



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: ANÁLISIS Y ESTUDIO DE HERRAMIENTAS LIBRES
PARA EL DESARROLLO DE APLICACIONES PARA
DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO
ANDROID APLICADO EN UN SISTEMA DE GESTIÓN DE
PEDIDOS PARA LA COMERCIALIZADORA “EL EDÉN”
UBICADA EN LA CIUDAD DE LATACUNGA.**

AUTORA: DIANA KATHERINE VELASCO GUANOLUISA

DIRECTOR: ING. MARCELO ÁLVAREZ

CODIRECTOR: ING. JAVIER MONTALUISA

LATACUNGA

2015

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE**CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA****CERTIFICADO**

ING. MARCELO ÁLVAREZ

ING. JAVIER MONTALUISA

CERTIFICAN

Que el trabajo titulado “ANÁLISIS Y ESTUDIO DE HERRAMIENTAS LIBRES PARA EL DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID APLICADO EN UN SISTEMA DE GESTIÓN DE PEDIDOS PARA LA COMERCIALIZADORA EL EDÉN UBICADA EN LA CIUDAD DE LATACUNGA” realizado por la Srta. Diana Velasco ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Universidad de las Fuerzas Armadas-ESPE.

Debido al avance de tecnologías de desarrollo hoy por hoy es la movilidad y más que relevante aplicaciones que se ejecutan sobre un Sistema Operativo Android recomiendan su publicación.

El mencionado trabajo consta de un documento empastado y un disco compacto el cual contiene los archivos en formato portátil de Acrobat (pdf). Autorizan a la Srta. Diana Velasco que lo entregue a Ing. Lucas Garcés, en su calidad de Director de la Carrera.

Latacunga, Mayo del 2015

Ing. Marcelo Álvarez
DIRECTOR

Ing. Javier Montaluisa
CODIRECTOR

UNIVERSIDAD DE LA FUERZAS ARMADAS - ESPE
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

DECLARACIÓN DE RESPONSABILIDAD

Diana Katherine Velasco Guanoluisa

DECLARO QUE:

El proyecto de grado denominado “ANÁLISIS Y ESTUDIO DE HERRAMIENTAS LIBRES PARA EL DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID APLICADO EN UN SISTEMA DE GESTIÓN DE PEDIDOS PARA LA COMERCIALIZADORA “EL EDÉN” UBICADA EN LA CIUDAD DE LATACUNGA”, ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan en el documento y cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de mi autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Latacunga, Mayo del 2015

Diana Katherine Velasco Guanoluisa

C.C.: 0502296015

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

AUTORIZACIÓN

Yo, Diana Katherine Velasco Guanoluisa

Autorizo a la Universidad de las Fuerzas Armadas – ESPE la publicación en la biblioteca virtual de la Institución del trabajo “ANÁLISIS Y ESTUDIO DE HERRAMIENTAS LIBRES PARA EL DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID APLICADO EN UN SISTEMA DE GESTIÓN DE PEDIDOS PARA LA COMERCIALIZADORA EL EDÉN UBICADA EN LA CIUDAD DE LATACUNGA”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.

Latacunga, Mayo del 2015

Diana Katherine Velasco Guanoluisa

C.C.: 0502296015

DEDICATORIA

Dedico este trabajo de investigación principalmente a Dios, por el grandioso regalo de la vida, por la fortaleza del día a día para continuar, por la humildad que de mi corazón nace al permitirme haber llegado hasta este momento tan importante de mi formación profesional y por mi maravillosa familia que siempre fueron un pilar fundamental y un apoyo incondicional en todo momento, sin importar nuestras diferencias de opiniones.

A mis padres, que han sabido formarme con excelentes valores y buenos sentimientos, lo cual me ha ayudado a salir adelante en los momentos más difíciles, pero sobretodo, a mi madre, porque la amo y por ser una prueba viviente de cariño, lucha y abnegación digna de reconocimiento y merecedora de cualquier sacrificio.

AGRADECIMIENTO

Mi agradecimiento infinito a Dios por ser mi guía y fortaleza en este arduo camino a lo largo de mi carrera, y por llenar mi vida de experiencias colmadas de aprendizaje y ante todo felicidad.

Mil gracias a mis padres Jorge y Evangelina por su ejemplo de vida, valioso en mi desarrollo personal, por el amor y los valores inculcados y por darme la oportunidad de tener una excelente educación.

A mis hermanos por ser mi motor y principal motivación para cumplir esta meta.

A mi familia, mis amigos, y las personas que quiero mucho, mis ángeles, por confiar y creer en mí, por el constante apoyo, paciencia y por no dejar que olvidara cómo volar y soñar a la vez.

A los ingenieros de la ESPE que me brindaron su apoyo en cada una de las actividades requeridas, al igual que a los ingenieros Marcelo Álvarez y Javier Montaluisa, que estuvieron involucrados directamente y fueron parte muy importante en el desarrollo de esta tesis de grado, les agradezco su tiempo, dedicación, confianza, apoyo, conocimientos y sobre todo su amistad.

ÍNDICE DE CONTENIDOS

PORTADA	i
CERTIFICADO.....	ii
DECLARACIÓN DE RESPONSABILIDAD	iii
AUTORIZACIÓN.....	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE DE CONTENIDOS	vii
ÍNDICE DE TABLAS	xii
ÍNDICE DE GRÁFICOS	xiii
RESUMEN	xvi
ABSTRACT	xvii
INTRODUCCIÓN	xviii
CAPÍTULO I.....	1
GENERALIDADES DE LA INVESTIGACIÓN	1
1.1. Antecedentes	1
1.2. Área de influencia	2
1.3. Planteamiento del problema.....	2
1.4. Justificación e importancia	3

1.5.	Objetivos	5
1.5.1.	Objetivo General	5
1.5.2.	Objetivos Específicos	5
1.6.	Hipótesis	5
1.7.	Variables	6
1.7.1.	Variable Independiente	6
1.7.2.	Variable Dependiente	6
 CAPÍTULO II.....		7
CONCEPTOS Y DEFINICIONES.....		7
2.1.	Comunicaciones móviles.....	7
2.1.1.	Servicios Interpersonales	7
2.1.2.	Servicios no interpersonales	8
2.2.	Dispositivos móviles	8
2.2.1.	Aplicaciones móviles	9
2.3.	Software libre	10
2.4.	Sistema operativo.....	11
2.5.	Sistemas operativos para dispositivos móviles	11
2.5.1.	Android	12
2.5.2.	iOS (anteriormente denominado iPhone OS).....	12

2.5.3. BlackBerry OS.....	13
2.5.4. Windows Phone	13
2.5.5. Symbian OS	13
2.5.6. webOS	13
2.5.7. Bada	14
2.6. Análisis de herramientas	14
2.7. Estudio de las herramientas.....	20
2.7.1. Java	20
2.7.2. JDK	21
2.7.3. SDK de Android	22
2.7.4. IDE de programación	22
2.7.5. Máquina Virtual Dalvik.....	23
2.7.6. Base de datos:	24
2.8. Metodología del diseño	24
2.8.1. DCU Diseño centrado en el usuario.....	25
2.9. Sistema operativo Android	26
2.9.1. Historia	26
2.9.2. Definiciones.....	27
2.9.3. Características de Android.....	28
2.9.4. Arquitectura de Android.....	29
2.9.5. Herramientas para el desarrollo de Aplicaciones	34

2.10. Sistemas de comunicaciones móviles.....	47
2.10.1. Funcionamiento de comunicaciones móviles.....	47
2.10.2. Comunicaciones inalámbricas WIFI.....	48
2.10.3. Aplicaciones para dispositivos móviles.....	49
2.10.4. Selección de la solución de la comunicación inalámbrica.....	51
CAPÍTULO III.....	53
DESARROLLO E IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL... 53	
3.1. Análisis.....	53
3.1.1. Requerimientos del usuario.....	53
3.2. Diseño.....	60
3.2.1. Casos de uso.....	60
3.2.2. Diagramas.....	65
3.3. Codificación e implementación.....	78
3.4. Plan de pruebas.....	80
3.4.1. Pruebas unitarias.....	80
3.4.2. Pruebas de aceptación.....	83

CAPÍTULO IV	85
CONCLUSIONES Y RECOMENDACIONES.....	85
4.1. Conclusiones.....	85
4.2. Recomendaciones.....	86
LINKOGRAFÍA	87

ÍNDICE DE TABLAS

Tabla 2.1	Benchmark de Lenguajes de Programación	16
Tabla 2.2	Benchmark de IDEs de Programación.....	17
Tabla 2.3	Benchmark de Bases de Datos Embebidas.....	18
Tabla 2.4	Comparativa de los distintos estándares de 802.11	52
Tabla 3.1	Pruebas unitarias Gestión de Clientes.....	82
Tabla 3.2	Pruebas unitarias Gestión de Productos.....	82
Tabla 3.3	Pruebas unitarias Gestión de Pedidos.....	82
Tabla 3.4	Resultados de la Prueba.....	84

ÍNDICE DE GRÁFICOS

Figura 2.1	Porcentaje de cuota de mercado	15
Figura 2.2	Estructura de directorios del JDK	21
Figura 2.3	Arquitectura de Android	29
Figura 2.4	Diagrama de la Arquitectura del S. O. Android	33
Figura 2.5	Paso 1 Instalación Java	34
Figura 2.6	Paso 2 Instalación Java	35
Figura 2.7	Paso 3 Instalación Java	35
Figura 2.8	Paso 4 Instalación Java	36
Figura 2.9	Archivo Zip de eclipse descomprimido.....	37
Figura 2.10	Ejecutar archivo eclipse.exe	37
Figura 2.11	Pantalla bienvenida	38
Figura 2.12	Archivo Zip de SDK descomprimido	38
Figura 2.13	Instalación paquetes necesarios del SDK manager.exe.....	39
Figura 2.14	Configuración plugin de Android para eclipse.....	40
Figura 2.15	Lista de herramientas a descargarse para el plugin	40
Figura 2.16	Actualizar el ADT Android en eclipse.....	41
Figura 2.17	Crear un nuevo dispositivo virtual AVD.....	42
Figura 2.18	Configurando el AVD	42
Figura 2.19	Inicializando el AVD.....	43

Figura 2.20	Pantalla inicial del emulador	43
Figura 2.21	Descargando el SQLite manager para Firefox.....	45
Figura 2.22	Abrir base de datos SQLite.....	45
Figura 2.23	Área de trabajo de SQLite	46
Figura 2.24	Arquitectura de una red GSM	48
Figura 3.1	Los elementos que intervienen en este tipo de diagrama...	60
Figura 3.2	Sistema de Gestión de Pedidos.....	61
Figura 3.3	Gestionar Inicio de Aplicación.....	62
Figura 3.4	Gestionar Datos de Clientes	62
Figura 3.5	Gestionar Datos de Productos.....	63
Figura 3.6	Gestionar enlaces de proveedores	63
Figura 3.7	Gestionar Pedidos	64
Figura 3.8	Visualizar Listado de Pedidos.....	65
Figura 3.9	Diagrama de secuencia para la Gestión de Clientes	66
Figura 3.10	Diagrama de secuencia para la Gestión de Productos.....	68
Figura 3.11	Diagrama de secuencia para la Gestión de Pedidos	69
Figura 3.12	Diagrama Secuencia Visualización de Pedidos.....	71
Figura 3.13	Diagrama de Estado.....	73
Figura 3.14	Diagrama de Clases	74
Figura 3.15	Diagrama Lógico.....	75
Figura 3.16	Diagrama Físico.....	76

Figura 3.17 Código Java equivalente al Layout ACERCA DE.. 79

Figura 3.19 Pruebas de Caja Blanca - White Box Software Testing..... 81

RESUMEN

Hoy en día, la tecnología móvil prácticamente ha invadido la vida cotidiana de las personas, haciéndonos testigos partícipes de todos y cada uno de los cambios y beneficios que conlleva. Gracias a las mejoras en cuanto a capacidad de memoria y procesamiento de datos de los dispositivos disponibles en el mercado, se presenta una gran oportunidad para desarrollar aplicaciones básicas y avanzadas para móviles, y todo se hizo factible cuando apareció Android, una herramienta libre, versátil y firme. En vista de todo lo expuesto, surge la propuesta, “ANÁLISIS Y ESTUDIO DE HERRAMIENTAS LIBRES PARA EL DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID APLICADO EN UN SISTEMA DE GESTIÓN DE PEDIDOS PARA LA COMERCIALIZADORA “EL EDÉN” UBICADA EN LA CIUDAD DE LATACUNGA”. El proyecto permitirá optimizar la administración de los pedidos que los clientes realizan a la empresa y cumplirlos de manera eficiente, aprovechando en la aplicación móvil los múltiples beneficios que presentan las herramientas a utilizarse así como la portabilidad que brinda Android, la adaptabilidad de Eclipse y el magnífico soporte y tratamiento que SQLite da a la información.

PALABRAS CLAVE:

- **SISTEMA OPERATIVO ANDROID**
- **DISPOSITIVOS MÓVILES**
- **SISTEMAS DE COMUNICACIÓN**
- **COMUNICACIONES MÓVILES**

ABSTRACT

Now a day, the mobile technology has invaded practically the everyday lives of people, making us witnesses partakers of all and each of the changes and benefits that entails. Thanks to improvements in terms of memory and data processing of the devices on the market, presents a great opportunity to develop basic and advanced mobile applications, and everything this became feasible when it appeared Android, a free tool, versatile and firm. In view of all the exposed, arises the proposal, "ANALYSIS AND STUDY OF FREE TOOLS FOR APPLICATION DEVELOPMENT FOR MOBILE DEVICES WITH ANDROID OPERATING SYSTEM APPLIED IN THE MANAGEMENT SYSTEM FOR THE DISTRIBUTOR 'EL EDÉN' LOCATED IN THE LATACUNGA CITY". The project will optimize the management of customer orders made to the company and perform them efficiently, leveraging in the mobile application, the multiple benefits that have the tools to use so: portability that provides Android, adaptability of Eclipse, excellent support and treatment that gives SQLite to the information.

KEYWORDS:

- **Android OS**
- **MOBILE**
- **COMMUNICATION SYSTEMS**
- **MOBILE COMMUNICATIONS**

**ANÁLISIS Y ESTUDIO DE HERRAMIENTAS LIBRES PARA EL
DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES CON
SISTEMA OPERATIVO ANDROID APLICADO EN UN SISTEMA DE
GESTIÓN DE PEDIDOS PARA LA COMERCIALIZADORA “EL EDÉN”
UBICADA EN LA CIUDAD DE LATACUNGA**

Basado en los datos obtenidos en el “V Estudio Anual IAB Spain Mobile Marketing: Informe de Resultados realizado por **IAB Spain**, a través de **The Cocktail Analysis**, sobre una muestra de más de 1000 personas entre 18 y 55 años” [1], se ha demostrado que el 86% de personas acceden a aplicaciones o sistemas, sean estos empresariales o de entretenimiento con mayor frecuencia a través de dispositivos móviles, además, la revista líderes de Ecuador apoyada en la información proporcionada por “la consultora Flurry Analytics, con oficinas en EE.UU., Reino Unido e India, ayuda a entender la relevancia de las 'apps': el 86% del tiempo que un usuario dedica a su 'smartphone' es para las aplicaciones; el 14% restante se va en usar un navegador de Internet” [2] fundamentando el presente proyecto en este enfoque, se concluye que con el desarrollo de aplicaciones o sistemas para dispositivos móviles se pondrá a disposición de los usuarios varios servicios, permitiéndole así, acelerar sus procesos habituales y por ende incrementar sus ingresos, todo esto, sin coartar su movilidad.

La presente tesis explica el fundamento teórico, básico y necesario para el análisis y estudio de las herramientas libres, y la construcción de un sistema de gestión de pedidos móvil que se llevará a cabo en el presente trabajo de investigación propuesto. Para ello se utiliza las siguientes fuentes: libros, revistas, Internet, además de opiniones de expertos en el tema con experiencia en desarrollo de software para estos dispositivos.

CAPÍTULO I

GENERALIDADES DE LA INVESTIGACIÓN

1.1. Antecedentes

Alrededor de los años 80, el software en general, disponible para el desarrollo de aplicaciones era propietario lo que hacía que los costos sean altos y muy limitados, además su código también fue privatizado lo que restringía la cooperación para que las aplicaciones disponibles evolucionen para servir de mejor manera a los usuarios y se ajusten a sus requerimientos.

En vista de esto surgieron como respuesta a esta problemática las herramientas libres y de código abierto que facilitaron el desarrollo de las aplicaciones a gran escala, permitiendo que los usuarios satisfagan sus necesidades de software a la medida.

La Comercializadora “El Edén” nació desde hace más de 30 años, en los cuales su labor ha sido decorar y dar alegría a la vida de las personas gracias a las flores que comercializa, para esto ha brindado siempre un gran servicio a todos sus clientes los cuales han sabido agradecer todos y cada uno de los detalles recibidos. A pesar de su efectivo esmero sus actividades no están muy a la par de la tecnología, necesaria en la actualidad puesto que hoy por hoy existen dispositivos que facilitarían y mejorarían cada una de las gestiones a realizarse en la empresa.

1.2. Área de influencia

Las instituciones hoy en día, han sentido la necesidad de hacer que la tecnología sea parte primordial de su administración, y para avanzar de forma paralela han debido adaptar su entorno y sus procesos. El presente proyecto está dirigido a la Comercializadora El Edén, ubicada en la ciudad de Latacunga, Provincia de Cotopaxi; y permitirá optimizar la Gestión de Pedidos realizados por sus clientes. En vista de lo expuesto se determina que su área de influencia de manera directa será la empresa dentro de su ubicación, puesto que la aplicación móvil estará en manos de la persona administradora y el área de influencia indirecta será los clientes en cada una de sus localidades, dentro o fuera de la provincia porque mejorará el servicio para con ellos.

1.3. Planteamiento del problema

En la actualidad la tecnología presenta muchos recursos para que nuestro trabajo mejore de manera eficaz, pero así como la tecnología avanza a pasos agigantados, las personas nos alejamos más de estar a la par con esos avances porque inmersos en los problemas cotidianos como falta de tiempo, cumplir con los pedidos exigentes de los clientes, fluctuación de la demanda de los productos en el mercado, entre otras y perdemos el sentido de autoeducación y restringimos a estos dispositivos, que nos facilitarían la vida, a un simple uso básico, es decir los subutilizamos, sin tomar en cuenta las ventajas reales que podemos explotar para nuestro beneficio y principalmente el de la empresa.

Desde el anterior punto de vista, es posible decir que la Comercializadora “El Edén”, en su largo camino y hasta la presente fecha ha evolucionado de tal manera que se hace evidente la necesidad de sistematizar sus procesos, principalmente la gestión de pedidos, puesto que actualmente existe la facilidad de los dispositivos móviles y sobretodo la necesidad de la portabilidad para adaptarse a un mundo competitivo, logrando así que sus pedidos se los realice de manera cumplida, eficiente y en cualquier lugar, como lo requiere la propietaria, a la que sus labores diarias como viajar al control de calidad de los productos en su lugar de procedencia, no le permiten estar estática dentro de su empresa y sin embargo debe receptar pedidos en el lugar en donde esté, siendo notorio que no siempre contará con la disponibilidad de un lápiz y un papel para recibir los pedidos o peor aún, las hojas volantes que llevaría a la mano corren el riesgo de perderse, esto pone en evidencia la principal problemática del presente proyecto.

1.4. Justificación e importancia

Los dispositivos móviles representan una de las innovaciones tecnológicas de mayor relevancia en la actualidad y gracias a su masificación, las empresas han logrado romper la brecha digital, relacionando a sus usuarios comunes y llegando a nuevos usuarios, para esto han invertido mucho tiempo y sobretodo dinero en el desarrollo de aplicaciones que se enfoquen al modelo de negocio de sus clientes, para así conquistar todo el potencial mercado que los rodea.

El desarrollo de aplicaciones propias y a la medida, ofrece muchas oportunidades para una empresa, entre otras, sitúan a la compañía a la vanguardia en tecnología y sistemas de información, reforzando su visibilidad y presencia en el mercado. Además de contribuir a ampliar mercado, las estrategias de desarrollo de estas aplicaciones permiten tener la información siempre actualizada al estar siempre en línea.

Por otra parte, muchas aplicaciones están diseñadas para ahorrar costes a las empresas, por ejemplo sustituyendo la toma de datos en papel por una aplicación que, además los almacena de forma inmediata. Así se ahorra en el acceso remoto a la información.

También sirven para mejorar la calidad de servicio a los clientes gracias a una atención inmediata y efectiva. En cuanto a ventajas internas para la compañía, ayudan a aumentar la productividad de sus empleados por agilizar los procesos.

En base a lo expuesto, gracias a los conocimientos adquiridos y a la disponibilidad de la tecnología, lo más adecuado para la Comercializadora “El Edén” es sistematizar su principal proceso para el servicio de sus clientes que es el de Gestión de Pedidos, tomando en cuenta que debido a que existen las herramientas y los dispositivos el costo será el mínimo.

1.5. Objetivos

1.5.1. Objetivo General

Analizar y estudiar las herramientas libres para desarrollar aplicaciones para dispositivos móviles con Sistema Operativo Android proyectando los resultados en un sistema de gestión de pedidos para la Comercializadora “El Edén” ubicada en la ciudad de Latacunga.

1.5.2. Objetivos Específicos

- Sintetizar el marco teórico acerca de las herramientas libres para el desarrollo de aplicaciones móviles.
- Instalar y configurar la herramienta más adecuada para el desarrollo de las aplicaciones sobre un S.O. Android.
- Analizar y diseñar los requerimientos para la gestión de pedidos.
- Codificar y probar la aplicación móvil para gestión de pedidos de la Comercializadora “El Edén” ubicada en la ciudad de Latacunga.
- Implementar el sistema de gestión de pedidos.

1.6. Hipótesis

¿El Sistema de Gestión de Pedidos aprovechará los beneficios que proporcionan las herramientas libres y los dispositivos móviles?

Los beneficios brindados por las herramientas libres combinados con los dispositivos móviles serán completamente aprovechados para hacer eficiente el trabajo de la Comercializadora “El Edén”, ya que mejorará la

administración de los pedidos de la extensa nómina de clientes de diferentes ciudades, adaptándose al complicado ritmo de trabajo de la empresa.

1.7. Variables

1.7.1. Variable Independiente

Mejora de la eficiencia de la Gestión de Pedidos

1.7.2. Variable Dependiente

Sistema de Gestión de Pedidos

CAPÍTULO II

CONCEPTOS Y DEFINICIONES

2.1. Comunicaciones móviles

“Las comunicaciones móviles permiten que un usuario pueda utilizar servicios de telecomunicaciones mientras se desplaza a lo largo de un territorio. La comunicación móvil es posible gracias a tecnologías como antenas, satélites, repetidores de señal y para ello debe existir un emisor y un receptor para que la comunicación sea posible.” [3]

Los diferentes servicios que se prestan en movilidad se pueden clasificar en dos grandes grupos:

2.1.1. Servicios Interpersonales

El mejor ejemplo es la telefonía móvil, porque en este tipo de servicio los corresponsales de la comunicación son quienes aportan el contenido. La evolución de la tecnología ha permitido la propagación de nuevos servicios interpersonales de comunicación en movilidad: a través de mensajería instantánea, correo electrónico, chats y juegos a distancia. Hoy en día la telefonía es el servicio más utilizado de los que ofrecen los sistemas de comunicaciones móviles.

2.1.2. Servicios no interpersonales

En este tipo de servicios, no hay una persona corresponsal o se utilizan contenidos que están proporcionados por terceros, como las aplicaciones empresariales, la televisión móvil o el acceso a internet.

Actualmente la comunicación móvil está en auge en todo el mundo debido a que la tecnología permite sorprender en tiempo record, con la creación de varios aparatos móviles que brindan múltiples utilidades a una persona que puede estar comunicada en cualquier lugar y de acuerdo a la cobertura del dispositivo.

2.2. Dispositivos móviles

“Los dispositivos móviles también conocidos como computadora de mano, son aparatos de tamaño pequeño, con algunas capacidades de procesamiento, con conexión intacta a una red, con memoria limitada, diseñados específicamente para una función, pero que pueden llevar a cabo otras funciones más generales, como transmitir señales acústicas por medio de señales eléctricas a distancia.” [4]

Al hablar de dispositivos móviles lo primero que se nos viene a la mente son teléfonos celulares, pero también son considerados móviles, las tablets, laptops, DVD portátil, etc.

La gama de dispositivos móviles es casi infinita, cada uno con una gran capacidad y velocidad para elegir, sobre todo por el impacto que estos dieron al mundo en cuanto a portabilidad.

2.2.1. Aplicaciones móviles

Una aplicación móvil es un software o aplicación informática, escrita para dispositivos móviles (teléfonos inteligentes, tablets, etc.) que realizan una tarea específica, como un juego, un calendario, un reproductor de música, un sistema de pedidos, etc. Existen dos tipos de aplicaciones móviles:

- **Aplicación nativa:** Está diseñada específicamente para ejecutarse en el firmware de la máquina y en el sistema operativo de un dispositivo, por lo general debe ser adaptada para distintos dispositivos.
- **Aplicación Web o aplicación de navegador:** De este tipo de aplicación, la totalidad o ciertos módulos se descargan de la Web cada vez que se ejecuta. Generalmente, es posible acceder desde dispositivos móviles con capacidad Web.

Existen aplicaciones móviles que se las pueden obtener de forma gratuita o también hay por las que se debe cancelar un valor, que muchas veces resulta elevado. Al igual que los sistemas de escritorio estas aplicaciones tienen sus limitaciones al momento de desarrollarlas, como tamaño del dispositivo, pantalla, software, configuraciones, entre otras.

Las aplicaciones móviles se caracterizan porque primero se las ejecuta en emuladores, esto, para probar su funcionalidad y luego en el dispositivo físico que se esté utilizando. Actualmente un gran número de personas han optado por desarrollar o crear aplicaciones móviles ya que esto genera un gran margen de ganancia con la menor inversión.

2.3. Software libre

“Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.” [5]

Ante todo esto el usuario debe cumplir cuatro libertades¹ al momento de desarrollar u obtener una aplicación sea web, de escritorio o para dispositivos móviles.

- **Libertad 0.**-La libertad de usar el programa con diferentes propósitos.
- **Libertad 1.** -La libertad de estudiar el funcionamiento del programa y adaptarlo a sus necesidades. Para esto la condición previa es el acceso al código fuente.
- **Libertad 2.** -La libertad de redistribuir copias a quienes lo requieran.
- **Libertad 3.** -La libertad de modificar el programa y hacer públicas las reformas a los demás, de modo que toda la comunidad se beneficie. El acceso al código fuente es un requisito previo para realizarlo.

Es software libre solo si cumple con las libertades antes mencionadas. Para que las libertades de hacer modificaciones y de publicar versiones

¹ La libertad de poseer un código fuente de una aplicación o software poderla modificarla al gusto de uno y después distribuirla con la que podamos ayudar a otra gente y sentirnos bien con uno mismo.

STALLMAN, Richard. Software libre y sus libertades. Boston, Massachusetts

mejoradas tengan sentido, se debe tener acceso al código fuente del programa. Por lo tanto, la posibilidad de acceder al código fuente es una condición necesaria para el software libre.

Software libre no significa 'no comercial'. Un programa libre debe estar disponible para su uso, desarrollo y distribución comercial. Cuando se trata de software libre, es mejor evitar términos como: “regalo” o “gratis”, porque esos términos implican que lo importante es el precio, y no la libertad misma.

2.4. Sistema operativo

Es el software que administra los recursos de hardware del sistema, sus principales funciones son ejercer control y coordinar el uso del hardware entre los diferentes programas de aplicación y sus respectivos usuarios.

El sistema operativo es la parte esencial de todo tipo de dispositivo móvil o fijo, porque actúa como intermediario entre el usuario y la aplicación que se encuentra en el dispositivo.

Un desarrollador debe seleccionar minuciosamente el sistema operativo con el cual trabajar, para evitar problemas en el transcurso del desarrollo de las aplicaciones.

2.5. Sistemas operativos para dispositivos móviles

Al igual que un computador necesita de un sistema operativo para funcionar como Windows, Linux, Mac, también lo necesita un dispositivo móvil, este es conocido como Sistema Operativo Móvil que es el encargado

de realizar el control. Las características principales son su simplicidad, que están más orientados a la conectividad inalámbrica y a la introducción de información en ellos. Entre los sistemas operativos móviles se puede encontrar los siguientes:

2.5.1. Android

“Es un sistema operativo móvil basado en Linux, que junto con aplicaciones middleware² está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos. Es desarrollado por la Open Handset Alliance basado en el Kernel de Linux, el cual es liderado por Google. El sistema de distribución de aplicaciones es Android Market.” [6]

2.5.2. iOS (anteriormente denominado iPhone OS)

Es un sistema operativo móvil desarrollado por Apple Inc., para el uso exclusivo del iPhone, aunque también sirvió para dispositivos como el iPad, iPod Touch, y el Apple TV. Apple Inc., no permite la instalación de su iOS en hardware de terceros. El sistema de distribución de aplicaciones es el App Store, el cual contiene más de 250.000 aplicaciones.

² Un middleware permite al desarrollador acceder a una base de datos que él desarrolló en cierto software a través de otros.

2.5.3. BlackBerry OS

RIM (Research In Motion), desarrolló un sistema operativo móvil multitarea, para dispositivos BlackBerry. Este sistema cuenta con diferentes métodos de entrada para su uso en smartphones, como la trackball, trackwheel, touchpad y ciertas pantallas táctiles.

2.5.4. Windows Phone

WP sucesor de la plataforma Windows Mobile, fue desarrollado por Microsoft, es un sistema operativo móvil orientado para el mercado de consumo. En este S.O., Microsoft integra varios de sus servicios propios como Xbox Live, Skype y OneDrive, para competir con el iOS de Apple y Android de Google.

2.5.5. Symbian OS

Es un sistema operativo para terminales móviles propiedad de Nokia, creado para rivalizar con sus principales contendores, Palm, BlackBerry OS, Windows Mobile o Windows Phone de Microsoft. Symbian tendrá soporte únicamente hasta el 2016 porque no pudo seguirle el paso a los requerimientos de los smartphones, lo que sí pudieron satisfacer Android de Google, iOS de Apple o Windows Phone.

2.5.6. webOS

Es un sistema operativo multitarea para sistemas embebidos basado en Linux, desarrollado por Palm, ahora propiedad de Hewlett-Packard

Company. La interfaz gráfica de usuario de webOS fue diseñada para dispositivos con pantalla táctil. Incluye un conjunto de aplicaciones para el manejo de la información personal y usa tecnologías web como HTML5, JavaScript y CSS.

Los dispositivos que actualmente utilizan webOS son el Palm Pre y el Palm Pixi, así como sus versiones "Plus", el Palm Pre2, HP Pre 3 y el HP Veer, así como la tablet HP Touchpad. [7]

2.5.7. Bada

“Es un sistema operativo para smartphones tanto de gama baja como de gama alta, desarrollado por Samsung Electronics, esta empresa afirma que Bada logrará convertir teléfonos básicos en teléfonos inteligentes al reemplazar su plataforma. Se basa en el sistema operativo propiedad de Samsung SHP OS, utilizado en muchos de sus teléfonos como el Samsung Wave.” [8]

2.6. Análisis de herramientas

El desarrollo de aplicaciones móviles, hoy en día, es reconocido por la mayor parte del mundo ya sean usuarios o profesionales, que crean sistemas de información, utilizando herramientas sofisticadas, actuales y rápidas en la transferencia de información.

Al momento de analizar las herramientas de programación para dispositivos móviles se debe considerar ventajas y limitaciones que tienen

cada una de ellas para luego utilizarlas de manera correcta en el proceso de desarrollo.

A continuación se hará un análisis exhaustivo de las herramientas existentes para determinar las necesarias para el desarrollo del sistema de gestión de pedidos móvil para la comercializadora El Edén.

En primera instancia, se analizará la gama de sistemas operativos móviles y se elegirá el que mejor cuota de mercado tenga, es decir que sea lo suficientemente conocido en el mercado de desarrollo de aplicaciones, además debe presentar confiabilidad y rapidez.

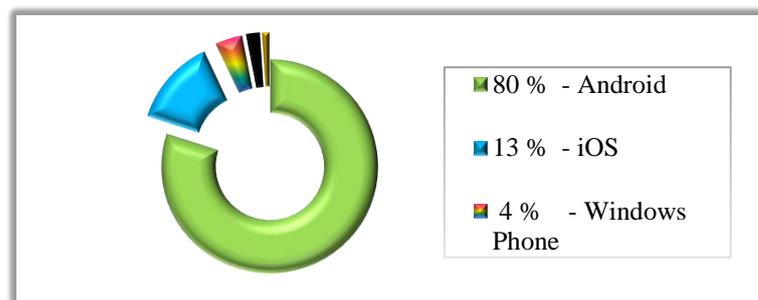


Figura 2.1 Porcentaje de cuota de mercado

Según el informe de **IDC** (International Data Corporation), para el cuarto semestre del 2013 Android se ha apoderado del 80% de la cuota del mercado mundial de dispositivos móviles y uno de los principales beneficios tangibles es su libre disponibilidad y sus extensas posibilidades en cuanto a aplicaciones se trata, por esta y muchas otras razones se ha optado por este sistema operativo para realizar el presente proyecto.

En segundo lugar, se analizará el lenguaje de programación y sus características, que deberán estar acordes a las necesidades que exija el sistema operativo a utilizar.

Tabla 2.1

Benchmark de Lenguajes de Programación

LENGUAJE DE PROGRAMACIÓN	ANÁLISIS
APP INVENTOR	<ul style="list-style-type: none"> • Software libre. • Programación Gráfica Drag and Drop. • Útil para desarrollo de aplicaciones estáticas.
BASIC4 ANDROID	<ul style="list-style-type: none"> • Software propietario • Poco soporte técnico. • Para programadores en Visual Basic. • El lenguaje es diferente al de Microsoft.
ADOBE INDESIGN CS6	<ul style="list-style-type: none"> • Software Propietario, costo según el número de publicaciones • Desarrollo para aplicaciones estáticas. • Multiplataforma.
JAVA	<ul style="list-style-type: none"> • Software libre. • Nativo de Android. • Programación orientada a objetos. • Útil para desarrollo de aplicaciones dinámicas. • Extenso soporte técnico. • Google utiliza Java por defecto.
LIVE CODE	<ul style="list-style-type: none"> • Software propietario de alto costo. • Compatible con varias plataformas: Windows Phone, iOS, Android, Linux, iPhone, iPad, Web y servidores. • Programación orientada a eventos.
MONO PARA ANDROID	<ul style="list-style-type: none"> • Software propietario. • Usa Lenguajes de programación de Microsoft .Net y C#

La opción más acertada en este caso es Java, puesto que se requiere realizar el desarrollo de la aplicación de forma nativa en la plataforma Android, además de sus múltiples ventajas y sobretodo que es de libre distribución, por lo que no implica costo alguno para su obtención.

Como tercer punto, es necesario seleccionar un IDE (Integrated Development Environment ó Entorno de Desarrollo Integrado) de programación que facilite el desarrollo de la aplicación Gestión de Pedidos, y como se va utilizar un dispositivo móvil con sistema operativo libre entonces se necesita un IDE de programación libre, con la condición que debe satisfacer las necesidades del usuario.

Tabla 2.2

Benchmark de IDEs de Programación

IDES DE PROGRAMACIÓN	ANÁLISIS
Android estudio	<ul style="list-style-type: none"> • Software Propietario • IDE basado en IntelliJ IDEA • Provoca ciertos inconvenientes al trabajar en Windows • Manejo por código en ciertas tareas.
BlueJ	<ul style="list-style-type: none"> • Software libre • Dirigido a un entorno académico • Limitado en cuanto a funcionalidad y diseño
Eclipse	<ul style="list-style-type: none"> • Software Libre. • Amplio y potente entorno de desarrollo integrado, incluso a nivel profesional. • Funcionalidad expansible mediante plugins. • Facilita el diseño gráfico.
JBUILDER	<ul style="list-style-type: none"> • Software Propietario. • Permite desarrollos gráficos.
JCreator	<ul style="list-style-type: none"> • Software exclusivamente Propietario. • Omite herramientas para desarrollos gráficos.
JDeveloper	<ul style="list-style-type: none"> • Software Propietario hasta el 2005 • Funciones deficientes • Es de Oracle.
Netbeans	<ul style="list-style-type: none"> • Software libre. • Utilizado a nivel profesional. • Facilita el diseño gráfico asociado a aplicaciones Java.

La principal razón para utilizar Eclipse es que cumple con la condición de ser software libre, además, en él se encuentran todas las funciones y herramientas necesarias para desarrollar una aplicación completa.

Cuarto punto, se precisa realizar una base de datos en la cual sea posible almacenar información importante y delicada, tanto para el desarrollador como para la empresa, teniendo en cuenta siempre, que se aplicará para dispositivos portátiles y no pc's, además es importante que la obtención de la misma no tenga costos elevados.

Tabla 2.3

Benchmark de Bases de Datos Embebidas

BASES DE DATOS EMBEBIDAS	ANÁLISIS
Apache Derby	<ul style="list-style-type: none"> • Software libre • Nativa de Java • Soporta Arquitectura Cliente – Servidor • Sólo trabaja con lenguaje de Java • No trabaja con tablas enlazadas
H2	<ul style="list-style-type: none"> • Software libre • Escrito en Java • Alta integración • Base de datos encriptada
Mysql embebido	<ul style="list-style-type: none"> • Escrito en C y C++ • Multiplataforma • Licencia GLP • Escalable • Poco fiable (transacciones, auditorias) • No ha habido mejoras

Continúa



SQLite	<ul style="list-style-type: none"> • Software Libre • Escrita en C. • Soporte de tablas e índices en un único archivo. • Soporte transaccional. • Ultra liviano • Absoluta portabilidad. • Soporta SQL. • Base de datos relacional. • Soporta perfil de usuarios. • Alto rendimiento y estabilidad • Diferentes interfaces API
SQL Server Compact	<ul style="list-style-type: none"> • SQL Server Compact únicamente soporta tipos de datos de cadena compatibles. • No soporta perfiles de usuario. • Limitado en el archivo de base de datos. • Compatible sólo con aplicaciones propietarias.

Después de presentar las características de ciertos motores de base de datos embebidas más utilizados en la actualidad, se puede determinar el uso indiscutible de SQLite, debido a sus ventajas notorias al momento de compararlas, resumiendo sus ventajas se podría decir que es compatible con ACID, esto significa que cumple con criterios de Atomicidad, Consistencia, Aislamiento y Durabilidad.

Resultado: Una vez analizadas las herramientas existentes tanto libres como propietarias, y sus respectivas características, para desarrollar el sistema de gestión de pedidos para la Comercializadora El Edén, se determinó usar las siguientes herramientas:

- Sistema operativo Android
- Lenguaje de programación Java
- IDE de programación Eclipse
- Base de datos embebida SQLite

2.7. Estudio de las herramientas

Para el desarrollo de aplicaciones para dispositivos móviles tenemos diferentes herramientas que cumplen funciones específicas ya sea como máquina virtual, base de datos, emuladores, y/o lenguajes de programación etc.

Para iniciar con la programación bajo cualquier sistema operativo, en este caso Android, es necesario empezar por el estudio de un buen lenguaje de programación.

2.7.1. Java

“Java es un lenguaje de programación orientado a objetos y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. (...) Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes”. [9]

Java es un lenguaje de programación con el que es posible realizar todo tipo de programas o aplicaciones móviles. Java es muy importante en el

desarrollo de aplicaciones para dispositivos móviles por su seguridad y eficiencia. Por estas y varias razones se determinó el uso de Java como lenguaje para el desarrollo de la presente tesis.

2.7.2. JDK

“JDK (Java Development Kit). Software que provee herramientas de desarrollo para la creación de programas en Java. Puede instalarse en una unidad de red o de forma local en un computador. En la unidad de red se pueden tener las herramientas distribuidas en varias computadoras y trabajar como una sola aplicación.” [10] A continuación se muestra una gráfica de la Estructura de directorios del JDK

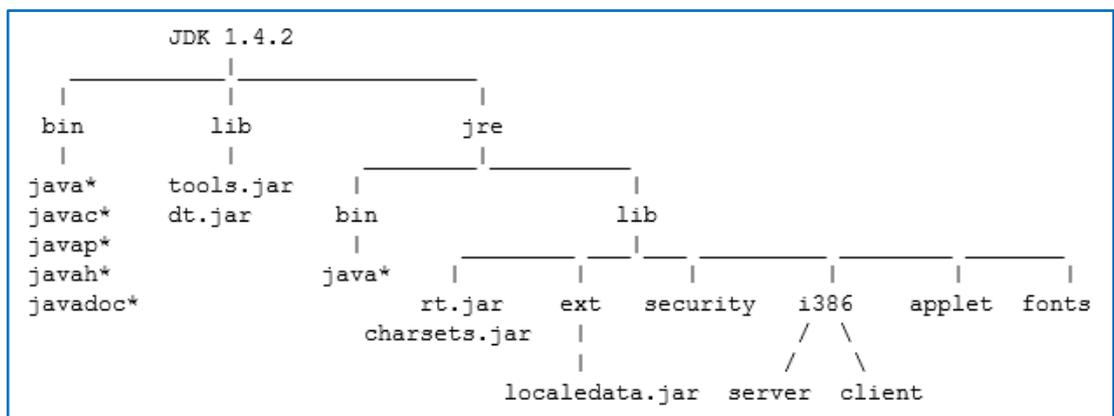


Figura 2.2 Estructura de directorios del JDK

Fuente: <http://javabasico.osmosislatina.com/cursos/objetos.htm>

Como se puede observar, la estructura se encuentra dividida en tres grandes segmentos:

- **bin:** Contiene los ejecutables para la generación de programas Java.
- **lib:** Incluye las librerías base (Clases) necesaria para la generación de la mayor parte de programas Java.
- **jre:** Comprende el ambiente preciso para ejecutar programas Java, el cual a su vez se encuentra sub-dividido en distintos directorios.

2.7.3. SDK de Android

“SDK Kit de Desarrollo de Software de las siglas en inglés Software Development Kit. Es un conjunto de herramientas de desarrollo de software que le permite al programador crear aplicaciones para un sistema concreto. Por ejemplo ciertos paquetes de software, frameworks, plataformas de hardware, computadoras, videoconsolas, sistemas operativos, etc.” [11]

Este conjunto de herramientas sofisticadas que ofrece el paquete SDK tiene muchas ventajas para su uso, ya que permite que mediante esta, el desarrollo de aplicaciones móviles, que es lo innovador al momento, sea más rápido y confiable, logrando que la mayoría de programadores opten por esta opción.

2.7.4. IDE de programación

2.7.4.1. Eclipse

“Eclipse es un IDE (*Integrated Development Environment* ó *Entorno de Desarrollo Integrado*) para el desarrollo de software de código abierto está escrito totalmente en Java permite crear aplicaciones empresariales,

móviles, etc. Una de sus principales ventajas es que además de ser multiplataforma es también multilenguaje. Permite desarrollar aplicaciones escritas en Java, C++, PHP, Ruby, Python, JavaScript y Perl.” [12]

La utilización de Eclipse en el desarrollo de aplicaciones es debido a que cuenta con un entorno amigable y permite el manejo de los objetos existentes y necesarios para el programador. Además, se caracteriza por ser un IDE sin costo de obtención y que está escrito en un lenguaje que es conocido por la mayor parte del mundo Java.

2.7.5. Máquina Virtual Dalvik

“Es la máquina virtual que utiliza la plataforma para dispositivos móviles Android. Dalvik ha sido diseñada por Dan Bornstein con contribuciones de otros ingenieros de Google.” [13]. Dalvik es una máquina virtual diseñada específicamente para dispositivos de mano es decir tablets, teléfonos inteligentes con sistema operativo Android, enfocándose principalmente en su capacidad de procesamiento y efectividad a la hora de ejecutar una aplicación.

Dalvik permite ejecuciones simultáneas de la máquina virtual ya que requiere poca memoria y delega al sistema operativo principal la gestión de memoria e hilos y el soporte de aislamiento de procesos.

Es posible afirmar que mediante el uso de un emulador con esta función se logra un dispositivo móvil práctico dentro del sistema operativo, PC o

Mac, para el desarrollo de las aplicaciones. Este emulador es conocido dentro el paquete del SDK como AVD (Dispositivo Virtual de Android).

2.7.6. Base de datos:

“Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.” [14]

2.7.6.1. SQLite

SQLite es un motor de base de datos relacionales gratis y de código abierto. A diferencia del sistema de gestión de base de datos cliente-servidor, esta no necesita de un servidor para comunicarse con los datos sino que es directamente entre la aplicación y los datos, esta es portable y puede ser embebida en cualquier tipo de software, ya sea web, de escritorio o para tecnologías móviles.

La principal ventaja de trabajar con SQLite es que no requiere de un servidor para su uso, basta con que se instale y ejecute en el dispositivo móvil para poder manipular los datos de forma correcta y sin inconvenientes.

2.8. Metodología del diseño

Conocida también como diseño metodológico, es la descripción de las estrategias y procedimientos a utilizarse en cada etapa de la investigación.

2.8.1. DCU Diseño centrado en el usuario

Es el proceso en el cual el elemento principal es el usuario, porque son quienes usan un producto o servicio y conocen sus necesidades al igual que sus objetivos y requerimientos, así, cada una de las decisiones para lograr el producto terminado están dirigidas por y hacia él, es decir se convierte en el co-creador. En este tipo de diseño la usabilidad es evaluada y mejorada durante el proceso por el diseñador, quien reúne toda la información proporcionada por el usuario y realiza el diseño para ellos, asumiendo la función de ayudar al usuario a lograr sus objetivos.

La metodología se compone de cuatro fases:

- a. Definición: Identificar a los usuarios finales, es decir las personas que utilizarán el producto.
- b. Análisis: Investigar minuciosamente los requerimientos de la empresa y los usuarios.
- c. Diseño: Generar soluciones de diseño.
- d. Evaluación: Evaluar los diseños a partir de los requerimientos de los usuarios.

MARCO TEÓRICO

2.9. Sistema operativo Android

2.9.1. Historia

RAMIREZ, Enrique: Desarrollo de Aplicaciones para dispositivos, menciona que “En Julio de 2005, Google adquirió Android, una pequeña empresa de California. En esos momentos, la compañía se dedicaba a la creación de software para teléfonos móviles. Una vez en Google, el equipo desarrolló un sistema operativo basado en Linux para dispositivos móviles. Más adelante, Google adaptó su buscador y sus aplicaciones para su uso en móviles.

En septiembre del 2007, Google tenía varias patentes de aplicaciones sobre el área de la telefonía móvil. El 5 de noviembre del mismo año, se anunció la fundación de la Open Handset Alliance al mismo tiempo que la creación de la plataforma Android.

La Open Handset Alliance está formada por un consorcio de 34 compañías de hardware, software y telecomunicaciones, entre las cuales se incluyen Google, HTC, Intel y Motorola entre otras, dedicadas a investigar estándares abiertos para dispositivos móviles.

El primer teléfono en el mercado que posee Android es el T-Mobile G1(también conocido como Dream), que viene con la versión Android 1.0 preinstalada. Este móvil es el resultado conjunto de T-Mobile, HTC y Google.

Desde el 21 de octubre de 2008, Android está disponible como código abierto.” (2009, pág. 15)

Es muy importante conocer la historia de un sistema operativo, su creador y su aplicabilidad a futuro de manera que se evidencien sus bondades y limitaciones, logrando total familiaridad con la plataforma. A continuación breves definiciones del sistema operativo Android

2.9.2. Definiciones

RAMÍREZ, Enrique: Desarrollo de Aplicaciones para Dispositivos con Sistema Operativo Android, manifiesta “Algunas de las definiciones de Android:

Android es un conjunto de herramientas y aplicaciones vinculadas a una distribución Linux para dispositivos móviles.

Android es de código abierto, gratuito y no requiere pago de licencias.

Android es una plataforma de código abierto para dispositivos móviles que está basada en Linux y desarrollada por Open Handset Alliance.

Es un paquete de software para dispositivos móviles que incluye un sistema operativo, actúa como intermediario para comunicarse con otras aplicaciones. Los desarrolladores pueden crear aplicaciones para la plataforma usando el SDK de Android.

Las solicitudes se han escrito utilizando el lenguaje de programación Java y se ejecutan en Dalvik, una máquina virtual personalizada que se ejecuta en

la parte superior de un núcleo de Linux la cual es diseñada para uso en dispositivos móviles.” (2009, pag.16)

2.9.3. Características de Android

Así mismo el autor antes mencionado hace referencia algunas de las características de Android que son las siguientes:

- Código Abierto
- Núcleo basado en el Kernel de Linux
- Adaptable a muchas pantallas y resoluciones, desde 2,5” hasta 10” (librerías de gráficos 2D y 3D)
- Almacenamiento de datos en SQLite
- Conectividad (CDMA, Bluetooth, Wi-Fi, GSM/EDGE, IDEN, EV-DO, UMTS, Bluetooth, LTE, y WiMAX)
- Mensajería (SMS y MMS)
- Navegador Web basado en WebKit incluido.
- Soporte de Java gracias a la máquina virtual Dalvik.
- Soporte de varios formatos multimedia (MPEG-4, MP3, JPG, GIF)
- Entorno de desarrollo (emulador, herramientas de depuración, perfiles de memoria y funcionamiento, plugin para Eclipse).
- Soporte de HTML, HTML5, Adobe Flash Player, etc.
- Catálogo de aplicaciones gratuitas o pagadas listas para descargar e instalar (Google Play).

- Soporte para hardware adicional y su respectivo software, así por ejemplo cámaras, GPS, aceleración 2d y 3d, entre otros avanzados elementos propios del dispositivo.
- Para realizar video llamadas usa Google Talk desde su versión Honey Comb.
- Multitarea real de aplicaciones.

2.9.4. Arquitectura de Android

La arquitectura de Android se caracteriza por estar constituida por varias capas, las mismas que facilitan al desarrollador la creación de aplicaciones. Gracias a esta distribución, al desarrollar una aplicación se puede hacer uso de los componentes de hardware de los dispositivos móviles sin tener que programar a bajo nivel las funcionalidades necesarias, ya que es posible acceder a las capas inferiores haciendo uso de las librerías.

A este tipo de arquitectura se la conoce también con el nombre de Pila porque cada capa puede efectuar sus funciones utilizando elementos de la capa inferior.



Figura 2.3 Arquitectura de Android

Fuente: <http://androideity.com/wp-content/uploads/2011/07/arquitecturaandroid.png>

A continuación se describen detalladamente cada una de las capas del sistema operativo Android en forma descendente:

Aplicaciones: Es la última capa, comprende todas las aplicaciones del dispositivo, sin importar el tipo, así: nativas, administradas, preinstaladas o que el usuario haya instalado, ya que Android entrega al usuario total control para optimizar un dispositivo a su medida.

Marco de trabajo de aplicaciones o Framework: Permite el acceso completo a los APIs del framework, es decir clases y servicios, usados por las aplicaciones base para realizar funciones, esto es posible a través de las librerías Java, permitiendo la reutilización de componentes.

Una capa de servicios disponibles para las aplicaciones incluye:

- Administrador de actividades. Gestiona la pila de actividades y el ciclo de vida de las aplicaciones.
- Administrador de ventanas. Organiza lo que se muestra en pantalla.
- Proveedores de contenidos, permite compartir los datos encapsulados entre aplicaciones, permitiendo mantener el control sobre el acceso a la información.
- Vistas del sistema. Los elementos necesarios para elaborar interfaces de usuario: botones, listas, cuadros de texto e incluso un visor de Google Maps o un navegador Web, entre otros.

- Administrador de notificaciones. Permite a todas las aplicaciones mostrar alertas personalizables en la barra de estado, incluso es posible hacer uso de sonidos, LED's o activar el vibrador.
- Administrador de paquetes. Biblioteca que permite gestionar información de paquetes instalados o a su vez instalación de nuevos
- Administrador de telefonía. Gracias a esta librería, es posible enviar o recibir mensajes de texto o multimedia, además de realizar llamadas.
- Administrador de recursos. Administra los elementos que están fuera del código pero son parte de la aplicación, así: imágenes, textos en otros idiomas, sonidos, etc.
- Administrador de ubicaciones. Trabaja directamente con el GPS o las redes disponibles y los mapas para determinar la posición geográfica del dispositivo Android.
- Administrador de sensores. Permite manipular los sensores propios del dispositivo tales como el de temperatura, proximidad, presión, luminosidad, de campo magnético, además, de la brújula, giroscopio, acelerómetro, etc.
- Cámara. Librería que permite utilizar la o las cámaras para tomar fotografías o videos.
- Multimedia. Es posible, visualizar y reproducir video, imágenes y audio en el dispositivo.

Entorno de ejecución o Runtime de Android: Está formado por un conjunto de librerías base que proporcionan muchas de las funcionalidades propias de Java y otras específicas de Android. Las aplicaciones se las codifica en Java, luego se las compila una sola vez con la extensión .dex, para garantizar su ejecución en la máquina virtual Dalvik, que es un elemento propio de los dispositivos Android.

Librerías: Android incluye librerías C/C++ tales como: SQLite, System C, librerías de gráficos 3D y de medios, entre otras. A través de ciertos componentes del sistema como el framework de aplicaciones los desarrolladores pueden hacer uso de estas capacidades.

Núcleo - Linux: Está basado en el Kernel de Linux 2.6. Actúa como una capa de abstracción entre hardware y software, gracias a esto es posible gestionar ciertos servicios del sistema como modelo de drivers, seguridad, procesos, memoria y red.

El siguiente gráfico muestra los principales componentes del sistema operativo Android:

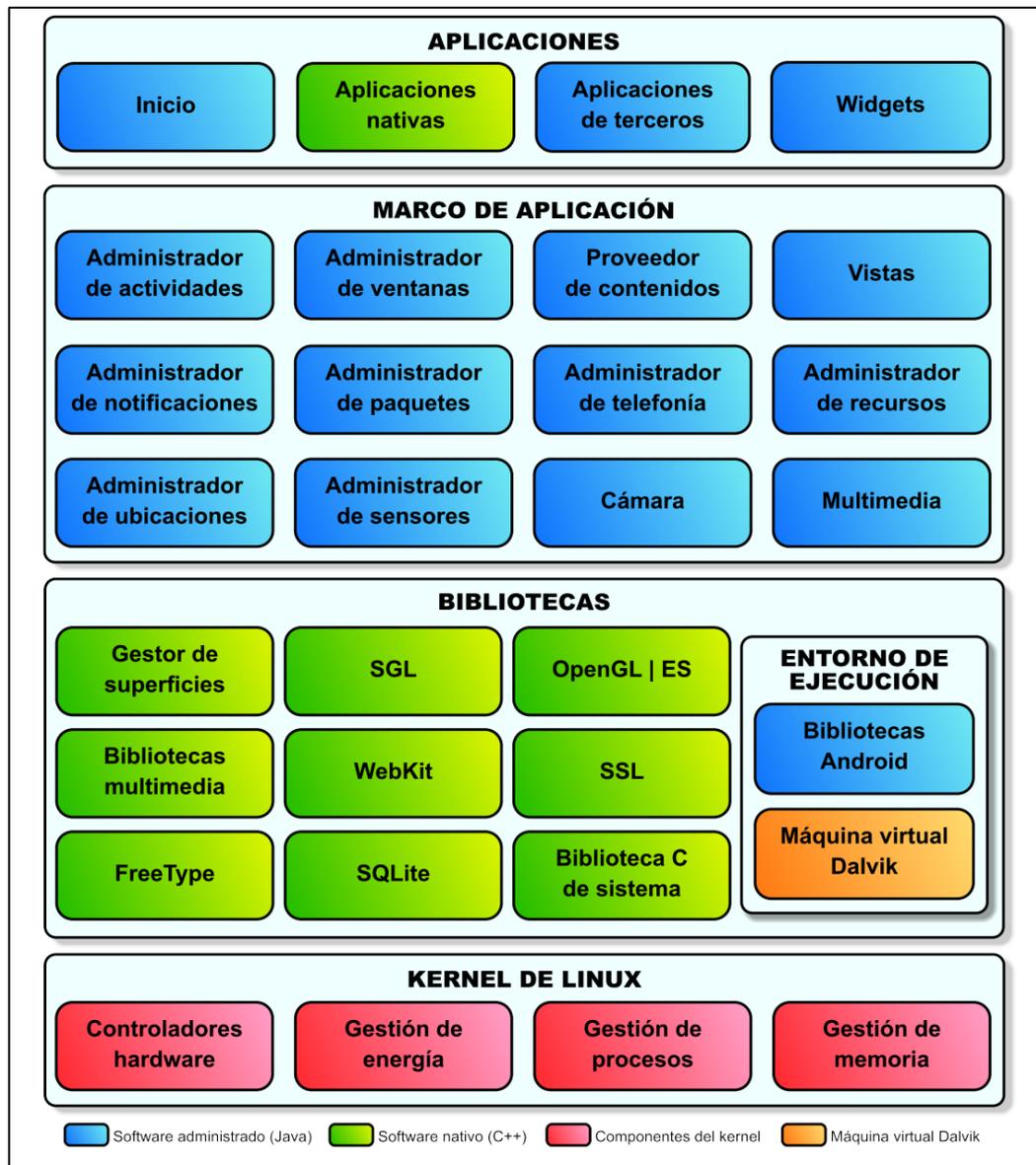


Figura 2.4 Diagrama de la Arquitectura del Sistema Operativo Android

FUENTE: <http://columna80.files.wordpress.com/2011/02/0013-01-pila-software-android.png>

2.9.5. Herramientas para el desarrollo de Aplicaciones

JDK (Java Development Kit), provee herramientas para el desarrollo de aplicaciones en Java. Contiene el compilador que convierte el código fuente en código “máquina” de la máquina virtual de Java o bytecode. Es posible instalarla en una unidad de red o en una computadora local.

Instalación y configuración JDK

- Descargar Java SE Development Kit (JDK), del sitio <http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>
- Ejecutar el archivo descargado, en la pantalla que se muestra continuación, Clic en Aceptar.



Figura 2.5 Paso 1 Instalación Java

- Aceptar el path predeterminado en donde se instalará el software.
C:/Archivos de programas/, clic en Next

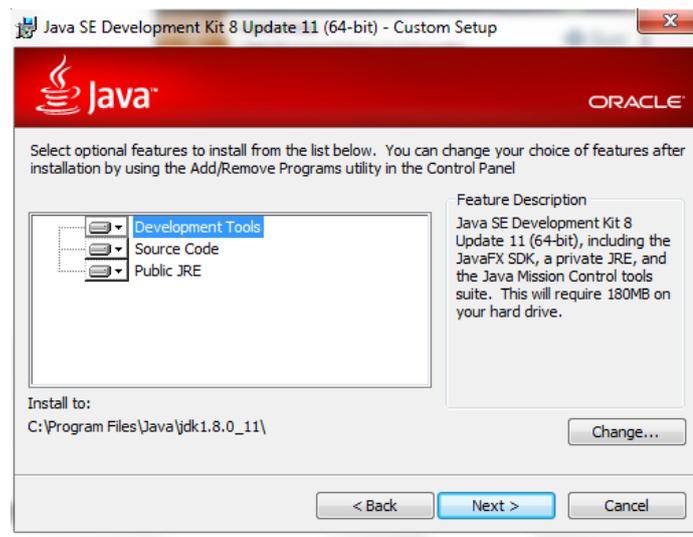


Figura 2.6 Paso 2 Instalación Java

- Automáticamente empieza la instalación de Java Development Tools:

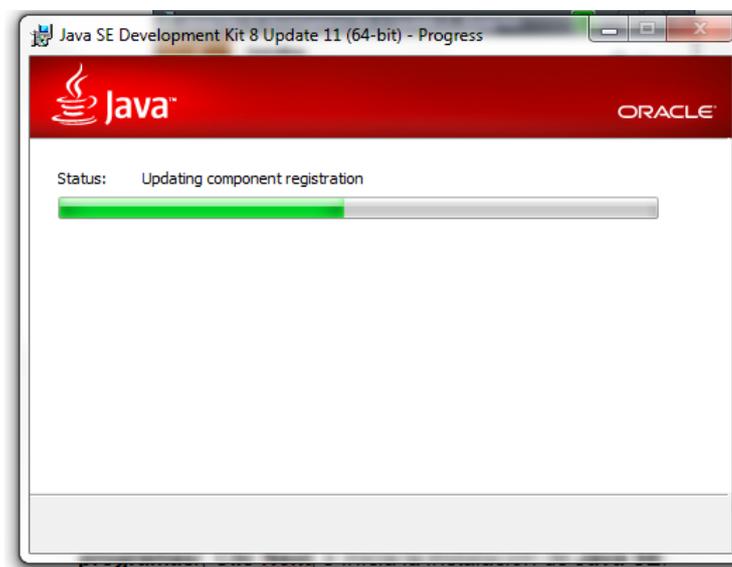


Figura 2.7 Paso 3 Instalación Java

- Solicita dirección para instalar el Java SE. Es aconsejable dejar la dirección que aparece de forma automática: **C:/Archivos de programas/**, **Clic Next**, e inicia la instalación de **Java SE**.

- Terminada la instalación, se muestra la siguiente ventana, **Clic en Finish.**



Figura 2.8 Paso 4 Instalación Java

2.9.5.1. IDE Eclipse

Eclipse es un Entorno de Programación Integrado ó IDE, es utilizado para el desarrollo de software de código abierto, está escrito totalmente en Java y gracias a sus recursos disponibles, permite crear aplicaciones empresariales, para dispositivos móviles, etc.

Instalación y configuración de eclipse

- Descargar el archivo ejecutable de:
<https://www.eclipse.org/downloads/>
- Descomprimir el archivo descargado, pegar la carpeta eclipse en el disco **C:**

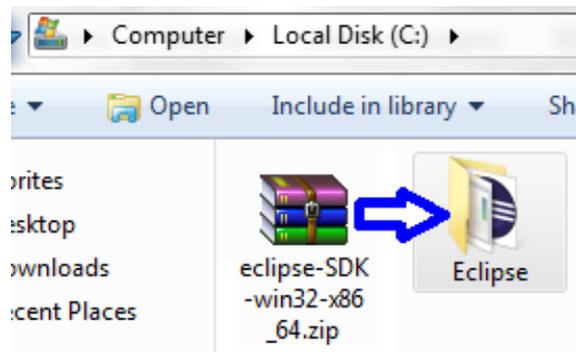


Figura 2.9 Archivo Zip de eclipse descomprimido

- Una vez pegada la carpeta en el disco **C:** ejecutar **eclipse.exe**:

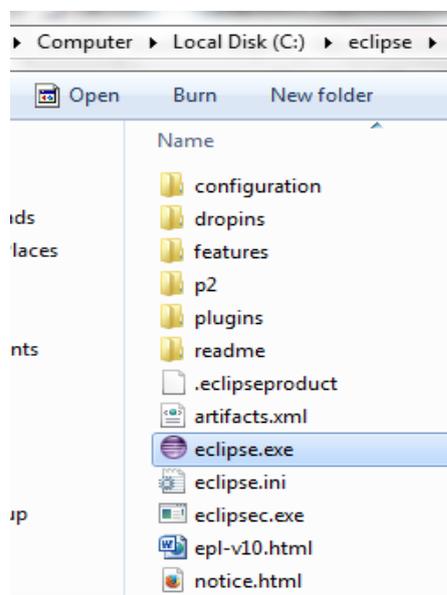


Figura 2.10 Ejecutar archivo eclipse.exe

- Al ejecutar el archivo se habilita el área de trabajo de eclipse.
- Es necesario seleccionar un área de trabajo, se recomienda conservar la carpeta por default, clic en **OK** y listo para crear un nuevo proyecto Android.

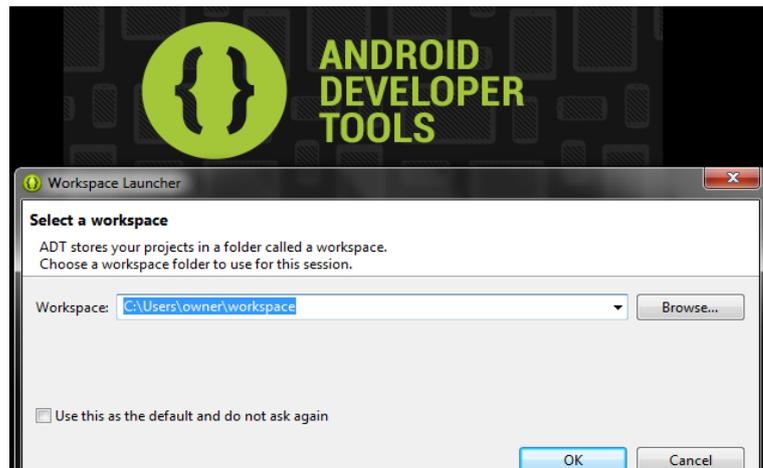


Figura 2.11 Pantalla bienvenida

2.9.5.2. SDK Software Development Kit

Conocida también como Plataforma de Desarrollo de Software, es un conjunto de compiladores, Intérpretes, librerías, herramientas de desarrollo, códigos de ejemplo y ayudas, que sirven para crear aplicaciones completas y concretas. Es la máquina virtual de Android para Java

Instalación SDK Manager

- Descargar el instalador del **SDK**.
- Descomprimir el archivo en la carpeta **SDK** en el disco **C:**

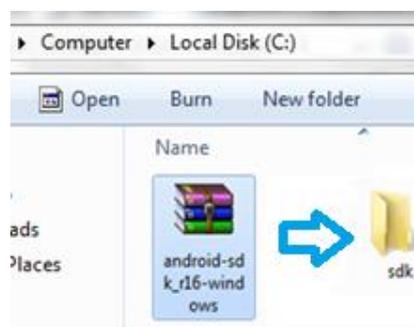


Figura 2.12 Archivo Zip de SDK descomprimido

- Abrir la carpeta y clic en **SDK de Manager.exe**
- Clic en **Instalar** para los paquetes recomendados. El Administrador de Android SDK instala los paquetes seleccionados en su entorno de Android SDK.

Nota: Para la instalación de los paquetes la computadora debe tener conexión a internet para las descargas.

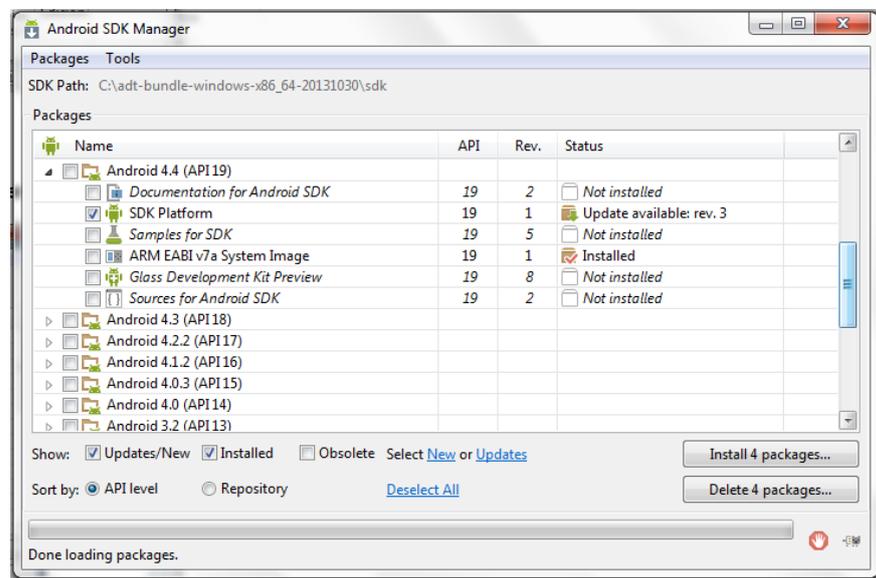


Figura 2.13 Instalación paquetes necesarios del SDK manager.exe

Instalación plugin ADT Android para Eclipse

- Abrir eclipse menú **Help->Install New Software**. Clic en **Add** y agregar :

Name: Android ADT

Location: <https://dl-ssl.google.com/android/eclipse/> y **OK**.

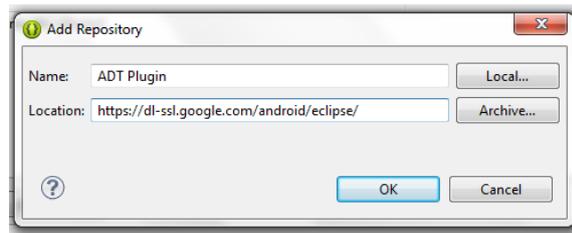


Figura 2.14 Configuración plugin de Android para eclipse

- Aparecerá una lista con el elemento "**Developer Tools**". Y las herramientas que serán descargadas. Clic en el botón **Next**.

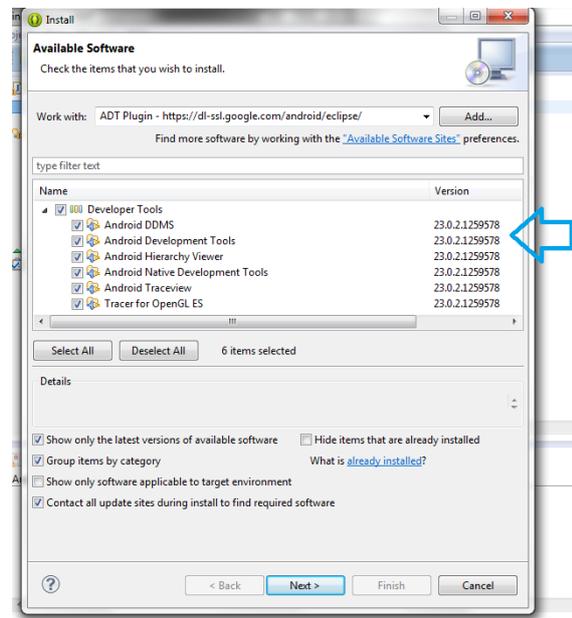


Figura 2.15 Lista de herramientas a descargarse para el plugin

- Aceptar los acuerdos de la licencia **Finish**, **OK** y reiniciar Eclipse.

Configuración plugin ADT en Eclipse

En eclipse se configura el ADT y se actualiza los datos de la ubicación del SDK.

- Abrir eclipse menú **Windows->Preferences ->opción Android**
- En **SDK Location:** Introducir el path de la carpeta SDK de Android.
- Clic en **Apply** y luego **OK**.

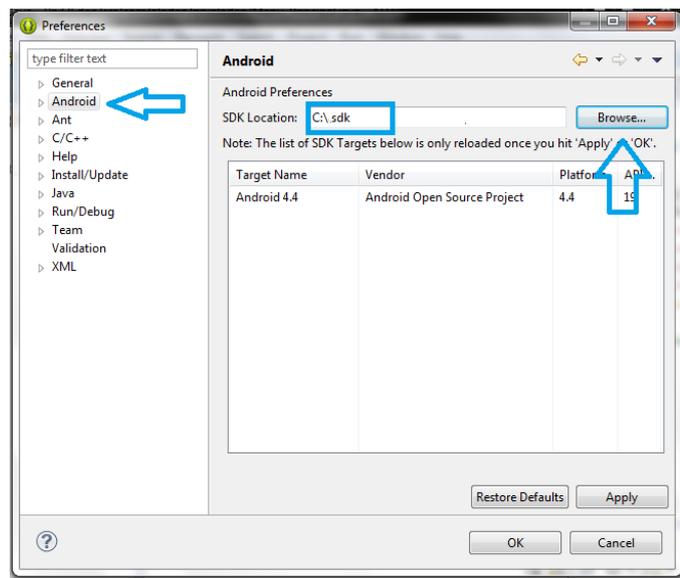


Figura 2.16 Actualizar el ADT Android en eclipse

Configuración AVD manager

Finalmente, crear un dispositivo virtual de Android que emule el funcionamiento del mismo. De esta forma se podrá ejecutar las aplicaciones.

- Abrir Eclipse menú **Window->AVD Manager**. Seleccionar la opción **New**.

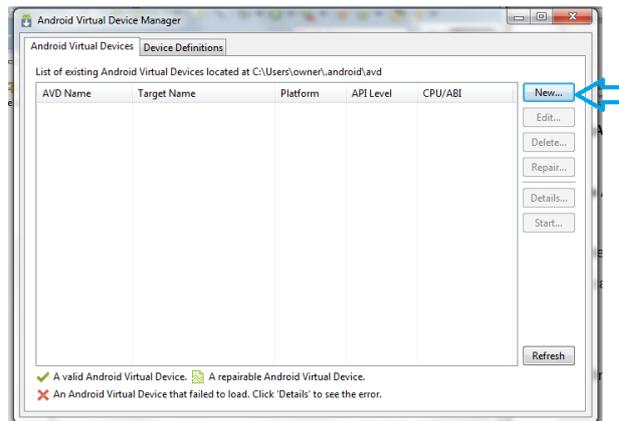


Figura 2.17 Crear un nuevo dispositivo virtual AVD

- Clic en Create new Android Virtual Device (AVD) llenar los datos:

Name: Android4.4

Target: Indica la versión de Android a emular 4.4

- Clic en **OK** para finalizar

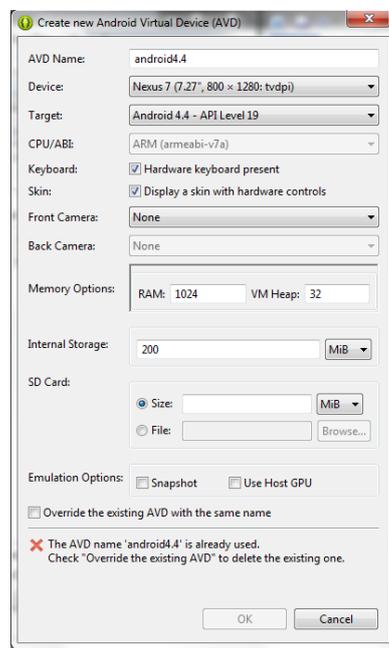


Figura 2.18 Configurando el AVD

Aparecerá la siguiente pantalla. Para iniciar, dar clic sobre el botón **Start**

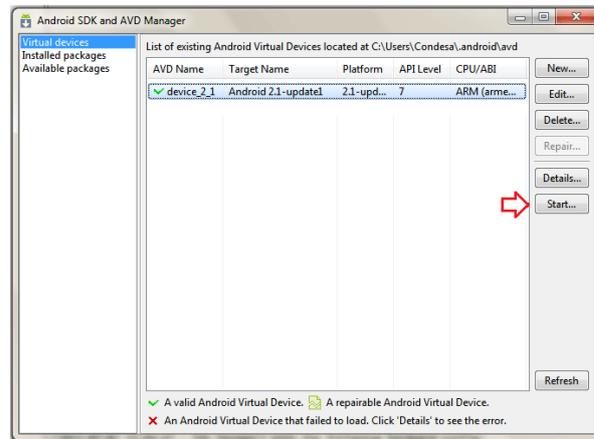


Figura 2.19 Inicializando el AVD

- Una vez configurado todo aparecerá la pantalla del dispositivo virtual donde se ejecutaran las aplicaciones

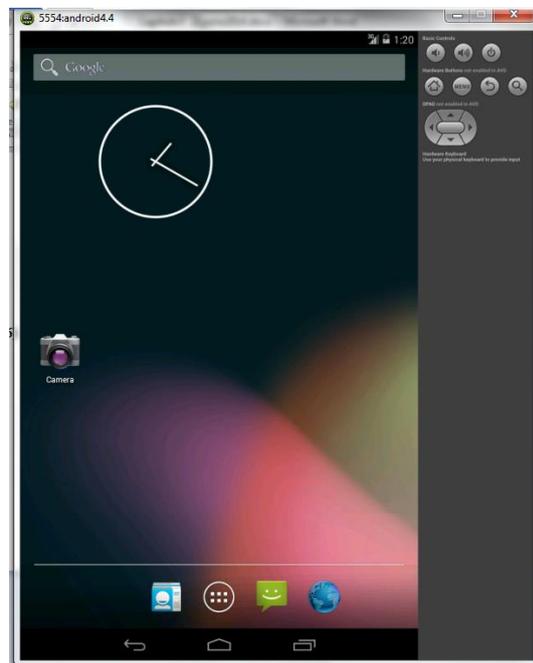


Figura 2.20 Pantalla inicial del emulador

2.9.5.3. BASE DE DATOS SQLite

INTEF: En Servicio a los usuarios, menciona que “SQLite es una pequeña librería programada en lenguaje C que implementa un completo motor de base de datos multiplataforma que no precisa configuración. Se distribuye bajo licencia de dominio público. Es muy rápido y la ventaja fundamental es que permite utilizar un amplio subconjunto del lenguaje estándar SQL. SQLite destaca también por su versatilidad. El motor de PHP 5 incluye soporte interno para SQLite.

Combina el motor y el interfaz de la base de datos en una única biblioteca, y almacena los datos en un único archivo de texto plano. Esto hace que cada usuario pueda crear tantas bases de datos como desee sin la necesidad de la intervención de un administrador de bases de datos que gestione los espacios de trabajo, usuarios y permisos de acceso. El hecho de almacenar toda la base de datos en un único archivo, facilita la portabilidad de los datos, y solamente tiene la restricción del espacio de disco asignado al usuario en el servidor.

Su potencia se basa fundamentalmente en la simplicidad, lo que hace que no sea una buena solución en entornos de tráfico muy elevado y/o alto acceso concurrente a datos. SQLite encapsula toda la base de datos en un único fichero.” [15]

Configuración SQLite

- Abrir Firefox menú ->**herramientas** escribir el nombre del plugin SQLite una vez encontrada instalar y saldra una pantalla como la siguiente luego reiniciar el firefox

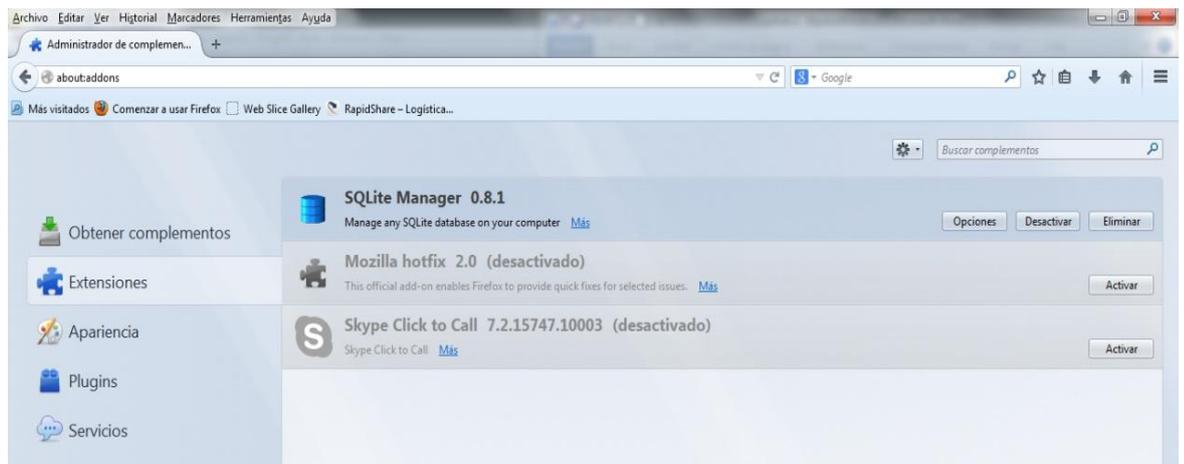


Figura 2.21 Descargando el SQLite manager para Firefox

- Para abrir la base menú **Herramientas**->SQLite manager

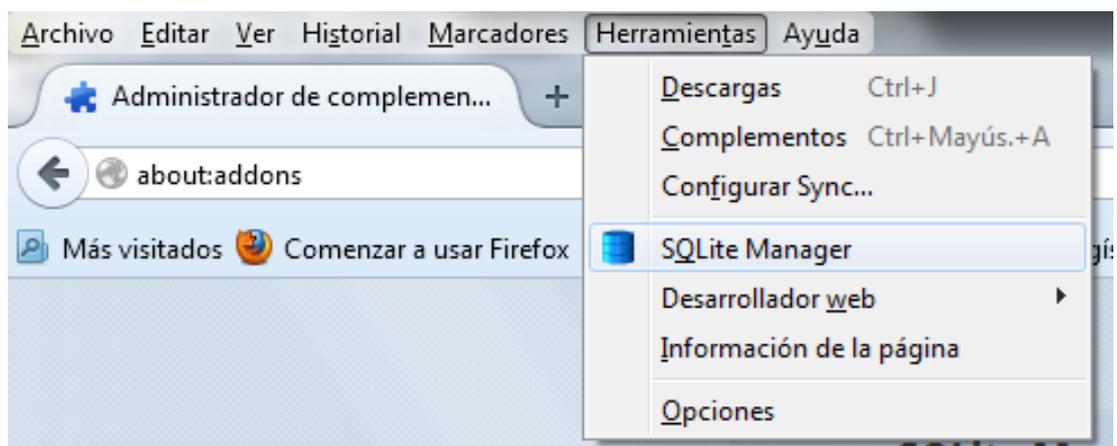


Figura 2.22 Abrir base de datos SQLite

- Se muestra el área de trabajo de SQLite.

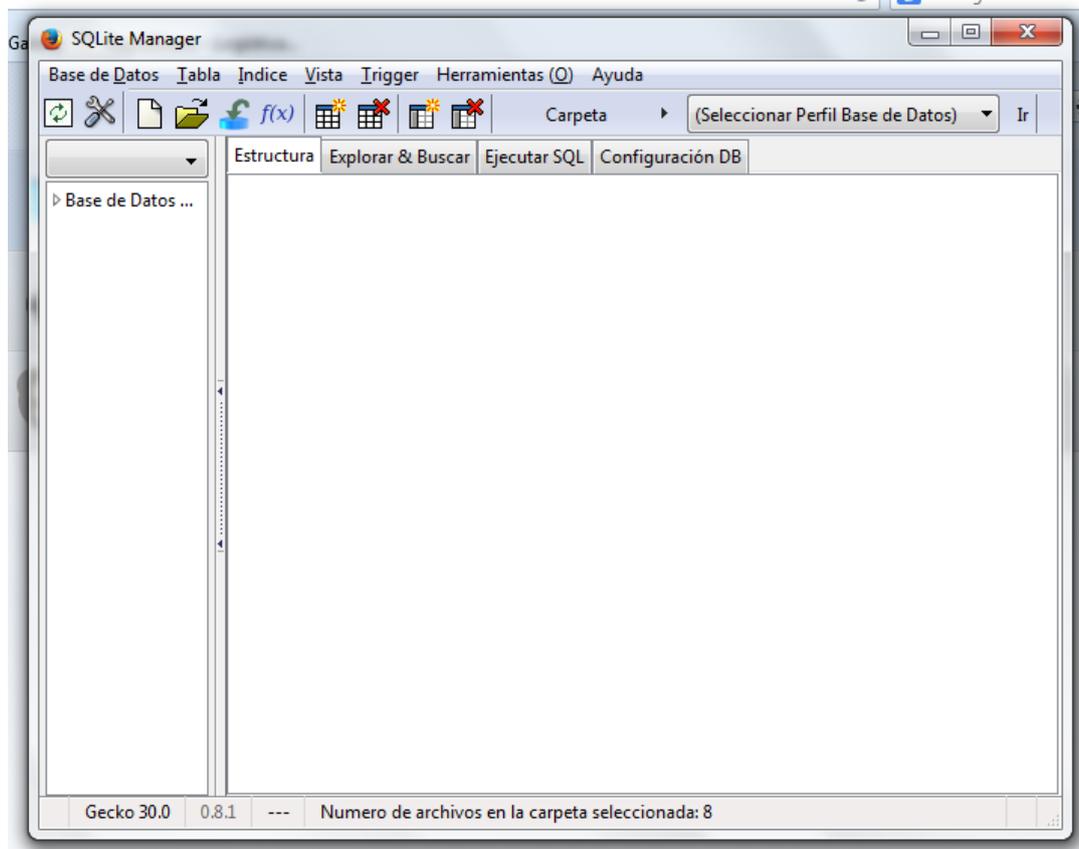


Figura 2.23 Área de trabajo de SQLite

2.9.5.4. Android Dataframework

“Mediante esta librería es posible gestionar una base de datos SQLite dentro de Android de una manera sencilla. Simplemente es necesario escribir un archivo XML y la librería se encargará de crear las tablas y facilitar una API mediante la cual se puede crear, modificar y eliminar registros de una manera sencilla y eficaz en la aplicación a desarrollarse.”

[16]

2.10. Sistemas de comunicaciones móviles

Las comunicaciones móviles, permiten el intercambio de datos o información entre terminales móviles y/o terminales fijos con una calidad determinada. Se caracterizan principalmente por su movilidad, es decir tanto el emisor como el receptor están, o pueden estar, en movimiento. Dicho de otra manera, la comunicación de los dos elementos que se encuentran en los extremos de la comunicación es factible sin la necesidad de utilizar un medio físico o cable que los conecten ya que se realiza a través de ondas de radio.

2.10.1. Funcionamiento de comunicaciones móviles

ARREGOCÉS Benyi: Cómo funcionan las redes móviles, menciona que: "Las comunicaciones móviles, funcionan a través de ondas de radio. Cada terminal actúa al mismo tiempo como emisor y receptor, que se comunica con la antena de telefonía más próxima, a la que se denomina estación base o BTS. Las antenas se sitúan en sitios altos para alcanzar mayor cobertura y conforman lo que se conoce como celdas, que se agrupan entre sí y dan cobertura a los dispositivos móviles. Cada una de las celdas se comunica en una frecuencia en concreto, para que no haya interferencias con las adyacentes, y enlaza con los terminales de los clientes.

De los BTS la señal se traslada a los BSC (siglas en inglés de "controlador de estaciones base"), que actúan como concentradores de varias de estas antenas. Por último, la señal llega a los MSC (siglas en inglés de "centro de conmutación móvil"), que se encargan de establecer la

comunicación cuando uno de los clientes marca un número, de enviar los mensajes cortos y de cobrar las llamadas por su duración. Todas estas conexiones se pueden realizar de diversas maneras, no sólo mediante ondas sino también a través de redes fijas y otros tipos de enlaces.” [17]

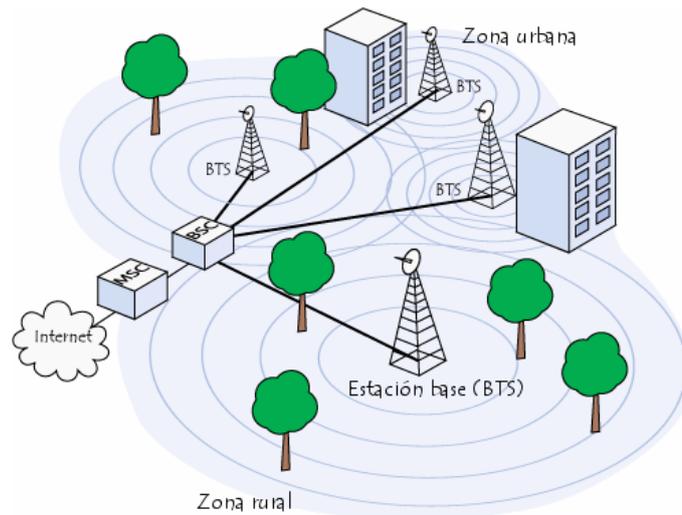


Figura 2.24 Arquitectura de una red GSM

FUENTE: <http://static.commentcamarche.net/es.kioskea.net/pictures/telephonie-mobile-images-reseau-cellulaire.png>

2.10.2. Comunicaciones inalámbricas WIFI

WIFI (Wireless Fidelity) llamada también WLAN (wireless lan, red inalámbrica) o estándar IEEE 802.11 es una de las tecnologías de comunicación inalámbrica mediante ondas más utilizada hoy en día. Gracias a su velocidad y alcance (100-150mts) es un medio ideal para acceder a internet sin cables.

Existen dos tipos de comunicación WIFI:

- 802.11b, que emite a 11 Mb/seg, y
- 802.11g, más rápida, a 54 MB/seg.

El funcionamiento de la red es sencillo, sólo es necesario conectar los dispositivos e instalar el software. Los enrutadores o routers WIFI incorporan herramientas de configuración para el control de acceso a la información que se transmite por el aire. Puesto que la información a transmitirse de esta manera se vuelve vulnerable, se recomienda la encriptación de la transmisión para emitir en un entorno seguro. WPA permite transmitir de forma segura, mucho más que su predecesor WEP y con nuevas características de seguridad, como la generación dinámica de la clave de acceso.

Para usuarios más avanzados existe la posibilidad de configurar el punto de acceso para que emita sólo a ciertos dispositivos. Usando la dirección MAC, un identificador único de los dispositivos, asignado durante su construcción, y permitiendo el acceso solamente a los dispositivos instalados. [18]

2.10.3. Aplicaciones para dispositivos móviles

Las aplicaciones móviles son aquellas que están contigo en todo momento, siempre y cuando la persona cuente con un dispositivo móvil, así por ejemplo: páginas web, aplicaciones para teléfonos Inteligentes, aplicaciones para tabletas, GPS, entre otras.

Las aplicaciones móviles fueron lanzadas al mercado informático como un complemento de los dispositivos móviles, pero con el pasar del tiempo, han ido evolucionando y se han convertido en una parte esencial de la experiencia de usuario.

Los ingresos económicos provenientes del mercado de las aplicaciones móviles han crecido exponencialmente con el pasar del tiempo y se estima que su porcentaje de crecimiento sea mayor al 50% cada año.

2.10.3.1. Características de una aplicación para dispositivos móviles

“Las aplicaciones móviles deben cumplir ciertas características para ser útiles en dispositivos móviles, así: [19]

- Portables
- Personales
- De uso diario ó regular
- Gratuitas ó Pagas
- Web ó Nativas
- Marketing, Ventas ó Corporativas

Todas estas características están consideradas según el mercado que rige las aplicaciones móviles, así como:

- Cerca del 60% de usuarios es del sexo masculino
- Un 42% del mercado de usuarios está entre los 25 a 34 años
- El 45% de usuarios usan aplicaciones principalmente de música y descargas.

Las aplicaciones que tienen más acogida y muestran un gran nivel de acogida a futuro son:

- Compartir información y Redes sociales (Twitter - Facebook)

- Publicidad en móviles
- Gestión Corporativa
- Tabletas, Teléfonos y Nube interconectados

2.10.4. Selección de la solución de la comunicación inalámbrica

Entre los tipos de comunicación con mejores características son bluetooth y 802.11 o WIFI, no se puede asegurar que una tecnología sea mejor que otra de forma absoluta. La elección de una tecnología determinada vendrá dada por las especificaciones de diseño de la aplicación. Lo que sí es posible afirmar es que bluetooth y 802.11 están más orientadas a dispositivos de consumo ya que son tecnologías complementarias aunque no sustitutorias. Por lo tanto después de un exhaustivo análisis, se ha considerado usar comunicación inalámbrica WIFI en el presente proyecto debido a las siguientes razones:

- Soporta grandes transferencias de información
- Conecta dispositivos con distancias relativamente largas a una velocidad elevada y puede llegar hasta los 11 Mbps.
- Útil cuando el empleo de cables es inconveniente o caro.
- Sustituye a interfaces de red como Ethernet.
- Soporta conexiones de usuarios simultáneamente.
- En caso de falta de cobertura.
- No genera costo de ingreso frecuente al servidor
- Permite respaldos en caso de pérdida y se lo realizará vía http.
- Reportes nocturnos localmente en el servidor de la empresa

Tabla 2.4**Comparativa de los distintos estándares de 802.11**

ESTÁNDAR	RANGO	FRECUENCIA	VELOCIDAD
802.11 a	10 m	5 GHz	5.4 Mb/s
802.11 b	100 m	2.4 GHz	11Mb/s
802.11 g	100 m	3.4 GHz	54 Mb/s
802.11 n	10 – 100 m	2.4 / 5 GHz	540 Mb/s

FUENTE: <http://www.tecnicaindustrial.es/tifrontal/a-4273-Bluetooth-criterios-seleccion-comparativa-tecnologias-inalambricas.aspx>

CAPÍTULO III

DESARROLLO E IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL

3.1. Análisis

3.1.1. Requerimientos del usuario

Un requerimiento es una declaración en lenguaje natural y en diagramas de cualquier característica, acción o política de negocio que sea relevante para el desarrollo de un sistema, es decir, los servicios que debe proveer y las restricciones que debe cumplir.

- **Funcionales.** Son aquellos que representan la interacción directa entre el usuario y el software, en otras palabras, describen toda la funcionalidad del sistema para con el usuario.
- **No funcionales.** Son los requerimientos que determinan el desempeño del sistema, pero no tienen una relación directa con el usuario, por ejemplo: tiempo de respuesta, seguridad, precisión, etc.

Una vez presentada la definición, se procede a determinar los requerimientos de usuario específicamente para la aplicación móvil del Sistema de Gestión de Pedidos de la Distribuidora Comercial “El Edén”.

Detalle de las actividades

1. Gestionar inicio de la aplicación
2. Gestionar datos de Clientes
3. Gestionar datos de Productos
4. Gestionar datos de Enlaces de proveedores
5. Gestionar Pedidos
6. Visualizar Listado de Pedidos

TÍTULO	GESTIONAR INICIO DE LA APLICACIÓN
DESCRIPCIÓN	Al iniciar la aplicación se debe visualizar información acerca de la aplicación y de la empresa en cuestión, así como una breve información de contacto de la empresa y una opción SIGUIENTE para continuar con la aplicación.
TAREAS	<ol style="list-style-type: none"> 1. Visualizar la opción siguiente para continuar con el menú de opciones (procesos) de la aplicación móvil. 2. Para terminar con la aplicación sin ingresar al menú de opciones, se habilitará la opción BACK.
RESPONSABLE	Tarea 1; Desarrollador – Diana Velasco Tarea 2; Desarrollador – Diana Velasco
PRODUCTOS	ENTRADA <ol style="list-style-type: none"> 1. Pantalla inicial de la aplicación SALIDA <ol style="list-style-type: none"> 1. Menú de opciones

TÍTULO	GESTIONAR DATOS DE CLIENTES
DESCRIPCIÓN	Al ingresar al menú, presionando el botón SIGUIENTE en la pantalla de inicio, se presentarán varias opciones, entre ellas la opción CLIENTES, dentro de ésta, será posible VISUALIZAR un listado de clientes (en caso de existir), CREAR un nuevo cliente, MODIFICAR la información de uno existente y ELIMINAR un cliente del listado.
TAREAS	<ol style="list-style-type: none"> 1. Visualizar el menú de opciones. 2. Mostrar las alternativas dentro de la opción CLIENTES (Listado, Modificar-Eliminar, Nuevo) y presentar el listado (en caso de existir). 3. Al seleccionar LISTADO, se desplegarán la lista de clientes ingresados (en caso de existir) y junto al nombre del cliente se mostrará un ícono de TELÉFONO que permitirá llamar al cliente. 4. Al elegir NUEVO (a través del ícono de GUARDAR), se presenta un formulario para ingresar la información necesaria, relacionada con el cliente a crearse. 5. Al dar clic en GUARDAR se almacenará la información y se presentará el LISTADO actualizado. 6. Al optar por la opción MODIFICAR-ELIMINAR (ícono hoja y lápiz), se mostrará el formulario en el que se visualizará la información del cliente requerido. 7. Para buscar un cliente se ingresará la cédula y clic en BUSCAR?. 8. Si el cliente existe, su información se presentará en el formulario que tiene al final dos botones, ACTUALIZAR y ELIMINAR, para escoger lo que se desea realizar. 9. Si desea modificar, es posible cambiar la información en los campos (a excepción de la cédula) y se presionará el botón ACTUALIZAR. 10. Si presiona la opción ELIMINAR, el usuario se eliminará junto con su información, del listado de clientes de la aplicación. 11. Para terminar con la aplicación sin ingresar al menú de opciones, se habilitará la opción BACK.
RESPONSABLE	Tarea 1 - 12; Desarrollador – Diana Velasco
PRODUCTOS	<p>ENTRADA</p> <ol style="list-style-type: none"> 1. Menú de opciones 2. Opción – cliente 3. Alternativa a. Listado 4. Alternativa b. Modificar-Eliminar 5. Alternativa c. Nuevo <p>SALIDA</p> <ol style="list-style-type: none"> 1. Opciones de la aplicación Sistema de Gestión de Pedidos 2. Pantalla de alternativas de Cliente 3. Alternativa a. Visualizar el listado de clientes (si acaso existe) 4. Alternativa b. Presentar formulario datos de clientes 5. Alternativa c. Presentar formulario datos de clientes

TÍTULO	GESTIONAR DATOS DE PRODUCTOS
DESCRIPCIÓN	Al ingresar al menú, presionando el botón SIGUIENTE en la pantalla de inicio, se presentarán varias opciones, entre ellas la opción PRODUCTOS, dentro de ésta, será posible VISUALIZAR un listado de PRODUCTOS (en caso de existir) y CREAR un nuevo producto
TAREAS	<ol style="list-style-type: none"> 1. Visualizar el menú de opciones. 2. Mostrar las alternativas dentro de la opción PRODUCTOS (Listado, Nuevo). 3. Al seleccionar LISTADO, se desplegarán la lista de productos ingresados, en caso de existir. 4. Al elegir NUEVO (a través del ícono de GUARDAR), se presenta un formulario para ingresar la información necesaria, relacionada con el producto a crearse y dos botones GRABAR y OBTENER IMAGEN. 5. Al dar clic en OBTENER IMAGEN, será posible seleccionar un gráfico o imagen localizada en el dispositivo que muestre el producto deseado completando la información a guardarse. 6. Al presionar GRABAR se almacenará la información y se presentará el LISTADO actualizado 7. Al mantener presionado un enlace se desplegará dos opciones: ELIMINAR ENLACE que permitirá borrar el producto y VISUALIZAR que presentará la información completa del producto almacenado. 8. Para terminar se habilitará la opción back.
RESPONSABLE	Tarea 1 - 12; Desarrollador – Diana Velasco
PRODUCTOS	<p>ENTRADA</p> <ol style="list-style-type: none"> 1. Menú de opciones 2. Opción – producto 3. Alternativa a. Listado 4. Alternativa b. Nuevo 5. Selección VISUALIZAR 6. Selección ELIMINAR <p>SALIDA</p> <ol style="list-style-type: none"> 1. Opciones de la aplicación Sistema de Gestión de Pedidos 2. Pantalla de alternativas de Producto 3. Alternativa a. Visualizar el listado de Productos (si existe) 4. Alternativa b. Presentar formulario datos de productos 5. Información completa del producto 6. Listado de productos actualizado

TÍTULO	GESTIONAR DATOS DE ENLACE DE PROVEEDORES
DESCRIPCIÓN	Al ingresar al menú, presionando el botón SIGUIENTE en la pantalla de inicio, se presentarán varias opciones, entre ellas la opción ENLACES, dentro de ésta se mostrará un formulario para ingresar información del enlace y un botón + (MÁS) para guardar la información. Adicional se VISUALIZARÁ el listado de enlaces (en caso de existir).
TAREAS	<ol style="list-style-type: none"> 1. Visualizar el menú de opciones. 2. Visualizar un listado de enlaces a sitios web de proveedores (en caso de existir), debajo del formulario de la opción ENLACES. 3. Mostrar el formulario de la opción ENLACES, en el que será posible ingresar el NOMBRE DEL ENLACE y la DIRECCIÓN DEL ENLACE. 4. Presentar el botón + (MÁS) que permitirá guardar la información 5. Luego de guardar la información, se presentará el listado actualizado de enlaces ingresados. 6. Al mantener presionado un enlace se desplegará dos opciones, ELIMINAR ENLACE que permitirá borrar el enlace y VER ENLACE que direccionará al sitio web almacenado. 7. Para terminar se habilitará la opción BACK.
RESPONSABLE	Tarea 1 - 7; Desarrollador – Diana Velasco
PRODUCTOS	<p>ENTRADA</p> <ol style="list-style-type: none"> 1. Menú de opciones 2. Opción – enlaces 3. Clic botón + <p>SALIDA</p> <ol style="list-style-type: none"> 1. Opciones de la aplicación Sistema de Gestión de Pedidos 2. Formulario ingreso de enlaces 3. Formulario vacío y listado actualizado

TÍTULO	GESTIONAR PEDIDOS
DESCRIPCIÓN	Al ingresar al menú, presionando el botón SIGUIENTE en la pantalla de inicio, se presentarán varias opciones, entre ellas la opción PEDIDO, dentro de ésta se mostrará un formulario en el que será posible elegir al cliente y la ciudad a la que pertenecen, además seleccionar el transporte en el que se enviará la encomienda para conocer las referencias para una posterior información, la forma de pago del pedido para un control contable, y lo más importante de este formulario, los pormenores del pedido como cantidad, precio, detalle y el valor total del pedido. Adicional existe un botón GUARDAR para archivar el pedido y así posteriormente poder imprimirlo para su despacho.
TAREAS	<ol style="list-style-type: none"> 1. Visualizar el menú de opciones. 2. Visualizar formulario completo para ingresar datos del pedido 3. Desplegar los clientes existentes 4. Mostrar la ciudad de destino 5. Seleccionar la empresa de transporte a utilizar para enviar la encomienda, especificando: descripción, valor, fecha y hora. 6. Registrar la forma de pago, crédito o depósito especificando: número de documento y valor 7. Seleccionar el estado del pedido, considerando: enviado, no enviado, pendiente de pago 8. Permitir ingresar los detalles del producto en el pedido, considerando: observación, cantidad precio referencial, precio a facturar y total 9. Visualizar un total general, para poder almacenar el pedido. 10. Para terminar con la aplicación sin ingresar al menú de opciones, se habilitará la opción BACK.
RESPONSABLE	Tarea 1 - 10; Desarrollador – Diana Velasco
PRODUCTOS	<p>ENTRADA</p> <ol style="list-style-type: none"> 1. Menú de opciones 2. Opción – pedido 3. Desplegar clientes 4. Mostrar ciudad 5. Seleccionar transporte 6. Registrar Pago 7. Seleccionar Estado de pedido 8. Ingresar Pedido 9. Calcular total factura 10. Almacenar pedido <p>SALIDA</p> <ol style="list-style-type: none"> 1. Opciones de la aplicación Sistema de Gestión de Pedidos 2. Formulario pedido 3. Selección de cliente o ingresar nuevo cliente 4. Selección ciudad 5. Selección transporte 6. Selección forma de pago 7. Selección estado de pedido 8. Ingresar datos de pedido 9. Visualizar total factura 10. Visualizar pedido

TÍTULO	VISUALIZAR PEDIDOS
DESCRIPCIÓN	Al ingresar al menú, presionando el botón SIGUIENTE en la pantalla de inicio, se presentarán varias opciones, entre ellas REPORTE, dentro de ésta se muestra una opción para seleccionar el cliente, mostrar sus pedidos y el detalle de cada una al elegirlo (en caso de existir).
TAREAS	<ol style="list-style-type: none"> 1. Visualizar el menú de opciones. 2. Visualizar un listado de clientes junto con su número de cédula y su ciudad. 3. Al elegir un cliente se desplegará los pedidos realizados por éste 4. Al escoger un pedido se mostrará en la parte inferior los detalles de este pedido. 5. Para terminar con la aplicación sin ingresar al menú de opciones, se habilitará la opción BACK.
RESPONSABLE	Tarea 1 - 4; Desarrollador – Diana Velasco
PRODUCTOS	<p>ENTRADA</p> <ol style="list-style-type: none"> 1. Menú de opciones 2. Opción – Reportes <p>SALIDA</p> <ol style="list-style-type: none"> 1. Opciones de la aplicación Sistema de Gestión de Pedidos 2. Visualización de reportes y detalles.

3.2. Diseño

3.2.1. Casos de uso

Un caso de uso es una representación de las secuencias de interacciones entre un sistema y sus actores y/u otro sistema, que se realizarán para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores.

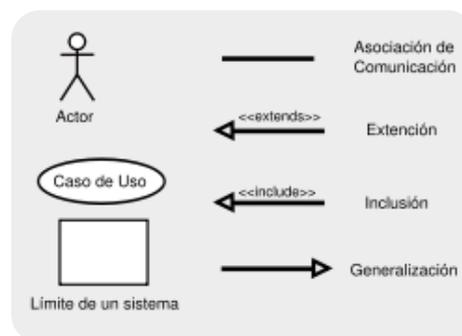


Figura 3.1 Los elementos que intervienen en este tipo de diagrama

FUENTE: http://es.wikipedia.org/wiki/Actor_%28UML%29#/media/File:Notacion_Caso_de_Uso.svg

Actor: entidad externa al sistema a desarrollarse y que puede interactuar con él. Ejemplo Usuario, Administrador, Sistema Externo.

Relaciones: Existen entre casos de uso y actores, así por ejemplo:

- Actor se comunica con un caso de uso.
- Caso de uso extiende otro caso de uso.
- Caso de uso usa otro caso de uso.

En base a lo previamente expuesto se puede proceder al desarrollo de los casos de uso del Sistema de Gestión de Pedidos de la Empresa “El Edén”.

Detalle de los casos de uso

1. SISTEMA DE GESTIÓN DE PEDIDOS DE LA DISTRIBUIDORA COMERCIAL “EL EDÉN”
2. Gestionar inicio de la aplicación
3. Gestionar datos de Clientes
4. Gestionar datos de Productos
5. Gestionar datos de Enlaces de proveedores
6. Gestionar Pedidos
7. Visualizar Listado de Pedidos

CASO DE USO - SISTEMA DE GESTIÓN DE PEDIDOS

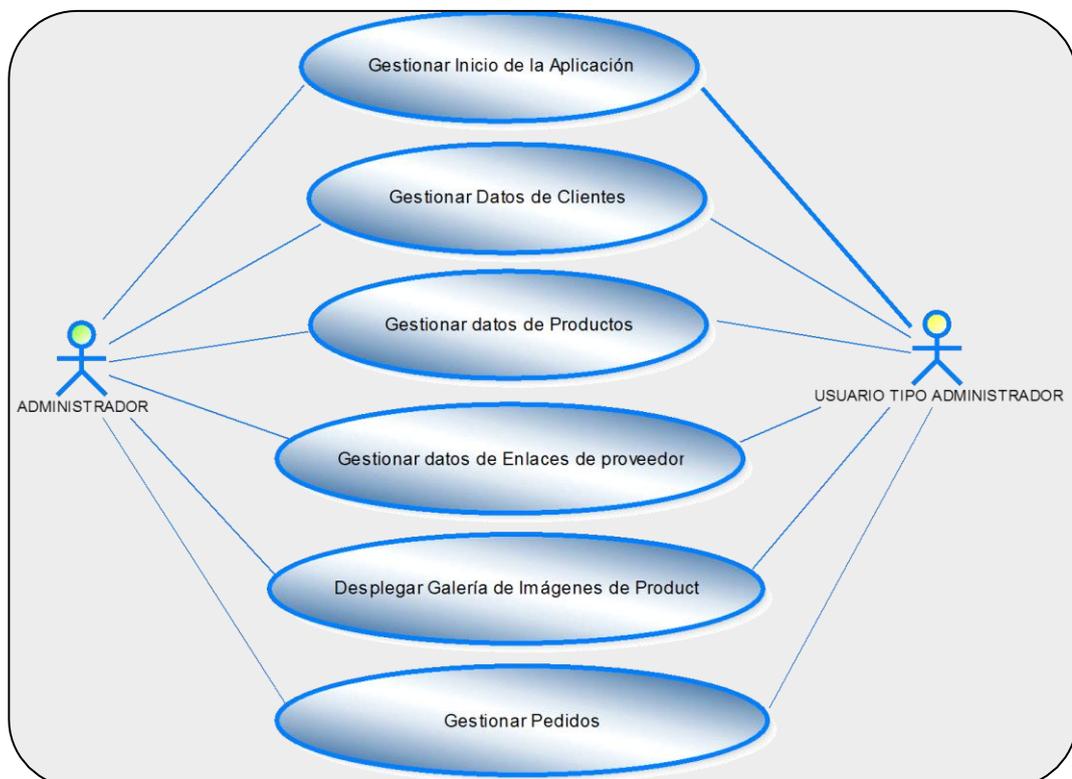


Figura 3.2 Sistema de Gestión de Pedidos

CASO DE USO – GESTIONAR INICIO DE LA APLICACIÓN

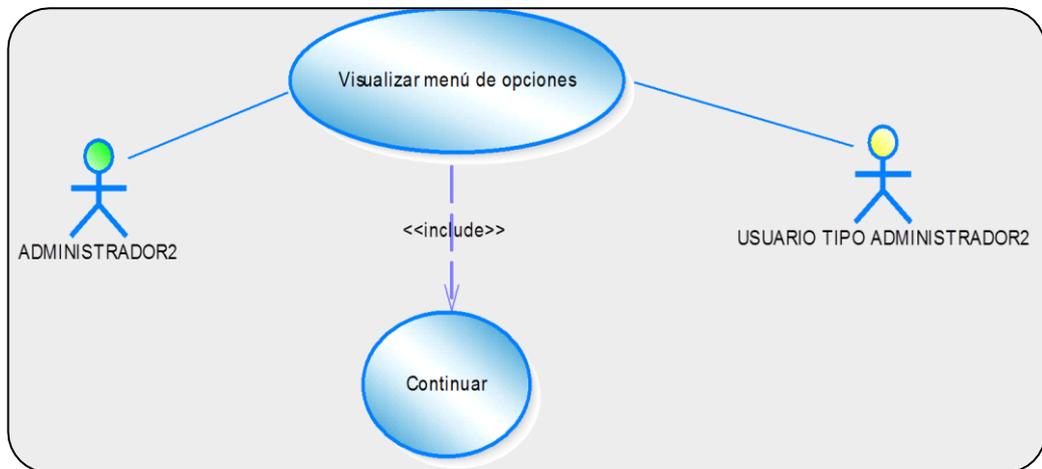


Figura 3.3 Gestionar Inicio de Aplicación

CASO DE USO - GESTIONAR DATOS DE CLIENTES

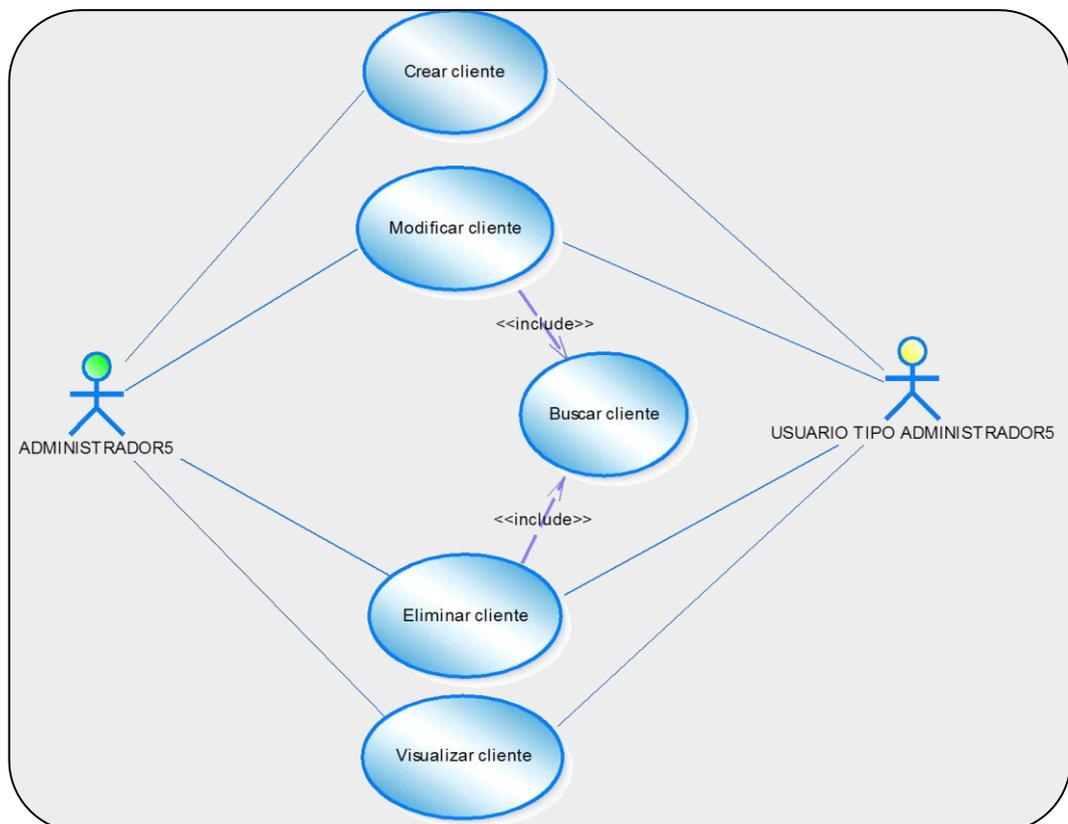


Figura 3.4 Gestionar Datos de Clientes

CASO DE USO - GESTIONAR DATOS DE PRODUCTOS

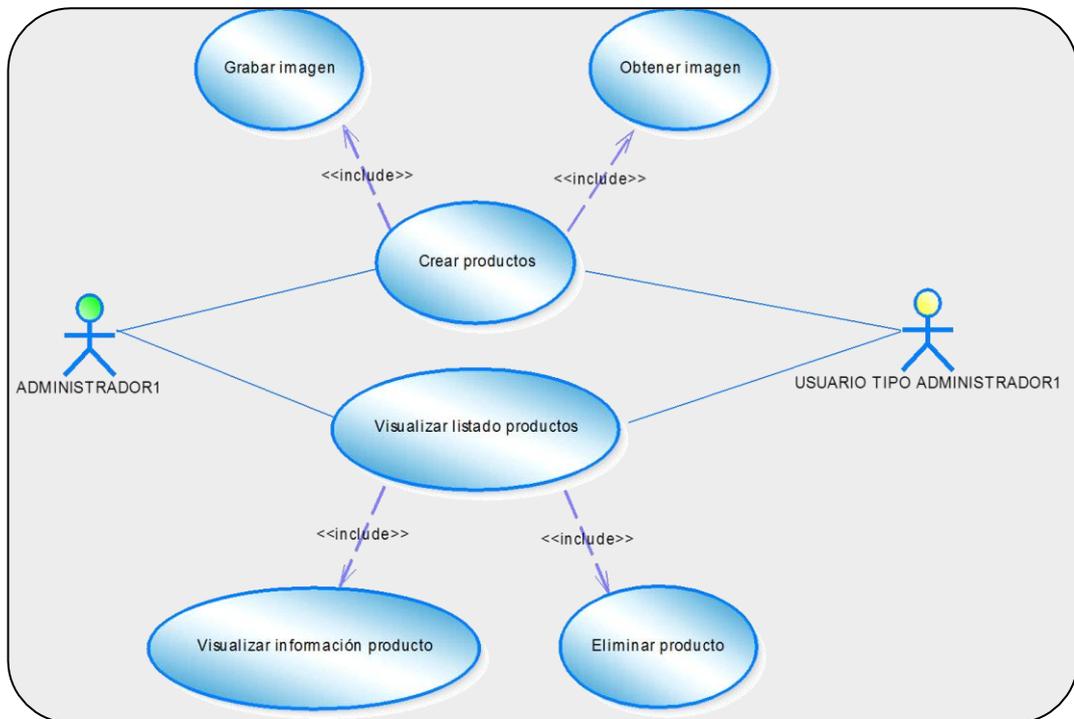


Figura 3.5 Gestionar Datos de Productos

CASO DE USO - GESTIONAR DATOS DE ENLACES DE PROVEEDORES

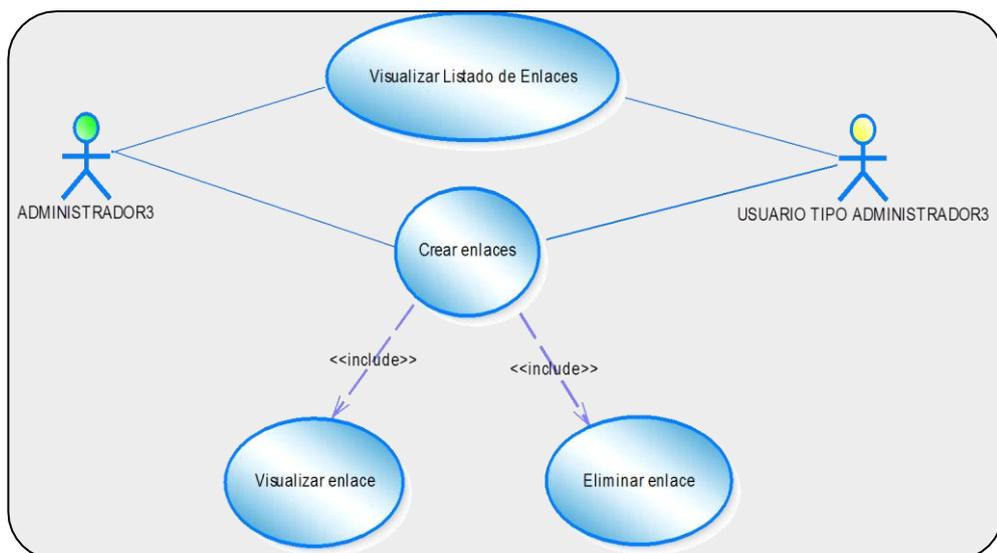
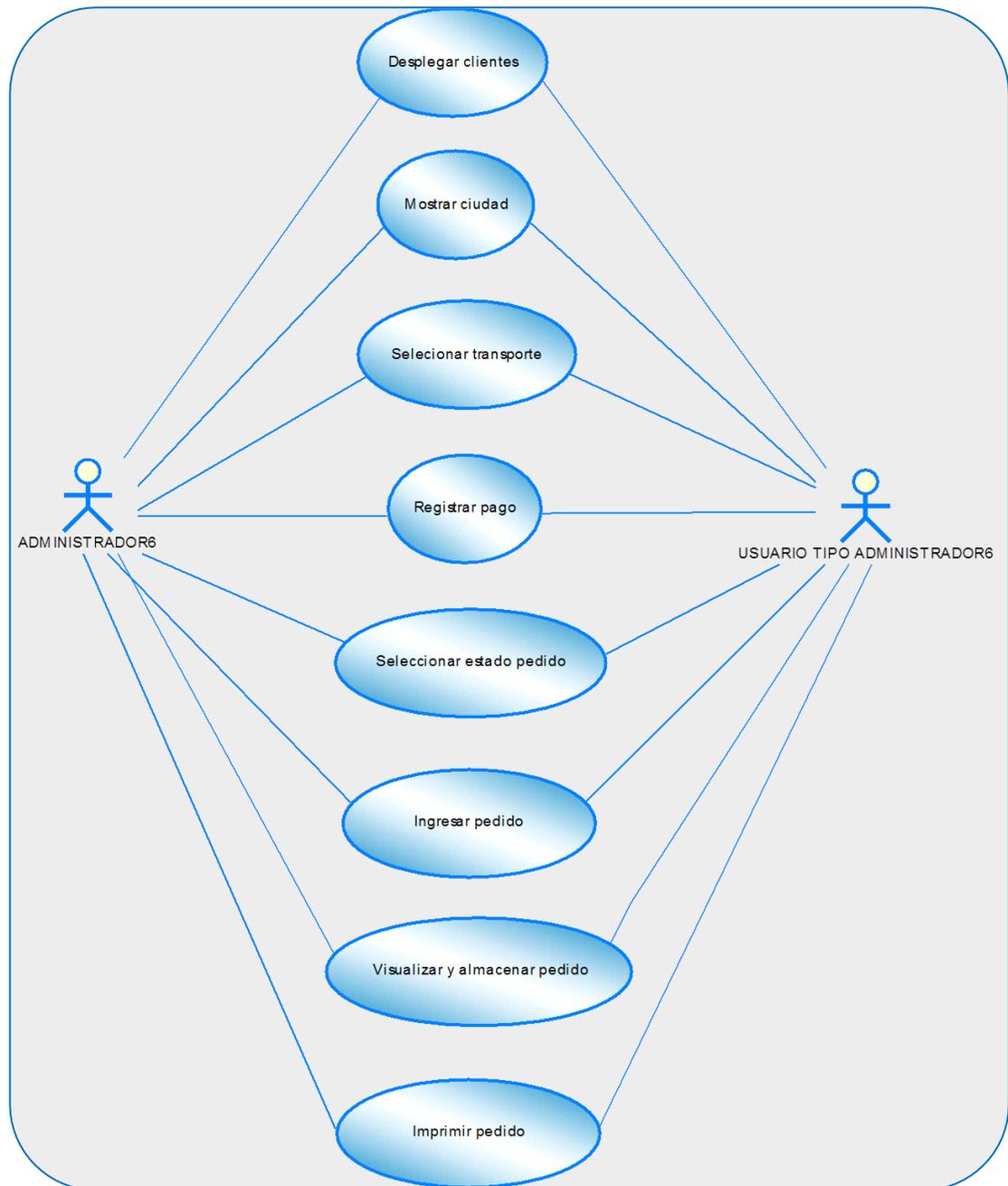


Figura 3.6 Gestionar enlaces de proveedores

CASO DE USO – GESTIONAR PEDIDOS**Figura 3.7 Gestionar Pedidos**

CASO DE USO – VISUALIZAR LISTADO DE PEDIDOS

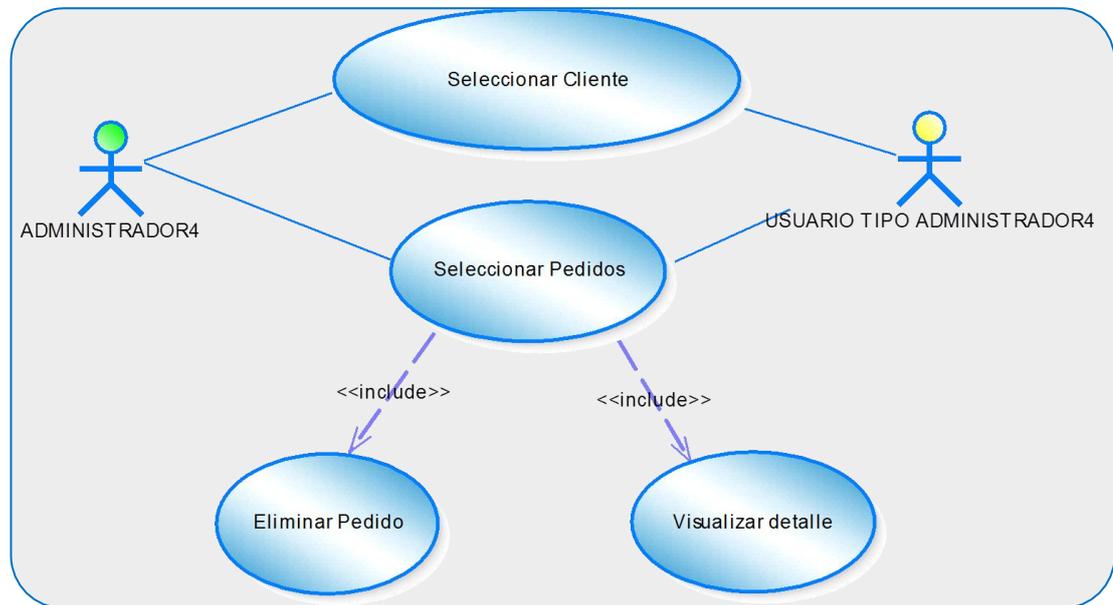


Figura 3.8 Visualizar Listado de Pedidos

3.2.2. Diagramas

Diagrama es una representación gráfica de los pasos a seguir en una secuencia de actividades, dentro de un proceso o un procedimiento, identificándolos mediante símbolos de acuerdo con su naturaleza; incluye, además, toda la información que se considera necesaria para el análisis.

Un diagrama usado en el desarrollo de software demuestra la estructura de la aplicación, es importante que muestre la jerarquización, las condiciones del usuario, las estructuras a utilizarse, todos los accesos a las bases de datos, las llamadas a subrutinas entre otras.

3.2.2.1. Colaborativo – secuencial

Un diagrama colaborativo-secuencial es usado para modelar o mostrar las interacciones que se podrán realizar en la aplicación a desarrollarse. Se mostrarán las clases o módulos que integran el programa y las llamadas que se realizan en cada uno de ellos para realizar una tarea determinada. Los elementos usados en este tipo de diagramas constan de objetos, una línea vertical que representa la línea de vida del mismo y los mensajes entre estos.

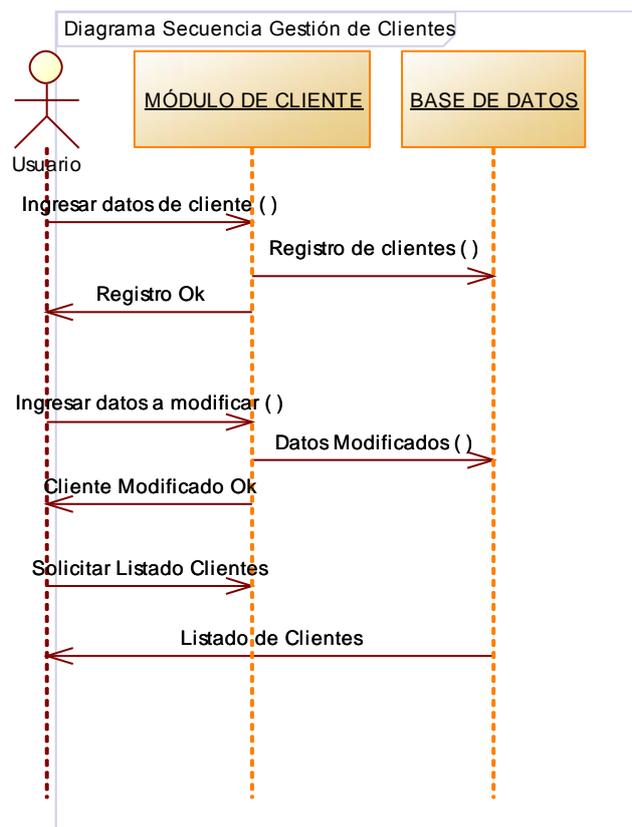


Figura 3.9 Diagrama de secuencia para la Gestión de Clientes

Escenario simple de Gestión de Clientes

- El usuario ingresa al módulo de clientes del sistema.
- El usuario ingresa los datos del cliente a través del módulo de cliente.
- El módulo de cliente realiza el registro del cliente en la base de datos.
- El módulo de cliente entrega una respuesta satisfactoria del registro del cliente al usuario.
- El usuario ingresa los datos a modificar del cliente a través del módulo de cliente.
- El módulo de cliente transmite los datos modificados del cliente a la base de datos.
- El módulo de cliente entrega una respuesta satisfactoria de los datos modificados del cliente al usuario.
- El usuario solicita un listado de clientes a través del módulo de cliente.
- La base de datos presenta el listado de clientes al usuario.

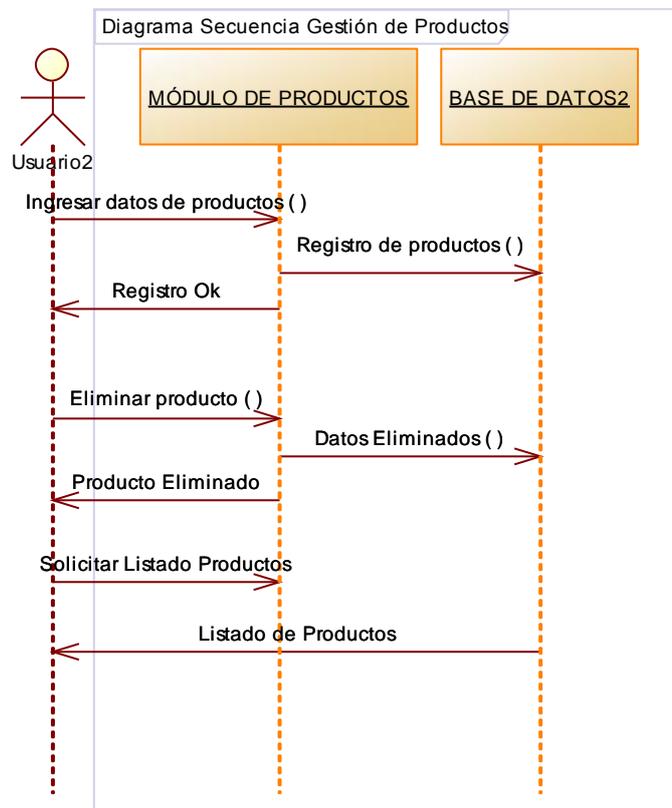


Figura 3.10 Diagrama de secuencia para la Gestión de Productos

Escenario simple de Gestión de Productos

- El usuario ingresa al módulo de productos del sistema.
- El usuario ingresa los datos del producto a través del módulo de productos.
- El módulo de productos realiza el registro del producto en la base de datos.
- El módulo de productos entrega una respuesta satisfactoria del registro del producto al usuario.
- El usuario selecciona el producto a eliminar a través del módulo de productos.

- El módulo de productos elimina los datos del producto de la base de datos.
- El módulo de productos entrega una respuesta satisfactoria del producto eliminado al usuario.
- El usuario solicita un listado de productos a través del módulo de productos.
- La base de datos presenta el listado de productos al usuario.

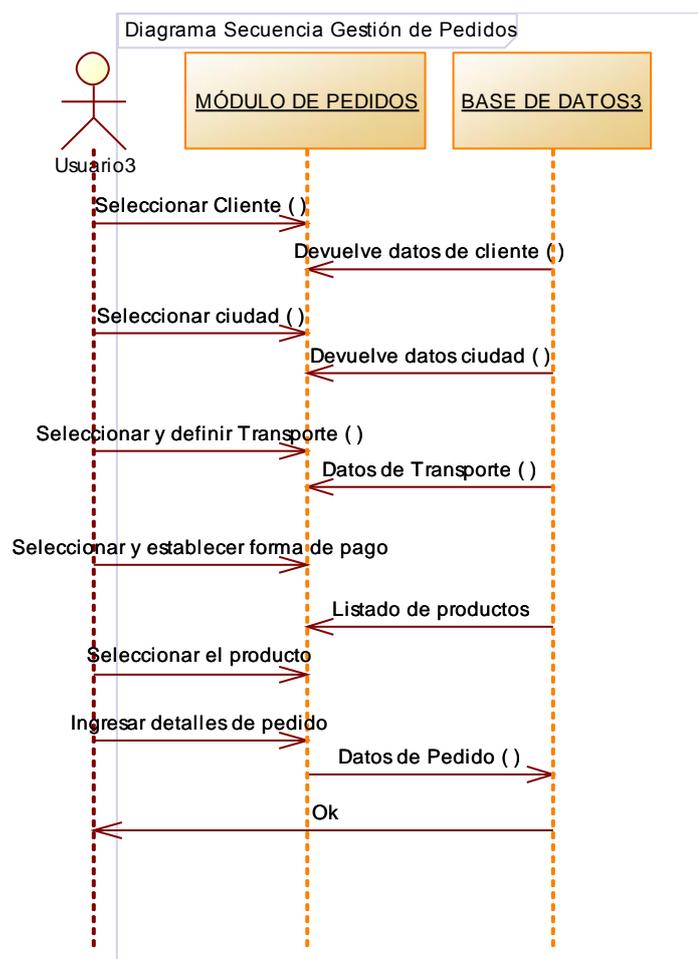


Figura 3.11 Diagrama de secuencia para la Gestión de Pedidos

Escenario simple de Gestión de Pedidos

- El usuario ingresa al módulo de pedidos del sistema.
- El usuario selecciona el cliente a través del módulo de pedidos.
- La base de datos devuelve al módulo de pedidos los datos del cliente seleccionado.
- El usuario selecciona la ciudad del cliente a través del módulo de pedidos.
- La base de datos devuelve al módulo de pedidos los datos de la ciudad seleccionada.
- El usuario selecciona y define el transporte a utilizarse a través del módulo de pedidos.
- La base de datos devuelve al módulo de pedidos los datos del transporte seleccionado.
- El usuario selecciona y establece la forma de pago a realizarse a través del módulo de pedidos.
- La base de datos devuelve al módulo de pedidos un listado de productos.
- El usuario selecciona el producto del listado presentado en el módulo de pedidos.
- El usuario ingresa los detalles del pedido al módulo de pedidos.
- El módulo de pedidos transmite los datos del pedido a la base de datos.
- La base de datos emite al usuario una respuesta positiva del pedido almacenado.

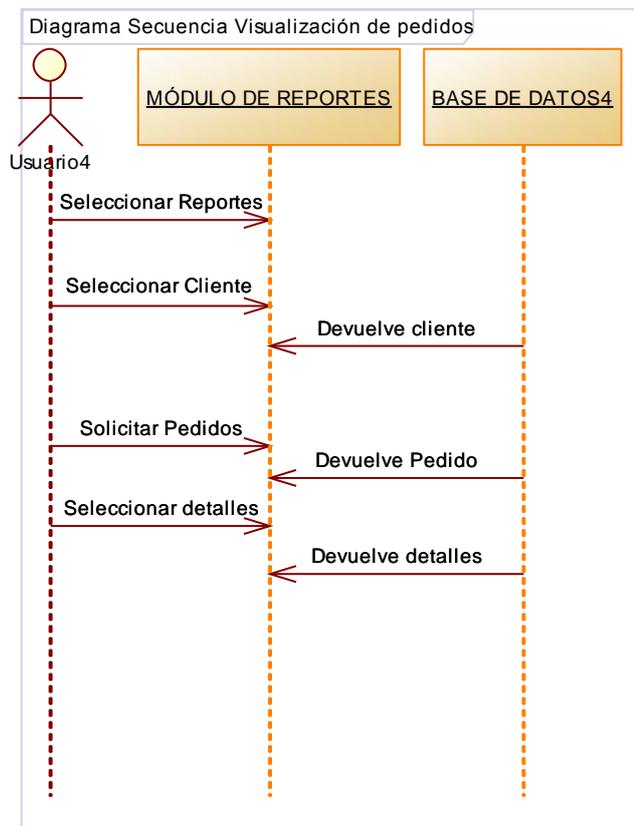


Figura 3.12 Diagrama Secuencia Visualización de Pedidos

Escenario simple de Visualización de Pedidos

- El usuario ingresa al módulo de Reportes del sistema.
- El usuario selecciona el cliente a través del módulo de Reportes.
- La base de datos devuelve al módulo de Reportes los pedidos realizados por el cliente seleccionado.
- El usuario selecciona el pedido del cliente del cliente a través del módulo de Reportes.
- La base de datos devuelve al módulo de Reportes los datos del pedido seleccionado.
- El usuario selecciona detalles del pedido a través del módulo de Reportes.

- La base de datos devuelve al módulo de Reportes los detalles del pedido seleccionado.

3.2.2.2. Estado

MAREÑO, Iván: Diagramas de Estado, menciona que “Los diagramas de estado son una técnica conocida para describir el comportamiento de un sistema. Describen todos los estados posibles en los que puede entrar un objeto particular y la manera en que cambia el estado del objeto, como resultado de los eventos que llegan a él. En la mayor parte de las técnicas Orientadas a Objetos, los diagramas de estado se dibujan para una sola clase, mostrando el comportamiento de un solo objeto durante todo su ciclo de vida.

Existen muchas formas de diagramas de estado, cada una con semántica ligeramente diferente. La más popular que se emplea en las técnicas de OO se basa en la tabla de estados de David Harel (Vol. 8). OMT fue quien la uso por primera vez para los métodos de OO y fue adoptada por Grady Booch en su segunda edición 1994.

El estado en el que se encuentra un objeto determina su comportamiento. Cada objeto sigue el comportamiento descrito en el Diagrama de Estados asociado a su clase. Los Diagramas de Estados y escenarios son complementarios, los Diagramas de Estados son autómatas jerárquicos que permiten expresar concurrencia, sincronización y jerarquías de objetos, son grafos dirigidos y deterministas. La transición entre estados es instantánea y se debe a la ocurrencia de un evento.” (2007, pág. 2.)

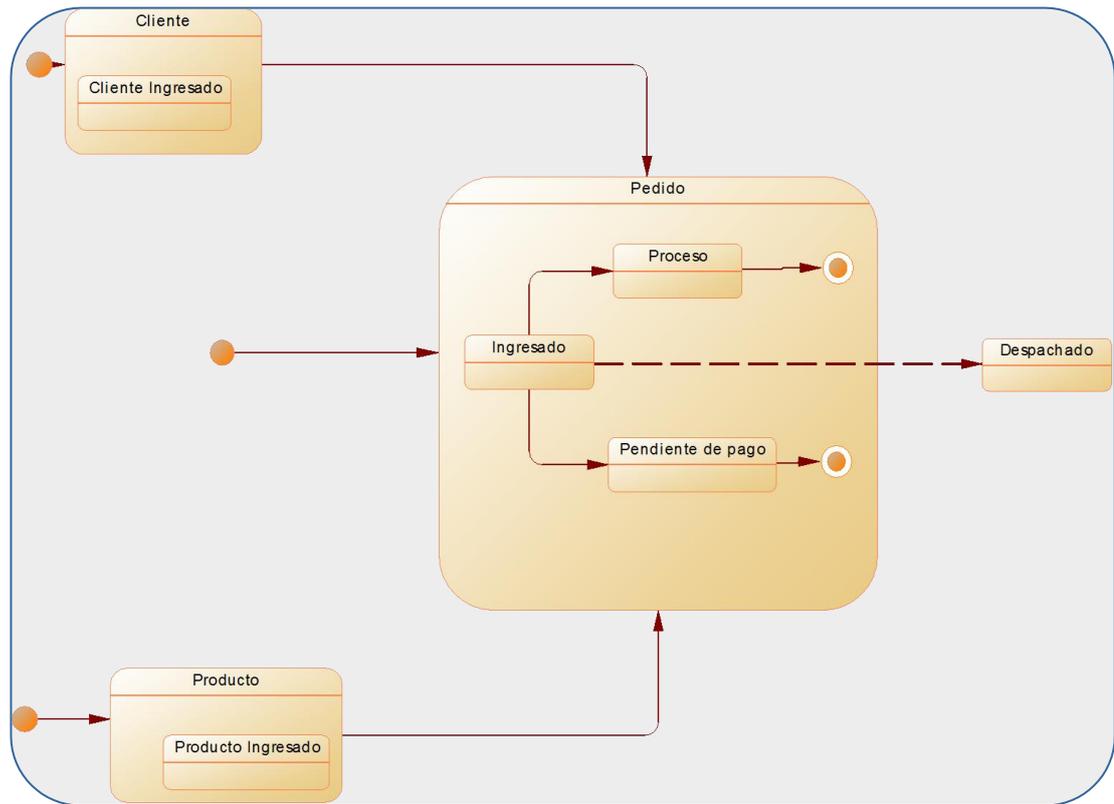


Figura 3.13 Diagrama de Estado.

Clases

Un diagrama de clases es un diagrama de tipo estático, sirve para visualizar o representar las relaciones de los objetos fundamentales del sistema o clases, es decir los que percibe el usuario y con los que espera tratar para completar su tarea en vez de objetos del sistema o de un modelo de programación.

Un diagrama de clases está compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
 - La clase define el ámbito de definición de un conjunto de objetos.
 - Cada objeto pertenece a una clase.

- Los objetos se crean por instanciación de las clases.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

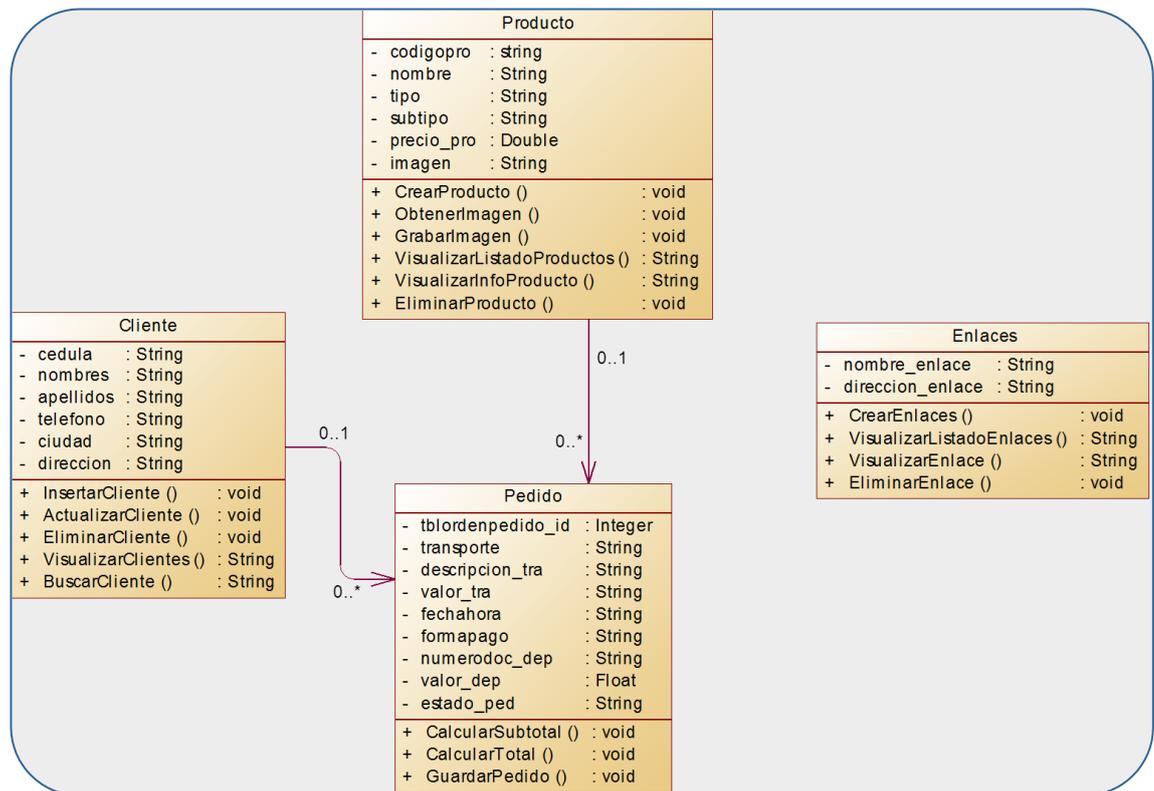


Figura 3.14 Diagrama de Clases

3.2.2.3. Entidad – relación

Un diagrama entidad-relación, también conocido como DER, es una herramienta de modelado de bases de datos pertenecientes a un sistema, describe la distribución, el almacenamiento que se realiza de forma abstracta de los datos y las relaciones entre éstos.

Existen definiciones que plantean diferencias entre el modelo entidad-relación y el diagrama entidad-relación, donde el modelo entidad-relación sería el "lenguaje" utilizado para crear los diagramas de entidad-relación.

Componentes de un DER

- Tipos de objetos o entidades.
- Relaciones: conectan los objetos o entidades.

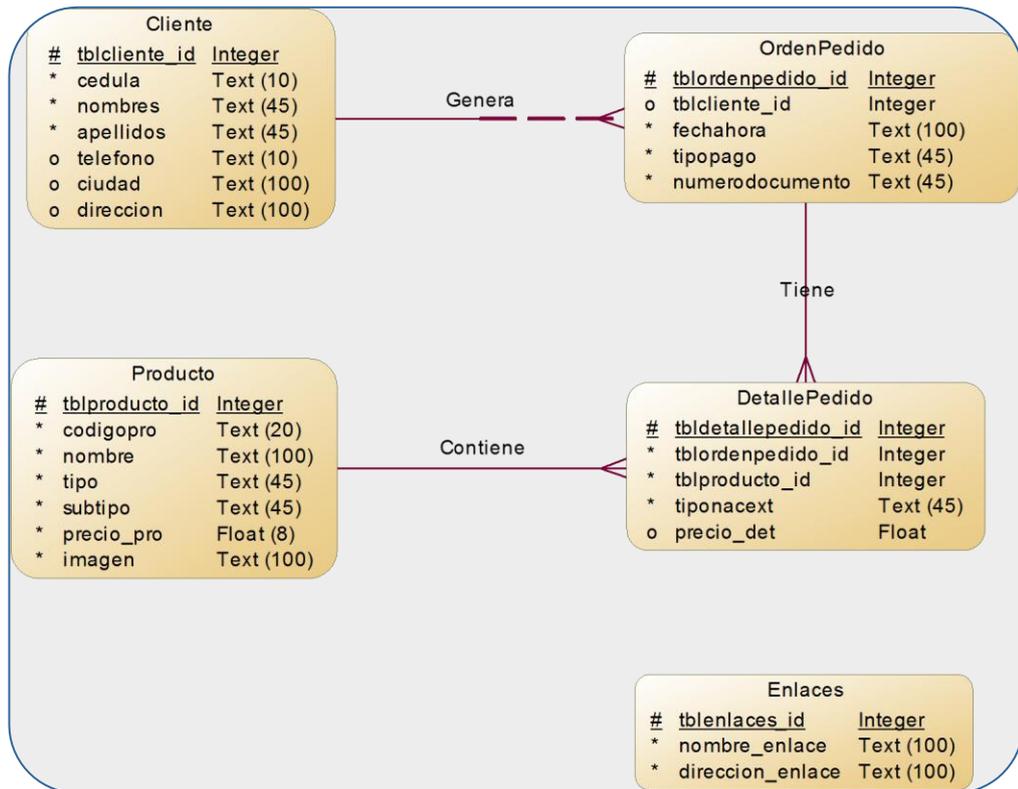


Figura 3.15 Diagrama Lógico

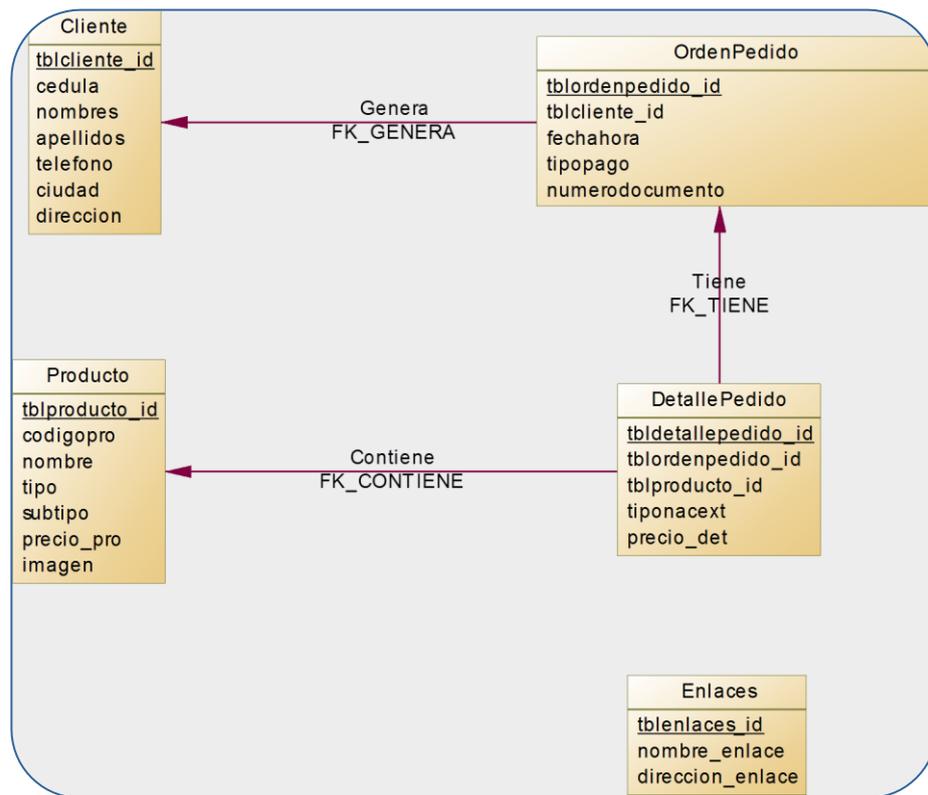


Figura 3.16 Diagrama Físico

Estructura de la BD SQLite para la aplicación Gestión de Pedidos

```

/*=====*/
/* DBMS name:   SQLLITE                               */
/* Created on:  23/09/2014 12:40:14                    */
/*=====*/
drop table if exists CLIENTE;

drop table if exists DETALLEPEDIDO;

drop table if exists ENLACES;

drop table if exists ORDENPEDIDO;

drop table if exists PRODUCTO;

```

```

/*=====*/
/* Table: CLIENTE */
/*=====*/
create table CLIENTE
(
  TBLCLIENTE_ID int not null,
  CEDULA text not null,
  NOMBRES text not null,
  APELLIDOS textnotnull,
  TELEFONO text,
  CIUDAD text,
  DIRECCION text,
  primarykey (TBLCLIENTE_ID)
);

```

```

/*=====*/
/* Table: DETALLEPEDIDO */
/*=====*/
createtable DETALLEPEDIDO
(
  TBLDETALLEPEDIDO_ID intnotnull,
  TBLORDENPEDIDO_ID intnotnull,
  TBLPRODUCTO_ID int not null,
  TIPONACEXT text not null,
  PRECIO_DET double,
  primary key (TBLDETALLEPEDIDO_ID)
);

```

```

/*=====*/
/* Table: ENLACES */
/*=====*/
create table ENLACES
(
  TBLENLACES_ID int not null,
  NOMBRE_ENLACE text not null,
  DIRECCION_ENLACE text not null,
  primary key (TBLENLACES_ID)
);

```

```

/*=====*/
/* Table: ORDENPEDIDO */
/*=====*/
create table ORDENPEDIDO
(
  TBLORDENPEDIDO_ID int not null,
  TBLCLIENTE_ID int,
  FECHAHORA text not null,
  TIPOPAGO text not null,
  NUMERODOCUMENTO text not null,
  primary key (TBLORDENPEDIDO_ID)
);

```

```

/*=====*/
/* Table: PRODUCTO */
/*=====*/
create table PRODUCTO
(
  TBLPRODUCTO_ID int not null,
  CODIGOPRO text not null,
  NOMBRE text not null,
  TIPO text not null,
  SUBTIPO text not null,
  PRECIO_PRO float(8) not null,
  IMAGEN text not null,
  primary key (TBLPRODUCTO_ID)
);

```

```

alter table DETALLEPEDIDO add constraint FK_CONTIENE foreign key
(TBLPRODUCTO_ID)
references PRODUCTO (TBLPRODUCTO_ID) on delete restrict on update restrict;

```

```

alter table DETALLEPEDIDO add constraint FK_TIENE foreign key
(TBLORDENPEDIDO_ID)
references ORDENPEDIDO (TBLORDENPEDIDO_ID) on delete restrict on update restrict;

```

```

alter table ORDENPEDIDO add constraint FK_GENERA foreign key (TBLCLIENTE_ID)
references CLIENTE (TBLCLIENTE_ID) on delete restrict on update restrict;

```

3.3. Codificación e implementación

Implementación es la etapa donde efectivamente se desarrolla el sistema, en otras palabras es la programación de un determinado algoritmo en un lenguaje específico.

3.3.1.1. Interfaces de usuario

La Interfaz de Usuario también conocida como IU, es un conjunto de elementos de una aplicación que permite que el usuario interactúe con la información a través de un dispositivo. La interfaz debe estar bien diseñada para que el usuario encuentre fácilmente la respuesta requerida de la aplicación desarrollada.

```

package com.eleden.pryeleden;

import com.eleden.pryeleden.R.id;

public class AcercaDe extends Activity {

    ListView acercaDe;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.acerca_de);
        acercaDe=(ListView) findViewById(id.Lv_acerca_de);
        final ClaseAcercaDe[] datos=
            new ClaseAcercaDe[]{new ClaseAcercaDe("Autor", "Diana Katherine Velasco Guanoluisa"),
                new ClaseAcercaDe("Sistema Operativo", "Android"),
                new ClaseAcercaDe("Version", "4.1.2"),
                new ClaseAcercaDe("Tema", "Gestión de pedidos."),
                new ClaseAcercaDe("Versión aplicación", "2.01"),
                new ClaseAcercaDe("Derechos reservados", "Autor D.K.V.G"),
            };

        AdaptadorAcercaDe adaptador=new AdaptadorAcercaDe(this,datos);
        acercaDe.setAdapter(adaptador);
    }
}

```

Figura 3.17 Código Java equivalente al Layout ACERCA DE..



Figura 3.18 Layout de ACERCA DE

3.4. Plan de pruebas

Al finalizar el desarrollo de una aplicación, después de un previo análisis de requerimientos de la situación actual de una empresa o de un sistema de información si existiera de antemano, es importante constatar si los parámetros preestablecidos por el usuario fueron cumplidos a cabalidad, para esto es necesario realizar un plan de pruebas.

3.4.1. Pruebas unitarias

“Las pruebas unitarias, modulares o de caja blanca (White Box), son casos de prueba escritos para cada función no trivial o método en el módulo, se las plantea de tal forma que cada caso sea independiente del resto. Estas pruebas permiten verificar el adecuado funcionamiento de cada uno de los módulos de código por separado para determinar si están listos, y correctamente terminadas las interfaces, o flujo de datos entre componentes.

El principal factor que se debe considerar al inicio de las pruebas es el tamaño del módulo a probar, se debe considerar si el tamaño del módulo permitirá probar adecuadamente toda su funcionalidad de manera sencilla y rápida. También es importante separar los módulos de acuerdo a su funcionalidad, si los módulos son muy grandes y contienen muchas funcionalidades, estos se volverán más complejos de probar y al encontrar algún error será más difícil ubicar la funcionalidad defectuosa y corregirla. Al hacer esta labor el analista de pruebas podrá recomendar que un módulo muy complejo sea separado en 2 o 3 módulos más sencillos”. [20]

Es posible realizar las pruebas antes de concluir todos los módulos, incluso, es oportuno iniciar la etapa de pruebas unitarias poco después del desarrollo, generalmente las pruebas modulares y las pruebas integrales se encubren.

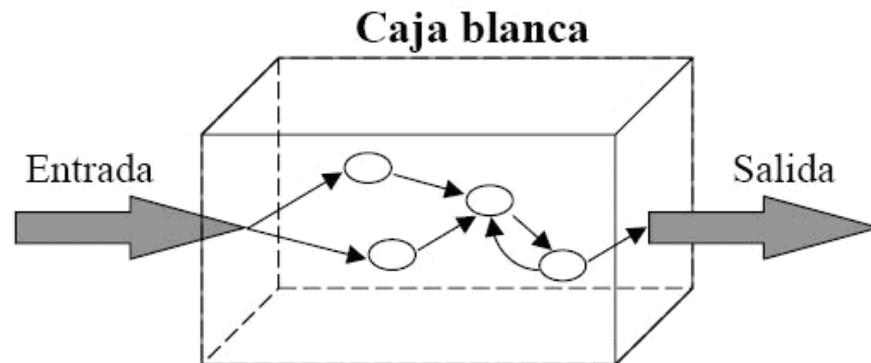


Figura 3.19 Pruebas de Caja Blanca - White Box Software Testing

FUENTE: http://www.calidadyssoftware.com/testing/pruebas_unitarias1.php

“En esta imagen se muestra lo que se considera una representación clásica de Software Testing White Box o pruebas de caja blanca, en este tipo de pruebas el cubo representaría un sistema en donde se pueden observar los diversos componentes que forman parte del mismo, cada uno de estos componentes debe ser probado en su totalidad (óvalos), y también sus interfaces o comunicaciones con los demás componentes (flechas), este tipo de pruebas también son llamadas pruebas de caja de cristal ya que este último término representa mejor el tipo de pruebas.” [20]

Requerimientos funcionales**Tabla 3.1****Pruebas unitarias Gestión de Clientes**

Entrada	Salida Esperada
Cédula Nombres Apellidos Teléfono E-mail Dirección	Datos del cliente almacenado Verificación en el listado de clientes

Tabla 3.2**Pruebas unitarias Gestión de Productos**

Entrada	Salida Esperada
Código Nombre del producto Tipo de Producto Precio Imagen	Datos del producto almacenado Verificación en el listado de productos

Tabla 3.3**Pruebas unitarias Gestión de Pedidos**

Entrada	Salida Esperada
Cliente Ciudad Transporte Detalle pedido	Datos del pedido almacenado Visualización del pedido

Requerimientos No Funcionales

- La aplicación debe ejecutarse en un dispositivo móvil con pantalla de 7" en orientación vertical para mayor legibilidad.
- La aplicación se desarrollará en la versión Android 4.2 para permitir la compatibilidad con las versiones similares o posteriores a ésta.

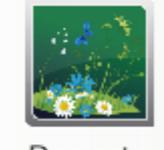
3.4.2. Pruebas de aceptación

Las pruebas de aceptación se las realiza para que el cliente certifique la validez o grado de calidad del sistema completo en su beneficio, es decir son pruebas funcionales que las ejecuta el cliente o personas involucradas directamente en el negocio, para comprobar si sus requisitos establecidos se satisfacen. Este tipo de pruebas se las realiza en etapas tempranas del desarrollo para utilizar los resultados como indicadores de su eficacia; si los resultados de dichas pruebas son satisfactorios para el cliente, el producto se considera terminado y, por tanto, propicio para su puesta en producción.

Los conocimientos informáticos previos no son necesarios para la preparación, ejecución y evaluación de las pruebas de aceptación pero ciertos conocimientos de métodos y técnicas de prueba y de la gestión de la calidad en general facilitarían esta labor.

Tabla 3.4

Resultados de la Prueba

MÓDULO	USUARIO	OBSERVACIONES
 Clientes	Byron Velasco	Fácil ingreso
 Productos	Byron Velasco	Entendible
 Pedidos	Byron Velasco	Excelente
 Reporte	Byron Velasco	Buena Opción
 Enlaces	Byron Velasco	Útil
 Acerca de	Byron Velasco	Ok

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- Entre la gran oferta de herramientas libres para el desarrollo de aplicaciones móviles en Android, Eclipse es la mejor IDE, puesto que presenta mayores beneficios y se adapta fácilmente al trabajo que se desea realizar, tanto así, que empresas reconocidas como Google la recomienda para la creación de Apps.
- Android es un Sistema Operativo óptimo que sirve de plataforma para desarrollar aplicaciones móviles con la ayuda de Eclipse, puesto que es de distribución libre considerando que siempre está disponible para su uso, desarrollo y distribución comercial.
- Utilizando Android, se desarrolló el sistema de gestión de pedidos para la Comercializadora “El Edén”, la cual será un aporte significativo para la empresa, ya que esto permitirá la eficiente y correcta administración de los pedidos de los clientes, gracias a la base de datos relacional y de dominio público SQLite.
- El Sistema de Gestión de Pedidos en las pruebas realizadas permitió agilizar los procesos cotidianos dentro de la Comercializadora “El Edén”, evitando la pérdida de información y facilitando su disponibilidad.

4.2.Recomendaciones

- Actualmente se recomienda optar por el IDE Android Estudio porque fue liberado y es posible aprovechar todas sus características como usabilidad y funcionalidades para el desarrollo de las aplicaciones, a diferencia de cuando se inició el presente estudio únicamente se disponía de la versión Beta, aunque aún hay que ser cuidadosos en cuanto a su estabilidad.
- Debe existir una conexión de internet con un ancho de banda amplio para una descarga eficiente de las API's de Android (versiones de Android).
- En el desarrollo de la aplicación es necesario definir y elegir de forma adecuada el dispositivo en el cual se va a ejecutar la aplicación móvil, puesto que de no ser así las interfaces gráficas tienden a distorsionarse.
- Seleccionar un adecuado SGBD para la administración de la BDD SQLite. (SQLite Manager de Firefox Mozilla presenta excelentes beneficios).
- Determinar adecuadamente, cuando es conveniente desarrollar una aplicación móvil o para PC, debido a la complejidad de ubicar muchos campos de entrada en una aplicación.

LINKOGRAFÍA

- [1] J. Marquina, «julianmarquina,» 26 Septiembre 2013. [En línea]. Available: <http://www.julianmarquina.es/el-86-de-las-personas-accede-a-internet-a-traves-del-smartphone/>.
- [2] P. Maldonado, «Revista Líderes,» 4 Enero 2015. [En línea]. Available: <http://www.revistalideres.ec/lideres/mundo-utiliza-apps.html>.
- [3] M. O. Cruz Ortega, «Slideshare,» 07 Julio 2013. [En línea]. Available: <http://es.slideshare.net/DriverNet/comunicaciones-moviles-23997054>.
- [4] Wikipedia, «Wikipedia,» 14 07 2014. [En línea]. Available: http://es.wikipedia.org/wiki/Dispositivo_m%C3%B3vil.
- [5] Hispalinux, «Hispalinux,» S/F. [En línea]. Available: <http://hispalinux.es/SoftwareLibre>.
- [6] G. O. Pedrozzo Petrazzini, 2012. [En línea]. Available: http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/Sistemas_Operativos_en_Dispositivos_Moviles.pdf.
- [7] Ecured, «Ecured,» [En línea]. Available: http://www.ecured.cu/index.php/HP_webOS.

- [8] Wikipedia, «Wikipedia,» 28 Julio 2014. [En línea]. Available:
<http://es.wikipedia.org/wiki/Bada>.
- [9] Java, «Java,» S/F. [En línea]. Available:
https://www.java.com/es/download/faq/whatis_java.xml.
- [10] Wikipedia, «Wikipedia,» 09 Diciembre 2013. [En línea]. Available:
http://es.wikipedia.org/wiki/Java_Development_Kit.
- [11] Wikipedia, «Wikipedia,» 26 Abril 2014. [En línea]. Available:
es.wikipedia.org/wiki/Kit_de_desarrollo_de_software.
- [12] Mundo Java, «Mundo Java,» 03 Octubre 2012. [En línea].
Available:
https://www.facebook.com/permalink.php?id=503295976366081&story_fbid=513489708680041.
- [13] Wikipedia, «Wikipedia,» 02 Diciembre 2013. [En línea]. Available:
<http://es.wikipedia.org/wiki/Dalvik>.
- [14] A. Aizaga, «Monografías.com,» [En línea]. Available:
<http://www.monografias.com/trabajos55/base-de-datos/base-de-datos.shtml>.
- [15] INTEF, «Intef,» [En línea]. Available:
<http://recursostic.educacion.es/usuarios/web/preguntas-frecuentes/45-sqlite->.

- [16] J. Pacheco, «GitHub,» 28 Marzo 2011. [En línea]. Available:
<https://github.com/javipacheco/Android-DataFramework/wiki/Android-Dataframework-->.
- [17] B. Arregocés Carrere, «Erosky Consumer,» 05 Enero 2009. [En línea]. Available:
<http://www.consumer.es/web/es/tecnologia/internet/2009/01/05/181935.php>.
- [18] Aulaclic, «Aulaclic,» Junio 2005. [En línea]. Available:
<http://www.aulaclic.es/articulos/wifi.html>.
- [19] R. Hernández, «S42,» 2011. [En línea]. Available:
<http://www.wwc.com.ve/ponencias2011/Aplicaciones%20Moviles.pdf>.
- [20] A. Ing. Oré, «CalidadSoftware.com,» 2009. [En línea]. Available:
http://www.calidadsoftware.com/testing/pruebas_unitarias1.php.
- [21] Wikipedia, «Wikipedia,» 10 Junio 2014. [En línea]. Available:
http://es.wikipedia.org/wiki/Entorno_de_desarrollo_integrado.

CERTIFICACIÓN

Certifico que el presente trabajo fue realizado en su totalidad por la Srta. Diana Katherine Velasco Guanoluisa como requerimiento parcial a la obtención del título de Ingeniero en Sistemas e Informática.

Director del Proyecto

Ing. Marcelo Álvarez

Codirector del Proyecto

Ing. Javier Montaluisa

Director de la Carrera

Ing. Lucas Garcés

Secretario Académico

Doctor Rodrigo Vaca Corrales