

**Universidad Nacional Autónoma de Nicaragua
Recinto Universitario Rubén Darío
Facultad de Ciencias e Ingenierías
Departamento de Computación**



Documento de Seminario para optar al título de Licenciatura en ciencia de la computación.

Tema:

Programación de Aplicaciones para Móviles

Tema delimitado:

Aplicación móvil bajo la plataforma Android para pre-matrícula de estudiantes de Oxford International School, en el año 2014.

Integrantes:

- ❖ Br. Luis Antonio Torres Gago.
- ❖ Br .Scarleth Elieth Rivas Borge.

Tutor:

Lic. Roberto Ezequiel Morales Suárez.



Dedicatoria

Este seminario se lo dedico a Dios primeramente por permitirme llegar hasta este punto de mi vida, brindándome: salud, fuerza, ánimo, sabiduría, además de ser mi guía durante este trayecto.

A mi madre Rosa Matilde Gago por haberme apoyado en todo momento, por sus consejos, sus valores, por su infinita motivación para encaminarme hacer un hombre de bien, pero más que nada su amor.

A mi esposa e hijo quienes han sufrido y permanecido junto conmigo, a los cuales les debo mi máximo esfuerzo, esperanzas para un porvenir óptimo para ellos, y como familia salir adelante.

A nuestros maestros por su gran apoyo, motivación, siendo nuestro ejemplos de culminación de una etapa más de nuestras vidas, por tenernos paciencia y dedicación, en especial al Licenciado Roberto Morales por su comprensión, ofreciendo sus conocimientos y transmitirlo a nosotros para la realización de este trabajo de seminario de graduación.

Br. Luis Antonio Torres Gago.



Dedicatoria

A mi madre por estar conmigo y apoyarme en cada una de las etapas de mi vida, enseñarme que el estudio es la mejor herencia que me puede dejar para salir adelante.

A mis tíos, abuelita, hermano que junto a mi madre me llenaron de consejo, palabras de aliento y fortaleza para llegar a este momento.

A mi esposo que siempre ha estado a lado, dándome su mano, teniendo paciencia, en momento que no pude estar presente.

A mi hija Nataly Silva que es la mayor de mis motivaciones para lograr este sueño para darle calidad de vida, pero sobre todo mucho amor.

A cada una de esas personas que no podre mencionar pero que estuvieron allí dándome la mejor de sus experiencias para culminar esta etapa de mi vida.

Br. Scarleth Rivas Borge



Agradecimientos

A Dios, quien nos dio la vida y nos ha acompañado durante nuestros estudios, brindándonos todo lo necesario hasta el día de hoy.

A nuestras familias que siempre han mostrado su apoyo incondicional.

A nuestro tutor Roberto Morales por resultar ser nuestra ayuda idónea en este seminario.



RESUMEN

El presente trabajo tuvo como objetivo fundamental el desarrollo de una aplicación móvil en Android para realizar inscripción de cursos técnicos, que ayuda a mantener los datos de los alumnos en orden al momento en que se realice la matrícula oficial para estudiantes que llevarán un curso técnico en Oxford international School. El estudio se enmarcó bajo la modalidad de proyecto factible sustentado por una investigación de campo de carácter aplicativo. La población que conforma el estudio estuvo constituida por todas las escuelas del municipio de Managua, y la muestra por el tercer año A del colegio Miguel de Cervantes del municipio de Managua. La necesidad fue presentada por el director del centro Oxford a quien junto con el comité Administrativo de dicho centro se aplicó el instrumento de la encuesta, las respuestas evidencian la necesidad de implementar la App que contribuya a mejorar el proceso de pre matrícula y asegurar los datos al momento de matricularse. El diseño de la aplicación móvil se centró en la captura de los datos en el momento de realizar la pre matrícula.



I.INTRODUCCION

En los últimos años las aplicaciones móviles han adquirido una gran importancia, gracias a la evolución que han tenido los teléfonos desde su aparición hoy en día se cuenta con dispositivos inteligentes o Smartphone que han permitido innovar la manera de comunicarse desde cualquier lugar del mundo ya sea por mensaje de texto video llamada o llamada de voz.

La demanda de estos dispositivos han causado desbordamiento de intereses económicos por parte de sus creadores, tanto así que se han desarrollado sistemas operativos más amigables, seguros y de mejor rendimiento, como el caso del sistema operativo Android, que en la actualidad es uno de los favoritos por ser un software de comercialización gratuita y brindar a los desarrolladores de App un sin número de herramientas. Otro punto no menos importante es que con la evolución de los dispositivos de comunicación específicamente móvil ha llevado al hombre a pensar en nuevas maneras de resolver problemas desde un dispositivo cómodo y portable.

Las App son de gran utilidad e importancia en la actualidad ya que nos vienen en muchas ocasiones a servir de canal para tener comercio así como a navegar y acceder a redes sociales y a realizar búsquedas en general, pero lo más importante es que según el fin nos acercan a quienes son los clientes o usuarios finales.

Android con su market al ser una plataforma open source permitió una mayor libertad, y con esto llegaron Smartphone de bajo costo. Ahora es normal de ver a personas de cualquier edad y estatus utilizando aplicaciones, y es por que poco a poco estos aparatos se están volviendo imprescindibles, pero serían nada sin el abundante y variado ecosistema de aplicaciones que existe para todas las plataformas.

El presente proyecto se enfoca en el desarrollo de una aplicación móvil bajo la plataforma Android, para realizar inscripciones (pre-matricular) de cursos que ofrece Oxford International School para todos los alumnos y personas que deseen aprender alguna de las carreras técnicas que tiene el centro.



II. JUSTIFICACION

Oxford International School nace con el objetivo de ayudar a la ciudadanía ofreciendo carreras técnicas al alcance de las personas con poco recursos económicos, es así que lleva operando por más de 4 años al servicio de la comunidad, cuenta con acreditación de INATEC, y esta está integrado al programa de gobierno llamado “Becas para el pueblo”.

Las becas que otorga son a estudiantes de colegios públicos de Managua y alrededores, las solicitudes de inscripción se hace de manera manual, recorriendo los colegios y haciendo encuestas a los estudiantes para conocer qué carrera le gustaría estudiar si fuera favorecido con dicha beca, el director es quien planifica con fecha los colegios que se visitaran para hacer la selección de becados , posteriormente se le anuncia a los favorecidos que fueron aprobados, se les entrega una hoja el cual contiene los datos necesarios (pre-matricula) y estos se hacen presente en el instituto para realizar la matricula.

La administración del centro ha tomado la decisión de proveer a los encargados de recopilar la información de la pre-matricula una aplicación que conserve los datos de manera íntegra al momento de efectuar la matrícula y desde luego que esté al alcance del centro, pero a que se refieren con “conservar los dato de manera íntegra”, ellos explican que la información que se colecta muchas veces es grande y en el momento de disponer de ella provocan situaciones como las siguientes:

- Retraso posteriores en el proceso de matrícula.
- Gastos de insumos de papelería para la elaboración de las pre-matriculas.
- Falsificaciones de documentos de pre-matriculas.
- Desorden y extravíos de pre-matricula.

Con el desarrollo de una aplicación móvil, para realizar el proceso de pre-matricula se llevara un mejor control de la información levantada, seguridad de los datos recolectados, ahorro en gastos de papelería por lo que la información estará almacenada en una base de datos, mejor atención a los estudiantes al momento de realizar su matrícula por que no habrá redundancia de solicitud de información o incluso pérdida de tiempo por extravío de documentos requerido.



III.OBJETIVO

3.1 Objetivo general

Desarrollar una aplicación móvil bajo la plataforma Android para inscripción de pre matrícula de estudiantes de Oxford International School, año 2014.

3.2 Objetivos específicos

- Analizar las diferentes Tecnologías y herramientas utilizadas en el diseño y desarrollo de aplicaciones móviles.
- Investigar el proceso que se realiza al momento de pre-matricular a un alumno para el instituto Oxford.
- Diseñar las interfaces correspondientes para la pre-matricula de alumnos del instituto.
- Programar una aplicación móvil para inscribir alumnos en el instituto Oxford a fin de asegurar los datos que se necesitan al momento de realizar la matricula oficial.



IV. MARCO TEORICO

4.1 Dispositivos móviles

Concepto

Según Wikipedia es un tipo de computadora de tamaño pequeño, con capacidades de procesamiento, con conexión a Internet, con memoria, diseñado específicamente para una función, pero que pueden llevar a cabo otras funciones más generales.

4.1.1 Generación de dispositivos

Primera Generación 1-G

Hizo su aparición en 1979 y se caracterizó por ser analógica y estrictamente para voz, es decir que la transmisión y recepción de datos se apoyaba sobre un conjunto de ondas de radio que cambiaban de modo continuo. La calidad de los enlaces era muy baja, tenían poca velocidad (2400 bauds) y en cuanto a la transferencia era muy imprecisa.

El hecho de que fueran analógicos traía consigo una serie de inconvenientes, su seguridad nula permitía escuchar llamadas ajenas con un simple sintonizador de radio o, incluso hacer uso de las frecuencias cargando el importe de las llamadas a otras personas.

Los estándares más utilizados se mencionan a continuación y se definen según el Glosario técnico de Telecom.

* **NMT (Nordic Mobile Telephone).**

Es un sistema de telefonía móvil que se creó en 1981 como una respuesta a la creciente congestión y los requisitos pesados de la red de telefonía móvil ARP. Se basa en la tecnología analógica (la primera generación o 1G) y dos variantes existentes: NMT 450 y NMT 900. Los números indican las bandas de frecuencia utilizadas. NMT 900 se introdujo en 1986, ya que tiene más canales que la anterior red NMT 450.

* **AMPS (Advanced Mobile Phone System)**

Es un sistema analógico que utiliza diferentes portadoras de frecuencia para crear canales de comunicaciones en una técnica conocida como acceso múltiple por división de frecuencia (FDMA). El AMPS sigue utilizándose ampliamente y constituye la base de cierto número de otras normas de radiocomunicaciones celulares tales como TACS (Total Access Communications System) y D-AMPS (Digital Advanced Mobile Phone System).



Segunda Generación 2-G

Arribo hasta 1990 y a diferencia de la primera se definió por ser digital, permitiendo, mediante la introducción de una serie de protocolos, la mejora del manejo de llamadas, más enlaces simultáneos en el mismo ancho de banda y la integración de otros servicios adicionales al de la voz, de entre los que destaca el Servicio de Mensajes Cortos (Short MessageService).

Estos protocolos fueron implementados por diversas compañías, siendo este hecho el origen de uno de los principales problemas de esta generación la incompatibilidad entre protocolos, debido a que el radio de utilización del teléfono quedaba limitado al área en el que su compañía le diera soporte.

Estándares más utilizados.

GSM (Global Systemfor Mobile Communications/Sistema Global para Comunicaciones Móviles).

El GSM da soporte a voz, datos y mensajería de texto, y permite la itinerancia entre diferentes redes (lo que significa que los usuarios GSM pueden llevar consigo sus teléfonos a muchas partes del mundo). Los sistemas GSM operan actualmente a 800 MHz, 900 MHz, 1 800 MHz o 1 900 MHz.

CDMA (CodeDivisionMultiple Access/Acceso Múltiple por División de Código)

En el CDMA los canales de comunicaciones se crear asignando un esquema de codificación especial a los flujos de información. Los sistemas de radiocomunicaciones celulares de segunda generación basados en el AMDC se utilizan en partes de Asia y de América del Norte y del Sur.

Generación Dos Punto Cinco 2.5-G

Muchos de los proveedores de servicios de telecomunicaciones, se moderan a las redes 2.5G antes de entrar masivamente a la 3. La tecnología 2.5G es más rápida, y más económica para actualizar a 3G.

La generación 2.5G ofrece características extendidas, ya que cuenta con más capacidades adicionales que los sistemas 2G, como: GPRS (General Packet Radios System), HSCSD (High SpeedyCircuitSwitched), EDGE (Enhanced Data Rates for Global Evolution), IS-136B eIS-95m, etc.



Tercera Generación 3-G

El año 2001 fue un año revolucionario en el ámbito de la telefonía móvil ya que se especializa por contener a la convergencia de voz y datos con accesos inalámbrico a Internet; en otras palabras, es apta para aplicaciones multimedia y altas transmisiones de datos.

Todo este conjunto de nuevos servicios integrados en el terminal junto con un nuevo estándar dieron lugar a la denominada hoy en día "tercera generación de móviles" o móviles 3G, siendo esta la tecnología actual.

Estándar más utilizado:

UMTS (Universal Mobile Telecommunications System /Servicios Universales de Comunicaciones Móviles).

Es un llamado de "Tercera Generación (3G)," de banda ancha , de paquetes basado en la transmisión de texto, voz digitalizada, vídeo y multimedia a velocidades de datos hasta y posiblemente superior a 2 megabits por segundo (Mbps), que ofrece un conjunto consistente de servicios de informática móvil y los usuarios de teléfonos sin importar dónde se encuentren en el mundo.

4.1.2 Tipos de Dispositivos móviles.

La clasificación de los dispositivos móviles ha sido basada según la funcionalidad o servicio principal para la que ha sido diseñado el dispositivo móvil o bien aquel dispositivo del que directamente procede y del que se supone es una versión mejorada.

Los dispositivos móviles pueden ser categorizados en los siguientes grupos (Ortiz Pascual, 2011)

Dispositivos de comunicación

Un dispositivo de comunicación es aquel dispositivo móvil cuyo cometido principal es ofrecer una infraestructura de comunicación, principalmente telefónica. Estos dispositivos ofrecen además servicios como el envío de mensajes SMS y MMS, o acceso WAP. En esta categoría se incluiría el tradicional teléfono móvil, precursor indiscutible dentro de los dispositivos móviles, BlackBerry y Smartphone, que amplía



considerablemente las prestaciones del primero mediante pantalla táctil, conexión a Internet o la ejecución de aplicaciones (por ejemplo, iPhone o HTC G1).

Dispositivo de computación

Los dispositivos de computación son aquellos dispositivos móviles que ofrecen mayores capacidades de procesamiento de datos y cuentan con una pantalla y teclado más cercanos a una computadora de sobremesa. Dentro de este grupo encontramos a las PDA, muy populares a finales de los años 90 y que permitían al usuario disponer de un organizador mucho más completo que los ofrecidos por los teléfonos móviles del momento, e incluso en ocasiones la visualización de documentos o acceso a Internet. Por otro lado, dispositivo de computación también es un ordenador portátil o laptop, que dentro de los dispositivos móviles son sin duda los que mayores prestaciones hardware ofrecen (igualando o superando a los de sobremesa) pero también los que tienen, con diferencia, un mayor tamaño, peso y precio. Las calculadoras gráficas pueden ser igualmente incluidas en este grupo de dispositivos de computación.

Reproductor multimedia

Un reproductor multimedia es aquel dispositivo móvil que ha sido específicamente diseñado para proporcionar al usuario la reproducción de uno o varios formatos de datos digitales, ya sea audio, vídeo o imágenes. Dentro de estos dispositivos encontramos reproductores de MP3, los DVD portátiles, los eBooks, y en los últimos años los reproductores multimedia de la popular familia iPod de Apple, que ofrecen tanto audio y como vídeo. Estos dispositivos son con frecuencia los de más reducido tamaño y, junto a los teléfonos móviles y Smartphone, los más extendidos.

Consola portátil

Una consola portátil es un dispositivo móvil cuya única función es la de proporcionar al usuario una plataforma de juego. Las consolas portátiles fueron, junto a los teléfonos, los primeros dispositivos móviles en convertirse en un producto de masas. Hoy en día representan un importantísimo volumen de ventas dada su gran aceptación en la sociedad y son objeto de auténticas guerras comerciales entre las principales



compañías del sector. Algunos ejemplos de esta categoría son la Nintendo DS de Nintendo, o la PSP de Sony.

PDA (Personal Digital Assistant)

Un PDA (Personal Digital Assistant o Ayudante personal digital) es un dispositivo de pequeño tamaño que combina un ordenador, teléfono/fax, Internet y conexiones de red, véase Figura 1.1. 15

A los PDAs también se les llama palmtops, handheldcomputers (computadoras de mano) y pocketcomputers (computadoras de bolsillo). Un PDA típico puede funcionar como teléfono móvil, fax, explorador de internet, organizador personal, GPS, etc.

Su funcionalidad principal es servir como organizadores, con agenda, calendario, gestión de contactos, y posteriormente han ido creciendo, de forma que actualmente sirven tanto como aparatos en los que leer un libro como en los que encontrarse en un mapa. La línea que los separa de los teléfonos es cada vez más difusa.

La mayoría de PDAs empezaron a usarse con una especie de bolígrafo en lugar de teclado, por lo que incorporaban reconocimiento de escritura a mano. Hoy en día los PDAs pueden tener teclado y/o reconocimiento de escritura. Algunos PDAs pueden incluso reaccionar a la voz, mediante tecnologías de reconocimiento de voz.

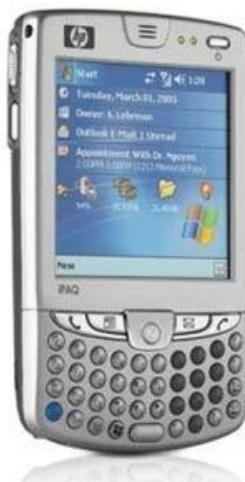


Figura 1. PDA

Fuente:<https://www.apple.com/es/ipod/>



Cámaras Digitales

Las cámaras digitales permiten tomar fotografías, pero almacenándolas en una memoria digital.

En las cámaras digitales, las imágenes, quedan registradas en una placa que posee millones de sensores, la cual traspasa la información, a la memoria. Elemento electrónico, donde se archivan o almacenan las imágenes obtenidas.

En la mayoría de las cámaras digitales, las fotografías se pueden observar de manera instantánea, en el visor de la misma. De aquella manera, se puede decidir, de manera inmediata, si la fotografía es del gusto de uno o no. Ya que de no serlo, se puede borrar de la memoria de manera instantánea.



Fig. 2. Fuente: www.sony.es/electronics/camaras

iPod

El iPod es un reproductor de música portátil y de pequeño tamaño creado por Apple Computer. Ver Figura1.3.

En realidad es un reproductor de audio digital y un disco duro portátil con capacidad de 10GB, 15GB o 30GB. Puede ser conectado con un ordenador a través de un puerto fireware o USB. Los usuarios pueden transferir canciones a su iPod con su ordenador y el software del iPod. Es posible cargar un CD entero en un iPod en tan sólo 10 segundos, y se pueden cargar hasta 10.000 canciones (o a audio libros) en el iPod.

Conocido como extremadamente fácil de usar, se navega en el iPod con una "rueda de tacto" diseñada para ser utilizada con una sola mano.



Además soporta la mayoría de formatos de archivos de audio, incluyendo MP3 y WAV. También incluye una variedad de funciones adicionales, incluyendo juegos, un calendario, un despertador, notas de voz, y notas del texto.



Fig.3 Fuente: <https://www.apple.com/es/ipod/>

🎮 Consola Portátil

En realidad esta categoría debería llamarse “dispositivos orientados a jugar”, porque son más que simples consolas. Los dos ejemplos actualmente en el mercado son la Sony PlayStation Portable (PSP) y la Nintendo DS, que no sólo sirven para jugar, sino que integran algunas de las funcionalidades típicas de una PDA, como reproducción de archivos multimedia, integración con agenda y calendario, o navegador de Internet.



Figura.4

Fuente: <http://www.atisal.com/jxd-s5110-consola-portatil-con-android-40>



Smartphone

Dentro de los dispositivos móviles, un Smartphone (cuya traducción en español sería “teléfono inteligente”) es una evolución del teléfono móvil tradicional que cuenta con ciertas características y prestaciones que lo acercan más a un ordenador personal que a un teléfono tradicional. Entre dichas características, se puede encontrar una mejora en la capacidad de proceso y almacenamiento de datos, conexión a Internet mediante Wi-Fi, pantalla táctil, acelerómetro, posicionado geográfico, teclado QWERTY y diversas aplicaciones de usuario como navegador web, cliente de correo, aplicaciones ofimáticas, reproductores de vídeo y audio, etc. incluyendo la posibilidad de descargar e instalar otras nuevas.

A pesar de estas importantes mejoras con respecto a sus predecesores móviles, el reducido tamaño de los Smartphone conlleva inexorablemente limitaciones de hardware que los mantienen claramente diferenciados de las computadoras convencionales. Estas limitaciones se reflejan principalmente en pantallas más pequeñas, menor capacidad del procesador, restricciones de memoria RAM y memoria persistente, y necesidad de adaptar el consumo de energía a la capacidad de una pequeña batería.

Estas limitaciones obligan a tener muy presente la capacidad real del dispositivo a la hora de desarrollar su software, ya sean aplicaciones de usuario o el propio sistema operativo.



Figura 5

Fuente: <http://www.samsung.com/global/microsite/galaxys5/>



4.2 Sistemas operativos para Móviles

Los dispositivos móviles también tienen sus propios sistemas operativos, aunque son menos famosos que los de PC. Por norma general, antes de la compra de un dispositivo son pocos los usuarios que atienden al sistema operativo que se instala en dicho dispositivo ya sea PDA, Pocket PC, PSP entre otros y mucho menos lo hacen en un teléfono inteligente.

A continuación se detallan los Sistemas Operativos Móviles más utilizados para el desarrollo de aplicaciones para dispositivos móviles

4.2.1 Symbian

Es el resultado de la unión de varias empresas en el mercado tales como Sony Ericsson, Samsung, Motorola, Nokia, Siemens, entre otras. Symbian se caracteriza por hacer

Uso eficiente de todos los recursos de la máquina, especialmente batería, RAM y ROM,

Adaptabilidad al hardware específico y a las pila de telefonía de los fabricantes,

Manejo fiable de los datos incluso en caso de fallo en la comunicación o falta de recursos, como memoria, disco o batería. Entre los servicios genéricos que brinda el sistema operativo, se encuentra una base de datos SQL, seguridad integrada contra malware y virus, y soporte para varias plataformas de desarrollo como C++, Visual Basic para dispositivos móviles y las variantes de Java (PJava, J2ME, etc.). Actualmente la empresa que contiene la mayor parte de sus dispositivos con Symbian es Nokia. (Caffa, A. 2008. pag. 208)

4.2.2 Windows Mobile

Es un SO de la familia Windows desarrollado por Microsoft, los primeros dispositivos móviles diseñados con este sistema operativo datan desde el año 2000. Para ese entonces fue lanzado como Pocket PC 2000 y estaba basado en Windows CE (Caffa, A. pag. 210)

Este sistema operativo está estrechamente vinculado a otros productos de la misma marca (Servicio Live, Office Mobile, Internet Explorer Mobile, etc.) y cuenta con una interfaz gráfica muy buena y similar a la de los sistemas operativos Windows, entre las características que resaltan en este sistema operativo está el tener un kernel unificado capaz de manejar 32000 procesos simultáneos cada uno con 2GB de memoria virtual compartida.



4.2.3 iPhone OS

El IOS es el sistema operativo que utiliza el iPhone y el iPod touch e Ipad. Está basado en una variante de mach kernel que se encuentra en Mac OS X. Este sistema operativo no tenía un nombre oficial hasta que salió la primera versión beta de iPhone SDK, el 6 de marzo de 2008. Antes de esto, Apple declaró, que “el iPhone controla un sistema operativo X, una referencia al padre de los sistemas operativos de los iPhone, el Mac OS X”. Sobre la versión modificada del kernel de Mac os X que corre el iPhone, se encuentran las capas de servicio que componen el teléfono móvil. Existe una gran inclinación en el desarrollo del SO a la interfaz de usuario y las cuestiones de usabilidad, con un modo de manejo revolucionario basado en su pantalla táctil e implementado mediante el frameworkcocoaTouch desarrollado por Apple.

4.2.4 Android

El sistema operativo Android ha expandido su mercado no solamente a Smartphone, sino también a Tablet y otro tipo de dispositivos, la evolución y reducción de costo de los Smartphone ha permitido que millones de personas tenga acceso a ese tipo de tecnología permitiendo llevar mucha información consigo, anteriormente esto era solo posible solo con laptop, pero hasta el día de hoy los celulares son las computadoras portátiles de nuestra era, millones de teléfono se activan diariamente y con el uso de googleplay anteriormente conocido como Androidmarket para la distribución de aplicaciones los desarrolladores pueden poner de inmediato sus aplicaciones a los usuarios ya sea aplicaciones o juegos creados en esta plataforma .



4.3 Sistema operativo Android

¿Qué es Android? Es un sistema operativo basado en Linux creado por google y utiliza como base el lenguaje de programación java para el desarrollo de aplicaciones, proyectándose así que sea el SO por default de muchos otros dispositivos, por ende el desarrollo de aplicaciones para de este SO es un mercado en crecimiento. Android fue creado con la finalidad de contar con una plataforma estándar, opens urce, robusta, que puedan ser ejecutadas en cualquier dispositivo móvil sin tener que volver a programar para un fabricante en específico, dones que brinda la máquina virtual de java.

La arquitectura de Android se divide en varias capas que hace uso de una máquina virtual llamada Dalvik con ello es posible abstraer el detalle del hardware y así desarrollar una vez la aplicación y ejecutarla en cualquier dispositivo que tenga una máquina virtual compatible, Android permite explorar los recursos de los dispositivos móviles a diferencia de otros SO móviles.

Debido a que la apps para pre-matricula correrá bajo este sistema operativo abordaremos un poco más a fondo cómo trabaja Android y el porqué de su elección para desarrollar la aplicación bajo este entorno.

Manuel Bejarano en su artículo tecnológico, cita que en los últimos años en Nicaragua los Consumidores han dejado de comprar computadoras de escritorios y portátiles, para sumarse a la tecnología de las tabletas y los teléfonos inteligentes [2],

El 33.90% de los usuarios Nicaraguenses que acceden a Facebook lo hacen por medio del sistema operativo Android, según el sitio web Owloo, y en segundo lugar, con el

8.98%, usan el sistema iOS, de Apple. En menor medida usan BlackBerry (4.92%) y Windows, de Nokia (2.71%).

Un restante 49.49% correspondería a usuarios que se conectan por medio de computadoras de escritorio y portátiles, explicó Julio Osorno, gerente comercial de Comtech.





4.3.1 Arquitectura Android

En las siguientes líneas se dará una visión global por capas de cuál es la arquitectura empleada en Android. Cada una de estas capas utiliza servicios ofrecidos por las anteriores, y ofrece a su vez los suyos propios a las capas de niveles superiores, tal como muestra la siguiente figura:

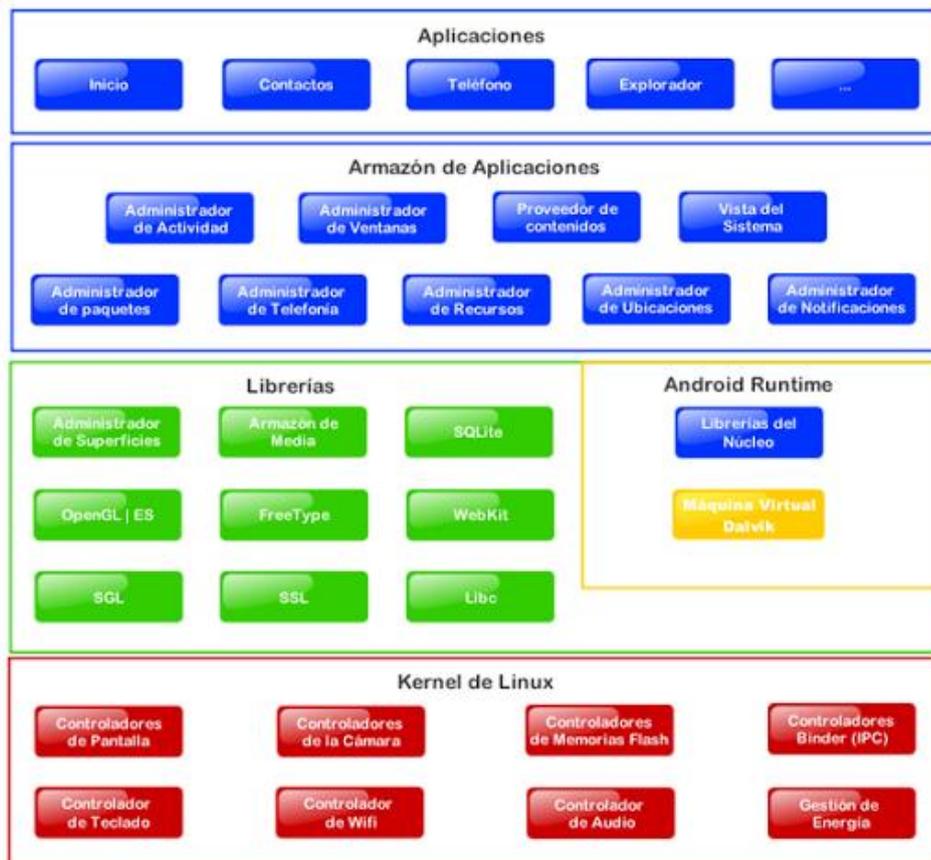


Figura 10.

Fuente: <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>

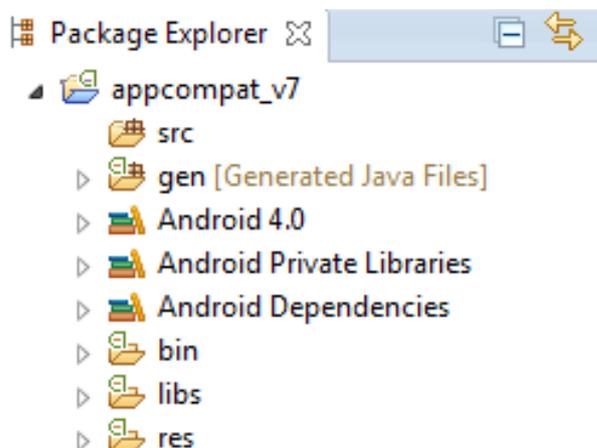
- **Aplicaciones:** Este nivel contiene, tanto las incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores.
- **Framework** de Aplicaciones: Representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de API y el mismo "framework", representado por este nivel.



- **Librerías:** La siguiente capa se corresponde con las librerías utilizadas por Android. Éstas han sido escritas utilizando C/C++ y proporcionan a Android la mayor parte de sus capacidades más características. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android.
- **Tiempo de ejecución de Android:** Al mismo nivel que las librerías de Android se sitúa el entorno de ejecución. Éste lo constituyen las CoreLibraries, que son librerías con multitud de clases Java y la máquina virtual Dalvik.
- **Núcleo Linux:** Android utiliza el **núcleo de Linux 2.6** como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente, hardware pueda ser utilizado mediante las llamadas correspondientes. Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerías de control o drivers necesarios dentro de este kernel de Linux embebido en el propio Android.

4.3.4 Estructura de un proyecto en Android

Para entender cómo se crea una aplicación en Android exploraremos la estructura general de un proyecto en Android con eclipse, al crear un nuevo proyecto en eclipse se genera automáticamente un árbol de carpetas, necesarias para el desarrollo de una aplicación.



Cada una de ellas tiene una tarea definida dentro de la estructura las cuales se describirán a continuación



Carpeta/res/: Contiene todos los ficheros de recursos necesarios para el proyecto: imágenes, videos, cadenas de texto, etc.

Carpeta/Gen: Contiene una serie de elementos de códigos generados, automáticamente al compilar el proyecto. Cada vez que generamos nuestro proyecto, la maquinaria de compilación de Android genera por nosotros una serie de ficheros fuentes en java dirigido al control de los recursos de la aplicación.

Carpeta/assets: Contiene todos los ficheros auxiliares necesarios para la aplicación (y que se incluirán en su propio paquete), como por ejemplo ficheros de configuración de datos, etc.[4].

Fichero AndroidManifest.xml: Contiene la definición en XML de los aspectos principales de la aplicación como por ejemplo su identificación (nombre, versión, icono, etc.), sus componentes (pantallas, mensajes), o los permisos necesarios para su ejecución.[5].

4.4 Programación de aplicaciones

¿Qué es la programación?

La programación es el proceso de diseñar, codificar, depurar y mantener el código fuente de programas computacionales. El código fuente es escrito en un lenguaje de programación. El propósito de la programación es crear programas que exhiban un comportamiento deseado. El proceso de escribir código requiere frecuentemente conocimientos en varias áreas distintas, además del dominio del lenguaje a utilizar, algoritmos especializados y lógica formal. Programar no involucra necesariamente otras tareas tales como el análisis y diseño de la aplicación (pero sí el diseño del código), aunque sí suelen estar fusionadas en el desarrollo de pequeñas aplicaciones. (profesorjavierguerrero.blogspot.com).

Breve historia

La telefonía móvil usa onda de radios para poder ejecutar todas y cada una de las operaciones ,realizar una llamada, mandar un mensaje de texto .La comunicación inalámbrica tiene sus raíces en la invención del radio por Nikola Tesla en los años 1880,aunque formalmente presentado en 1894 por un joven Italiano llamado Guglielmo Marconi .(Fernández & Fernández, 2004)



El teléfono se remonta a los inicio de la segunda guerra mundial, donde ya se veía que era necesaria la comunicación a distancia, por lo cual la compañía Motorola creó un quipo llamado HandieTalkie H12-16, que es un equipo que permitió el contacto con las tropas vía ondas de radio que en ese tiempo no superaban más de 600 khz.

En cuestión de tiempo se juntó la tecnología Tesla y Marconise y dieran a luz la comunicación mediante radio teléfonos-.Martin Cooper, considerado como el padre de la telefonía celular, al introducir el primer radio teléfono, en 1973, en Estados unidos, mientras trabajaba para Motorola.

En 1979 aparecieron los primeros sistemas a la venta en Tokio (Japón),fabricados por la compañía NTT.Los países Europeo no se quedan atrás y en 1981 se introdujo en Escandinavia un sistema similar a AMPS(advancedmobilephonessystem).En estados unidos fue hasta en 1983 que se puso en operación el primer sistema comercial en la ciudad de Chicago.

Así comenzó el desarrollo de una de la tecnología cuya aceptación fue la necesidad de desarrollar e implementar otras formas de comunicación así como transformar los sistemas analógicos y digitales dar mayor capacidad de usuarios.

4.4.1 lenguajes de programación

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente. Aunque muchas veces se usa lenguaje de programación y lenguaje informático como si fuesen sinónimos, no tiene por qué ser así, ya que los lenguajes informáticos engloban a los lenguajes de programación y a otros más, como, por ejemplo, el HTML. (jorgesaavedra.wordpress.com).

4.4.2 Eclipse

La plataforma de eclipse consiste en entorno de desarrollo integrado (IDE) abierto y extensible Eclipse sirve como IDE de Java y cuenta con numerosas herramientas de desarrollo de software. También da soporte otros lenguajes de programación como C, C++, Cobol, Fortran, etc. A la plataforma de eclipse se le pueden añadir pluggins (extensiones) para ampliar la funcionalidad.



Breve Historia

Eclipse fue realizada por IBM antes, de que surgiera el proyecto eclipse como tal. El antecesor de eclipse fue VisualAge y se construyó usando SmallTalk en un entorno de desarrollo llamado Envy, con la aparición de Java en la década de los 90, IBM desarrollo una máquina virtual valida tanto para SmallTalk y Java. La rápida expansión de Java y su ventaja con mira a un internet en plena expansión obliga a IBM a plantear el abandono de esta máquina virtual dual y la construcción de una nueva plataforma basada en Java desde el principio. El producto final fue eclipse el cual ya había tenido un coste de 40 millones de dólares a IBM en el año 2001[3].

Instalación de Android y eclipse

Lo primero que se debe hacer es descargar el archivo del correspondiente al SDK y ejecutar el SDK manager hay sistemas operativos como Windows que necesita que se configure una variable de entorno para que pueda funcionar apropiadamente denominado ANDROID_SDK_HOME, cuyo contenido ha de ser la ruta completa al directorio de instalación del SDK.

La instalación de eclipse es muy sencilla solo consiste en descomprimir el contenido del archivo Zip que contiene la aplicación y librerías en el directorio de preferencia aunque es recomendable que sea en C.

4.4.3Lenguaje de programación JAVA

Java es un lenguaje de programación originalmente desarrollado por James Gosling de Sun (adquirida por Oracle) y publicado en 1995 como un potente fundamental de la plataforma Java de Sun. El lenguaje deriva mucha sintaxis de C y C++, pero tiene menos facilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son generalmente compiladas en bytecodes que pueden correr en cualquier máquina virtual Java sin importar la arquitectura de la computadora, es un lenguaje de programación de propósito general concurrente, basado en clase y orientado a objetos.

Características y propiedades

Escribir software en una plataforma y ejecutarla virtualmente en otra.

Crear programas que se puedan ejecutar en un explorador y acceder a servicios Web disponibles

Desarrollar aplicaciones de servidor para foros en líneas, almacenes, encuestas, etc.



Combinar aplicaciones con servicios que utilizan lenguaje Java para crear aplicaciones o servicios con un gran nivel de personalización.

Escribir aplicaciones potentes y eficaces para teléfonos móviles, procesadores remotos, productos de consumo, y cualquier otro dispositivo electrónico.

Java para Android

Porque se utiliza java para Android.

- **Es el lenguaje oficial:** Google decidió que java era lo mejor para su nuevo sistema operativo.
- **Las mejores y más completas herramientas.:** Primero Eclipse ADT y ahora Android Studio son herramientas basadas en sendos y populares IDEs Java que tienen muchos años en el mercado, si quieres herramientas maduras, poderosas y que te hagan fácil la vida desarrollando en Android.
- **Miles de miles de librerías disponibles.:** Si quieres usar JSON, parsear XML o HTML, tener un ORM, conexión a base de datos, etc. La fauna y flora en Java es variada y extensa, encontraras librerías muy maduras y testeadas hasta la saciedad para casi cualquier cosa que quieras hacer. Y todas esas librerías las puedes usar en una aplicación Android.

4.5 Aplicación móvil.

Una aplicación móvil o *app* es una aplicación informática diseñada para ser ejecutada en teléfonos inteligentes, tabletas y otros dispositivos móviles. Por lo general se encuentran disponibles a través de plataformas de distribución, operadas por las compañías propietarias de los sistemas operativos; móviles como Android, iOS, BlackBerry OS, Windows Phone, entre otros.

4.5.1 Componentes de una aplicación en Android

Las aplicaciones Android son programas creados para funcionar en teléfonos móviles de última generación (Smartphone) con sistema operativo Android, una plataforma basada en Linux y de software libre, esto significa que cualquier programador Android puede acceder a su código y puede desarrollar aplicaciones para Android de forma gratuita.

Su enorme difusión vino de su compra por Google, que decidió extender su negocio a la telefonía móvil alcanzando un acuerdo con distintos fabricantes de terminales



móviles (HTC, Samsung, LG, etc.) para que desarrollaran modelos basados en Android.

Paralelamente se creó el AndroidMarket, una tienda virtual en Internet en la que se pueden descargar aplicaciones gratuitas y de pago. Las aplicaciones Android se aprovechan de las principales ventajas de los nuevos terminales (pantalla táctil, conexión de datos, geolocalización) para ofrecer funcionalidades que conectan de forma directa con los servicios de Google en Internet y con el mundo on-line en general, resultando una alternativa muy útil para empresas.

La alianza entre Android y empresa es provechosa por su concepto de software libre, que reduce los costes y asegura la calidad de las aplicaciones para Android desarrolladas.

Las aplicaciones en Android se estructuran en componentes, cada componente de una aplicación juega un papel específico dentro de esta, a la hora de desarrollar un aplicación para Android no se tiene una sola función principal, sino que se tendrán varios componentes independientes cuya comunicación será llevada a través del sistema operativo [3].

La manera en que funciona una aplicación en Android es de forma asíncrona, es decir que es el sistema operativo que ante peticiones del usuario van llamando a una y otra función de los diferentes componentes de la aplicación según sea convenga.

A través de las API de Android podemos declarar podemos declarar cinco tipos de componentes

- **Actividad:** Son los componentes más importantes. Una actividades una pantalla de la aplicación, con su interfaz de usuario, tiene un ciclo de vida muy definido, que será igual para todas las actividades. Este ciclo de vida es impuesto por el SDK de Android. Las actividades tienen cuatro posibles estados: Activa, pausada, parada y reiniciada.

- **Intent:** Un Intent consiste básicamente en la voluntad de realizar alguna acción, generalmente asociada a unos datos. Lanzando un Intent, una aplicación puede delegar el trabajo en otra, de forma que el sistema se encarga de buscar qué aplicación de entre las instaladas, es la que puede llevar a cabo la acción solicitada. Por ejemplo, abrir una URL en algún navegador web, o escribir un correo electrónico desde algún cliente de correo. Los Intents están incluidos en el AndroidManifest porque describen dónde y cuándo puede comenzar una actividad. Cuando una actividad crea un Intent, éste puede tener descriptores de lo que se quiere hacer. Una vez se está ejecutando la aplicación, Android compara esta información del Intent con los Intents de cada aplicación, eligiendo el más adecuado para realizar la operación especificada por el llamante.



- **BroadcastIntent Receiver:** Un componente BroadcastIntent Receiver se utiliza para lanzar alguna ejecución dentro de la aplicación actual cuando un determinado evento se produzca (generalmente, abrir un componente Activity). Por ejemplo, una llamada entrante o un SMS recibido

- **Servicio:** Un componente Service representa una aplicación ejecutada sin interfaz de usuario, y que generalmente tiene lugar en segundo plano mientras otras aplicaciones (éstas con interfaz) son las que están activas en la pantalla del dispositivo. Un ejemplo típico de este componente es un reproductor de música. La interfaz del reproductor muestra al usuario las distintas canciones disponibles, así como los típicos botones de reproducción, pausa, volumen, etc. En el momento en el que el usuario reproduce una canción, ésta se escucha mientras se siguen visualizando todas las acciones anteriores, e incluso puede ejecutar una aplicación distinta sin que la música deje de sonar. La interfaz de usuario del reproductor sería un componente Activity, pero la música en reproducción sería un
 - componente Service, porque se ejecuta en background. Este elemento está implementado por la clase de mismo nombre Service.

- **Content Provider:** Con el componente que permite almacenar una información para que después pueda ser accedida desde varias aplicaciones, cualquier aplicación en Android puede almacenar datos en un fichero, en una base de datos SQLite o en cualquier otro formato que considere. Además, estos datos pueden ser compartidos entre distintas aplicaciones.

4.6 Análisis de Sistemas

Concepto

Es un conjunto o disposición de procedimientos o programas relacionados de manera que juntos forman una sola unidad. Un conjunto de hechos, principios y reglas clasificadas y dispuestas de manera ordenada mostrando un plan lógico en la unión de las partes. Un método, plan o procedimiento de clasificación para hacer algo. También es un conjunto o arreglo de elementos para realizar un objetivo predefinido en el procesamiento de la Información. Esto se lleva a cabo teniendo en cuenta ciertos principios:



- Debe presentarse y entenderse el dominio de la información de un problema.
- Defina las funciones que debe realizar el Software.
- Represente el comportamiento del software a consecuencias de acontecimientos externos.
- Divida en forma jerárquica los modelos que representan la información, funciones y comportamiento.

Proceso de Análisis de sistema

El proceso debe partir desde la información esencial hasta el detalle de la Implementación.

La función del Análisis puede ser dar soporte a las actividades de un negocio, o desarrollar un producto que pueda venderse para generar beneficios. Para conseguir este objetivo, un Sistema basado en computadoras hace uso de seis (6) elementos fundamentales:

- Software, que son Programas de computadora, con estructuras de datos y su documentación que hacen efectiva la logística metodología o controles de requerimientos del Programa.
- Hardware, dispositivos electrónicos y electromecánicos, que proporcionan capacidad de cálculos y funciones rápidas, exactas y efectivas (Computadoras, Censores, maquinarias, bombas, lectores, etc.), que proporcionan una función externa dentro de los Sistemas.
- Personal, son los operadores o usuarios directos de las herramientas del Sistema.
- Base de Datos, una gran colección de informaciones organizadas y enlazadas al Sistema a las que se accede por medio del Software.
- Documentación, Manuales, formularios, y otra información descriptiva que detalla o da instrucciones sobre el empleo y operación del Programa.
- Procedimientos, o pasos que definen el uso específico de cada uno de los elementos o componentes del Sistema y las reglas de su manejo y mantenimiento.

Un Análisis de Sistema se lleva a cabo teniendo en cuenta los siguientes objetivos en mente:

- Identifique las necesidades del Cliente.
- Evalúe que conceptos tiene el cliente del sistema para establecer su viabilidad.
- Realice un Análisis Técnico y económico.
- Asigne funciones al Hardware, Software, personal, base de datos, y otros elementos del Sistema.
- Establezca las restricciones de presupuestos y planificación temporal.
- Cree una definición del sistema que forme el fundamento de todo el trabajo de Ingeniería.



Para lograr estos objetivos se requiere tener un gran conocimiento y dominio del Hardware y el Software, así como de la Ingeniería humana (Manejo y Administración de personal), y administración de base de datos.

Objetivos del Análisis

Identificación de necesidades, es el primer paso del análisis del sistema, en este proceso en Analista se reúne con el cliente y/o usuario (un representante institucional, departamental o cliente particular), e identifican las metas globales, se analizan las perspectivas del cliente, sus necesidades y requerimientos, sobre la planificación temporal y presupuestal, líneas de mercadeo y otros puntos que puedan ayudar a la identificación y desarrollo del proyecto.

Algunos autores suelen llamar a esta parte " *Análisis de Requisitos* " y lo dividen en cinco partes:

- *Reconocimiento del problema.*
- *Evaluación y Síntesis.*
- *Modelado.*
- *Especificación.*
- *Revisión.*

Antes de su reunión con el analista, el cliente prepara un documento conceptual del proyecto, aunque es recomendable que este se elabore durante la comunicación Cliente – analista, ya que de hacerlo el cliente solo de todas maneras tendría que ser modificado, durante la identificación de las necesidades.

Análisis Económico y Técnico

El análisis económico incluye lo que llamamos, el análisis de costos – beneficios, significa una valoración de la inversión económica comparado con los beneficios que se obtendrán en la comercialización y utilidad del producto o sistema.

Muchas veces en el desarrollo de Sistemas de Computación estos son intangibles y resulta un poco dificultoso evaluarlo, esto varía de acuerdo a las características del Sistema. El análisis de costos - beneficios es una fase muy importante de ella depende la posibilidad de desarrollo del Proyecto.

En el Análisis Técnico, el Analista evalúa los principios técnicos del Sistema y al mismo tiempo recoge información adicional sobre el rendimiento, fiabilidad, características de mantenimiento y productividad.

Los resultados obtenidos del análisis técnico son la base para determinar sobre si continuar o abandonar el proyecto, si hay riesgos de que no funcione, no tenga el rendimiento deseado, o si las piezas no encajan perfectamente unas con otras.



4.7 Sistema de Información

Concepto

Por definición es un conjunto de elementos que interactúan entre sí con un fin común; que permite que la información esté disponible para satisfacer las necesidades en una organización, un sistema de información no siempre requiere contar con recuso computacional aunque la disposición del mismo facilita el manejo e interpretación de la información por los usuarios. (Kendall, 2005)

Los elementos que interactúan entre sí son: el equipo computacional (cuando esté disponible), el recurso humano, los datos o información fuente, programas ejecutados por las computadoras, las telecomunicaciones y los procedimientos de políticas y reglas de operación.

Un Sistema de Información realiza cuatro actividades básicas:

- Entrada de información: proceso en el cual el sistema toma los datos que requiere.
- Almacenamiento de información: puede hacerse por computadora o archivos físicos para conservar la información.
- Procesamiento de la información: permite la transformación de los datos fuente en información que puede ser utilizada para la toma de decisiones
- Salida de información: es la capacidad del sistema para producir la información procesada o sacar los datos de entrada al exterior.

Los usuarios de los sistemas de información tienen diferente grado de participación dentro de un sistema y son el elemento principal que lo integra, así se puede definir usuarios primarios quienes alimentan el sistema, usuarios indirectos que se benefician de los resultados pero que no interactúan con el sistema, usuarios gerenciales y directivos quienes tienen responsabilidad administrativa y de toma de decisiones con base a la información que produce el sistema. (Incap.init).

4.7.1 Tipos de sistema de información

Los sistemas de información son desarrollados con propósitos diferentes dependiendo de las necesidades del negocio o institución.

- **Sistema de Procesamiento de Transacciones (TPS)**

Un sistema de procesamiento de transacciones (TPS) es un tipo de sistema de información que recolecta, almacena, modifica y recupera toda la información generada por las transacciones producidas en una organización. Una transacción es un evento que genera o modifica los datos que se encuentran eventualmente almacenados en un sistema de información. Para que un sistema informático pueda ser considerado como un TPS, este debe superar el test ACID.

Desde un punto de vista técnico, un TPS monitoriza los programas transaccionales (un tipo especial de programas). La base de un programa transaccional está en que gestiona los datos de forma que estos deben ser siempre consistentes (por ejemplo, si se realiza un pago con una tarjeta electrónica, la cantidad de dinero de la cuenta sobre



la que realiza el cargo debe disminuir en la misma cantidad que la cuenta que recibe el pago, de no ser así, ninguna de las dos cuentas se modificará), si durante el transcurso de una transacción ocurriese algún error, el TPS debe poder deshacer las operaciones realizadas hasta ese instante. (Kendall, 2005)

- **Sistema de Manejo de Conocimiento (KWS)**

Dan soporte a los trabajadores profesionales, tales como científicos, ingenieros y doctores, les ayudan a crear un nuevo conocimiento que contribuya a la organización o a toda la sociedad. (Kendall, 2005)

- **Sistema de Automatización de Oficina (OAS)**

Dan soporte a los trabajadores de datos, quienes, por lo general, no crean un nuevo conocimiento sino que usan la información para analizarla y transformar datos, o para manejarla en alguna forma y luego compartirla o diseminarla formalmente por toda la organización y algunas veces más allá de ella. (Kendall, 2005).

- **Sistema de Apoyo a Decisiones (DSS)**

El sistema de apoyo de decisiones es similar al sistema de información gerencial tradicional en que ambos dependen de una base de datos como fuente. Un sistema de apoyo a decisiones se aparta del sistema de información gerencial tradicional en que enfatiza el apoyo a la toma de decisiones en todas sus fases, aunque la decisión actual todavía es del dominio del tomador de decisiones. (Kendall, 2005)

- **Sistema de Información Gerencial (MIS)**

Los sistemas de información gerencial (MIS) no reemplazan a los sistemas de procesamiento de transacciones, sino que todos los MIS incluyen procesamiento de transacciones. Los MIS son sistemas de información computarizada que trabajan debido a la interacción resuelta entre gentes y computadoras. Requieren que las gentes, el software (programas de computadora) y el hardware (computadoras, impresoras, etc.) trabajen al unísono. Los sistemas de información dan soporte a un espectro más amplio de tareas organizacionales que los sistemas de procesamiento de transacciones, incluyendo el análisis de decisiones y la toma de decisiones. (Kendall, 2005)

- **Sistema de Apoyo a Decisiones de Grupo (GDSS)**

Cuando los grupos necesitan trabajar juntos para tomar decisiones semi-estructuradas o sin estructura, un sistema de apoyo a decisiones de grupo puede plantear una solución. Los sistemas de apoyo a decisiones de grupo (GDSS) son usados en cuartos



especiales, equipados en varias configuraciones diferentes, que permiten que los miembros del grupo interactúen con apoyo electrónico, frecuentemente en forma de software especializado y con una persona que dé facilidades al grupo. Los sistemas para decisiones de grupo están orientados para reunir a un grupo, a fin de que resuelva un problema con la ayuda de varios apoyos como votaciones, cuestionarios, aportación de ideas y creación de escenarios. (Kendall, 2005)

- **Sistema Expertos e Inteligencia Artificial**

La inteligencia artificial (AI por sus siglas en inglés) puede ser considerada la meta de los sistemas expertos. Los sistemas expertos son un caso muy especial de un sistema de información, cuyo uso ha sido factible para los negocios a partir de la reciente y amplia disponibilidad de hardware y software. Un sistema experto (también llamado un sistema basado en conocimiento) captura en forma efectiva y usa el conocimiento de un experto para resolver un problema particular experimentado en una organización. Observe que a diferencia del DSS, que deja la decisión final al tomador de decisiones, un sistema experto selecciona la mejor solución a un problema o a una clase específica de problemas. (Kendall, 2005)

- **Sistema de Apoyo a Ejecutivo (ESS)**

Cuando los ejecutivos se acercan a la computadora, frecuentemente están buscando formas que les ayuden a tomar decisiones a nivel estratégico. Un sistema de apoyo a ejecutivos (ESS) ayuda a éstos, para organizar sus interacciones con el ambiente externo, proporcionando apoyo de gráficos y comunicaciones en lugares accesibles tales como salas de juntas u oficinas personales corporativas. (Kendall, 2005).

4.8 Ingeniería de Sistemas

Concepto

Se encarga del diseño, la programación, la implantación y el mantenimiento de sistemas. A diferencia de otras ramas de la ingeniería, esta disciplina no se ocupa de productos tangibles (los ingenieros civiles, por ejemplo, construyen edificios), sino de productos lógicos. También, la ingeniería de sistemas implica el uso de nociones matemáticas que permitan concretar la aplicación tecnológica de las teorías de los sistemas. Se trata de una ciencia interdisciplinaria, que requiere de diversos conocimientos para plasmar sus diseños en la vida práctica.

La ingeniería de sistemas permite transformar una necesidad operativa en una descripción de los parámetros del rendimiento de un sistema, con su correspondiente configuración. Por otra parte, posibilita la integración de los parámetros técnicos relacionados de modo tal que las interfaces de programa y funcionales sean compatibles y se garantice el funcionamiento del sistema total.



4.8.1 Modelo de desarrollo de sistemas

La ingeniería de software, con el fin de ordenar el caos que era anteriormente el desarrollo de software, dispone de varios modelos, paradigmas y filosofías de desarrollo, estos los conocemos principalmente como modelos o ciclos de vida del desarrollo de software, esto incluye el proceso que se sigue para construir, entregar y hacer evolucionar el software, desde la concepción de una idea hasta la entrega y el retiro del sistema y representa todas las actividades y artefactos (productos intermedios) necesarios para desarrollar una aplicación, entre ellos se puede citar:

➤ **Modelo en cascada**

En Ingeniería de software el modelo en cascada, también llamado desarrollo en cascada o ciclo de vida clásico, se basa en un enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, esto sugiere una aproximación sistemática secuencial hacia el proceso de desarrollo del software, que se inicia con la especificación de requerimientos del cliente y continúa con la planeación, el modelado, la construcción y el despliegue para culminar en el soporte del software terminado. (Pressman, 1997)

➤ **Desarrollo Incremental**

Desarrollo iterativo y creciente (o incremental) es un proceso de desarrollo de software, creado en respuesta a las debilidades del modelo tradicional de cascada, es decir, este modelo aplica secuencias lineales como el modelo en cascada, pero de una manera iterativa o escalada según como avance el proceso de desarrollo y con cada una de estas secuencias lineales se producen incrementos (mejoras) del software.

Se debe tener en cuenta que el flujo del proceso de cualquier incremento puede incorporar el paradigma de construcción de prototipos, ya que como se mencionó anteriormente, este tipo de modelo es iterativo por naturaleza, más sin embargo se diferencia en que este busca la entrega de un producto operacional con cada incremento que se le realice al software. Este desarrollo incremental es útil principalmente cuando el personal necesario para una implementación completa no está disponible. (Pressman, 1997)

➤ **Modelo de desarrollo rápido de aplicaciones (MDRA)**

El modelo DRA, es un modelo de proceso del desarrollo del software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto (Es una adaptación a alta velocidad del modelo lineal secuencial). El proceso DRA permite al equipo de desarrollo crear un sistema completamente funcional dentro de periodos muy cortos de tiempo.

El DRA comprende las siguientes etapas:

Modelado de Gestión: aquí se modela el flujo de información entre las funciones de gestión. Este flujo debe "responder" a preguntas tales como ¿Qué información



conduce el proceso de gestión?, ¿Quién la genera?, ¿A dónde va la información?, ¿Quién la procesa?

Modelado de datos: se definen las características (atributos) de cada objeto, formado a partir del flujo de información, y las relaciones entre ellos.

Modelado del proceso: las descripciones del proceso se crean para añadir, modificar, suprimir o recuperar un objeto de datos.

Generación de aplicaciones: en lugar de crear software, el RAD reutiliza componentes de programas ya existentes o crea componentes reutilizables.

Prueba y entrega: debido al punto anterior, los componentes ya han sido examinados y probados, lo cual permite que el tiempo de duración de las pruebas sea menor. Todo esto no impide que se tenga que probar cada uno de los nuevos componentes. (Pressman, 1997).

➤ **Construcción de prototipo**

El modelo de prototipos, en Ingeniería de software, pertenece a los modelos de desarrollo evolutivo. Este permite que todo el sistema, o algunos de sus partes, se construyan rápidamente para comprender con facilidad y aclarar ciertos aspectos en los que se aseguren que el desarrollador, el usuario, el cliente estén de acuerdo en lo que se necesita así como también la solución que se propone para dicha necesidad y de esta forma minimizar el riesgo y la incertidumbre en el desarrollo, este modelo se encarga del desarrollo de diseños para que estos sean analizados y prescindir de ellos a medida que se adhieran nuevas especificaciones, es ideal para medir el alcance del producto, pero no se asegura su uso real. Este modelo principalmente se lo aplica cuando un cliente define un conjunto de objetivos generales para el software a desarrollarse sin delimitar detalladamente los requisitos de entrada procesamiento y salida, es decir cuando el responsable no está seguro de la eficacia de un algoritmo, de la adaptabilidad del sistema o de la forma en que interactúa el hombre y la máquina. Este modelo se encarga principalmente de ayudar al ingeniero de sistemas y al cliente a entender de mejor manera cuál será el resultado de la construcción cuando los requisitos estén satisfechos. (Pressman, 1997).

➤ **Modelo en espiral**

El modelo en espiral, que Barry Boehm propuso originalmente en 1986, es un modelo de proceso de software evolutivo que conjuga la naturaleza iterativa de la construcción de prototipos con los aspectos controlados y sistemáticos del modelo en cascada, es decir, cuando se aplica este modelo, el software se desarrolla en una serie de entregas evolutivas (s ciclos o iteraciones), cada una de estas entregando prototipos más completas que el anterior, todo esto en función del análisis de riesgo y las



necesidades del cliente. Aunque el modelo espiral representa ventajas por sobre el desarrollo lineal, el cálculo de los riesgos puede ser muy complicado y por lo cual su uso en el ámbito real es muy escaso. (Pressman, 1997).

➤ **Modelo de desarrollo por etapas**

Es un modelo en el que el software se muestra al cliente en etapas refinadas sucesivamente. Con esta metodología se desarrollan las capacidades más importantes reduciendo el tiempo necesario para la construcción de un producto; el modelo de entrega por etapas es útil para el desarrollo de la herramienta debido a que su uso se recomienda para problemas que pueden ser tratados descomponiéndolos en problemas más pequeños y se caracteriza principalmente en que las especificaciones no son conocidas en detalle al inicio del proyecto y por tanto se van desarrollando simultáneamente con las diferentes versiones del código.

En este modelo Pueden distinguirse las siguientes fases:

- Especificación conceptual.
- Análisis de requisitos.
- Diseño inicial.
- Diseño detallado (codificación, depuración, prueba y liberación).

Cuando es por etapas, en el diseño global estas fases pueden repetirse según la cantidad de etapas que sean requeridas.

Entre sus ventajas tenemos:

- Detección de problemas antes y no hasta la única entrega final del proyecto.
- Eliminación del tiempo en informes debido a que cada versión es un avance.
- Estimación de tiempo por versión, evitando errores en la estimación del proyecto general.
- Cumplimiento a la fecha por los desarrolladores. (Monografias.com)

➤ **Desarrollo basado en componentes**

En esencia, un componente es una pieza de código pre elaborado que encapsula alguna funcionalidad expuesta a través de interfaces estándar . Los componentes son los "ingredientes de las aplicaciones", que se juntan y combinan para llevar a cabo una tarea. Es algo muy similar a lo que podemos observar en el equipo de música que tenemos en nuestra sala. Cada componente de aquel aparato ha sido diseñado para acoplarse perfectamente con sus pares, las conexiones son estándar y el protocolo de comunicación está ya preestablecido. Al unirse las partes, obtenemos música para nuestros oídos.

El paradigma de ensamblar componentes y escribir código para hacer que estos componentes funcionen se conoce como Desarrollo de Software Basado en Componentes. El uso de este paradigma posee algunas ventajas:



1. **Reutilización del software.** Nos lleva a alcanzar un mayor nivel de reutilización de software.
2. **Simplifica las pruebas.** Permite que las pruebas sean ejecutadas probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados.
3. **Simplifica el mantenimiento del sistema.** Cuando existe un débil acoplamiento entre componentes, el desarrollador es libre de actualizar y/o agregar componentes según sea necesario, sin afectar otras partes del sistema.

Mayor calidad. Dado que un componente puede ser construido y luego mejorado continuamente por un experto u organización, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo. (Monografias.com)

➤ **Modelo de Métodos formales**

En ingeniería de software un método formal es un camino a la construcción y análisis de modelos matemáticos que permitan una automatización del desarrollo de sistemas informáticos. Los métodos formales se caracterizan por emplear técnicas y herramientas matemáticas para lograr una facilitación a la hora de encarar la construcción o el análisis de un modelo matemático de un sistema. (Monografias.com)

➤ **Modelo Estructurado**

Este modelo como su nombre lo indica utiliza las técnicas del Diseño estructurado o de la Programación estructurada para su desarrollo, también se utiliza en la creación de los algoritmos del programa. Este formato facilita la comprensión de la estructura de datos y su control. Entre las principales características de este modelo se encuentran las siguientes:

- Generalmente se puede diferenciar de una forma más clara los procesos y las estructuras de datos.
- Existen métodos que se enfocan principalmente en ciertos datos.
- La abstracción del programa es de un nivel mucho mayor.
- Los procesos y estructuras de datos son representados jerárquicamente.

Este modelo también presenta sus desventajas entre las cuales podemos mencionar algunas:

- Se podría encontrar datos repetidos en diferentes partes del programa.



- Cuando el código se hace muy extenso o grande su manejo se complica demasiado.

En el modelo estructurado las técnicas que comúnmente se utilizan son:

- El Modelo entidad-relación, esta técnica se relaciona principalmente con los datos.
- El Diagrama de Flujo de Datos, esta es utilizada principalmente para los procesos. (Monografias.com)

➤ **Modelo Orientado a Objeto**

Estos modelos tienen sus raíces en la Programación orientada a objetos y como consecuencia de ella gira en torno al concepto de clase, también lo hacen el análisis de requisitos y el diseño. Esto además de introducir nuevas técnicas, también aprovecha las técnicas y conceptos del desarrollo estructurado, como diagramas de estado y transiciones. El modelo orientado a objetos tiene dos características principales, las cuales ha favorecido su expansión:

- Permite la reutilización de software en un grado significativo.
- Su simplicidad facilita el desarrollo de herramientas informáticas de ayuda al desarrollo, el cual es fácilmente implementada en una notación orientada a objetos llamado UML. (Monografias.com)

4.9 Base de datos

Concepto

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. (Wikipedia, [documento en línea]).

Otro concepto brindado por Marvin Garay en su artículo Que son las bases de datos sería una base de datos es un “almacén” que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente. (Maestro del web [Artículo en línea])

Sistema Gestor de bases de datos: Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones:



- Definición de los datos
- Mantenimiento de la integridad de los datos dentro de la base de datos
- Control de la seguridad y privacidad de los datos
- Manipulación de los datos

Un sistema gestor de base de datos está compuesto de:

El gestor de la base de datos

Se trata de un conjunto de programas no visibles al usuario final que se encargan de la privacidad, la integridad, la seguridad de los datos y la interacción con el sistema operativo. Proporciona una interfaz entre los datos, los programas que los manejan y los usuarios finales.

Cualquier operación que el usuario hace contra la base de datos está controlada por el gestor.

El gestor almacena una descripción de datos en lo que llamamos diccionario de datos, así como los usuarios permitidos y los permisos.

Tiene que haber un usuario administrador encargado de centralizar todas estas tareas.

Diccionario de datos

Es una base de datos donde se guardan todas las propiedades de la base de datos, descripción de la estructura, relaciones entre los datos, etc.

El diccionario debe contener:

- La descripción externa, conceptual e interna de la base de datos
- Las restricciones sobre los datos
- El acceso a los datos
- Las descripciones de las cuentas de usuario
- Los permisos de los usuarios
- Los esquemas externos de cada programa

Los lenguajes

Un sistema gestor de base de datos debe proporcionar una serie de lenguajes para la definición y manipulación de la base de datos. Estos lenguajes son los siguientes:

- Lenguaje de definición de datos (DDL). Para definir los esquemas de la base de datos
- Lenguaje de manipulación de datos (DML). Para manipular los datos de la base de datos



Lenguaje de control de datos (DCL). Para la administración de usuarios y seguridad en la base de datos.

4.10 Proceso de Normalización

La normalización es el proceso de organizar los datos de una base de datos. Se incluye la creación de tablas y el establecimiento de relaciones entre ellas según reglas diseñadas tanto para proteger los datos como para hacer que la base de datos sea más flexible al eliminar la redundancia y las dependencias incoherentes.

Los datos redundantes desperdician el espacio de disco y crean problemas de mantenimiento. Si hay que cambiar datos que existen en más de un lugar, se deben cambiar de la misma forma exactamente en todas sus ubicaciones. Un cambio en la dirección de un cliente es mucho más fácil de implementar si los datos sólo se almacenan en la tabla Clientes y no en algún otro lugar de la base de datos.

¿Qué es una "dependencia incoherente"? Aunque es intuitivo para un usuario mirar en la tabla Clientes para buscar la dirección de un cliente en particular, puede no tener sentido mirar allí el salario del empleado que llama a ese cliente. El salario del empleado está relacionado con el empleado, o depende de él, y por lo tanto se debería pasar a la tabla Empleados. Las dependencias incoherentes pueden dificultar el acceso porque la ruta para encontrar los datos puede no estar o estar interrumpida.

Hay algunas reglas en la normalización de una base de datos. Cada regla se denomina una "forma normal". Si se cumple la primera regla, se dice que la base de datos está en la "primera forma normal". Si se cumplen las tres primeras reglas, la base de datos se considera que está en la "tercera forma normal". Aunque son posibles otros niveles de normalización, la tercera forma normal se considera el máximo nivel necesario para la mayor parte de las aplicaciones.

Al igual que con otras muchas reglas y especificaciones formales, en los escenarios reales no siempre se cumplen los estándares de forma perfecta. En general, la normalización requiere tablas adicionales y algunos clientes consideran éste un trabajo considerable. Si decide infringir una de las tres primeras reglas de la normalización, asegúrese de que su aplicación se anticipa a los problemas que puedan aparecer, como la existencia de datos redundantes y de dependencias incoherentes.



Objetivo de la Normalización:

Las bases de datos relacionales se normalizan para:

- Evitar la redundancia de los datos.
- Evitar problemas de actualización de los datos en las tablas.
- Proteger la integridad de los datos.

En el modelo relacional es frecuente llamar tabla a una relación, aunque para que una tabla bidimensional sea considerada como una relación tiene que cumplir con algunas restricciones:

- Cada columna debe tener su nombre único.
- No puede haber dos filas iguales. No se permiten los duplicados.
- Todos los datos en una columna deben ser del mismo tipo.

(Kendall, 2005)& (magazine.joomla.org, 2013)

4.10.1 Formas normales

Primera forma normal (1FN)

En esta etapa debemos asegurarnos de que todos nuestras **campos** son únicos (**atómicos**) e indivisibles y eliminar todos los datos que sean repetidos o que tengan una dependencia funcional.

Dependencia funcional: Se pueden establecer asociaciones entre varios atributos de una misma relación llamada dependencia. Así, los valores de algunos atributos en una relación pueden determinar de forma única, el conocimiento de los valores de otros atributos de la misma relación. Estas dependencias traducen reglas semánticas que como se sabe, son muy comunes en el montaje del mundo real.

A partir de este concepto de dependencia funcional, se ha generado una serie de resultados teóricos que dan origen a lo que actualmente se conoce como Teoría Relacional de Bases de Datos. (González, 1996)



Segunda forma normal (2FN)

Dependencia completa. Está en 2FN si está en 1FN y si sus atributos no principales dependen de forma completa de la clave principal, es decir, **eliminar la redundancia** que se pueda observar, esto lo hacemos si al revisar las dependencias funcionales existentes notamos que un **subconjunto de nuestra tabla no depende de la llave en su totalidad (dependencia parcial de la llave)**. Para lograr esto debemos crear una tabla independiente para estos valores incluyendo algún campo que nos permita relacionarlo con la tabla original. (González, 1996)

Tercera forma normal Boyce-Codd (3FNBC)

Una relación se encuentra en tercera forma normal Boyce-Codd (3FNBC), si todos los atributos son determinados solo por llaves, es decir, si cada vez que $X \rightarrow A_1$, $A_1 \notin X$, se verifica en R, entonces X contiene una llave en R. (González, 1996)

4.11 Conectividad a la Base de Datos

Solicitud a la Base de datos

En base de datos, una consulta es el método para acceder a los datos en las bases de datos. Con las consultas se puede modificar, borrar, mostrar y agregar datos a una base de datos. Para esto se utiliza un lenguaje de consultas. (ALEGSA, 2014)

Inserción y Actualización de los datos a la base de datos

- **Insertar datos en una tabla:** describe las técnicas necesarias para insertar una o varias filas de datos en tabla.
- **Actualizar los datos de una tabla:** describe las técnicas necesarias para modificar datos de tablas. (Microsoft, 2014)



4.12 Gestor de base de datos SQL

Concepto

La sigla que se conoce como SQL corresponde a la expresión, inglesa Structured Query Language (entendida en español como Lenguaje de Consulta Estructurado), la cual identifica a un tipo de lenguaje vinculado con la gestión de bases de datos de carácter relacional que permite la especificación de distintas clases de operaciones entre éstas. Gracias a la utilización del álgebra y de cálculos relacionales, el SQL brinda la posibilidad de realizar consultas con el objetivo de recuperar información de las bases de datos de manera sencilla. Este gestor fue producido por Microsoft, sus lenguajes de consultas son T-SQL y ANSI SQL. Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como Oracle, PostgreSQL o MySQL.

Es un lenguaje para consulta de datos. Su finalidad básica es facilitar la manipulación de datos como si fuesen conjuntos, en contraposición al tratamiento que se efectúa desde los lenguajes de programación, donde cada dato se procesa de manera individual. (definicion.de/sql)

Características de SQL Server

- Soporte de transacciones.
- Soporta procedimientos almacenados.
- Incluye también un entorno gráfico de administración que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red solo acceden a la información.
- Permite administrar información de otros servidores de datos. (definicion.de/sql)

4.12.1 Partes de SQL Server

El lenguaje SQL está en realidad compuesto de varios sub-lenguajes entre los cuales se destacan los siguientes:

- DML: Es la parte más conocida del lenguaje SQL, contiene todas las sentencias de manipulación de datos (Extracción de información, actualización y eliminación).



- **DDL:** Con las sentencias DDL no se manipulan los datos propiamente dichos, sino la información de su estructura. Con las sentencias DDL puede definirse las columnas de una tabla y sus atributos (Eliminar o crear un índice, etc).
- **DCL:** Este lenguaje se compone de sentencias específicas para organizar la seguridad de acceso a los datos, facilitando la gestión de usuarios y el otorgamiento o denegación de los permisos necesarios para operar sobre cada elemento de la base de datos. (definicion.de/sql).

4.12.2 Porque utilizar SQL como gestor de base de datos

Las bases de datos son un componente imprescindible en cualquier sistema informático, por tal razón debe de llevarse en paralelo la experticia en desarrollo de aplicaciones en una plataforma de software y el desarrollo y modelado de bases de datos en un motor de base de datos, por lo cual Soluciones Aplicativas tiene experiencia trabajando con diversos motores de bases de datos, dando servicios como:

- Generación de modelos de datos para proyectos
- Administración de gestor de base de datos
- Instalación de gestor de base de datos
- Soporte de base de datos

A continuación se muestra una comparación entre los diferentes motores de datos utilizado en la actualidad:



Oracle, empresa prominente en tecnologías de servidores, aplicaciones empresariales y bases de datos, tiene su base de datos propia, la cual es una de las mas potentes, robustas y escalables a nivel mundial y de alta confiabilidad, Oracle Database. Oracle Database es una solución completa que incluye un motor de base de datos con posibilidad de crear sistemas de tablas relacionadas, índices, así como un lenguaje de programación para el desarrollo de procedimientos almacenados y triggers(PL/SQL) que permite el desarrollo del lado de la base de datos. Además debido a su afinidad con JAVA, es posible una cierta integración del lenguaje JAVA en el desarrollo de base de datos con Oracle, además del hecho que es multiplataforma, pues Oracle puede funcionar en Windows, Linux, Solaris. La mayoría de nuestros profesionales poseen amplia experiencia trabajando durante años con Oracle Database.



Del lado de las alternativas de código abierto u Open Source en el mercado de bases de datos, tenemos a PostgreSQL, un motor de base de datos que con los años y con el apoyo de la comunidad a nivel mundial se ha consolidado como un gestor de base de datos bastante robusto a pesar de su naturaleza gratuita. PostgreSQL tiene muchas características semejantes o iguales a la de motores de base de datos de pago como Oracle Database o DB2 como replicación, variedad de tipos de datos, capacidad de tuplas en las tablas, índices, lenguaje de scripting de procedimientos semejante en capacidades a PL/SQL, etc. Razón por la cual nos hemos enfocado mucho últimamente a utilizar esta base de datos en nuestros proyectos.



MySQL fue durante mucho tiempo el líder en motores de base de dato de código abierto entre la comunidad de desarrolladores. Sin embargo, debido a la compra de Sun Microsystems por parte de Oracle le dió un carácter privativo, ya que ahora hay algunas versiones de MySQL que son de pago. Sin embargo, MySQL es un gran referente en el mercado de base de datos, ya bastante maduro, contando desde hace mucho tiempo con características de las bases de datos de pago como sistemas de tablas relacionales, varios tipos de datos, desarrollo de procedimientos y triggers, etc. Y a pesar de la adquisición por parte de Oracle, esto también se traduce a un apoyo al desarrollo de la base de datos por parte de Oracle, lo que a la larga puede beneficiar a las empresas ya que viene a potenciar MySQL para un uso más empresarial.



DB2 es un potente motor de base de datos de pago que se compara al exitoso Oracle Database. Este es uno de los productos insignia de IBM, su motor de base de datos orientado a empresas. DB2 integra en su funcionamiento XML de manera nativa para operaciones de búsqueda con el propósito de optimizar el rendimiento en consultas. Además, es optimizado para ser ligero y almacenar datos sin necesidad de tanto espacio y sin tantas prestaciones de hardware o consumo de



recursos y servidores. Al igual que Oracle Database, DB2 puede funcionar en múltiples plataformas como Linux, Windows, Solaris, etc. Sin contar la compatibilidad que tiene con otros productos de la familia de IBM adicional a las características de un motor promedio de base de datos como clustering, replicación, desarrollo del lado de la base de datos, triggers, índices, etc.



SQLite es un motor de base de datos liviano que está basado en C, de tipo open source. Este difiere de los motores de base de datos tradicionales en que no se trata de un proceso independiente de la base de datos, sino de una interacción por parte de la aplicación con la librería de SQLite mediante la invocación de rutinas de la misma, lo cual la convierte en una base de datos bastante ligera. Está orientada al desarrollo de aplicaciones livianas y personales ya que la información de una base de datos se almacena en un solo archivo, características que si bien es cierto no son aptas para el desarrollo de aplicaciones empresariales, si pueden ser requeridas en otro tipo de escenarios, como el desarrollo de aplicaciones móviles para la plataforma iOS, donde el motor de base de datos usado por excelencia es SQLite.



La alternativa de Microsoft para el mercado de bases de datos es SQL Server. Optimizado principalmente para ser utilizado en conjunto con aplicaciones desarrolladas en la plataforma Microsoft .NET, SQL Server es una buena alternativa de pago un poco más accesible en términos económicos que otros motores como Oracle o DB2. Al igual que los otros motores de base de datos, cuenta con un lenguaje para el desarrollo de procedimientos almacenados o triggers (T-SQL) y es bastante estable hasta cierto punto y escalable.

4.13 Lenguaje de Modelado Unificado (UML)

¿Qué es UML?

El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas



notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas en los cuales modelar sistemas.

- Diagramas de Casos de Uso para modelar los procesos 'business'.
- Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
- Diagramas de Colaboración para modelar interacciones entre objetos.
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- Diagramas de Componentes para modelar componentes.
- Diagramas de Implementación para modelar la distribución del sistema.

UML es una consolidación de muchas de las notaciones y conceptos más usados en orientados a objetos. Empezó como una consolidación del trabajo de Grade Booch, James Rumbaugh, e Ivar Jacobson, creadores de tres de las metodologías orientadas a objetos más populares. (Schmuller, 1996).

Elementos estructurales

Las clases, objetos, actores, interfaces y caso de uso son cinco de los elementos estructurales en el UML. Aunque tiene diversas diferencias (mismas que, como ejercicio deberá indicar), son similares en el sentido de que representan partes ya sea físicas o conceptuales de un modelo. (Schmuller, 1996)



Elementos de comportamiento

Describen como funciona los cosas, es decir cuál es el comportamiento del sistema que información se necesita para realizar una función en un determinado modulo del sistema. Las interacciones y las máquinas de estado son ejemplos de cosas de comportamiento. Dentro de la que tiene que ver con la parte de elementos de comportamiento del sistema en UML, se encuentran los diagramas de comportamiento (que son todos aquellos diagramas que especifican el funcionamiento del sistema, las diferentes iteraciones de los usuarios con el software). (Kendall, 2005)

Elementos de agrupación

El paquete es el único elemento de agrupación en el UML, este le permite organizarlos elementos estructurales en un modelo. Un paquete puede contener cualquier tipo de elemento estructural, y diferentes tipos a la vez. (Schmuller, 1996)

Relaciones

La asociación, generalización, dependencia y realización, son las relaciones en el UML. (La inclusión y extensión son dos tipos de dependencias), sin las relaciones, los modelos UML no serían más que listas de elementos estructurales. Las relaciones conectan a tales elementos y de ese modo conectan los modelos con la realidad. (Schmuller, 1996)

4.13.1 Diagramas de UML

Es UML está compuesto por diversos elementos gráficos que combinan para conformar diagramas. Debido a que UML es un lenguaje, cuenta con reglas para combinar tales elementos.

A continuación se describirán brevemente los diagramas más comunes del lenguaje de modelado unificado (UML).



Diagrama de clase

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la Información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro. (Lucas)

Diagrama de objetos

Los diagramas de objetos son utilizados durante el proceso de Análisis y Diseño de los sistemas informáticos en la metodología UML.

Se puede considerar un caso especial de un diagrama de clases en el que se muestran instancias específicas de clases (objetos) en un momento particular del sistema. Los diagramas de objetos utilizan un subconjunto de los elementos de un diagrama de clase. Los diagramas de objetos no muestran la multiplicidad ni los roles, aunque su notación es similar a los diagramas de clase.

Una diferencia con los diagramas de clase es que el compartimiento de arriba va en la forma Nombre de objeto: Nombre de clase. (Lucas)

Diagrama de caso de uso

En el Lenguaje de Modelado Unificado, un diagrama de casos de uso es una especie de diagrama de comportamiento.

El Lenguaje de Modelado Unificado define una notación gráfica para representar casos de uso llamada modelo de casos de uso. UML no define estándares para que el formato escrito describa los casos de uso, y así mucha gente no entiende que esta notación gráfica define la naturaleza de un caso de uso; sin embargo una notación gráfica puede solo dar una vista general simple de un caso de uso o un conjunto de casos de uso. Los diagramas de casos de uso son a menudo confundidos con los casos de uso. Mientras los dos conceptos están relacionados, los casos de uso son mucho más detallados que los diagramas de casos de uso. (Lucas).



Diagrama de estado

En UML, un diagrama de estados es un diagrama utilizado para identificar cada una de las rutas o caminos que puede tomar un flujo de información luego de ejecutarse cada proceso.

Permite identificar bajo qué argumentos se ejecuta cada uno de los procesos y en qué momento podrían tener una variación.

El diagrama de estados permite visualizar de una forma secuencial la ejecución de cada uno de los procesos. (Lucas)

Diagrama de secuencia

El diagrama de secuencia es un tipo de diagrama usado para modelar interacción entre objetos en un sistema según UML. En inglés se pueden encontrar como "sequencediagram", "event-trace diagrams", "eventscenarios" o "timingdiagrams". (Lucas)

4.14 Internet

Definición

Es un conjunto de operaciones descentralizados de redes de comunicación interconectadas, que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. (Quispe-Otazu, 2014)

Servicios de Internet

Internet es mucho más que lo que permite WWW (Word Wide Web), aunque ciertamente es su servicio más exitoso. Los otros servicios de Internet que se identifican inmediatamente son: correo electrónico (SMTP), transmisión de archivos (FTP y P2P), conversaciones en línea (IRC e IM), telefonía (VoIP), televisión (IPTV), boletines electrónicos (NNTP) y acceso remoto a dispositivos (SSH y Telnet). (Castro, 2012)

Conectividad entre la aplicación y el servidor



La conectividad permite el uso y creación de bases de datos a la que puedan acudir los usuarios para hacer consultas y acceder a la información que les interese, es una herramienta imprescindible de cualquier sistema informativo sea en red o fuera de ella.

Protocolo SSL

SSL (Secure Socket Layers): Es un proceso que administra la seguridad de las transacciones que se realiza a través de internet. Se basa en un proceso de cifrado de clave pública que garantiza la seguridad de los datos que se envían a través de internet. Su principio consiste en el establecimiento de un canal de comunicación seguro (cifrado) entre dos equipos (el cliente y el servidor) después de una fase de autenticación.

4.14.1 Características del protocolo SSL

➤ Confidencialidad

Mediante el uso de la Encriptación se garantiza que los datos enviados y recibidos no podrán ser interpretados por ninguna persona que no sea ni el emisor ni el receptor.

➤ Integridad

Se garantiza que los datos recibidos son exactamente iguales a los datos enviados, pero no se le impide al receptor la posibilidad de modificar estos datos una vez recibidos.

➤ Autenticación

El vendedor se autentifica utilizando un certificado digital emitido por una empresa llamada Autoridad Certificadora, este documento es totalmente infalsificable y garantiza que el vendedor es quien dice ser. (Ramirez Ponferrada, 2014)

4.14.2 Funcionamiento del SSL

El protocolo SSL ha sido diseñado e implementado por Netscape para transferir información segura a través de Internet. Proporciona sus servicios de seguridad cifrando los datos intercambiando entre el servidor y el/la cliente con un algoritmo de cifrado simétrico y cifrando la clave de sesión mediante un algoritmo de cifrado de



clave pública. La clave de sesión es la que se utiliza para cifrar los datos que viene del y van al servidor seguro. Se genera una clave de sesión distinta para cada transacción, lo cual permite que aunque sea intervenida en una transacción dada para descifrar futuras transacciones.

Una conexión con SSL requiere que toda la información enviada entre el/la cliente y el servidor sea encriptado por el software de envío y des encriptado por el software que recibe, protegiendo la información ante intentos de interceptación. Además, todo dato enviado sobre una conexión encriptado por SSL, es protegido con un mecanismo que detecta si el dato ha sido alterado en su tránsito. Cuando el/la cliente pide al servidor seguro una comunicación segura, el servidor abre un puerto cifrado, gestionado por el software SSL Record, situado encima del protocolo TC/IP de comunicación en internet. Será el software de alto nivel, Protocolo SSL Handshake, quien utilice el Protocolo SSL Record y el puerto abierto para comunicarse de forma segura con el/la cliente.

4.15 Factibilidad

El estudio de factibilidad no se considera un estudio completo de un sistema, sino más bien una herramienta que se utiliza para recopilar una serie de datos que son necesarios para la administración, la cual facilita la toma de decisión referente a la continuidad del proyecto. Para los proyectos de sistemas la factibilidad es valorada de tres aspectos: técnica, económica y operacional. (Kendall, 2005)

4.15.1 Factibilidad Técnica

Gran parte de la determinación de los recursos tienen que ver con la evaluación de la factibilidad técnica. El analista debe averiguar si es posible actualizar o incrementar los recursos técnicos actuales de tal manera que satisfagan los requerimientos bajo consideración. Sin embargo, los “agregados” a los sistemas existentes son costosos y no redituables, simplemente porque no cumplen las necesidades con eficiencia. Si no es posible actualizar los sistemas existentes, la siguiente pregunta es si hay tecnología disponible que cumplan las especificaciones.

Durante esta fase del estudio de factibilidad se consideran todos aquellos recursos técnicos necesarios para el desarrollo del sistema, en otras palabras se valoraron dos enfoques hardware y software, esto se realiza para tener las bases necesarias que van a soportar al sistema, una vez este llegue a su fase de implementación e igualmente sea aprobado y aceptado por la institución idónea en el tema, ya que se deben detallar los requerimientos de las TIC'S. (Kendall, 2005)



4.15. 2 Factibilidad económica

La factibilidad económica es la segunda parte de la determinación de recursos. Los recursos básicos a considerar son: el tiempo propio y del equipo de sistema, el costo de hacer un estudio de sistema completo (incluyendo el tiempo de los empleados con los que se trabajara), el costo del tiempo de los empleados del negocio, el costo estimado de hardware y el costo estimado del software y/o desarrollo de software.

El negocio de que se trate deberá ser capaz de hacer el valor de la inversión en su ponderación antes de comprometerse a un estudio de sistemas completo. Si los costos a corto plazo no son sobrepasados por las ganancias a largo plazo, o no producen una reducción inmediata en los costos de operación, el sistema no es factible económicamente y el proyecto ya no debe continuar.

Es decir en lo que respecta a la factibilidad económica o financiera se estima los costos de los equipos informáticos y redes, de igual forma los recursos humanos establecidos, los cuales son necesarios tanto para el desarrollo e implementación del sistema. (Kendall, 2005)

4.15.3 Factibilidad operativa

Suponiendo por un momento que los recursos técnicos y económicos se evaluaron de manera adecuada. El analista de sistemas aún debe considerar la factibilidad operativa del proyecto solicitado. La factibilidad operativa depende de los recursos humanos disponibles para el proyecto e implica determinar si el sistema funcionara y será utilizado una vez que se instale. (Kendall, 2005).

4.16 OXFORD international School

Oxford international School se ubica en colonia Independencia frente al colegio Miguel de Cervantes en el municipio de Managua, departamento de Managua es un centro de capacitación técnica reconocido por formar técnicos profesionales en demanda del sector productivo promocional, en correspondencia a las demandas del sector productivo y social de todos los sectores, incluidos los más desprotegidos.

Objetivo

El objetivo principal es crear capacidades y competencia a jóvenes de escasos recursos en los diferentes modos de formación tales como aprendizajes, habilitación, complementación y especialización para poder enfrentar el mundo laboral.



Misión

Desarrollar técnicas de capacitación que permitan implementar enseñanzas del idioma inglés como segunda lengua beneficiando primordialmente a la población empresarial aportante del 2% y grupos de bajos recursos.

V. Diseño metodológico

La aplicación móvil para realizar inscripciones en instituto Oxford será una herramienta de gran apoyo al momento de realizar inscripciones de curso se espera que la App pueda mantener de forma íntegra y segura toda la información necesaria para el proceso de matrícula del estudiantado. En su creación se utilizó el método de desarrollo ágil (Programación extrema o XP) por encontrarlo oportuno debido al factor tiempo en que se desarrollaría la aplicación no obviando el mejoramiento continuo que puede presentar la misma para mantenerla viva.

5.1 Tipo de investigación (tipo de estudio)

La investigación utilizada tomando en cuenta la finalidad perseguida fue de carácter aplicada ya que se pusieron a pruebas todos los conocimientos informáticos adquiridos durante todos estos años de estudio y aprendizaje, como transversal por el tiempo determinado de desarrollo.

5.2 Población y muestra

La aplicación va dirigida a todos los centros escolares del municipio de Managua que son beneficiados con las becas, para los resultados finales se tomó como muestra el tercer año A del instituto Miguel de Cervantes.

5.3 Técnicas de recolección de datos

Para poder llegar a un producto final fue necesario tener un indicio el cual fue facilitado por los señores encargados del instituto Oxford para poder acercarnos al medio que se preparaba, en el momento de realizar las inscripciones de pre matrícula, fue necesario de valernos de los siguientes métodos de recolección de información:

- La observación: principalmente porque a través de ella nos dimos cuenta el procedimiento a realizar desde la preparación de la papelería hasta el momento de realizar una matrícula, empapándonos de todos y cada uno de los procesos llevados a cabo.



- La entrevista: Gracias a ellas despejamos dudas que surgieron al momento de ver algún procedimiento con el afán de tomarlo en cuenta para la realización de la apps. Esta fue realizada tanto a los responsables del instituto como a los usuarios de la apps que en este caso son los asesores del centro.
- La encuesta: A través de la encuesta se pudo consolidar los resultados finales así como establecer una conclusión, acerca del funcionamiento de la aplicación móvil.

5.4 Modelo de desarrollo

Para el desarrollo de la aplicación hicimos uso de la programación extrema o Extreme Programming (XP), porque consideramos que el tiempo de desarrollo se necesitaría trabajar de una manera acelerada, pero no obviando la secuencia de pasos necesaria para realizar un desarrollo agradable y sencillo, también porque es un método que se adapta al desarrollo de sistema pequeños y grandes; optimiza el tiempo de desarrollo, permite realizar el desarrollo de sistema en pareja para complementar conocimientos.

Fases del método de programación extrema

Fase de análisis y diseño

En esta fase se procedió a analizar toda la información que se pudo recolectar y se consideró de gran importancia, se comenzó a crear el diagrama entidad relación en el estor de base de datos SQL server 2012, luego se procedió a elaborar los diferentes diagramas de UML: Diagrama de casos de uso, diagrama de secuencia y diagrama de actividades.

Para el diseño y modelado de la aplicación móvil (apps) se utilizó Microsoft Visio 2007 y ER estudio (embarcadero), en pocas palabras se puede decir que en esta fase se hizo todo el diseño lógico de la aplicación.

- **Para diseñar las vistas se hizo uso de eclipse.**
- **Para el diseño de la base de datos SQL 2012 Enterprise edición**
- **Para la elaboración del documento se utilizó Microsoft office 2007.**



- **Para la elaboración del diccionario de datos se utilizó Embarcadero.**

Fase de código y prueba

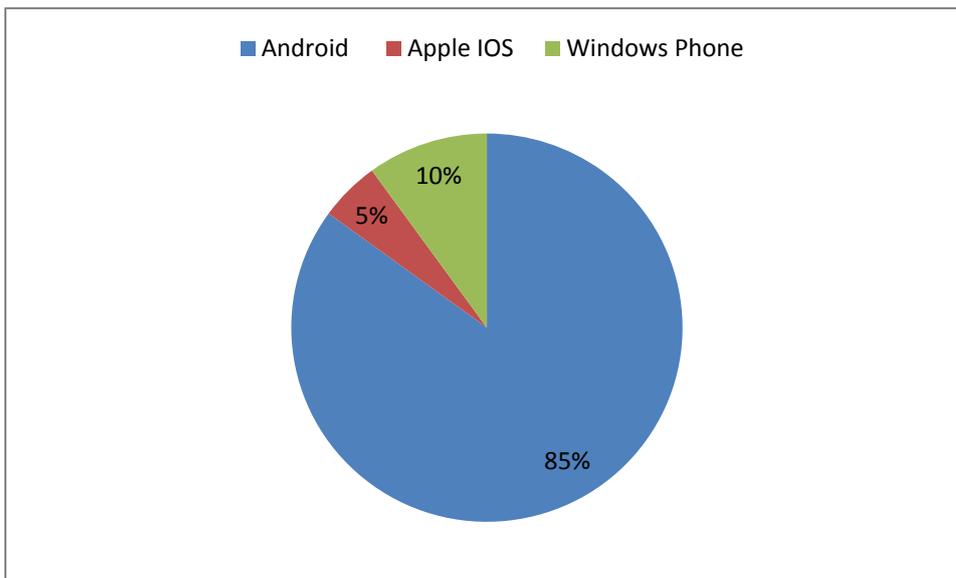
En esta fase se realizó la generación de los diferentes métodos que pondrían en funcionalidad la aplicación y se realizaron pruebas necesario con al menos un par de dispositivos móviles. En lo que respecta la creación de la base de datos se utilizó como gestor de base de datos SQL 2012, porque es capaz de mantener la integridad de los datos, permite deshacer las transacciones que dan problemas o que están en estado intermedio es decir que si algo falla al momento de realizar una transacción es capaz de deshacerlas y regresar a su estado inicial.



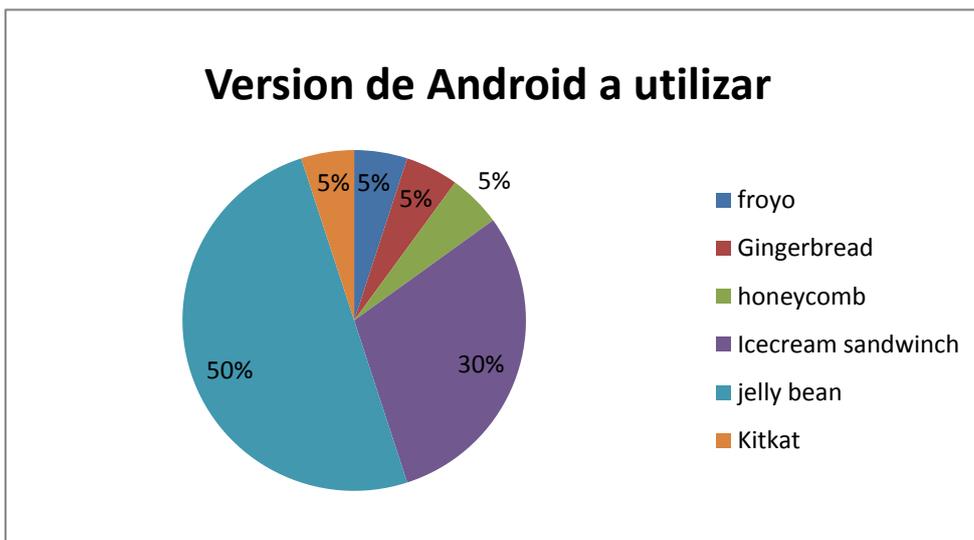
VI. Análisis de resultados

6.1 Encuestas previas al desarrollo de la aplicación

- ✓ Se solicitó marcar el sistema operativo de preferencia para la realización de la aplicación y los resultados fueron los siguientes:



- ✓ Se solicitó marcar la versión de Android favorita para adquirir siendo los resultados:



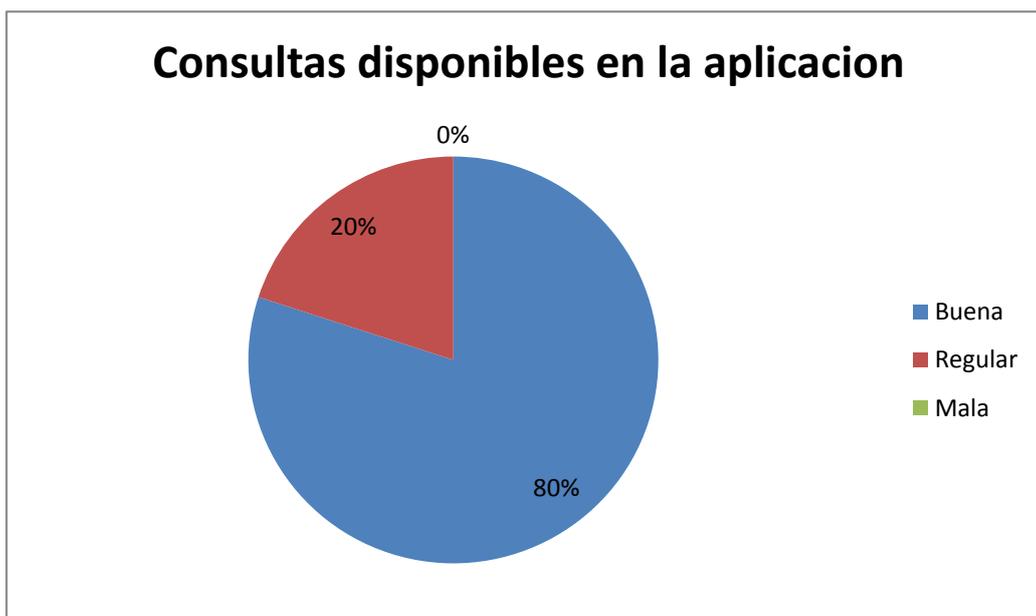


6.2 Encuestas realizadas durante el periodo de pruebas de la app móvil.

- ✓ El 90% de los usuarios opina que la información necesaria (la información de los estudiantes) si se maneja al momento de realizar la matricula oficial, solo un 10% considera que no está disponible.

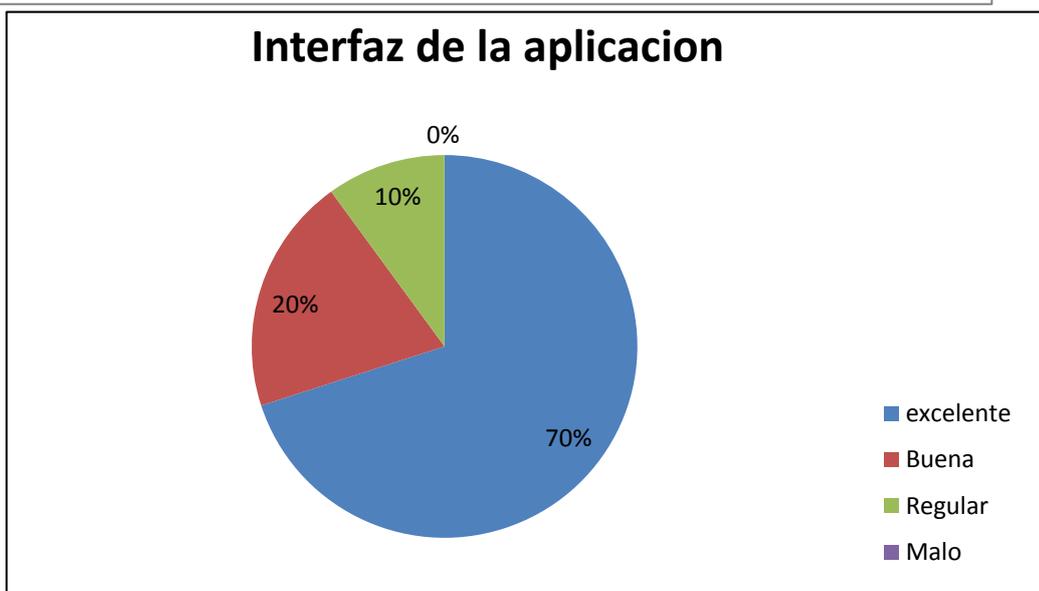
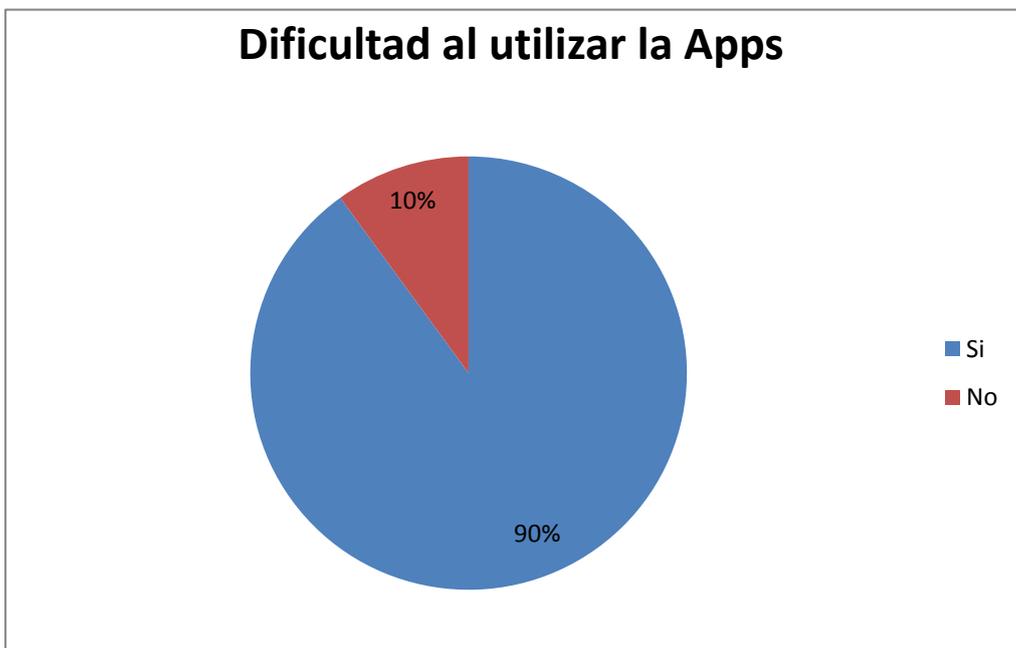


- ✓ El 80% de los usuario clasifica como buena la actividades disponibles en la aplicación, un 20% las considera regular





- ✓ El 70% de usuarios opina que la interfaz de la apps es amigable, un 20% considera que es buena y solo un 10% la considera regular.
- ✓ El 90% de los usuarios dijeron no tener problemas con el uso de la aplicación, un 10% comento si tener dificultades.





6.3 Proceso de pre matricula

Al realizar las debidas indagaciones sobre el proceso de matricula a un alumno se encontró:

1. Los promotores retiran la hoja de inscripción en la dirección del centro
2. Luego los promotores se dirigen al centro correspondiente para levantar la información del alumno respecto a la hoja.
3. Las hojas son llevadas de regreso a la secretaria del centro para ser ingresada por la secretaria al sistema del centro.
4. Cuando un alumno llega a ser efectiva su matrícula es se dirige a la secretaria y es buscado en el sistema por la secretaria del la dirección para efectuar el pago correspondiente.

6.4 Herramientas de desarrollo

La aplicación fue desarrollada con la ayuda de las siguientes aplicaciones o software



Eclipse IDE principal donde se programaron los diferentes módulos que componen la app.



Photoshop herramienta que facilito el retoque y diseño de la app.



Se utilizó como gestor de bases de datos.



Se realizaron los diferentes diagramas UML.



6.5 Descripción de la aplicación

La aplicación móvil Oxford permitirá registrar datos que llevaran a cabo la matrícula para recibir cursos técnicos impartidos en Oxford .La ventaja de utilizar la aplicación al momento de pre-matricular es que los datos necesarios al momento de matricular estarán en orden, sin alteraciones o equivocaciones y podrá ahorrar tiempo al evitar la perdidas de documentos durante la matricula .La aplicación puede correr en dispositivos móviles que funcionan bajo Android desde la 2.0 en adelante.

6.6 Estudio de factibilidad

El hardware que se propone es válido tanto para su desarrollo como para la implementación.

6.6.1 Factibilidad técnica

Cantidad	Descripción
1	Tablet marca Verykool
	Modelo: Versión de Android: Android OS, v4.2 JellyBean
	Tamaño: 7.0 pulgadas
	RAM: 512MB
	Memoria interna: Resolución de pantalla: 480 x 800 pixel
	Procesador: dual-core 1.3GHz
1	Laptop marca Hp
	Sistema operativo Windows 8
	RAM: 6GB
	DH: 500 GB
	Procesador: 2.2GHZ



Requerimientos de Software para el desarrollo de la apps móvil

Paquete	Descripción
Microsoft Windows 8	Sistema operativo para desarrollar la app
Microsoft Office 2007	Paquete de oficina
SQL 2012	Software de base de datos
Eclipse	IDE de desarrollo para la app
Visual Studio 2010	Desarrollo de la administración de la app.
ER Estudio	Software de desarrollo para diagramas

6.6.2 Factibilidad Económica

- Estimado de costos para los dispositivos donde correrá la aplicación móvil y donde se desarrollara la aplicación.

Equipo	Cantidad	Precio unitario	Sub-total
Tablet	7	U\$150	U\$1050
Laptop	1	U\$400	U\$400
Total			U\$1450

- Estimación de costo del software para el desarrollo de la aplicación móvil de inscripciones.

Paquete	Cantidad	Precio unitario	Sub-total
Microsoft Windows	1	U\$300	U\$300
Microsoft office	1	U\$250	US250
Microsoft SQL 2012	1	U\$600	U\$600
Hosting (tarifa mensual)		U\$17.92	\$215



Eclipse	1	U\$0	U\$0
Visual Studio	1	U\$ 799	U\$ 799
Total			U\$2164

- Estimación del costo del personal y otros insumos de desarrollo de la aplicación de inscripciones.

Costos operativos	Cantidad	Valor	Sub-Total
Costo de mano de obra	2	U\$500	U\$1000
Materiales y suministros	Variable	U\$150	U\$150
Total			US1150

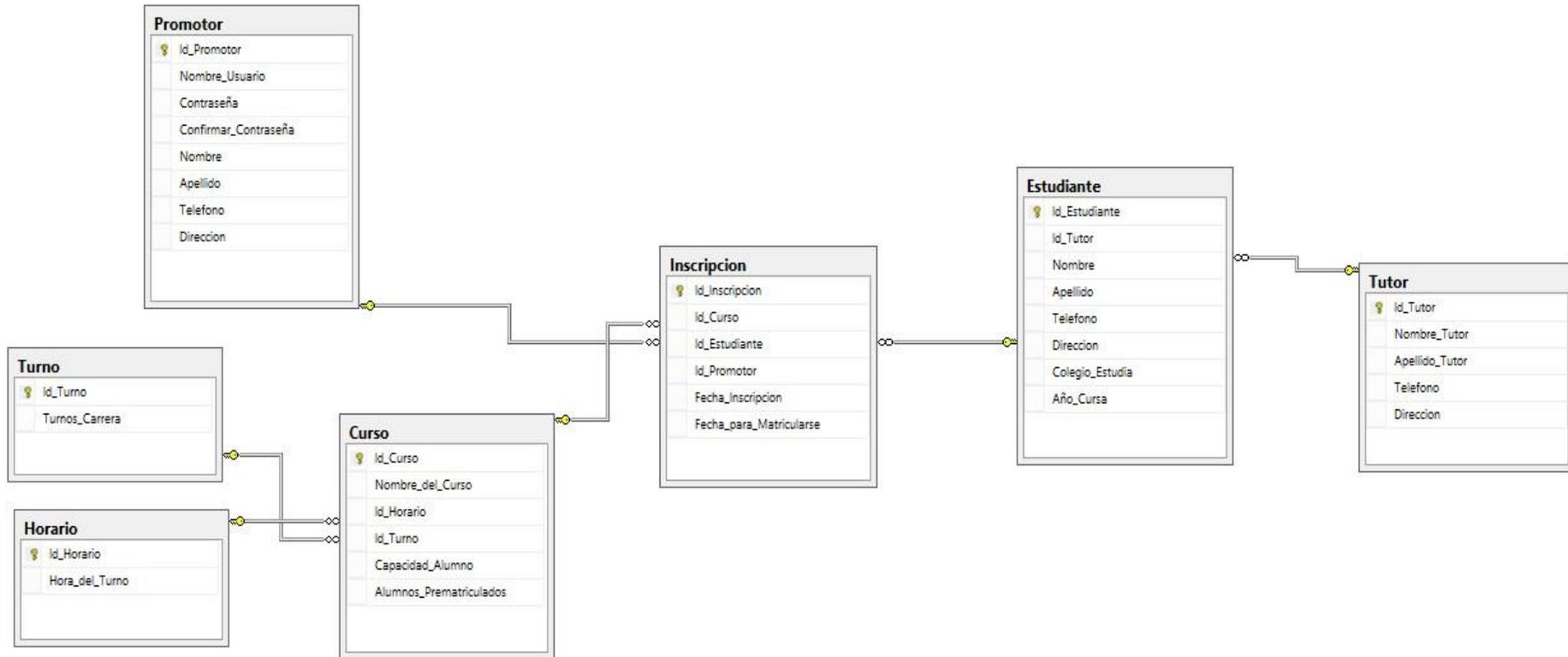
El costo total estimado para el desarrollo e instalación de la aplicación móvil para inscripciones es de **U\$4,764**.

6.6.3 Factibilidad Operacional

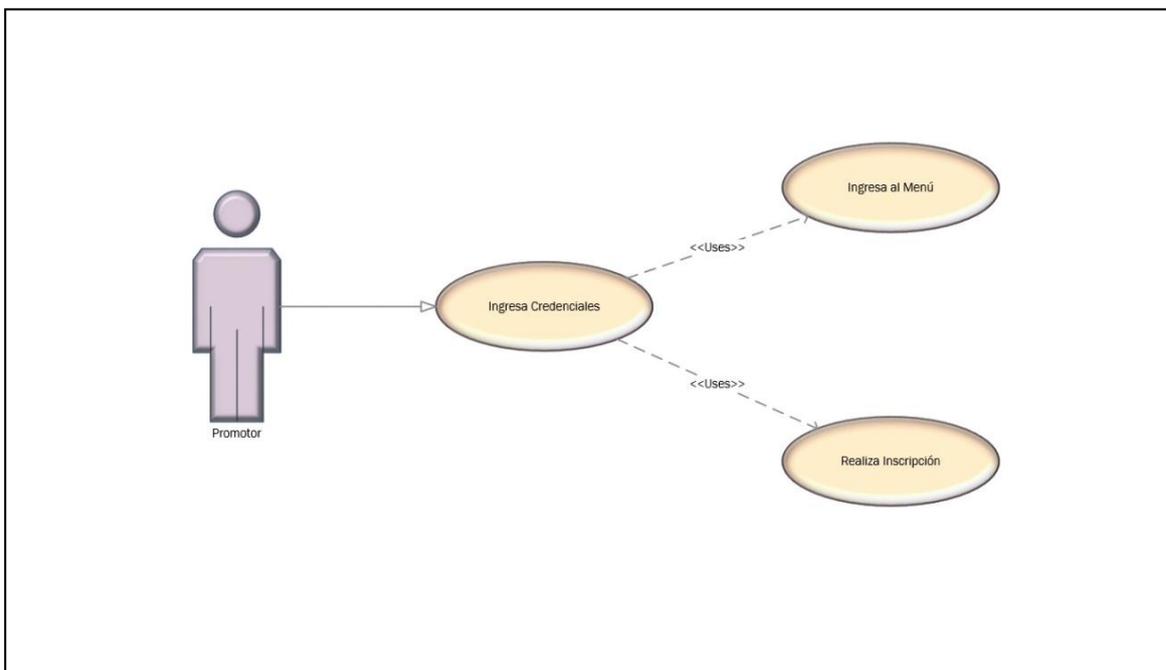
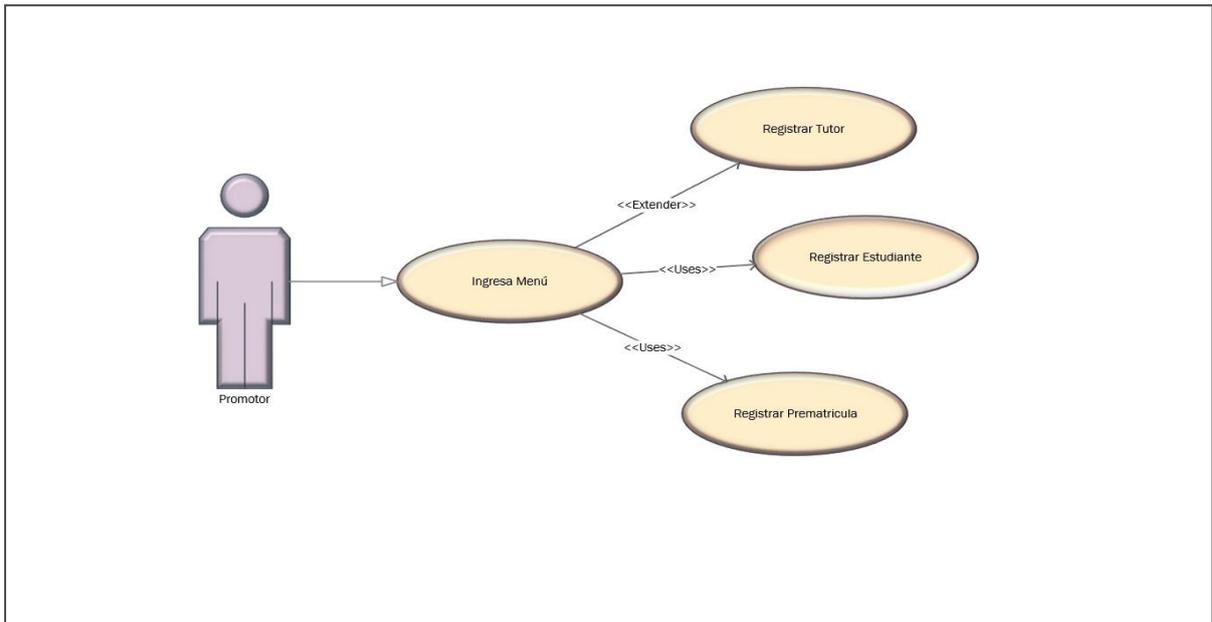
La aplicación móvil para la pre-matricula Oxford es una aplicación factibl porque si uso resuelve la problemática de la seguridad de los datos, es un aplicación sencilla y de practico uso, está diseñada de tal forma que su programación puede ser extendida a otros módulos según la necesidad del instituto. Los usuarios finales que serán los promotores de pre matriculas cuentan con la debida experiencia previa en manejar este tipo de aplicaciones, pero en tal caso que tengan alguna dificultad serán previamente capacitados para su manipulación.



6.7 Diagrama entidad-relación

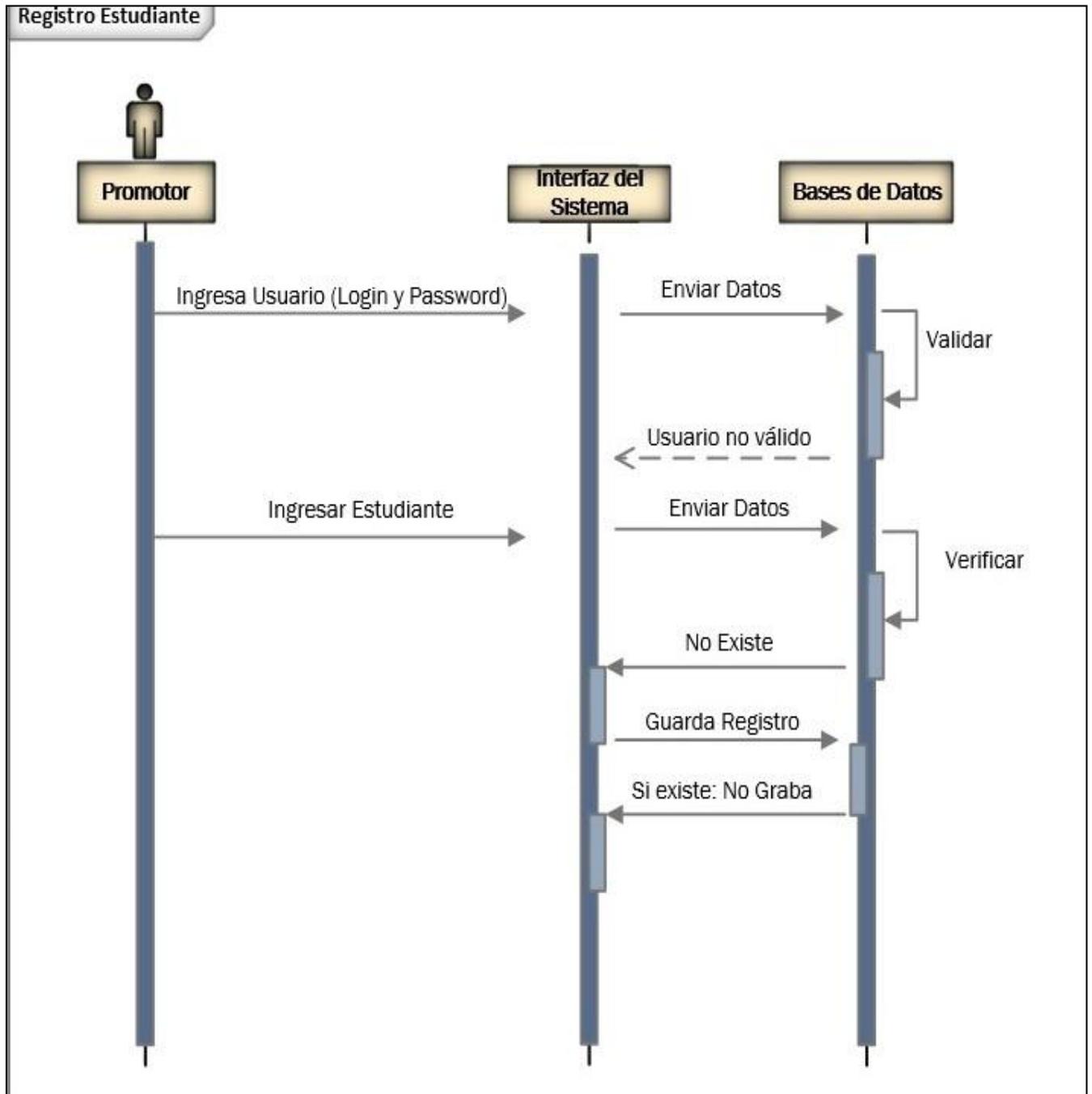


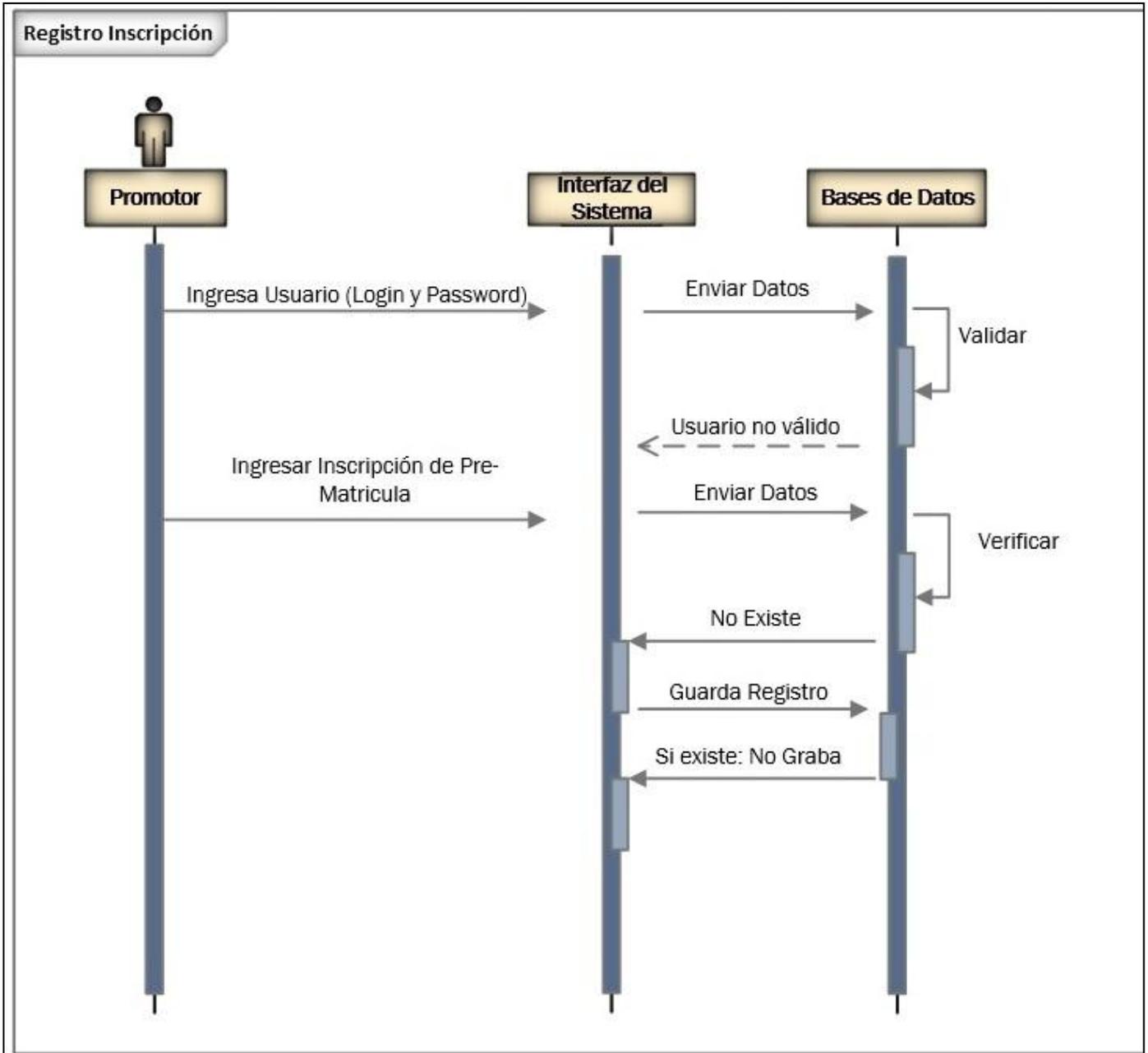
6.8 Diagrama de casos de usos

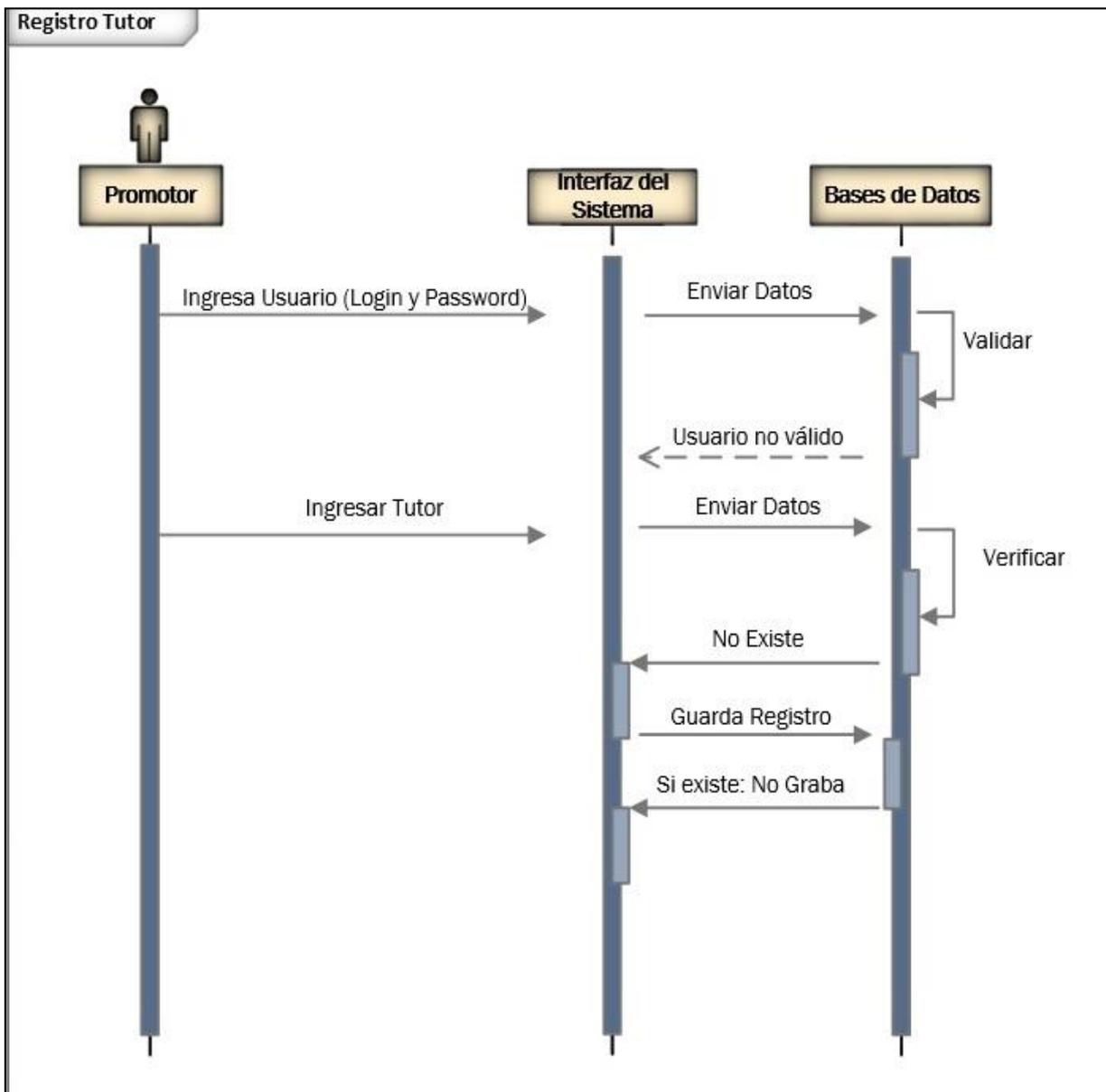




6.9 Diagrama de secuencia









6.10 Diccionario de datos

Nombre de la Tablas	Nombre del campo	Descripción	Tipo de Dato	Valores en blanco	Llave Primaria
Curso	Id_Curso	Almacena el código que identifica al curso técnico	INTEGER	NOT NULL	Yes
	Nombre_del_Curso	Almacena el nombre del curso técnico	VARCHAR (30)	NULL	No
	Alumnos_Prematriculados	Almacena	INTEGER	NULL	No
Estudiante	Id_Estudiante	Almacena el código del alumno registrado	INTEGER	NOT NULL	Yes
	Direccion	Guarda la dirección domiciliar del alumno	VARCHAR (100)	NULL	No
	Telefono	Guarda el número telefónico del estudiante	VARCHAR (8)	NULL	No
	Edad	Almacena la edad del estudiante	INTEGER	NULL	No
	Colegio_Estudia	Almacena el centro escolar donde estudia el alumno	VARCHAR (80)	NULL	No
	Año_Cursa	Almacena el año que cursa el alumno	INTEGER	NULL	No
	Nombre_Completo	Almacena el nombre del alumno a registrar	VARCHAR (90)	NULL	No
	Id_Tutor	Guarda el código del horario del curso	INTEGER	NOT NULL	No
Horario	Id_Horario	Almacena el horario que se imparte el curso	INTEGER	NOT NULL	Yes
Inscripcion	Id_Inscripcion	Almacena el código de registro de la inscripción de curso	INTEGER	NOT NULL	Yes
	Id_Promotor	Almacena el código del promotor que realizo la prematricula	INTEGER	NOT NULL	No
	Fecha_Incripcion	Almacena la fecha en que se inscribe el curso	DATETIME	NULL	No
	Fecha_Para_Matricularse	Almacena la fecha en que se realizara la prematricula	DATETIME	NULL	No
	Id_Estudiante	Almacena el código del estudiante	INTEGER	NOT NULL	No
	Id_Curso	almacena el código de	INTEGER	NOT NULL	No



		curso			
	Id_Turno	Almacena el código del turno en que se llevara el curso	INTEGER	NOT NULL	No
	Id_Horario	Almacena el horario en que se llevara el curso	INTEGER	NOT NULL	No
Promotor	Id_Promotor	Almacena el código del promotor	INTEGER	NOT NULL	Yes
	Nombres	Almacena el nombre del promotor	VARCHAR (40)	NULL	No
	Apellidos	Almacena el apellido del promotor	VARCHAR (40)	NULL	No
	Telefono	Almacena el teléfono del promotor	VARCHAR (8)	NULL	No
	Nombre_Usuario	almacena el usuario que se le asigna al promotor	VARCHAR (20)	NULL	No
	Contrasenia	Almacena la contraseña para ingresar a la aplicación	VARCHAR (20)	NULL	No
	Confirmar_Contrasenia	Almacena la contraseña para ingresar a la aplicación	VARCHAR (20)	NULL	No
	Direccion	almacena la dirección domiciliar del promotor	VARCHAR (100)	NULL	No
Turno	Id_Turno	Almacena el código asignado al curso	INTEGER	NOT NULL	Yes
	Turnos_Carrera	Almacena el turno en que se imparte el curso	VARCHAR (10)	NULL	No
Tutor	Id_Tutor	Almacena el código del tutor	INTEGER	NOT NULL	Yes
	Teléfono	almacena el nombre del tutor	VARCHAR (8)	NULL	No
	Direccion	Almacena la dirección domiciliar del tutor	VARCHAR (100)	NULL	No
	Nombre_Completo_Tutor	Almacena el nombre completo del tutor	VARCHAR (90)	NULL	No



6.11 Vistas de la aplicación

La figura muestra el acceso directo que se crea al instalar el APK de la aplicación móvil para acceder a la vista de login.

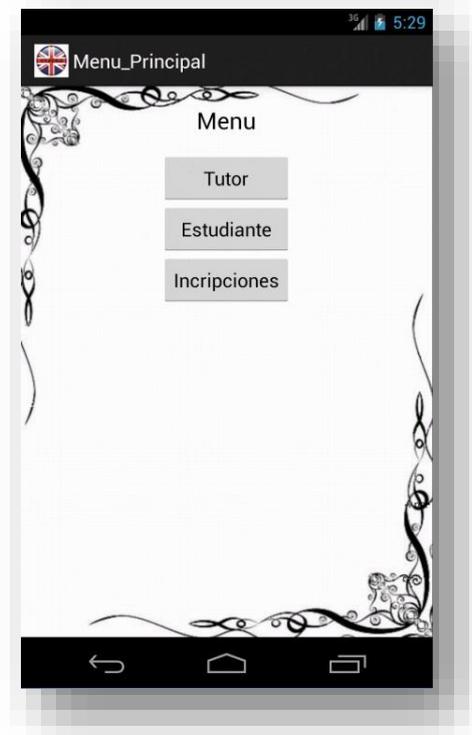


La siguiente figura, muestra la vista de login de la apps.

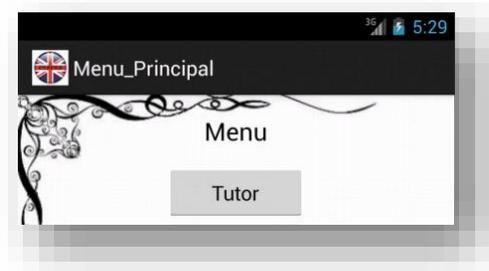




Una vez que usuario se haya verificado, se muestra la vista donde podrá realizar el registro correspondiente según el menú presionado.

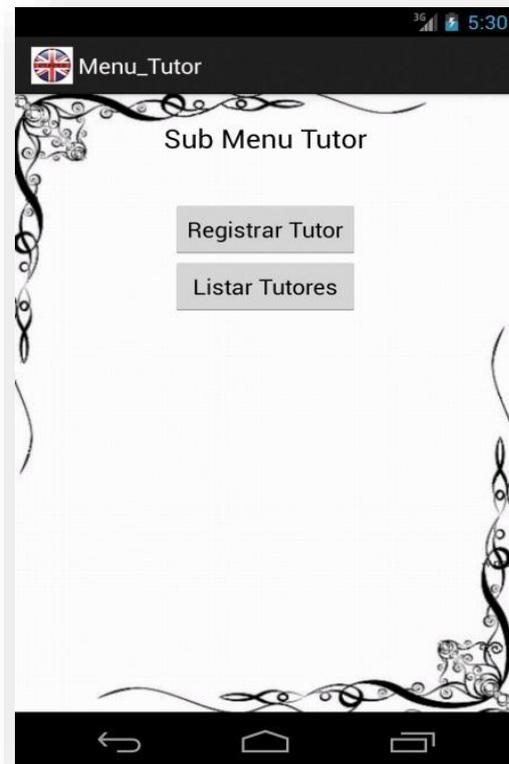


Si necesita registrar un padre de familia o responsable del estudiante deberá accesar al menú tutor

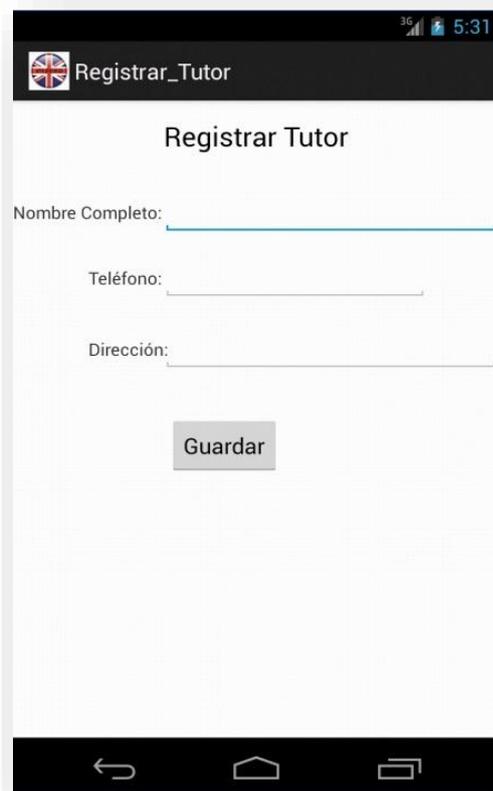




Luego de presionar el menú tutor aparece la vista de registro de tutor donde podrá registrar o listar los tutores que ya han sido registrados.



Al seleccionar registrar tutor se podrá ingresar los datos correspondientes al tutor y luego almacena.





Al presionar listar tutor se activa la siguiente vista donde se podrá visualizar los tutores y ingresados.



Al ingresar al menú estudiante



La funcionalidad es idéntica a la explicada anteriormente lo único que cambia es la vista.





- Vista del menú inscripciones

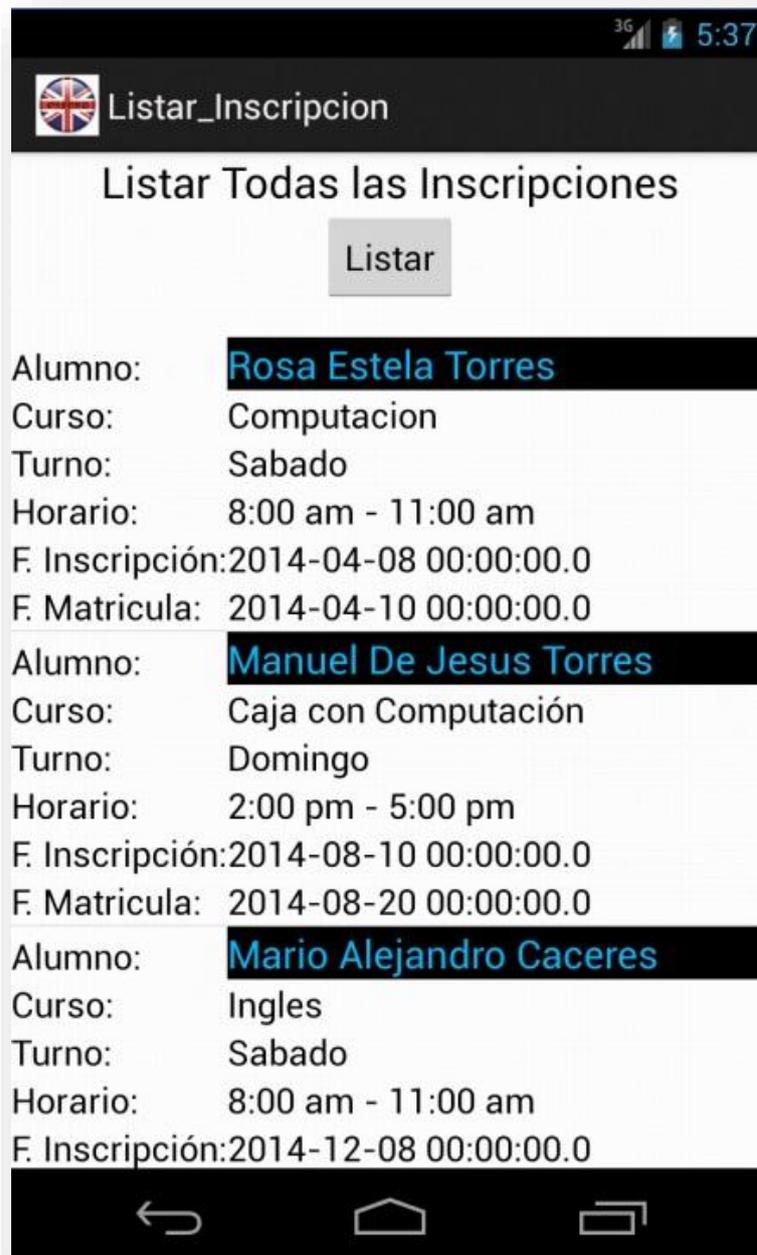


- Vista de la opción registro de inscripción





- Vista de la opción lista de registro de inscripción.





6.12 Pruebas Realizadas

Las pruebas se realizaron con las pre matriculas realizadas en el centro Miguel de Cervantes donde se registro a los alumnos del tercer año dentro de los resultados de las pruebas se obtuvieron los siguientes resultados:

- El tiempo de carga de la aplicación fue considerable.
- La aplicación se mantuvo constante es decir que en ningún momento se cayó.
- Su uso no mostró complicaciones por los usuarios.
- Los datos se mostraron y guardaron correctamente.



VII. Conclusiones

- Este trabajo nos permitió adentrarnos en cada una de las tecnologías y herramientas utilizadas en el desarrollo de aplicaciones móviles especialmente con las que se utilizan para desarrollar las aplicaciones para Android lo que esto refiere podemos concluir que: Android es la tecnología que más se ajusto al desarrollo de la app tomando en cuenta lo solicitado por la administración del centro.
- Por su parte el proceso de realizar la gestión de pre matricula, se define que no es la forma más segura de mantener la información recolectada, por tanto se llevará el mismo método tradicional de pre matricula utilizando ahora los beneficios de una app llamada Oxford.
- Tomando en cuenta los resultados de la encuesta, podemos reafirmar que el diseño que se realizo para la elaboración de la app fue todo un éxito ya que los usuarios no tuvieron problemas con la manipulación de la misma, así como incomodidad al momento de realizar la pre matricula de cursos Oxford gracias a la sencillez de la misma.
- Para Finalizar en lo que corresponde la programación de la aplicación móvil se llevo a cabo con gran logro, esto es argumentado y sostenido gracias a la parte de las pruebas realizadas al momento de efectuar pre-matriculas para Oxford.



VIII.Recomendaciones

- Se recomienda Extender la app Oxford de tal manera que beneficie al instituto, como podría ser el manejo de las matriculas a través de dicho dispositivo.



X. Bibliografía

- ALEGSA. (3 de Mayo de 2014). *Definicion de Consulta en Base de datos*. Obtenido de <http://www.alegsa.com.ar/Dic/consulta%20en%20base%20de%20datos.php>
- Blanco, L. M. (2005). *"Fundamentos de programacion con Visual Basic.NET"*.
- Blanco, L. M. *Fundamento de programación con Visual Basic.Net*.
- Carlos, A. G. (2002). *"Sistemas de Base de Datos"*. San Jose: Editorial Tecnológico de Costa Rica.
- Castro, L. (1 de Julio de 2012). *¿Que es Internet?* Obtenido de <http://aprenderinternet.about.com/od/ConceptosBasicos/g/Que-es-internet.htm>
- Cegarra Sánchez José ,2012,Los métodos de investigacion,editorial Díaz Sánchez
- Definición de docente - Qué es, Significado y Concepto*. (s.f.). Obtenido de <http://definicion.de/docente/#ixzz2zfRCJCQF>
- definicion.de/sql*. (s.f.). Obtenido de Definición de SQL - Qué es, Significado y Concepto : <http://definicion.de/sql/#ixzz305RsD5Qd>
- definicionabc.com- Def. TUTOR*. (s.f.). Obtenido de definicionabc.com- Def. TUTOR: <http://www.definicionabc.com/derecho/tutor.php#ixzz31q3vYIcz>
- definicionabc.com-Def. ESTUDIANTE*. (s.f.). Obtenido de definicionabc.com-Def. ESTUDIANTE: <http://www.definicionabc.com/general/estudiante.php#ixzz2zdNFYZZS>
- Gómez Marcelo M., 2006 introducción a la metodología de investigación científica, edicion córdobas.
- González, C. (1996). *Sistemas de Bases de Datos*. Costa Rica: Tecnológica de Costa Rica.
- Incap.init*. (s.f.). Obtenido de Incap.init: <http://www.incap.int/sisvan/index.php/es/acerca-de-san/conceptos/sistema-de-vigilancia>
- jorgesaavedra.wordpress.com*. (s.f.). Obtenido de Lenguajes de Programación: <http://jorgesaavedra.wordpress.com/2007/05/05/lenguajes-de-programacion/>
- Kendall, K. K. (2005). *Análisis y diseño de sistemas. Sexta edición*. México: McGraw Hill.
- Lucas. (s.f.). *ftp.inf.utfsm.cl*. Obtenido de Modelado Sistemas UML: <ftp://ftp.inf.utfsm.cl/pub/Linux/Docs/LuCaS/Tutoriales/doc-modelado-sistemas-UML/multiple-html/c12.htm>
- M.Kroenke, D. (2003). *"Procesamiento de Bases de Datos" Fundamentos, diseños e implementación. Octava Edición*. México: PEARSON EDUCACIÓN.
- magazine.joomla.org*. (Noviembre de 2013). Obtenido de desarrollo de componentes normalizacion de la base de datos: <http://magazine.joomla.org/es/ediciones->



anteriores/noviembre-2103/item/1610-desarrollo-de-componentes-normalizacion-de-la-base-de-datos

Microsoft, (. (3 de Mayo de 2014). *Cambiar datos en una base de datos*. Obtenido de Recuperado: <http://msdn.microsoft.com/eses/library/ms181096.aspx> ¿

mifuturo.cl. (s.f.). Obtenido de mifuturo.cl: <http://www.mifuturo.cl/index.php/media-conoce-tus-opciones/sabes-que-es-la-educacion-tecnica>

Monografias.com. (s.f.). Obtenido de Analisis de Sistema de Información: <http://www.monografias.com/trabajos55/analisis-sistemas-informacion/analisis-sistemas-informacion.shtml#ixzz2zg1HENmS>

Monografias.com . (s.f.). Obtenido de Monografias.com: <http://www.monografias.com/trabajos55/analisis-sistemas-informacion/analisis-sistemas-informacion.shtml#ixzz2zfzHrYCD>

Pressman, R. S. (1997). *Ingeniería del Software un enfoque práctico*. México: McGraw Hill.

profesorjavierruero.blogspot.com. (s.f.). Obtenido de Que es programación?: <http://profesorjavierruero.blogspot.com/2013/01/que-es-programacion.htm>

Quispe-Otazu, R. (18 de Mayo de 2014). *¿Que es Internet?* Obtenido de <http://www.rodolfoquispe.org/blog/que-es-internet.php>

Ramirez Ponferrada, E. (12 de Enero de 2014). *Protocolo SSL*. Obtenido de <http://www.pedroximenez.com/ssl.htm>

Schmuller, J. (1996). *Aprendiendo UML en 24 horas*. México: Pearson Educación S.A.

Senn, J. A. (1992). *Analisis y Diseño de Sistemas de Información*. México: McGraw Hill.

Sequeiro, V. y. (2007). *“Investigar es fácil”*. *Manual de investigación*. 5ta Edición. Managua: Oceano.

Sommerville, I. (2005). *Ingeniería del software*. Madrid: Pearson Educación.



IX. Anexos

9.1 Entrevista

Universidad Nacional autónoma de Nicaragua

UNAN-Managua

Recinto Universitario “Rubén Darío”

Estamos llevando a cabo una investigación sobre “El proceso de pre matriculas de alumnos que tomen cursos que imparte el Instituto”, por lo que pedimos nos permita realizar la presente entrevista, cuyo objetivo es: Desarrollar una aplicación móvil que mantenga seguro los datos al momento de realizar la matricula oficial.

I. Datos personales.

Nombre del entrevistado: _____

Ocupación: _____

II. Cuestionario

1. ¿Qué piensa usted acerca del proceso tradicional de pre matricula en el instituto Oxford?
2. ¿Nos podría mencionar algunas desventajas que tiene el proceso tradicional de prematricular?
3. ¿Qué opinión tiene sobre la iniciativa de prematricular desde una aplicación móvil?
4. ¿Cree que la aplicación móvil vendría a beneficiar el proceso de matrícula en el instituto?
5. ¿Cree que el instituto Oxford School está preparado para implementar el uso de una aplicación móvil para prematricular?



9.2 Encuesta

Encuesta de opinión realizada en Oxford International School

Nombre _____ Cargo _____

Teléfono _____

Marque con un x su respuesta:

1. ¿Qué le pareció la interfaz móvil de pre matricula?

Excelente Bueno Regular Malo

2. ¿La aplicación móvil de pre matricula cumple con las necesidades del instituto para realizar dicha pre-matricula?

Si No

3. ¿La aplicación móvil le dificulta su uso al momento de registrar una pre matricula?

Si No

4. ¿La aplicación móvil le facilita la matricula posterior de un alumno?

Si No

5. ¿Qué le pareció la consulta que puede hacer a través de la aplicación móvil?

Bueno Regular Malo

6. ¿Qué otra Funcionalidad le gustaría que realizara la aplicación móvil para el instituto?

¡Gracias por su participación



9.3 Fotos prueba de la aplicación





INDICE

INDICE

RESUMEN	5
I.INTRODUCCION	6
II. JUSTIFICACION.....	7
III.OBJETIVO.....	8
3.1 Objetivo general.....	8
3.2 Objetivos específicos.....	8
IV. MARCO TEORICO.....	9
4.1 Dispositivos móviles	9
4.1.1 Generación de dispositivos	9
4.1.2 Tipos de Dispositivos móviles.....	11
4.2 Sistemas operativos para Móviles.....	17
4.2.1 Symbian.....	17
4.2.2 Windows Mobile	17
4.2.4 Android.....	18
4.3 Sistema operativo Android.....	19
4.3.1 Arquitectura Android	20
4.3.4 Estructura de un proyecto en Android.....	21
4.4 Programación de aplicaciones	22
4.4.1 lenguajes de programación	23
4.4.2 Eclipse.....	23
4.4.3 Lenguaje de programación JAVA.....	24
4.5 Aplicación móvil.	25
4.5.1 Componentes de una aplicación en Android	25
4.6 Análisis de Sistemas	27
4.7 Sistema de Información	30
4.7.1 Tipos de sistema de información	30
4.8 Ingeniería de Sistemas	32
4.8.1 Modelo de desarrollo de sistemas	33
4.9 Base de datos	37
4.10 Proceso de Normalización.....	39
4.10.1 Formas normales.....	40



4.11	Conectividad a la Base de Datos	41
4.12	Gestor de base de datos SQL	42
4.12.1	Partes de SQL Server	42
4.12.2	Porque utilizar SQL como gestor de base de datos.....	43
4.13	Lenguaje de Modelado Unificado (UML)	45
4.13.1	Diagramas de UML	47
4.14	Internet.....	49
4.14.1	Características del protocolo SSL	50
4.14.2	Funcionamiento del SSL	50
4.15	Factibilidad	51
4.15.1	Factibilidad Técnica	51
4.15.2	Factibilidad económica.....	52
4.15.3	Factibilidad operativa.....	52
4.16	OXFORD international School	52
V.	Diseño metodológico	53
5.1	Tipo de investigación (tipo de estudio)	53
5.2	Población y muestra	53
5.3	Técnicas de recolección de datos.....	53
5.4	Modelo de desarrollo	54
VI.	Análisis de resultados.....	56
6.1	Encuestas previas al desarrollo de la aplicación	56
6.2	Encuestas realizadas durante el periodo de pruebas de la app móvil.....	57
6.3	Proceso de pre matricula	59
6.4	Herramientas de desarrollo	59
6.5	Descripción de la aplicación	60
6.6	Estudio de factibilidad.....	60
6.6.1	Factibilidad técnica.....	60
6.6.2	Factibilidad Económica.....	61
6.6.3	Factibilidad Operacional.....	62
6.7	Diagrama entidad-relación	63
6.8	Diagrama de casos de usos	64
6.9	Diagrama de secuencia	65
6.10	Diccionario de datos.....	68



6.11 Vistas de la aplicación	70
6.12 Pruebas Realizadas	76
VII. Conclusiones	77
VIII.Recomendaciones.....	78
X. Bibliografía	79
IX.Anexos.....	81