagen 47** Catálogo

TIPO	MARCA	ARTICULOS
APERITIVO 5	ANDES	<a href="#">Codigo :10063447</a> CERVEZA CORONITA 207 ML PVP : \$1.53
BRANDY 7	BUCKLER	Stock : 183 Unidades CORONA Unidades x caja: 24.0
CERVEZA 36	CORONA	<a href="#">Codigo :10063448</a> CERVEZA CORONA EXTRA 710 ML PVP : \$3.67
COGNAC 13	HEINEKEN	Stock : 5 Unidades CORONA Unidades x caja: 12.0
CORDIALES 5	KUNSTMANN	
CREMAS 36	LATITUD CERO	<a href="#">Codigo :10063450</a> CERVEZA CORONA EXTRA 355 ML

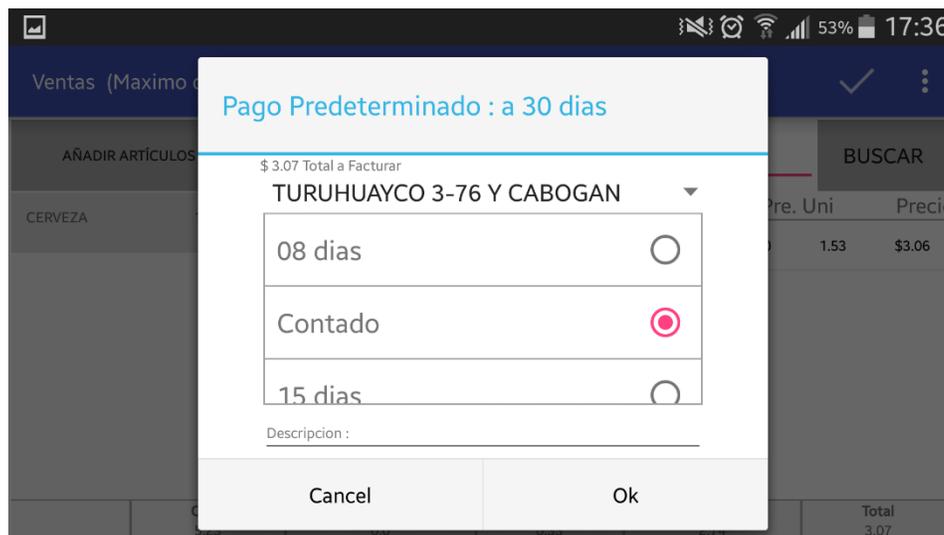
Introducimos la cantidad deseada.

**Imagen 48** Cantidad de producto



Ingresamos la forma de pago y una descripción si se cree conveniente

**Imagen 49** Forma de pago



Confirmamos la orden.

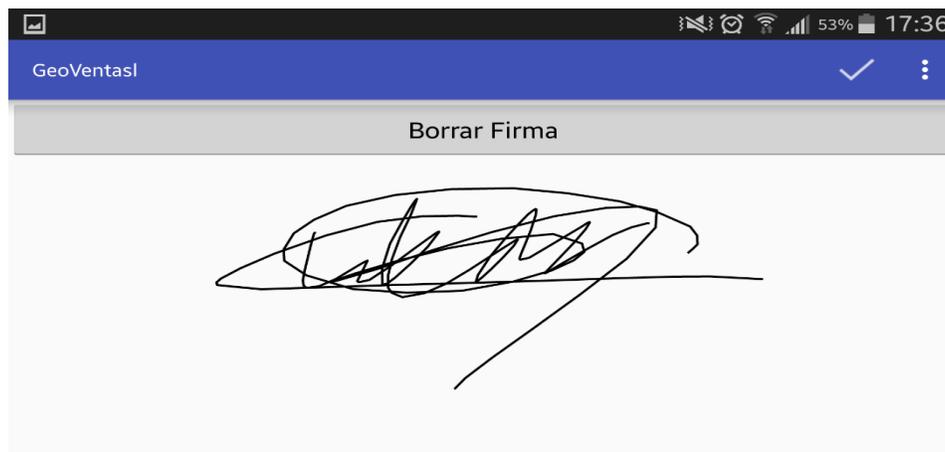
### Imagen 50 Confirmar pedido

Articulos	Cantidad	Precio	Total
CERVEZA CORONITA 207 ML	2	1.53	3.06

Subtotal : 2.74  
 Descuento : 0.0  
 Iva : 0.33  
 Total : 3.07

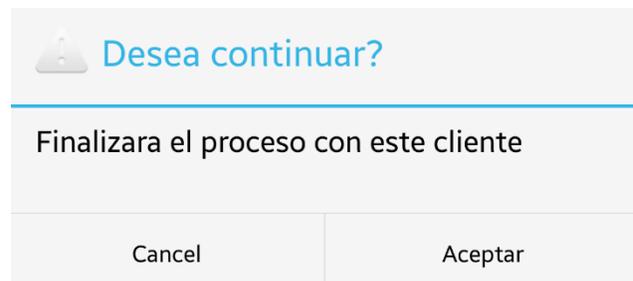
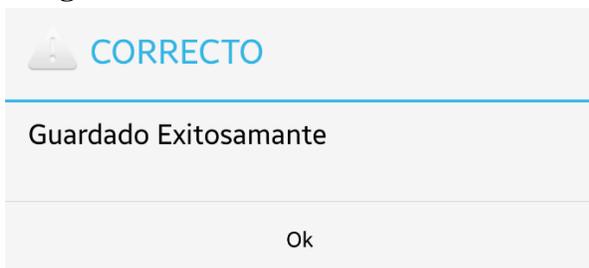
Se solicita al cliente la firma para que se tenga una constancia del pedido.

### Imagen 51 Firma



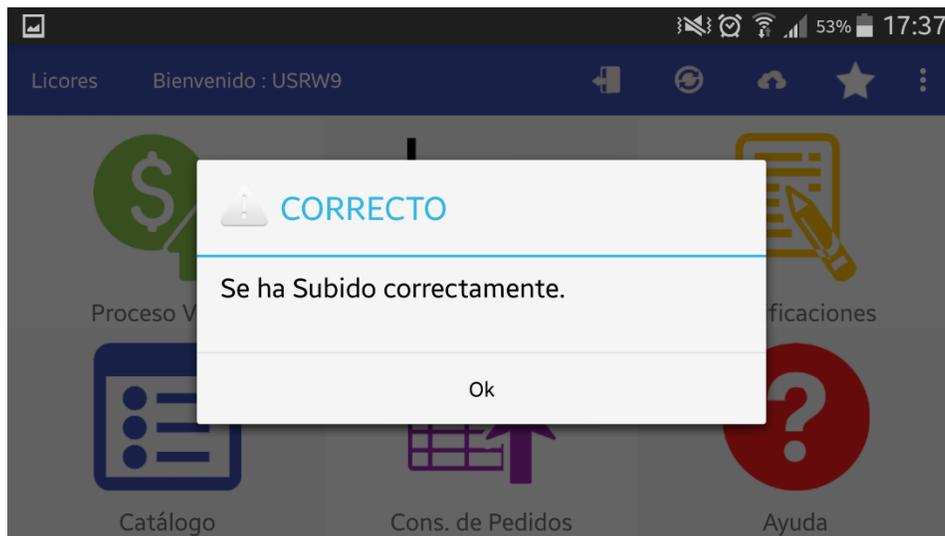
Confirmamos que el pedido se realizó.

### Imagen 52 Pedido realizado



Cerramos el proceso con el cliente, si el dispositivo posee internet los datos se sincronizarán con la base de datos central automáticamente, de lo contrario se guardará de forma temporal en la base de datos del dispositivo hasta cerrar sesión o hasta realizar el próximo pedido.

**Imagen 53** Datos sincronizados



#### 4.2 Desarrollo de pruebas con dispositivos móviles

Para el desarrollo de pruebas se utilizaron tablets y celulares los cuales cumplían con los requerimientos funcionales y no funcionales antes expuestos, se obtuvieron mejores resultados utilizando tablets debido al menor consumo de memoria RAM que estas presentan.

Se realizaron pruebas con los siguientes dispositivos obteniendo resultados satisfactorios.

## Tablet – Samsung Galaxy Tab 3

- Memoria: 16 Gb
- Pantalla: 7 pulgadas
- Procesador: 1,2 GHz doble núcleo
- Memoria RAM: 1 Gb
- Sistema Operativo: Jelly Bean

**Imagen 54** Galaxy Tab 3

## Tablet – Samsung Galaxy tab 4

- Memoria: 16 Gb
- Pantalla: 7 pulgadas
- Procesador: Quad-Core 1,2 GHz
- Memoria RAM: 1.5 Gb
- Sistema Operativo: KitKat

**Imagen 55** Galaxy Tab 4

Celular – Samsung Galaxy S4

- Memoria: 16 Gb
- Pantalla: 5 pulgadas
- Procesador: 1,6 GHz Quad + 1,2 GHz Quad (Octacore)
- Memoria RAM: 2 Gb
- Sistema Operativo: Jelly Bean

**Imagen 56** Galaxy S4



Celular – Samsung Galaxy S5

- Memoria: 16Gb
- Pantalla: 5.1 pulgadas
- Procesador: Qualcomm MSM8974AC  
Snapdragon 801 quad-core 2.5 GHz
- Memoria RAM: 2 Gb
- Sistema Operativo: KitKat

**Imagen 57** Galaxy S5



Celular – Samsung Galaxy s6

- Memoria: 32 Gb
- Pantalla: 5.1 pulgadas
- Procesador: Octaore 2.1GHz, 1.5GHz
- Memoria RAM: 3 Gb
- Sistema Operativo: lollipop

**Imagen 58** Galaxy S6



## Conclusiones

GeoVentas es una aplicación creada con la herramienta de desarrollo Android Studio, la cual permite a los agentes de ventas realizar los pedidos de forma digital y con geolocalización de cada uno de ellos, toda esta información se envía de manera automática a un servidor de base de datos para lo cual se utilizó tecnología JSON. Cuenta también con una página web creada con el lenguaje php la cual servirá para la visualización de los pedidos y el lugar en donde fueron realizados, para la geolocalización se utilizó el API de google Maps que muestra de forma gráfica el lugar donde fue realizado cada pedido, mediante el uso de estas distintas tecnologías se llegaron a las siguientes conclusiones.

Android Studio es la aplicación que se proyecta a ser la dominante en el mercado por ser un sistema operativo de código abierto y desarrollado por Google, la mayoría de desarrolladores preferirá el uso de esta herramienta debido a las facilidades que presta la misma al momento de diseño y programación.

La tecnología JSON es la que mejor se adecua a la transmisión de datos requerida, entre los dispositivos móviles y el servidor, debido a la velocidad de transferencia y al menor consumo de datos que presenta la misma.

La aplicación GeoVentas es amigable con el usuario, cuenta con una interfaz predecible razón por la cual puede ser utilizada fácilmente por el personal y no requiere de mayores explicaciones para su uso.

El excesivo consumo de energía al usar GPS representa un problema para la aplicación debido a que la batería de nuestro dispositivo no soportaría una jornada laboral completa razón por la cual se disminuyó precisión de dicho sistema.

El excesivo consumo de datos y de batería que produce la verificación periódica de la sincronización entre el dispositivo y el servidor se soluciona realizando una comprobación solo en el momento en el que el usuario hace uso del sistema al momento de enviar un nuevo pedido o cerrando la sesión.

El avance tecnológico se aplica directamente al sistema de pedidos con gran éxito, permitiendo controlar de mejor manera los recursos humanos y agilizando de manera notoria los siguientes procesos.

Con la aplicación GeoVentas fueron realizadas pruebas en febrero de 2016, dándonos como resultado que los pedidos han mejorado en un 30% en los ámbitos de control de inventarios, su-

pervisión de la fuerza de ventas, y varios procesos que no intervienen directamente con la aplicación, entre estos tenemos la aprobación de los pedidos, logística, envíos de mercadería, inventarios y la facturación.

Utilizando el método anterior de pedidos manuales se realizaban entre 80 y 90 pedidos diarios, los cuales tardaban entre 5 y 6 días en ser aprobados y entregados a los clientes, después de realizar algunas pruebas con la aplicación GeoVentas se obtuvieron de 150 a 200 pedidos al día, permitiendo una rápida aprobación y envío de los mismo, obteniendo el cliente sus productos entre 2 y 3 días después de haber realizado el pedido.

## Bibliografía

Ableson, F., Collins, C., & Sen, R. (2009). *Unlocking Android: A Developer's Guide*. Manning Publications.

*AcademiaAndroid*. (11 de 12 de 2014). Obtenido de <http://academiaandroid.com/android-studio-v1-caracteristicas-comparativa-eclipse/>

Balaguero, J. (2008). *Estudio de la plataforma Android*. España: epsc.

Buzay, G., & Ruiz, E. (2012). GEOTECNÓSFERA. TECNOLOGÍAS DE LA INFORMACIÓN. *ANEKUMENE*, 4, 88-106.

Durán, Á., Peinado, L., & Alosnso, A. (2015). *Comparación de dos tecnologías de desarrollo de aplicaciones móviles desde la perspectiva del rendimiento como atributo de calidad*. Colombia: Scientia et Technica.

FedeProEx. (16 de Mayo de 2013). *ANDROCODE*. Obtenido de <http://androcode.es/2013/05/android-studio-el-nuevo-ide-para-android/>

Goggin, G. (2006). *Cell Phone Culture Mobile technology in everyday life*. New York: Routledge.

*IformAndroid*. (12 de Septiembre de 2014). Obtenido de <http://androideno.blogspot.com/2014/09/sistemas-operativos-android.html>

Invarato, R. (17 de 03 de 2012). *Jarroba*. Obtenido de <http://jarroba.com/instalar-el-sdk-de-android-y-vincularlo-con-eclipse/>

- López Michelone, M. (23 de Septiembre de 2013). *UNOCERO*. Obtenido de <https://www.unocero.com/2013/09/23/la-historia-de-android/>
- Masumoto, Y. (1992). Global Positioning System. 1-10.
- Ocaña, S. (s.f.). *Telefonia Celular(Móvil) funcionamiento y generaciones*. malxmail.com.
- Purcell, K. (November 2 2011). *Half of adult cell phone owners have apps on their phones*. Washington: pewinternet.org.
- Ruano, S. (s.f.). Nanotecnología. *Razón y Palabra*, 68, 1-16.
- Sabater, C. (2015). *No, sin mi móvil. Diferencias de genero y uso de las nuevas Tecnologías*. ICONO.
- Tecnología y Actualidad*. (16 de 10 de 2014). Obtenido de <http://blogs.uab.cat/tecnologiayactualidad/2014/10/16/cuantas-apps-tenemos-en-play-store-android/>
- Torres, C. (6 de Junio de 2014). *Andriod SIS*. Obtenido de <http://www.androidsis.com/la-verdadera-historia-de-android-nacimiento-del-sistema-operativo-2003/>
- Vacas Aguilar, F. (2007). *Mobile Telephony: The Fourth Window*. 1-19.

## **Anexos**

Aplicación GeoVentas