



UNIVERSIDAD DEL AZUAY

FACULTAD DE: Ciencias de la Administración

ESCUELA: Ingeniería de Sistemas

**“SISTEMA INFORMÁTICO PARA LA ATENCIÓN DE UN
RESTAURANTE”**

**Monografía previa a la obtención del título de:
Ingeniero en Sistemas**

Autor: César Tenesaca Narváez

Director: Oswaldo Merchán Manzano

Cuenca, Ecuador

2015

DEDICATORIA

Dedico esta monografía a mi familia, A mi esposa, quien ha estado a mi lado todo este tiempo en que he estado realizando este trabajo. A mis padres que me han apoyado en el ámbito económico y emocional. A mi hermana por la insistencia de que lo haga y a todas las personas que estuvieron y estarán siempre a mi lado, dedico este trabajo con cariño y un muy grande agradecimiento.

ÍNDICE DE CONTENIDOS

CONTENIDOS	PÁGINAS
Portada	i
Dedicatoria	ii
Índice	iii
Resumen	v
Abstract	vi
Introducción	vii
Objetivo General	vii
Objetivos Específicos	vii
1. Capítulo 1: Marco Teórico	1
1.1. Introducción	1
1.2. ¿Qué es sistema Android?	1
1.3. Versiones de Android que existen	2
1.3.1. Android versión 1.0. Nivel de API 1, (septiembre 2008)	2
1.3.2. Android versión 1.1. Nivel de API 2, (febrero 2009)	2
1.3.3. Android versión 1.5. Nivel de API 3 (Cupcake, abril 2009)	2
1.3.4. Android versión 1.6. Nivel de API 4 (Donut, septiembre 2009)	3
1.3.5. Android versión 2.0. Nivel de API 5 (Éclair, octubre 2009)	3
1.3.6. Android versión 2.1. Nivel de API 7 (Éclair, enero 2010)	3
1.3.7. Android versión 2.2. Nivel de API 8 (Froyo, mayo 2010)	4
1.3.8. Android versión 2.3. Nivel de API 9 (Gingerbread, diciembre 2010)	5
1.3.9. Android versión 3.0. Nivel de API 9 (Honeycom, febrero 2011)	5
1.3.10. Librerías de compatibilidad	6
1.3.10.1. Android versión 3.1. Nivel de API 12 (mayo 2011)	6
1.3.10.2. Android versión 3.2. Nivel de API 13 (julio 2011)	6
1.3.11. Android versión 4.0. Nivel de API 14 (Ice Cream Sandwich, octubre 2012)	7
1.3.12. Android versión 4.0.3. Nivel de API 15 (diciembre 2011)	7
1.3.13. Android versión 4.1. Nivel de API 16 (Jelly Bean, julio 2012)	8
1.3.14. Android versión 4.2. Nivel de API 17 (noviembre 2012)	8
1.3.15. Android versión 4.3. Nivel de API 18 (julio 2013)	9
1.3.16. Android versión 4.4. Nivel de API 19 (KitKat, octubre 2013)	9
1.3.17. Android versión 5.0. Nivel de API 21 (Lollipop, noviembre 2014)	10
1.4. Conclusiones	14
2. Capítulo 2: Análisis de herramientas para la programación para Android	15
2.1. Introducción	15
2.2. Adobe Flash Builder	15

2.3. Eclipse	16
2.4. Conclusiones	18
3. Capítulo 3: Propuesta técnica	19
3.1. Introducción	19
3.2. Instalación de la herramienta elegida	19
3.2.1. Instalación máquina virtual Java	16
3.2.2. Instalación de Eclipse	19
3.2.3. Instalación del plugin ADT en eclipse	20
3.2.4. Instalación del SDK de Android	21
3.2.5. Instalamos el SDK, para Android, con el Android SDK Manager.	22
3.2.6. Vinculación de Eclipse con los SDK	23
3.2.7. Instalación del Emulador de Android	24
3.3. Desarrollo de la aplicación	25
3.3.1. Introducción	25
3.3.2. Análisis:	26
3.3.2.1. Elaboración de Diagrama de flujos de datos	26
3.3.2.2. Elaboración de Modelo Entidad Relación	26
3.3.2.2. Elaboración del Diccionario de Datos	27
3.3.2.4. Elaboración de Interfaces del Usuario	28
3.3.3. Diseño	29
3.3.3.1. Elaboración del servicio web	30
3.3.3.2. Elaboración del sistema de generación de la factura	31
3.3.3.3. Elaboración del sistema para el dispositivo móvil	31
3.3.4. Codificación o construcción	33
3.3.4.1. Pruebas Unitarias	33
3.3.4.2. Pruebas de Interconexión	34
3.3.4.3. Pruebas de Integración	34
3.3.4.4. Implementación del prototipo	34
3.3.5. Conclusiones	35
4. Conclusiones	35
5. Recomendaciones	35
6. Bibliografía	36

RESUMEN

Este trabajo de desarrollo generará una herramienta informática que nos permita mejorar los procesos de atención en un restaurante evitando la problemática en la coordinación entre: cliente, mesero, cocinero y el cajero.

La herramienta permitirá generar el pedido del cliente, la orden para la elaboración del pedido del cliente y la facturación de dicho pedido.

Esta aplicación desarrollada funcionará en cualquier dispositivo móvil que tenga instalado como sistema operativo base Android 2.2 o superior, esta aplicación permitirá al mesero ingresar el pedido de los clientes que visitan el restaurante, información que estará a disposición del cocinero, mesero y cajero que forman parte del proceso de atención al cliente.

Con la elaboración de este sistema se tendrá todo integrado desde la toma de pedido hasta la elaboración de la factura.

El desarrollo del prototipo proporcionará una alternativa para que el personal de los restaurantes pueda atender mejor a sus clientes y realizar eficazmente su trabajo con la utilización de mínimos recursos.

INFORMATION SYSTEM FOR RESTAURANTS

ABSTRACT

This work will generate a software tool to allow improving care processes in a restaurant, avoiding the coordination problems that occur between customer, waiter, cook and cashier. This tool will generate the customer's order, the request for processing the client's order, and the order invoicing. This application will work on any mobile device that has installed Android 2.2 or higher as base operating system; it will also allow the waiter to enter the customer's order. This information will be available to the cook, waiter and cashier, all of which are part of the customer's service process. With the development of this system, every process will be integrated, from the order taking to invoicing. The development of the prototype will provide an alternative to the restaurant staff to better serve its customers, and do their job effectively with minimal use of resources.



Translated by,
Lic. Lourdes Crespo

Introducción

Siguiendo el criterio de: “El cliente siempre tiene la razón”, surge la necesidad de realizar este trabajo de investigación para determinar la mejor opción de programación para desarrollar un sistema de toma de pedidos en un restaurante, ya que en la actualidad el proceso es muy demorado ya que el mesero tiene que avisar al cocinero que preparar y a su vez al cajero que es lo que tiene que cobrar.

El sistema completo tendrá la siguiente estructura. (Gráfico #1)

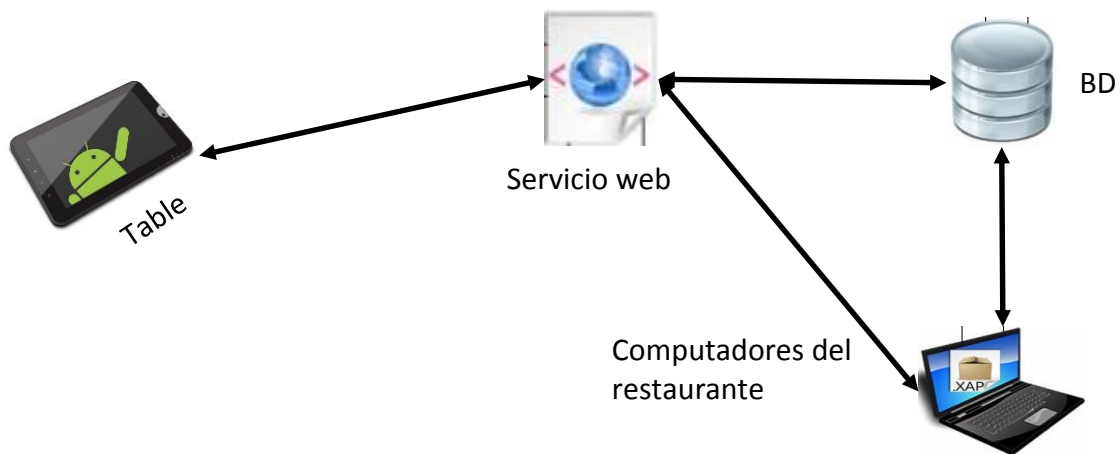


Gráfico #1. Estructura del sistema.

Objetivo general: Desarrollar una aplicación para mejorar la atención de un restaurante en la toma de pedidos de sus clientes.

Objetivos específicos:

- Fundamentar teóricamente la propuesta.
- Investigar las mejores alternativas para el desarrollo de la aplicación.
- Desarrollo de la aplicación.

1. Capítulo 1: Marco Teórico

1.1. Introducción

El marco teórico que fundamentará este proyecto proporcionará a las personas que lo lean una idea más clara de lo que es Android, y como funciona, encontrando los aspectos y conceptos básicos.

La información que encontramos en este marco teórico fue revisada en internet.

Toda la información encontrada y analizada, servirá para conocer más sobre el sistema operativo Android, el mismo que albergará nuestra aplicación, y determinaremos cuál es la versión idónea para la que debemos desarrollar la aplicación.

1.2. ¿Qué es sistema Android?

Android es un sistema operativo, desarrollado para funcionar especialmente en teléfonos móviles y tablets, no es el único, existen otros como: iOS, desarrollado por Apple, Symbian, desarrollado por Nokia y Blackberry OS, desarrollado por Blackberry.

Android se diferencia de los demás, debido a que emplea a Linux como núcleo del sistema, el cual es libre, gratuito y multiplataforma.

El sistema Android nos da la facilidad de programar funciones con una variación de Java llamada Dalvik¹, dejándonos utilizar todas las interfaces necesarias para el desarrollo de aplicaciones, las cuales podrán tener acceso a las funciones de los dispositivos móviles, tales como: GPS, llamadas, agenda, etc., todo de una forma muy sencilla, y usando java como lenguaje de programación.

¹ **Dalvik:** es la máquina virtual de proceso dentro de Android, es el software que ejecuta las aplicaciones en los dispositivos con Android.

1.3. Versiones que existen

Android apareció en septiembre de 2008, hasta la fecha se conocen 16 versiones del sistema operativo, a continuación revisaremos brevemente cada una de ellas.

1.3.1. Android versión 1.0. Nivel de API 1, (septiembre 2008)

Es la primera versión de Android, apareció en septiembre del 2008, nunca se utilizó correctamente debido a las limitaciones que tenía el sistema operativo para el desarrollo de aplicaciones.

1.3.2. Android versión 1.1. Nivel de API 2, (febrero 2009)

Poco después se lanzó la versión 1.1, no varía mucho con la versión anterior ya que se la hizo para corregir los errores de la versión anterior a la cual no se añadió ninguna funcionalidad adicional.

1.3.3. Android versión 1.5. Nivel de API 3 (Cupcake, abril 2009)

A esta versión se le dio el nombre clave de Cupcake, la novedad principal era la incorporación de un teclado en pantalla, con sus respectivas ventajas como la predicción de texto, y así eliminamos la necesidad de un teclado físico, entre otras novedades tiene la capacidad de grabación avanzada de audio y vídeo.

Una novedad fue la incorporación de *widjets*² y *live folders*³, los cuales se podían personalizar a gusto del usuario, la posibilidad de conectar auriculares *bluetooth* estéreo además de las posibilidades de tener transiciones entre las ventanas que se realizan mediante animaciones.

² **Widjets:** Es una pequeña aplicación o programa que tienen como objetivos dar fácil acceso a funciones frecuentemente usadas y proveer de información visual.

³ **Live Folder:** Carpetas de acceso directo.

1.3.4. Android versión 1.6. Nivel de API 4 (Donut, septiembre 2009)

Donut, la siguiente versión, tiene características un poco más elaboradas, como la capacidad de búsqueda avanzada en todo el dispositivo, gestures⁴, conversión de texto a voz, personalización de diferentes densidades de pantalla, soporte de pantallas WVGA, soporte para tecnologías CDMA/EVDO, 802.1x y VPNs.

1.3.5. Android versión 2.0. Nivel de API 5 (Éclair, octubre 2009)

Éclair, fue el nombre clave para esta versión de Android, el API de esta versión permite controlar el *bluetooth 2.1*, esto permite la sincronización de dispositivos que usen dicha tecnología, otras mejoras son la incorporación de un servicio centralizado de manejo de cuentas de usuario, soporte para Microsoft Exchange, mejora en la gestión de contactos y el calendario, más ajustes para la cámara de los dispositivos, aumento en el número de tamaños de ventana y resoluciones soportadas, nueva interfaz para el navegador y soporte para HTML5, esto complementándose con la optimización de la velocidad de *hardware* y el soporte de eventos en pantallas multitáctil.

1.3.6. Android versión 2.1. Nivel de API 7 (Éclair, enero 2010)

Es una versión que no tiene cambios significativos y por eso se sigue llamando Éclair, se destaca por el reconocimiento de voz, el mismo que permite introducir texto en un campo sin necesidad de utilizar el teclado, el desarrollo de fondos de pantalla animados y la opción de obtener información sobre la señal de la red actual que posea el dispositivo. En el paquete WebKit, se incluye nuevos métodos para manipular bases de datos almacenadas en Internet.

⁴ **Gesture:** Es un movimiento pregrabado sobre la pantalla táctil, que la aplicación puede reconocer

1.3.7. Android versión 2.2. Nivel de API 8 (Froyo, mayo 2010)

Froyo fue una versión que impuso nuevas características y mejoras al sistema operativo, de tal forma que se mejoró la velocidad en las aplicaciones entre 2 a 5 veces más que en la versión anterior, esto de acuerdo a varios *benchmarks*⁵ realizados por los *testers del sistema*.

Se agregan mejoras relacionadas con el navegador Web, como la incorporación de Adobe Flash 10.1 y Javascript V8 utilizado en Chrome.

Esta versión permite la instalación y almacenamiento de aplicaciones desde un medio externo como una memoria expandible, además de que se puede usar la memoria interna del dispositivo, la aplicación se actualiza de forma automática cuando se dispone de una nueva versión.

Cuenta con un servicio para realizar copias de seguridad desde la propia aplicación para comodidad del usuario, se facilita el uso de aplicaciones y programas a través del reconocimiento de voz.

Permite definir modos de interfaz del usuario en cuanto a nitidez de la pantalla para que las aplicaciones sean más cómodas y estén a gusto del usuario gracias a las mejoras en el API gráfica, la cual usaría OpenGL ES 2.0.

Por último, al utilizar el reconocimiento de voz genera comodidad para el usuario y de igual manera interactúa con varios programas de reconocimiento de voz también utilitarios.

⁵ **Benchmarks:** Son programas que se utilizan como banco de pruebas para evaluar determinadas capacidades del hardware o del software de forma objetiva.

1.3.8. Android versión 2.3. Nivel de API 9 (Gingerbread, diciembre 2010)

Esta versión, denominada Gingerbread, apareció ya que para ese entonces Android ya ha tenido mucha acogida en estos últimos años.

Incluye una interfaz de usuario con diseño actualizado, las funciones de “cortar, copiar y pegar” ahora son más fáciles de usar y el teclado en pantalla es multifuncional.

Se incluye soporte nativo para varias cámaras, considerando una segunda cámara que se usa para una videoconferencia, otro uso de la segunda cámara es reconocimiento facial para identificar el usuario del terminal.

Esta versión mejora el consumo de energía, y eso ayuda a garantizar una mejor respuesta en juegos y aplicaciones similares al igual que su capacidad de respuesta.

Además de permitir el cierre de aplicaciones que se encuentran en desuso se han sido ejecutadas de forma previa.

Razón por la cual se cambia el sistema de ficheros de YAFFS a ext4.

1.3.9. Android versión 3.0. Nivel de API 9 (Honeycom, febrero 2011)

Honeycom, rediseñada la interfaz de usuario de forma completa creando paradigmas nuevos para la navegación e interacción. La novedad destacada son los *fragments*, que sirve para diseñar distintos elementos del interfaz de usuario.

La nueva interfaz se encuentra a la disposición de todas las aplicaciones, y esto incluye las versiones anteriores creadas por la misma plataforma. Pero esto solo se puede conseguir a través de la introducción de librerías de compatibilidad que pueden ser utilizadas en versiones anteriores a la 3.0.

1.3.10. Librerías de compatibilidad

1.3.10.1. Android versión 3.1. Nivel de API 12 (mayo 2011)

Esta versión permite el manejo de dispositivos conectados por USB (tanto host como dispositivo). De forma adicional se implementa la transferencia de fotos y vídeos (PTP/MTP) en tiempo real (RTP).

1.3.10.2. Android versión 3.2. Nivel de API 13 (julio 2011)

Esta versión de Android se hizo para la optimización de las distintas tabletas, compatibilidad para aplicaciones de tamaño fijo y la sincronización automática de la información multimedia desde SD.

Se mejora los gráficos 2D/3D, gracias a la animación en 3D como en el OpenGL, siendo acelerado por hardware y el nuevo generador de gráficos Rederscript, que incorpora su propia API, la animación es mucho más flexible, también conocida como animación de propiedades.

Esta es la primera versión de la plataforma Android que soporta procesadores multinúcleo. La máquina virtual Dalvik está optimizado para un funcionamiento de multiprocesos.

Muchas de las mejoras afectan a los documentos multimedia como las listas de reproducción generadas a través de HTTP Live Streaming, considerando los protocolos de protección de derechos de autor, con una mayor capacidad de transferencia de archivos por medio del USB con los parámetros de MTP y PTP.

Se consideran muchas otras alternativas de conectividad, como la conectividad a través del Bluetooth y HSP de audio que permite la comunicación simultánea entre otros dispositivos.

Una funcionalidad adicional es la de poder conectar teclados completos a través de los mismos medios como el USB y/ o Bluetooth.

Al crear nuevas tecnologías como la estamos tratando se puede tener más dispositivos que aligeren el entorno empresarial, pero de igual manera eso hace hincapié para introducir nuevas políticas administrativas como la encriptación de la información, su caducidad, cambios de contraseña y mejoras para administrar los dispositivos de forma efectiva y segura.

1.3.11. Android versión 4.0. Nivel de API 14 (Ice Cream Sandwich, octubre 2012)

Ice Cream Sandwich unifica dos de las versiones anteriores (2.x para teléfonos y 3.x para tabletas) en una sola que se pueda utilizar en cualquier dispositivo. Entre sus características tenemos: nueva interfaz de usuario, botones en pantalla, reconocimiento facial y reconocimiento de voz.

Aparece un menú dentro de las configuraciones donde nos muestra el tráfico de datos por Internet, donde se puede constatar gráficamente el consumo y los límites de un plan de datos y esto sirve para evitar recargos innecesarios por parte de las operadoras móviles. En esta versión se dispone de múltiples herramientas para la edición de imágenes en tiempo real, se mejora el API para comunicaciones por el uso de medios inalámbricos como el Bluetooth y la integración de redes sociales.

1.3.12. Android versión 4.0.3. Nivel de API 15 (diciembre 2011)

Se introducen pocas mejoras en algunas APIs incluyendo el de las redes sociales, el calendario, el revisor ortográfico, de texto a voz y bases de datos entre otros.

1.3.13. Android versión 4.1. Nivel de API 16 (Jelly Bean, julio 2012)

Jelly Bean se enfoca en el mejoramiento del uso de la interfaz de usuario, de esta manera se prueban varias técnicas, como: sincronización vertical, el triple búfer y el aumento de la velocidad del procesador al manipular la pantalla.

Se deja ver un mejor sistema de notificaciones expandible que se pueden personalizar. Los Widgets de escritorio se pueden ajustar según el tamaño requerido y tener un sitio de forma automática en la pantalla del dispositivo. El comando de voz puede ser ejecutado sin conexión a Internet (solo en inglés).

Se mejora la aplicación Google Search, que potencia la búsqueda por voz con resultados fáciles de entender, siendo un rival al sistema SIRI del iOS. La aplicación Google Now utiliza información del GPS para los resultados, los datos de la agenda y la hora frecuente de la búsqueda.

Se agregan soportes para usuarios internacionales: como texto de doble entrada, soporte en otros idiomas y teclados rápidos de instalar. Se brindará mayor seguridad con aplicaciones cifradas.

1.3.14. Android versión 4.2. Nivel de API 17 (noviembre 2012)

Permite tener varios usuarios con sus respectivas cuentas dentro del mismo dispositivo, aunque cabe destacar que esta aplicación está disponible en las Tablet. Cada cuenta tendrá sus aplicaciones y configuración propias.

Permite la utilización de Widgets en la pantalla de bloqueo y se incorpora un teclado deslizante al estilo Skype, el mismo que es basado en gestos.

Se pueden conectar dispositivos diferentes y TV mediante la redes inalámbricas, la aplicación cámara tiene un nuevo modo, Photo Sphere, que permite hacer fotos panorámicas en realidad virtual, es decir a 360°.

1.3.15. Android versión 4.3. Nivel de API 18 (julio 2013)

Los programadores tienen la capacidad de definir restricciones en las apps, que cada uno de los usuarios pueden activar de ser su parecer. También se puede dar soporte a través del Bluetooth en la cual los dispositivos Android pueden interactuar entre los mismos con un consumo mínimo de recursos.

Para esta ejecución se considera tres características importantes como la codificación, la transmisión y la recolección de archivos multimedia. Esta versión permite el soporte para OpenGL versión 3.0.

Esta versión se encarga de mejorar la seguridad para gestionar y mantener ocultas las claves privadas de cada usuario.

1.3.16. Android versión 4.4. Nivel de API 19 (KitKat, octubre 2013)

Aunque se esperaba la versión número 5.0 con el nombre *Key Lime Pie*, Google sorprendió con el cambio de nombre, que se debió a un acuerdo con Nestlé para asociar ambas marcas, por lo que esta versión se terminó denominada KitKat.

KitKat tenía como principal objetivo estar a la disposición de varios dispositivos expandiendo su capacidad inclusive en los de memoria RAM de tan solo 512 MB.

Los componentes de Android han sido de alguna manera para decir recortados para que los requerimientos en la memoria no sean tan pesados y afecte a otros procesos de los dispositivos para ello se ha creado la nueva API que permite manejar el comportamiento de los dispositivos con poca memoria.

La mejor novedad es la interfaz que permite el modo de pantalla completa, WebViews (componentes de la interfaz de usuario para mostrar las páginas Web), que ahora se basan en el navegador Chrome que permite la búsqueda de contenido basado en HTML5, son algunas de las mejoras que tiene.

La conectividad, con soporte de NFC, para la emulación de tarjetas de pago tipo HCE, protocolos sobre Bluetooth, soporte para mandos infrarrojos, sensores para disminuir el consumo de energía y sensor para contar pasos, son otras mejoras que con las nuevas tecnologías son aprovechadas por los usuarios, que le dan un nuevo atractivo a los terminales móviles.

Con la utilización de herramientas y servicios en la nube, esta versión tiene un acceso de forma rápida a dicha herramienta, en un marco de almacenamiento muy fácil de entender y de usar. Esta herramienta incluye el conocido document provider dentro de un tipo específico de content provider que sirve para abrir, crear y modificar documentos y a su vez seleccionar diferentes ficheros.

Son solo algunas de las mejoras que contiene también se ha incorporado un administrador de impresión para enviar documentos a través de WiFi a una impresora.

Desde una perspectiva técnica, se destaca la nueva máquina virtual ART, que consigue tiempos de ejecución muy superiores a la máquina Dalvik. Sin embargo, todavía está en una etapa experimental. Por defecto se utiliza la máquina virtual de Dalvik, permitiendo a los programadores activar opcionalmente ART para verificar que sus aplicaciones se ejecutan de forma correctamente.

1.3.17. Android versión 5.0. Nivel de API 21 (Lollipop, noviembre 2014)

La novedad más importante de Lollipop es la extensión de Android a nuevas plataformas, incluyendo Google Wear, Google TV y Google Card. Hay un cambio significativo en la arquitectura, al utilizar la máquina virtual ART en lugar de Dalvik. Esta






novedad ya había sido incorporada en la versión anterior a modo de prueba. ART mejora de forma considerable el tiempo de ejecución del código escrito en Java. Además se soporta dispositivos de 64 bits en procesadores ARM, x86, y MIPS. Muchas aplicaciones del sistema (Chrome, Gmail, etc.) se han incorporado en código nativo para una ejecución más rápida.




Desde el punto de vista del consumo de batería, hay que resaltar que en Lollipop el modo de ahorro de batería se activa por defecto. Este modo desconecta algunos componentes en caso de que la batería esté baja. Se incorpora una nueva API (*android.app.job.JobScheduler*) que nos permite que ciertos trabajos se realicen solo cuando se cumplan determinadas condiciones (por ejemplo con el dispositivo cargando). También se incluyen completas estadísticas para analizar el consumo que nuestras aplicaciones hacen de la batería.

En el campo Gráfico Android Lollipop incorpora soporte nativo para OpenGL ES 3.1. Además esta versión permite añadir a nuestras aplicaciones un paquete de extensión con funcionalidades gráficas avanzadas (fragment shader, tessellation, geometry shaders, AST, etc.).

Otro aspecto innovador de la nueva versión lo encontramos en el diseño de la interfaz de usuario. Se han cambiado los iconos, incluyendo los de la parte inferior (Retroceder, Inicio y Aplicaciones), que ahora son un triángulo, un círculo y un cuadrado.

En esta nueva versión se introduce un modo de bloqueo que impide al usuario salir de una aplicación y bloquea las notificaciones. Esto podría utilizarse, por ejemplo, para que mientras un usuario realiza un examen, no pueda ver las notificaciones, acceder a otras aplicaciones, o volver a la pantalla de inicio.

PRINCIPALES VERSIONES DE ANDROID	
VERSION	CARACTERISTICAS
<p>1.0 Cupcake</p> 	<ul style="list-style-type: none"> ▶ Está basada en el kernel 2.6.27 de Linux ▶ Ofrece mejoras en materia multimedia, como grabación de videos y sincronización de fotos y videos con YouTube y Picasa
<p>1.6 Donut</p> 	<ul style="list-style-type: none"> ▶ Incluye la búsqueda mediante voz ▶ Mejoras en el Android Market ▶ Mayor velocidad.
<p>2.0 Eclair</p> 	<ul style="list-style-type: none"> ▶ Llega la compatibilidad con HTML5 ▶ mejoras en los mapas de Google y Bluetooth 2.1.
<p>2.2 Froyo</p> 	<ul style="list-style-type: none"> ▶ Nuevo aumento de velocidad, funcionalidad hotspot para acceso WiFi ▶ Actualizaciones automáticas desde el Android Market.
<p>2.3 Gingerbread</p> 	<ul style="list-style-type: none"> ▶ Tiene mayor velocidad ▶ Mejoras en el sistema de introducción de texto intuitivo ▶ Más control sobre aplicaciones y funciones de red.

<p style="text-align: center;">3.0. Honeycomb</p> <div style="text-align: center;">  </div>	<ul style="list-style-type: none"> ▶ Se incorporan varias mejoras multimedia, como listas de reproducción M3U ▶ Transferencia de archivos multimedia a través de USB con los protocolos MTP y PTP. ▶ Permite conectar teclados completos por USB o Bluetooth.
<p style="text-align: center;">4.0 Ice Cream Sandwich</p> <div style="text-align: center;">  </div>	<ul style="list-style-type: none"> ▶ Nueva API de reconocedor facial, permite entre otras muchas aplicaciones desbloquear el teléfono a su propietario. ▶ Aparece un nuevo gestor de tráfico de datos por Internet, donde podremos ver el consumo de forma gráfica y donde podemos definir los límites a ese consumo para evitar cargos inesperados con la operadora.
<p style="text-align: center;">4.1 Jelly Bean</p> <div style="text-align: center;">  </div>	<ul style="list-style-type: none"> ▶ Se incorporan varias técnicas, como: sincronismo vertical, triple búfer y aumentar la velocidad del procesador al tocar la pantalla. ▶ Se mejoran las notificaciones con un sistema de información ▶ El dictado por voz puede realizarse sin conexión a ▶ Se introducen varias mejoras en Google Search.



<p style="text-align: center;">4.4KitKat</p> 	<ul style="list-style-type: none"> ▶ Disponible en una amplia de dispositivos, incluyendo aquellos con tamaños de memoria RAM de solo 512 MB. ▶ Permite adaptar el comportamiento de la aplicación en dispositivos con poca memoria.
<p style="text-align: center;">5.0 Lollipop</p> 	<ul style="list-style-type: none"> ▶ La extensión de Android a nuevas plataformas, incluyendo Google Wear, Google TV y Google Card. ▶ Cambio significativo en la arquitectura, al utilizar la máquina virtual ART en lugar de Dalvik.

Tabla #1

1.4. Conclusiones

Hemos aprendido que versiones existen del sistema operativo Android, lo cual nos permitirá, conocer que ventajas y que limitaciones tienen cada una de ellas y determinar cuál es la versión más eficaz para la cual vamos a programar y así aprovechar las funciones que esta versión posee.

La versión escogida es la 4.2, API 17, la misma que al momento del desarrollo de la aplicación es la más común y es la que se tiene instalada en la mayoría de los dispositivos móviles.

2. Capítulo 2: Análisis de herramientas para la programación para Android

2.1. Introducción

En este capítulo se va a analizar dos programas (Eclipse, Adobe Flash Builder) que permiten programar aplicaciones para sistemas operativos Android, y así determinaremos cual es la mejor opción para el desarrollo de la aplicación, tomando en cuenta, precios de los dos programas analizados y rapidez de programación.

2.2. Adobe Flash Builder



Adobe Flash Builder, es un entorno de desarrollo para la creación de juegos y aplicaciones utilizando el lenguaje ActionScript y el marco Flex de código abierto. Flash Builder incluye herramientas de pruebas profesionales, como supervisión de la red, y soporte las pruebas unitarias (*Ref. UMAN CONSULTORIA Y DESARROLLO, S.L.L, <http://www.formacionprofesional.info/manuales-de-todos-los-programas-de-adobe/>*).

Este programa se llamaba Flex 4, porque usa lenguaje Flex, que es un conjunto de librerías – o framework – para desarrollo de UI (user interface por sus siglas en ingles). Aunque ActionScript 3.0 (AS3) es el lenguaje para programar aplicaciones usando el Flex framework, AS3 no es Flex. Si el código AS3 que se programe utiliza los componentes y métodos definidos en el Flex framework, entonces se está usando Flex.

FlexBuilder es solo una herramienta para crear aplicaciones Flex, este no es un lenguaje de programación ni una plataforma. Tiene una vista de diseño, que puede usarse rápidamente para crear un UI en Flex, pero solo sirve para cosas simples. Flex es un SDK, y adobe nos ofrece comandos para poder compilarlo, así que se puede usar cualquier editor de texto en lugar de Flex Builder.

(*Ref. EPIWIKI, <http://www.epidataconsulting.com/tikiwiki/tiki-index.php?page=Flex>*).

El precio para poder utilizar este programa en la versión más completa es de 627,99 €, que a dólares sería de: 661,59, sin tomar en cuenta que cada actualización tiene su precio, la actual está en: 458,59 €. (Gráfico #2)

Versión Completa	Actualización
<p>Flash Builder 4.7 Premium Rapidly develop high-performing mobile, web, and desktop applications</p> <p>Quiero Versión completa <input type="checkbox"/></p> <p>comprar:</p> <p>Plataforma: Mac/Win</p> <p>Idioma: Inglés</p> <p>Entrega: Descargar</p> <p>Cantidad: 1</p> <p> 627,99 € IVA incl.</p> <p><input type="button" value="Añadir al carro"/></p>	<p>Flash Builder 4.7 Premium Rapidly develop high-performing mobile, web, and desktop applications</p> <p>Quiero Actualización <input type="checkbox"/></p> <p>comprar:</p> <p>Tengo: Flash Builder 4.7...</p> <p>Plataforma: Mac/Win</p> <p>Idioma: Inglés</p> <p>Entrega: Descargar</p> <p>Cantidad: 1</p> <p> 458,59 € IVA incl.</p> <p><input type="button" value="Añadir al carro"/></p>

(Gráfico #2. Precios de la aplicación)

(Ref. <http://www.adobe.com/es/products/flash-builder.html>)

2.3. Eclipse

Eclipse es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma, para desarrollar lo que el grupo de eclipse llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores.

(Ref. http://es.wikipedia.org/wiki/Eclipse_%28software%29)

Esta plataforma, ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado *Java Development Toolkit* (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent o Azureus.

(Ref. http://es.wikipedia.org/wiki/Eclipse_%28software%29)

Eclipse es también una comunidad de usuarios, extendiendo constantemente las áreas de aplicación cubiertas. Un ejemplo es el recientemente creado Eclipse Modeling Project, cubriendo casi todas las áreas de Model Driven Engineering, que se basa para el modelado de aplicaciones.

Desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge, es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.

Eclipse fue liberado originalmente bajo la Common Public License, pero después fue re-licenciado bajo la Eclipse Public License. La Free Software Foundation ha dicho que ambas licencias son licencias de software libre, pero son incompatibles con Licencia pública general de GNU, ya que Eclipse no es de código modificable.

Las características principales de este software:

1. Es gratuito,
2. Dispone de un editor de texto con un analizador sintáctico.
3. La compilación es en tiempo real.
4. Tiene pruebas unitarias con Junit,
5. Controla versiones con CVS,

Eclipse es una herramienta es gratuita, lo que hace más fácil utilizarla.

Para la programación móvil, necesitamos de Android SDK, que incluye una completa variedad de herramientas que ayudarán en el desarrollo de aplicaciones móviles para la plataforma Android. Entre las más importantes encontramos el Emulador de Android y el plugin que se integra con Eclipse, Android Development Tools, pero el SDK

también incluye una variedad de herramientas para debugging, packaging, e instalación de aplicaciones en el emulador.

2.4. Conclusiones

Después de determinar las dos opciones que tenemos en el mercado la una pagada y la otra gratuita se llegó a determinar que la mejor es ECLIPSE, por ser gratuita, tener integración, es decir se puede interactuar con cualquier otro tipo de lenguaje y base de datos y control de versiones de las aplicaciones desarrolladas en este programa.

Eclipse tiene un manejo de versiones, para llevar un control de las aplicaciones, con un ID llamado subversión.

Por lo tanto, en lo que se va a desarrollar la aplicación de esta monografía es en Eclipse.

3. Capítulo 3: Propuesta técnica

3.1. Introducción

En este capítulo vamos a preparar e instalar todo lo necesario para que la aplicación de esta monografía, que es la que va a tomar el pedido del cliente, sea desarrollada en Eclipse.

Conoceremos algunos conceptos básicos para que podamos entender la programación para dispositivos móviles.

3.2. Instalación de la herramienta elegida

Para poder instalar Eclipse y tener todo funcionando, debemos realizar los siguientes pasos, los mismos que nos indica Edwin Martínez Hernández, en su página web:

http://www.academia.edu/3439525/Programaci%C3%B3n_para_Dispositivos_M%C3%B3viles_Android_

3.2.1. Instalar máquina virtual de Java

Debemos tener instalado la máquina de Java la misma que la podemos descargar de: https://www.java.com/es/download/windows_xpi.jsp?locale=es

3.2.2. Instalación de Eclipse

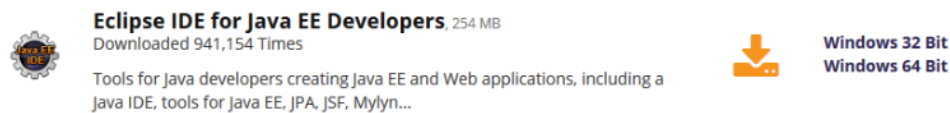


(Gráfico #1. Nombre del programa Eclipse)

Eclipse es un entorno de desarrollo de aplicaciones muy potente y que dispone de plugins para la mayoría de lenguajes de programación, entre ellos Android, que de hecho es un subconjunto de un lenguaje mayor denominado java, que fue desarrollado originalmente por Sun y adaptado posteriormente por Google.

La instalación es muy sencilla y consiste en descomprimir el contenido de un archivo que contiene la aplicación y las librerías en cualquier parte de nuestro computador.

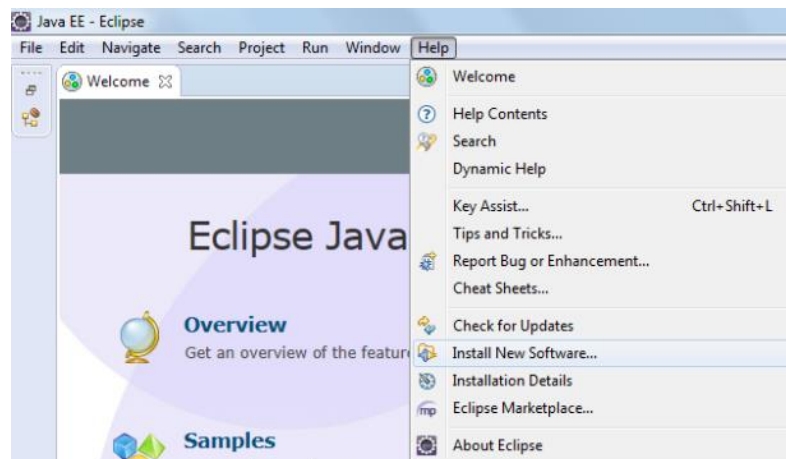
El link de descarga es <http://www.eclipse.org/downloads/>



(Gráfico #2. Descripción del sistema)

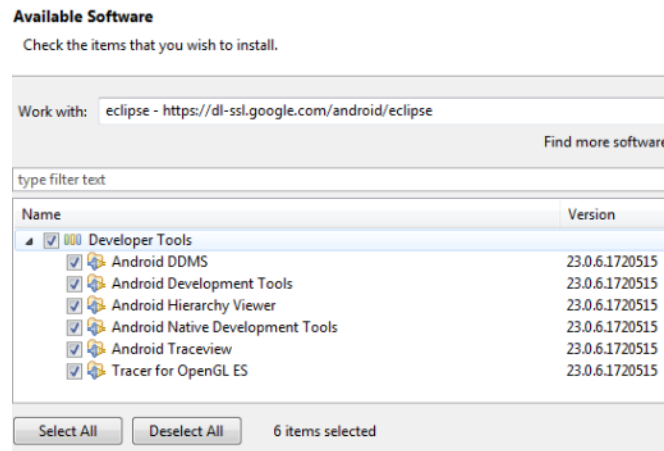
3.2.3. Instalación del plugin ADT en eclipse

Con el sistema abierto damos clic en el menú Help, seleccionamos *Install new software*, colocamos la siguiente dirección <https://dl-ssl.google.com/android/eclipse>, que corresponde a la ruta de los repositorios. (Gráfico #3)



(Gráfico #3. Instalación de los repositorios.)

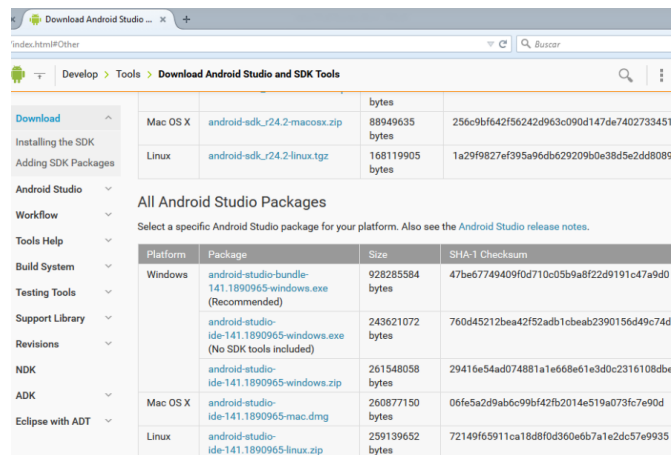
Realizamos el proceso de Instalación, tal como se indica en el Gráfico #4.



(Gráfico #4. Instalación de repositorios.)

3.2.4. Instalación del SDK de Android

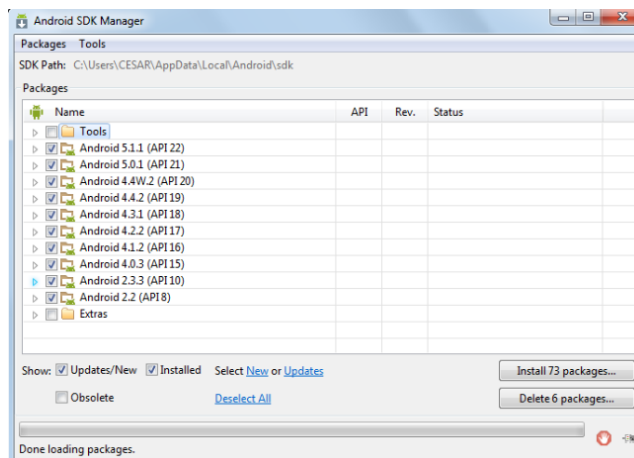
Para ello descargamos el siguiente archivo *android-studio-ide-135.1740770-windows.exe*, el mismo que lo podemos descargar de: <http://developer.android.com/sdk/index.html#Other>, descomprimos el archivo donde sea nuestra preferencia (Gráfico #5).



(Gráfico #5. Página de la cual tenemos que descargar el archivo)

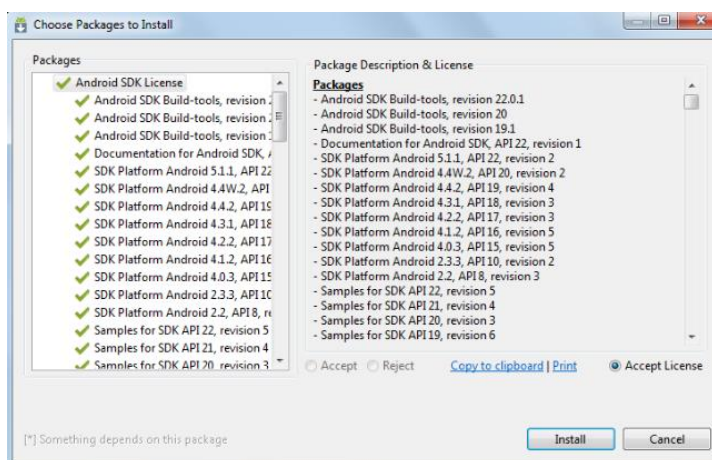
3.2.5. Instalamos el SDK, para Android, con el Android SDK Manager

Como nos indica la gráfica #6, aquí nos permitirá instalar las versiones de eclipse,



(Gráfico #6. Android SDK Manager)

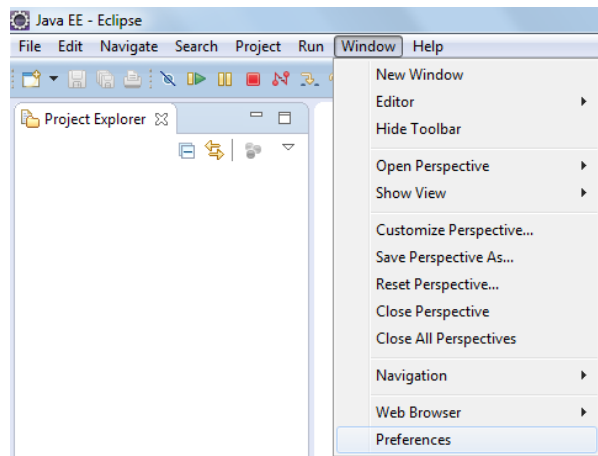
Seleccionamos las versiones que queremos instalar para nuestra aplicación, y damos clic en Install. (Gráfico #7)



(Gráfico #7. Versiones que se van a instalar)

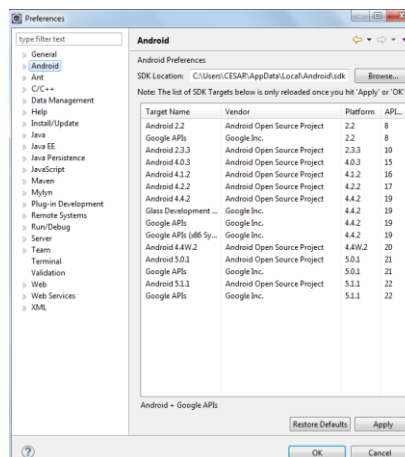
3.2.6. Vinculación de Eclipse con los SDK

Para ello seleccionamos Windows -> Preferencias, dentro de la aplicación Eclipse.
(Gráfico #8)



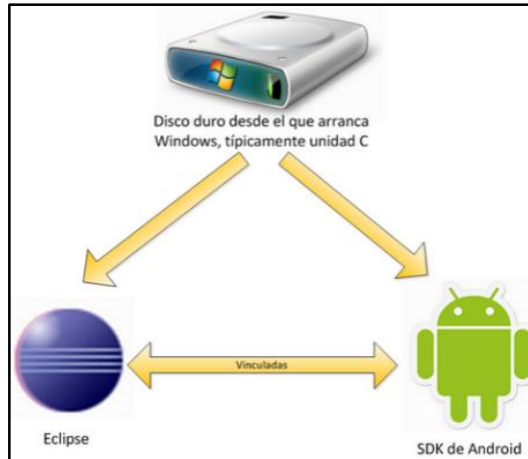
(Gráfico #8. Vinculación de Eclipse con los SDK.)

En la opción Android, seleccionamos la ruta en donde esta instalados los SDK
(Gráfico 9), los mismos que descargamos como nos indica el gráfico #5.



(Gráfico #9. Preferencia de ruta de los SDK)

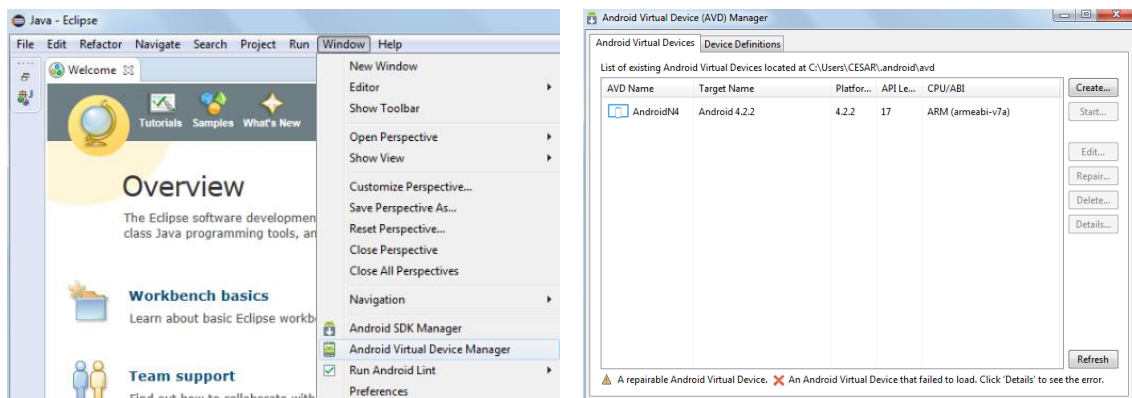
Con todo esto tenemos la siguiente relación (Gráfico #10):



(Gráfico #10. Vinculación entre Eclipse y SDK)

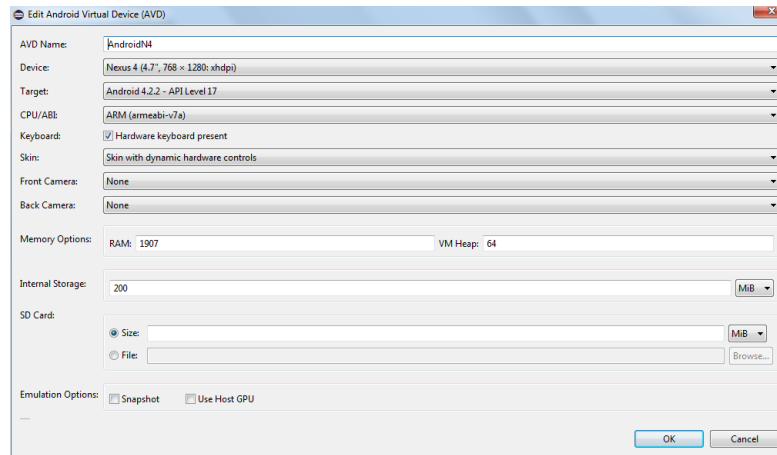
3.2.7. Instalación del Emulador de Android

Para ello damos clic en Windows, Android Virtual Device Manager (Gráfico #11)



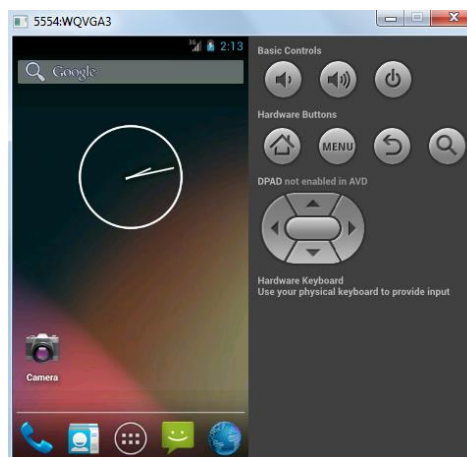
(Gráfico #11. Instalación del Emulador de Android)

Damos clic en Create, escribimos todos los datos que nos pide tal como nos indica el gráfico # 12 y ponemos en OK.



(Gráfico #12. Datos para la creación del Emulador)

Si iniciamos el Emulador creado, tendríamos algo parecido a esto. (Gráfico #13.)



(Gráfico #13. Emulador iniciado)

3.3. Desarrollo de la aplicación

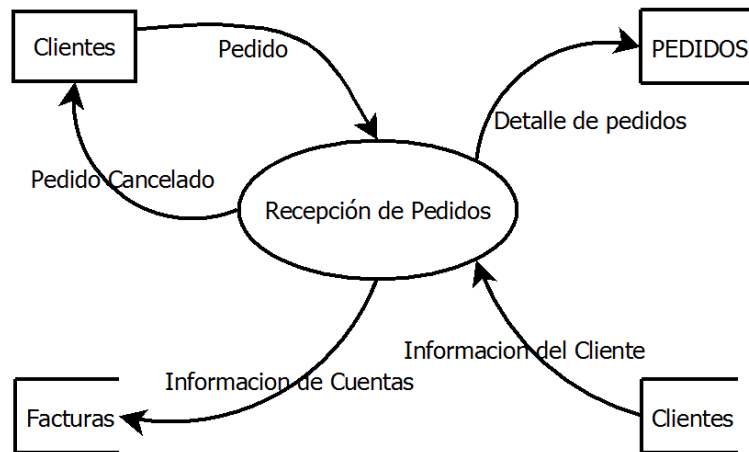
3.3.1. Introducción

En esta etapa de la monografía se analizará todo lo necesario para el desarrollo de la aplicación, se seguirá las siguientes etapas: Análisis, Diseño, Codificación o construcción, Implementación del Prototipo.

3.3.2. Análisis:

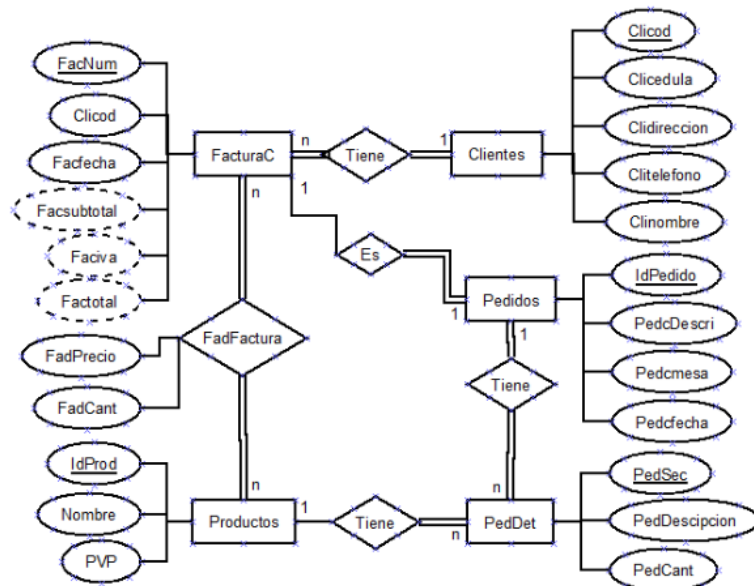
3.3.2.1. Elaboración de Diagrama de flujos de datos

La estructura del diagrama nos muestra de manera general el funcionamiento del sistema.



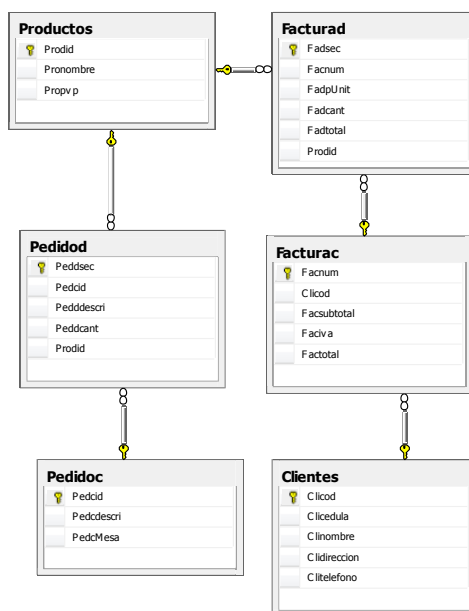
3.3.2.2. Elaboración de Modelo Entidad Relación

Nuestra la estructura de la base de datos.



3.3.2.3. Elaboración del Diccionario de Datos

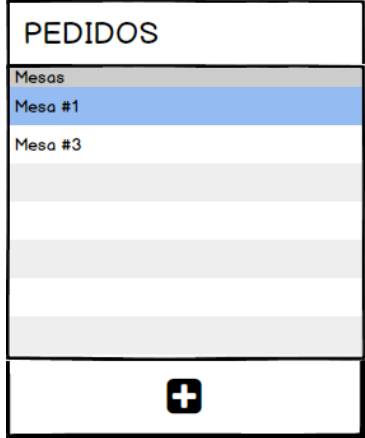
El desarrollo del diccionario de datos es baso en la siguiente base de datos:

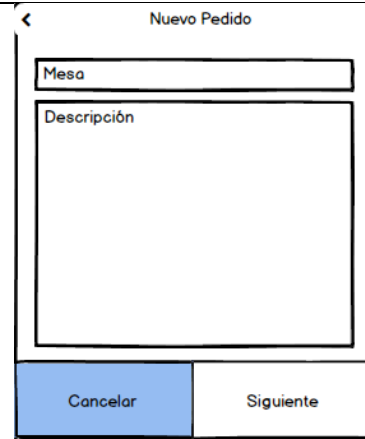
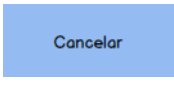
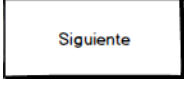


CAMPO	DESCRIPCIÓN	TAMAÑO	TIPO DE DATO	ES AUTONUMERICO	LLAVE PRIMARIA	LLAVE FORANEA	TABLA REFERENCIA	COLUMNA REFERENCIA
TABLA: CLIENTES								
Clicod	Llave primaria de la tabla de clientes	4	int	SI	SI	NULL	NULL	NULL
Clicedula	Cédula del cliente	20	nvarchar	NO	NO	NULL	NULL	NULL
Clinombre	Nombre del cliente	60	nvarchar	NO	NO	NULL	NULL	NULL
Cldireccion	Dirección del cliente	60	nvarchar	NO	NO	NULL	NULL	NULL
Clictelefono	Teléfono del cliente	20	nvarchar	NO	NO	NULL	NULL	NULL
TABLA: FACTURAC								
Facnum	Identificador de la tabla de factura	4	int	SI	SI	NULL	NULL	NULL
Facfecha	Fecha en la cual se hace la factura	3	datetime	NO	NO	NULL	NULL	NULL
Clicod	Identificador de la tabla del cliente	4	int	NO	NO	FK_Facturac_Clientes	Clientes	Clicod
Facsubtotal	Subtotal de la factura	9	numeric	NO	NO	NULL	NULL	NULL
Faciva	IVA de la factura	9	numeric	NO	NO	NULL	NULL	NULL
Factotal	Total de la factura	9	numeric	NO	NO	NULL	NULL	NULL
TABLA: FACTURAD								
Fadsec	Identificador del detalle de la factura	4	int	SI	SI	NULL	NULL	NULL
Facnum	Identificador de la tabla de la cabecera de la factura	4	int	NO	NO	FK_Facturad_Facturac	Facturac	Facnum
FadpUnit	Precio unitario del plato que consumió el cliente	9	numeric	NO	NO	NULL	NULL	NULL
Fadcant	Cantidad de platos que pidió el cliente	4	int	NO	NO	NULL	NULL	NULL
Fadtotal	Total de ese plato que pidió el cliente	9	numeric	NO	NO	NULL	NULL	NULL
ProdId	Identificador de la tabla de productos	4	int	NO	NO	FK_Facturad_Productos	Productos	ProdId
TABLAS: PEDIDOC								
PeddId	Identificador de la tabla de pedido	4	int	SI	SI	NULL	NULL	NULL
Peddfecha	Fecha en la cual se hace el pedido	4	datetime	NO	NO	NULL	NULL	NULL
PeddDescrip	Descripción del pedido, la misma que es opcional	60	nvarchar	NO	NO	NULL	NULL	NULL
PeddMesa	Describe el número de la mesa en donde está ubicado el cliente.	4	int	NO	NO	NULL	NULL	NULL
PeddEstado	Determina el estado del pedido, si está facturado o no	4	int	NO	NO	NULL	NULL	NULL
TABLA: PEDIDOD								
Peddsec	Identificador del detalle del pedido	4	int	SI	SI	NULL	NULL	NULL
PeddId	Identificador de la tabla de pedido	4	int	NO	NO	FK_Peidod_Peidoc	Peidoc	PeddId
PeddDescrip	Descripción del plato que pidió el cliente	16	texto	NO	NO	NULL	NULL	NULL
PeddCant	Cantidad de platos que van a pedir el cliente	4	int	NO	NO	NULL	NULL	NULL
ProdId	Identificador de la tabla de productos	4	int	NO	NO	FK_Peidod_Productos	Productos	ProdId
TABLA: PRODUCTOS								
ProdId	Identificador de productos o platos que tiene el restaurante para la venta	4	int	SI	SI	NULL	NULL	NULL
Pronombre	Nombre del plato	60	nvarchar	NO	NO	NULL	NULL	NULL
Propvp	Precio de venta al público de ese plato	9	numeric	NO	NO	NULL	NULL	NULL

3.3.2.4. Elaboración de Interfaces del Usuario

El diseño de la aplicación del dispositivo móvil sería de la siguiente manera

PAN01	
	Pantalla principal, ésta es la pantalla principal de la aplicación, aparece cuando se abre la aplicación.

PAN02	
	Cuando da click en la opción de agregar pedidos, se abre esta pantalla que nos permite ingresar un nuevo pedido en el cual se va a colocar el número de la mesa en la que se tomó el pedido y una descripción del mismo.
Si da clic en 	regresa a la PAN01.
Si da clic en 	Avanza a la PAN03

PAN03

Mesa:

Producto	Cantidad
Chaulafan	2
Colas	2

Agregar Preparar

Esta pantalla es para agregar los platos al pedido de la mesa que nos indica en la parte superior.

Agregar

Si pone en , aparece la pantalla que nos permite seleccionar los platos a consumir, abriéndonos la pantalla PAN04.

Preparar

El botón es para confirmar el pedido.

PAN04

< Nuevo Plato

Cantidad

Platos

Descripción

Cancelar Listo

En esta pantalla se ingresa los platos que están a disposición del restaurante.

Listo

El botón sirve para confirmar el plato escogido.

Cancelar

El botón para cancelar el ingreso de ese plato.

3.3.3. Diseño

En esta etapa vamos a programar todo lo necesario para que el prototipo de toma de pedidos funcione correctamente.

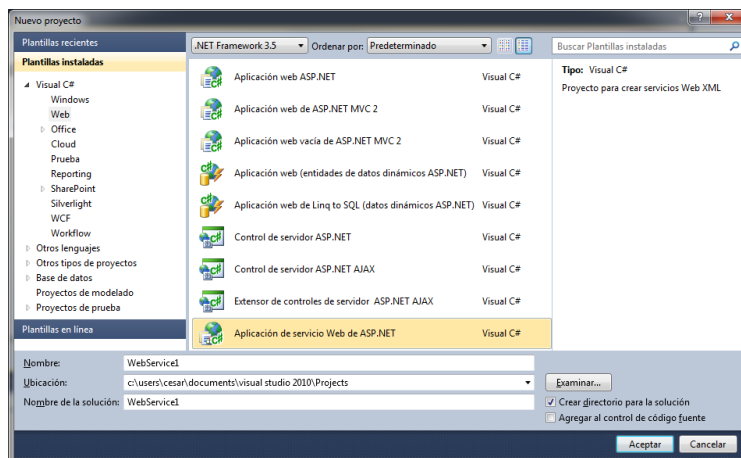
3.3.3.1. Elaboración del servicio web

Es la codificación del servicio web el mismo que se desarrolló en Microsoft Visual Studio 2010, con lenguaje C#.

Para que la aplicación del dispositivo móvil pueda interactuar con el servicio web debe estar publicado en IIS 7, y en la aplicación tener descargada la librería *ksoap2-android*, que es la que nos permitirá ahorrar código ya que lo tiene integrado para poder acceder a los servicios web. Para poder tener acceso a esta librería la descargaremos de la siguiente dirección:

<http://ksoap2-android.googlecode.com/svn/m2-repo/com/google/code/ksoap2-android/ksoap2-android-assembly/3.4.0/ksoap2-android-assembly-3.4.0-jar-with-dependencies.jar>

Para crear el servicio web, abrimos Visual Studio 2010, y elegimos la plantilla de *Aplicación de Servicio Web de Asp.Net*, el mismo que funciona solo con Framework 3.5. (Gráfico #14)



(Gráfico #14. Pantalla de inicio para crear el servicio web)

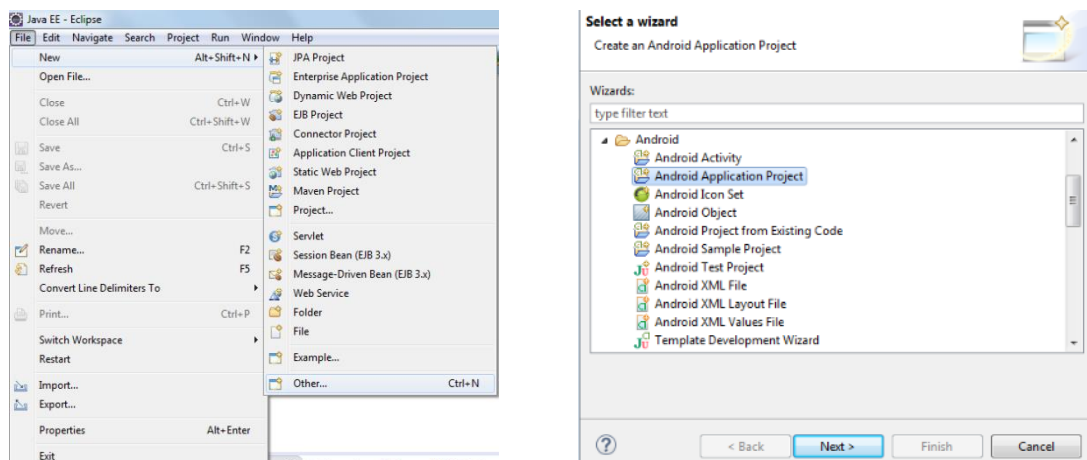
3.3.3.2. Elaboración del sistema de generación de la factura

El desarrollo de esta aplicación va a ser en Visual Basic 2010, con lenguaje C#, y va a estar funcionando en los computadores del restaurante, esta también será la encargada de mostrar los pedidos que se vayan generando en el restaurante.

Usa el servicio web para acceder a la base de datos y realizar las operaciones correspondientes.

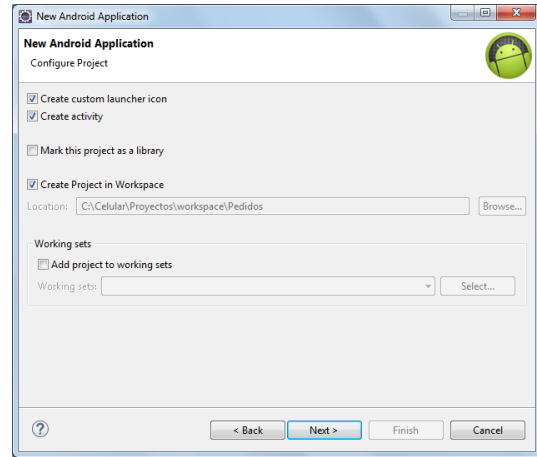
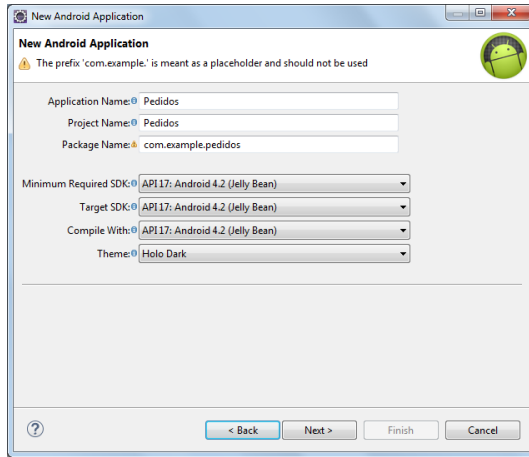
3.3.3.3. Elaboración del sistema para el dispositivo móvil

Para crear la aplicación en Eclipse damos clic en File -> New -> Other, y seleccionamos Android -> Android Application Project, como lo se indica en el gráfico # 15



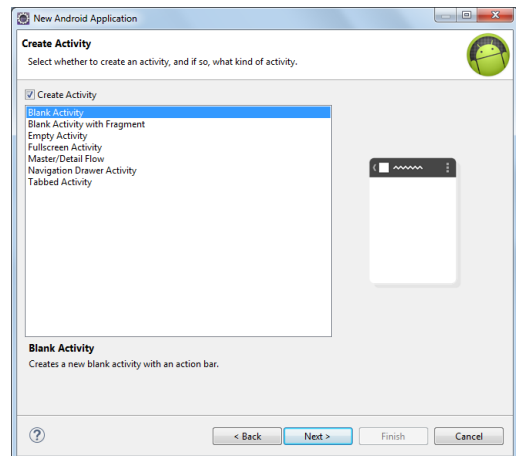
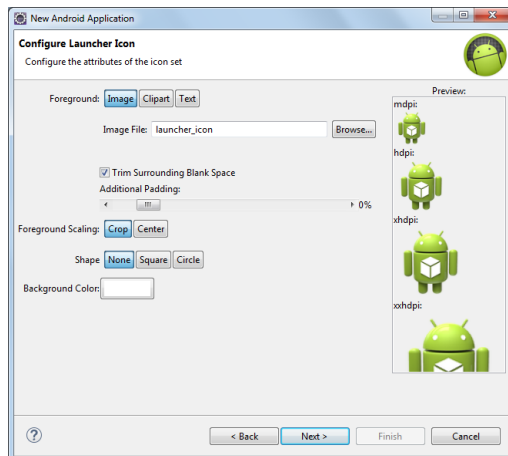
(Gráfico #15. Creación de la aplicación móvil)

Colocamos los datos del proyecto, gráfico #16



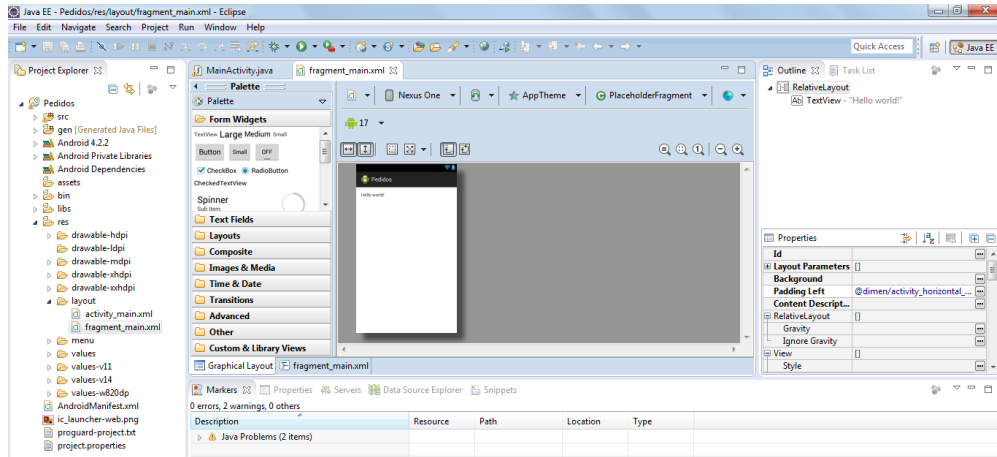
(Gráfico #16. Datos de la aplicación móvil)

Seleccionamos el icono que representará a nuestra aplicación y la creación de la actividad principal, gráfico # 17.



(Gráfico #17. Datos de la aplicación)

Después de crear la aplicación nos muestra el entorno de desarrollo, gráfico #18.



(Gráfico #18. Entorno de desarrollo)

3.3.4. Codificación o construcción

3.3.4.1. Pruebas Unitarias

Las pruebas unitarias de los sistemas y su funcionamiento es hacer una prueba con la siguiente información:

Información	Resultado esperado
<p>Generar el siguiente pedido con la siguiente información:</p> <p>Para la mesa 2:</p> <p>1 churrasco,</p> <p>3 filete de carne:</p> <p style="padding-left: 40px;">1 a término medio,</p> <p style="padding-left: 40px;">2 a término cocido, sin pimienta y sin sal.</p> <p>4 gaseosas:</p> <p style="padding-left: 40px;">1 Coca – Cola.</p> <p style="padding-left: 40px;">2 Fiora.</p> <p style="padding-left: 40px;">1 Fanta.</p>	
La aplicación se abre correctamente	<input checked="" type="checkbox"/>
Permite el ingreso de datos	<input checked="" type="checkbox"/>

El servicio web funciona correctamente	<input checked="" type="checkbox"/>
Graba correctamente en la base de datos	<input checked="" type="checkbox"/>
El sistema web devuelve correctamente los datos procesados, para poder visualizarlo en los computadores del restaurante.	<input checked="" type="checkbox"/>
El sistema que genera la factura funciona correctamente generando la factura y permite cobrar al cliente	<input checked="" type="checkbox"/>

3.3.4.2. Pruebas de Interconexión

Realizando la prueba unitaria se determinó que se conecta correctamente los tres sistemas, de la siguiente manera.

Del dispositivo móvil al servicio web y del servicio web al sistema de facturación.

3.3.4.3. Pruebas de Integración

Al realizar las pruebas respectivas vemos que el servicio web, el prototipo de aplicación móvil para la toma de pedidos, y el sistema de facturación funciona correctamente. En todos los casos el funcionamiento y el procesamiento de la información ingresada es el correcto.

3.3.4.4. Implementación del prototipo

El prototipo funciona en un simulador en una máquina de sistema operativo Windows 7 Ultimate.

Para realizar la instalación en el dispositivo móvil de este prototipo, debemos copiar el archivo .apk, que se crea en la carpeta del proyecto de eclipse.

En el dispositivo instalamos el programa que lo copiamos.

3.3.5. Conclusiones

En este capítulo pudimos determinar cómo va a ser el funcionamiento de los sistemas, tanto el que va a funcionar en el dispositivo móvil, como el sistema para las máquinas del restaurante, y sabemos con qué aplicación serán desarrolladas.

4. Conclusiones

Con la realización de este prototipo, podemos determinar que la creación de aplicaciones para móviles en Eclipse es mucho más rápido y resulta más económica que con los demás programas que nos permite realizar esto. Tomando en cuenta los tiempos empleados en la elaboración de sistemas similares en el transcurso de la vida universitaria.

Saber cómo funciona los servicios web SOAP y como es la integración para otros sistemas clientes.

5. Recomendaciones

Con la elaboración de este prototipo y en base al trabajo realizado se obtiene que es necesario tener bases sólidas en programación java, para que el trabajo sea mucho más fácil de realizarlo.

Como recomendación final, este prototipo es fácil de realizarlo y que queda a libre exposición para tomarlo como base para sistemas en el futuro, tomado en cuenta las necesidades que se puedan mejorar, tales como ponerla a disposición de todos los usuarios de Android, subiendo esta aplicación a Google play, y subir a un servidor web, para que se pueda acceder desde cualquier lugar y a cualquier momento a través de internet.

6. Bibliografía

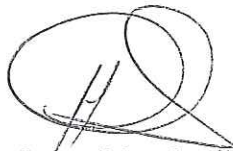
- Android, C. d. (s.f.). Obtenido de <http://www.tuprogramacion.com/programacion/comunicacion-android-con-webservices-soap/>
- Códigos de programación de interes.* (s.f.). Obtenido de Pensamientos varios sobre programacion: <https://github.com/sgolivernet/curso-android-src/blob/master/android-ws-soap/src/net/sgoliver/android/soap/MainActivity.java>
- Developers. (s.f.). *Android Developers*. Obtenido de <http://developer.android.com/reference/android/provider/LiveFolders.html>
- Developers. (s.f.). *Developers*. Obtenido de SDK, Android: <http://developer.android.com/sdk/installing/index.html?pkg=studio>
- Eclipse, M. d. (s.f.). *Eclipse.org*. Obtenido de <http://eclipse.org/subversive/>
- Epiwiki. (s.f.). *Epiwiki*. Obtenido de <http://www.epidataconsulting.com/tikiwiki/tiki-index.php?page=Flex>.
- Gabriela. (12 de 02 de 2014). *Hipertextual*. Obtenido de Hipertextual: <http://hipertextual.com/archivo/2014/02/que-es-dalvik-android/>
- Garzas, J. (s.f.). *Tipos de pruebas*. Obtenido de JavierGarzas: <http://www.javiergarzas.com/2014/07/tipos-de-pruebas-10-min.html>
- Github. (s.f.). *Repositorios de codigos de programación*. Obtenido de <https://github.com/sgolivernet/curso-android-src>
- González, G. (14 de Febrero de 2014). *Hipertextual*. Obtenido de <http://hipertextual.com/archivo/2014/02/que-es-dalvik-android/>
- Gueyraud, M. (27 de Febrero de 2015). *Programador Android Py*. Obtenido de http://programadorandroidpy.blogspot.com/2015_02_01_archive.html
- Hernandez, E. M. (s.f.). *Academia*. Obtenido de http://www.academia.edu/3439525/Programaci%C3%B3n_para_Dispositivos_M%C3%B3viles_Android_
- HtcMania. (s.f.). *Programación de aplicaciones para móviles Android - Unidad 1*. Obtenido de http://www.htcmania.com/mediawiki/index.php/Programaci%C3%B3n_de_aplicaciones_para_m%C3%B3viles_Android_-_Unidad_1
- Invarato, R. (s.f.). *Instalar el SDK de Android y Vincularlo con Eclipse*. Obtenido de <http://jarroba.com/instalar-el-sdk-de-android-y-vincularlo-con-eclipse/>
- LoMeli, F. (19 de Febrero de 2014). *FabiiLomeli*. Obtenido de http://fabiilomeli.blogspot.com/2014_02_01_archive.html

- Montar un entorno de programación para android.* (s.f.). Obtenido de <http://www.terminalesandroid.com/montar-el-entorno-de-programacion-para-android-con-eclipse>
- Pensamientos varios sobre programación.* (s.f.). Obtenido de Sgoliver: <http://www.sgoliver.net/blog/acceso-a-servicios-web-soap-en-android-22/>
- Pérez, G. (Dirección). (s.f.). *Conectar una Aplicación Android a un WebService con ASP.NET* [Película].
- Pérez, G. (s.f.). *Conectar una Aplicación Android a un WebService con ASP.NET.* Obtenido de <https://www.youtube.com/watch?v=jXyt-BwOdcE>
- Valencia, U. P. (s.f.). *Diploma de especialización en desarrollo de aplicaciones para android.* Obtenido de <http://www.androidcurso.com/index.php/tutoriales-android/36-unidad-5-entradas-en-android-teclado-pantalla-tactil-y-sensores/153-gestures>
- Valencia, U. P. (s.f.). *Diploma de especialización en desarrollo de aplicaciones para Android.* Obtenido de <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/146-las-versiones-de-android-y-niveles-de-api>
- Valencia, U. P. (s.f.). *Diplomado de Especialización en desarrollo de aplicaciones para Android.* Obtenido de <http://www.androidcurso.com/index.php/tutoriales-android/36-unidad-5-entradas-en-android-teclado-pantalla-tactil-y-sensores/153-gestures>
- Web, C. A.-S. (s.f.). Obtenido de <http://androideity.com/2012/06/03/consultar-un-servicio-web-en-android-parte-i/>
- Web, V. d.-c. (s.f.). Obtenido de <https://www.youtube.com/watch?v=jXyt-BwOdcE>
- Wikipedia. (s.f.). *Wikipedia.* Obtenido de Wikipedia: http://es.wikipedia.org/wiki/Eclipse_%28software%29
- Wikipedia. (s.f.). *Wikipedia.* Obtenido de https://es.wikipedia.org/wiki/Eclipse_%28software%29
- Wikipedia. (s.f.). *Wikipedia.* Obtenido de https://es.wikipedia.org/wiki/GNU_General_Public_License

CONVOCATORIA

Por disposición de la Junta Académica de Ingeniería de Sistemas y Telemática, se convoca a los Miembros del Tribunal Examinador, a la sustentación del Protocolo del Trabajo de Titulación : **"Sistema informático para la atención de un Restaurante"**, presentado por el estudiante César Xavier Tenesaca Narváez con código 33926, previa a la obtención del grado de Ingeniero de Sistemas, para el día **MARTES 24 DE FEBRERO DE 2015 A LAS 08H30.**

Cuenca, 20 de febrero de 2015



Dra. Jenny Ríos Coello
Secretaria de la Facultad

Ing. Oswaldo Merchán Manzano

Ing. Juan Carlos Salgado Arteaga



comisab
el Seno

12-02-2015

ESCUELA DE INGENIERIA DE SISTEMAS

*Diseños de Monografía
Escuela de Ingeniería de Sistemas*

Estudiante: César Xavier Tenesaca Narvárez con código 33926.

Tema: "SISTEMA INFORMATICO PARA LA ATENCION DE UN RESTAURANTE"

Para: La obtención del título de Ingeniero en Sistemas

Director: Ing. Oswaldo Merchán Manzano.

Tribunal: Ing. Juan Carlos Salgado

DIA: *MARTE*

FECHA: *2 DE FEBRERO 2015.*

HORA: *08h30.*



ACTA

SUSTENTACIÓN DE PROTOCOLO/DENUNCIA DEL TRABAJO DE TITULACIÓN

1.1 Nombre del estudiante: César Xavier Tenesaca Narváez

Código 33926

1.2 Director sugerido: Ing. Oswaldo Merchán Manzano

1.3 Codirector (opcional): _____

1.4 Tribunal: Ing. Juan Carlos Salgado Arteaga

1.5 Título propuesto: "Sistema informático para la atención de un Restaurante"

1.6 Resolución:

1.6.1 Aceptado sin modificaciones

1.6.2 Aceptado con las siguientes modificaciones:

- Responsable de dar seguimiento a las modificaciones Ing. Oswaldo Merchán Manzano

1.6.3 No aceptado

- Justificación:

Tribunal

Ing. Oswaldo Merchán Manzano

Sr. César Xavier Tenesaca Narváez

Ing. Juan Carlos Salgado Arteaga

Dra. Jenny Ríos Coello
Secretario de Facultad

Fecha de sustentación: Martes 24 de febrero de 2015



RÚBRICA PARA LA EVALUACIÓN DEL PROTOCOLO DE TRABAJO DE TITULACIÓN

1.1 Nombre del estudiante: César Xavier Tenesaca Narváez

1.1.1 Código 33926

1.2 Director sugerido: Ing. Oswaldo Merchán Manzano

1.3 Codirector (opcional):

1.4 Título propuesto: "Sistema informático para la atención de un Restaurante"

1.5 Revisores (tribunal): Ing. Juan Carlos Salgado Arteaga

1.6 Recomendaciones generales de la revisión:

	Cumple totalmente	Cumple parcialmente	No cumple	Observaciones (*)
Línea de investigación				
1. ¿El contenido se enmarca en la línea de investigación seleccionada?	✓			
Título Propuesto				
2. ¿Es informativo?	✓			
3. ¿Es conciso?				
Estado del arte				
4. ¿Identifica claramente el contexto histórico, científico, global y regional del tema del trabajo?	✓			
5. ¿Describe la teoría en la que se enmarca el trabajo	✓			
6. ¿Describe los trabajos relacionados más relevantes?	✓			
7. ¿Utiliza citas bibliográficas?	✓			
Problemática y/o pregunta de investigación				
8. ¿Presenta una descripción precisa y clara?	✓			
9. ¿Tiene relevancia profesional y social?	✓			
Hipótesis (opcional)				
10. ¿Se expresa de forma clara?	✓			
11. ¿Es factible de verificación?	✓			
Objetivo general				
12. ¿Concuerda con el problema formulado?	✓			
13. ¿Se encuentra redactado en tiempo verbal infinitivo?	✓			
Objetivos específicos				
14. ¿Concuerdan con el objetivo	✓			



general?				
15. ¿Son comprobables cualitativa o cuantitativamente?	✓			
Metodología				
16. ¿Se encuentran disponibles los datos y materiales mencionados?	✓			
17. ¿Las actividades se presentan siguiendo una secuencia lógica?	✓			
18. ¿Las actividades permitirán la consecución de los objetivos específicos planteados?	✓			
19. ¿Los datos, materiales y actividades mencionadas son adecuados para resolver el problema formulado?	✓			
Resultados esperados				
20. ¿Son relevantes para resolver o contribuir con el problema formulado?	✓			
21. ¿Concuerdan con los objetivos específicos?	✓			
22. ¿Se detalla la forma de presentación de los resultados?	✓			
23. ¿Los resultados esperados son consecuencia, en todos los casos, de las actividades mencionadas?	✓			
Supuestos y riesgos				
24. ¿Se mencionan los supuestos y riesgos más relevantes?				No hay
25. ¿Es conveniente llevar a cabo el trabajo dado los supuestos y riesgos mencionados?				
Presupuesto				
26. ¿El presupuesto es razonable?	✓			
27. ¿Se consideran los rubros más relevantes?	✓			
Cronograma				
28. ¿Los plazos para las actividades son realistas?	✓			
Referencias				
29. ¿Se siguen las recomendaciones de normas internacionales para citar?	✓			
Expresión escrita				
30. ¿La redacción es clara y fácilmente comprensible?	✓			
31. ¿El texto se encuentra libre de faltas ortográficas?	✓			

(*) Breve justificación, explicación o recomendación.

- Opcional cuando cumple totalmente,



- Obligatorio cuando cumple parcialmente y NO cumple.

.....

.....

.....

Ing. Oswaldo Merchán Manzano

Ing. Juan Carlos Salgado Arteaga

Oficio Nro. 015-2015-DIST-UDA

Cuenca, 10 de Febrero de 2015

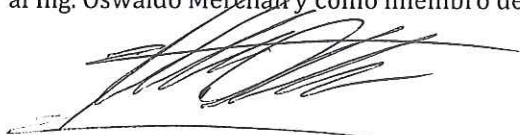
Señor Ingeniero
Xavier Ortega Vázquez
DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN
Presente.-

De nuestras consideraciones:

La Junta Académica de la Escuela de Ingeniería de Sistemas y Telemática, reunida el día 10 de Febrero del 2015, recibió el proyecto de monografía titulado "Sistema Informático para la atención de un Restaurante", presentada por el estudiante César Xavier Tenesaca Narváez, estudiante de la Escuela de Ingeniería de Sistemas y revisado por el Ing. Oswaldo Merchan, previo a la obtención del título de Ingeniero de Sistemas.

La Junta solicita por su digno intermedio notificar al tribunal designado y determinar lugar, fecha y hora de sustentación.

Por lo expuesto, y de conformidad con el Reglamento de Graduación de la Facultad, recomienda como director y responsable de aplicar cualquier modificación al diseño del trabajo de graduación posterior al Ing. Oswaldo Merchán y como miembro del Tribunal al Ing. Juan Carlos Salgado.


Atentamente,

Ing. Marcos Orellana Cordero
Director Escuela de Ingeniería de Sistemas y Telemática
Universidad del Azuay

Cuenca, 06 de Febrero del 2015.

Ingeniero Xavier Ortega Vásquez
DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN
Ciudad.

De mis consideraciones:

Yo, **CÉSAR XAVIER TENESACÁ NARVÁEZ**, con código estudiantil 33926, estudiante de la escuela de Ingeniería de Sistemas, solicito a usted de la manera más comedida y respetuosa que por su intermedio al Honorable Consejo de Facultad, se sirvan revisar el diseño de monografía titulado "Sistema Informático para la atención de un Restaurante", previa a la obtención del Título de Ingeniero de Sistemas.

Permitame, sugerir el nombre del Ingeniero, Oswaldo Merchán como director de esta monografía, por cuanto nos ha asesorado en la elaboración del presente diseño y además cuento con su aceptación firmada en carta adjunta.

Por la favorable acogida que se sirva a la presente, suscribo de usted.


César Tenesaca N.
0104112941

Cuenca, 6 de febrero de 2015

Señor Ingeniero

Xavier Ortega Vásquez

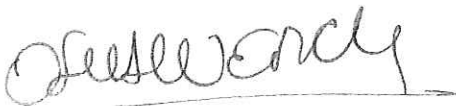
Decano de la Facultad de Ciencias de la Administración

Presente.

De mi consideración:

Una vez revisado el diseño del trabajo de graduación presentado por el señor César Xavier Tenesaca Narváz (código 33926), estudiante de la carrera de Ingeniería de Sistemas y Telemática, sobre el tema "Sistema informático para la atención de un restaurante", considero que el diseño cumple con las especificaciones legales y reglamentarias que exige la Universidad, razón por la cual, me permito recomendar su aprobación.

Atentamente,

A handwritten signature in black ink, appearing to read "Oswaldo Merchán", written over a horizontal line.

Oswaldo Merchán



UNIVERSIDAD DEL
AZUAY

DOCTORA JENNY RIOS COELLO SECRETARIA, DE LA FACULTAD DE
CIENCIAS DE LA ADMINISTRACIÓN DE LA UNIVERSIDAD DEL AZUAY

CERTIFICA:

Que, el Señor César Xavier Tenesaca Narváez, registrado con código 33926
perteneciente a la Escuela de Ingeniería de Sistemas luego de cumplir con todas las
asignaturas de su Pensum de estudios, egresó de la Facultad el 14 de Marzo de 2014.

Cuenca, Febrero 05 de 2015

UNIVERSIDAD DEL
AZUAY
FACULTAD DE
ADMINISTRACION
SECRETARIA

Derecho 114618

vcf.-



1. DATOS GENERALES

1.1. Nombre del estudiante: Tenesaca Narváez César Xavier.

1.1.1. Código: 33926

1.1.2. Contacto: 0995771218. cesarxtn@gmail.com

1.2. Director sugerido: Merchán Manzano Oswaldo.

1.2.1. Contacto: omerchan@uazuay.edu.ec

1.3. Título propuesto: Sistema Informático para la atención de un restaurante.

1.4. Línea de la Investigación de la carrera:

1.4.1. Código del UNESCO: 1203 Informática de computadores
1203.17. Informática

1.5. Área de Estudio: Programación y Base de Datos.

2. CONTENIDO

2.1 Motivación de la investigación: Basándonos en la ideología del "Cliente siempre tiene la razón", tomaremos pie para nuestro proyecto, ya que es lo primero que tenemos que tomar en cuenta cuando tengamos un negocios en el que se base la compra o venta de bienes o servicios, para que todo nos salga bien y tener a los clientes satisfechos.

2.2 Problemática: En la ciudad de Cuenca existe muchos restaurantes que no poseen de un sistema que permita al personal: cocinero, cajero, mesero automatizar el proceso desde el pedido del cliente, la orden al cocinero y la facturación.

En los procesos que actualmente se están llevando a cabo en los restaurantes, el de tomar el pedido, comunicar al cocinero para que prepare los alimentos e informar al cajero para que genere la factura, se estar perdiendo mucho en la adquisición de papel, y se utiliza demasiado tiempo, haciendo demorar al cliente.

2.3 Pregunta de investigación: Para el desarrollo de la aplicación hay que resolver una pregunta: cuál es la mejor alternativa para dar solución a los procesos de atención en un restaurante.

2.4 Resumen: Este trabajo de desarrollo es para generar una herramienta informática que nos permita mejorar la atención en un restaurante, y así evitar la problemática en la coordinación entre: cliente, mesero, cocinero y el cajero.

La herramienta permitirá generar la factura automáticamente con los precios de los alimentos que se sirvieron los clientes en el restaurante y va a permitir que el cocinero sepa que es lo que tiene que preparar.

Esta aplicación será desarrolla para que funcione en cualquier dispositivo móvil que tenga instalado como sistema operativo base Android 2.2 o superior, este sistema le permitirá al mesero ingresar el pedido de los clientes que visitan el restaurante, este mismo pedido estar disponible para que todo el personal del restaurante puedan verlo.

Con la elaboración de este sistema tendríamos todo integrado desde la toma de pedido has la elaboración de la factura.

El desarrollo de la aplicación proporciona una alternativa para que el personal de los restaurantes pueda atender mejor a sus clientes y realizar mejor y más eficaz su trabajo con la utilización de mínimos recursos.

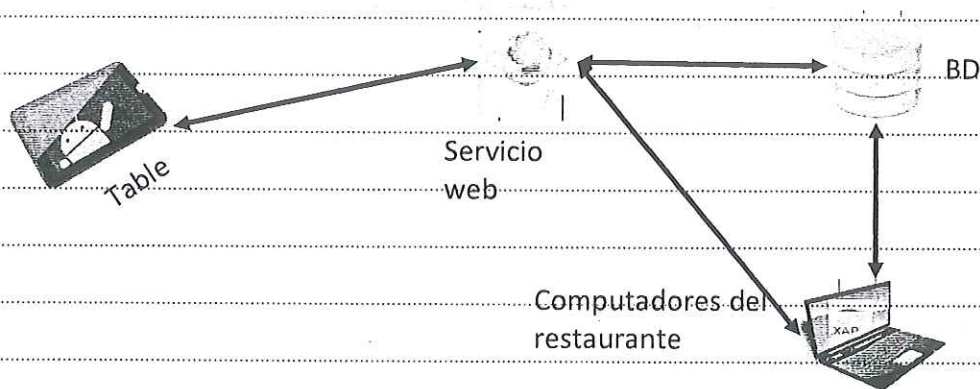
2.5 Estado del Arte y marco teórico: La aplicación, será desarrolla para cualquier dispositivo móvil que tenga instalado como sistema base Android 2.2 o superior, tales como tablet's o celulares.

El sistema Android es un sistema pensado para teléfonos móviles a igual que los sistemas iOS, Symbian y Blackberry OS. Lo que le hace diferente a todos estos sistemas es que está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma.

Para la elaboración de esta aplicación, se analizará dos opciones de programación: Eclipse y Adobe Flash Builer, y así determinar la mejor, tomando en cuenta la rapidez para el desarrollo y funciones que nos permiten hacer mejor nuestra aplicación.

La comunicación entre dispositivo móvil y los computadores del restaurante, será un servicio web y otra aplicación que genera la factura.

El sistema completo tendrá la siguiente estructura.



Para la elaboración del servicio web y la aplicación que generará la factura se desarrollará en Visual Studio 2010, con código C#.

La base de datos, será desarrollada en Sql Server 2008.

2.6 Objetivo general: Desarrollar una aplicación para mejorar la atención de un restaurante en la toma de pedidos de sus clientes.

2.7 Objetivos específicos:

- Fundamentar teóricamente la propuesta.
- Investigar las mejores alternativas para el desarrollo de la aplicación.
- Desarrollo de la aplicación.

2.8 Metodología: Para fundamenta la propuesta y conocer las alternativas que nos ofrece el mercado para el desarrollo de aplicaciones móviles las vamos a encontrar en internet.

Con toda la información que recolectemos vamos a desarrollar la aplicación, usando las técnicas investigadas.

2.9 Alcances y resultados esperados: Con la elaboración de la aplicación se pretende alcanzar a resolver la problemática que tienen los restaurantes al momento de tomar el pedido al cliente, y la generación de la factura con sus respectivos valores.

2.10 Presupuesto: Estaría conformado de la siguiente manera:

Rubro – Denominación	Costo USD (Detalle)	Justificación ¿Para qué?
Computador	700	Para desarrollar las aplicaciones necesaria.
Tablet	98	Para poner en funcionamiento la aplicación.
Internet	25 mensuales	Para investigar todo lo que se necesite saber para la elaboración de la aplicación
Total	823	

2.11. Financiamiento: El aporte será personal.

2.12. Esquema tentativo:

El esquema tentativo seria el siguiente.

Introducción

Resumen

Objetivos

Capítulo 1: Marco Teórico

Introducción

Que es sistema Android.

Versiones que existen.

Conclusiones

Capítulo 2: Análisis de herramientas para la programación para Android

Introducción



Eclipse
 Adobe Flash Builer
 Conclusiones
 Capítulo 3: Propuesta técnica
 Introducción
 Instalación de la herramienta elegida
 Desarrollo de la aplicación.
 Análisis:
 Elaboración de Diagrama de flujos de datos.
 Elaboración de Modelo Entidad Relación.
 Elaboración del Diccionario de Datos.
 Elaboración de Interfaces del Usuario.
 Diseño.
 Elaboración del servicio web.
 Elaboración del sistema de generación de la factura.
 Codificación o construcción
 Pruebas Unitarias
 Pruebas de Interconexión
 Pruebas de Integración
 Implementación del prototipo.
 Conclusiones
 Recomendaciones
 Bibliografía

2.13. Cronograma

Objetivo Especifico	Actividad	Resultado Esperado	Tiempo
Fundamentar teóricamente la propuesta.	Investigar todo lo relacionado sobre el sistema Android.	Estar con todo el conocimiento sobre el sistema para el cual se va a desarrollar.	1 semana.
Investigar las mejores alternativas para el desarrollo de la aplicación.	Buscar toda tipo de información sobre las herramientas que nos permiten desarrollar este tipo de aplicaciones.	Conocer todo lo que se tiene que saber para el desarrollo de la aplicación.	1 semana.
Desarrollo de la aplicación.	Debemos tomar en cuenta el ciclo de vida de una aplicación: Análisis: - Elaboración de	Tener todo analizado y comprender completamente como fusiona las aplicaciones que vamos a desarrollar. Tonto la	4 semana.



diagrama de flujos de datos. - Elaboración de Entidad Relación. - Elaboración del diccionario de Datos. - Elaboración de Interfaces del Usuario.	comunicación entre base de datos, y aplicaciones.	
Diseño. - Elaboración del servicio web. - Elaboración del sistema de generación de la factura.	Conocer cómo va estar elaboradas las aplicaciones necesarias para la interpretación de los datos.	
Codificación o construcción - Pruebas Unitarias - Pruebas de Interconexión - Pruebas de Integración	Determinar y solucionar problemas que se puedan presentar en el funcionamiento de las aplicaciones.	
Implementación del prototipo.	Facilitar el ingreso detallado de pedidos.	

2.14. Referencias

¿Qué es Android?, <http://www.xatakandroid.com/sistema-operativo/que-es-android>, fecha de consulta, 27/01/15.

Pasos para crear aplicaciones para Android, http://www.pcactual.com/articulo/zona_practica/paso_a_paso/paso_a_paso_software/12544/crea_una_aplicacion_para_los_dispositivos_android.html, fecha de consulta, 27/01/15.

Lenguajes de programación para Andoid, <http://androideity.com/2012/07/16/5-lenguajes-para-programar-en-android/>, fecha de consulta, 27/01/15

Fases para el desarrollo de un programa informático, <http://www.foro.lospillaos.es/fases-para-la-elaboracion-de-un-programa-informatico-vt3968.html>, fecha de consulta, 02/02/2015



UNIVERSIDAD DEL
AZUAY

2.15. Firma de responsabilidad (estudiante)

César Tenesaca Narváz

2.16. Firma de responsabilidad (director sugerido)

Ing. Oswaldo Merchán Manzano

2.17. Fecha de entrega: