



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“Aplicación Móvil de Solicitud de Cotizaciones”

INFORME DE PROYECTO DE GRADUACIÓN

Previo a la obtención del título de:

INGENIERO EN CIENCIAS COMPUTACIONALES

ORIENTACIÓN SISTEMAS MULTIMEDIA

Presentado por:

Ricardo Antonio Márquez Enríquez

Guayaquil – Ecuador

2013

AGRADECIMIENTO

A dios por darme fuerza y discernimiento para poder terminar con éxito este proyecto y haberme ayudado a través de toda mi formación universitaria, a mis padres por haberme dado el soporte necesario para culminar mis estudios, a mis profesores por darme su paciencia en el momento de enseñarme cada una de las materias, a mi director de tesis por haberme guiado tan bien durante todo el proceso y a mis amigos y familiares por haberme ayudado con sus experiencias, ánimos y guías durante todo el proceso de mi tesis.

Ricardo Márquez Enríquez

DEDICATORIA

A mi Dios por darme sabiduría y su infinito amor en toda mi vida, a mis padres por todo lo que me han enseñado y han logrado hacerme la persona que soy, a mi hermana que siempre ha estado ahí para cuando la necesito y también me ha guiado, a mis amigos que aunque algunos se han apartado igual supieron llenar esa necesidad y me han apoyado en su momento y por último a mi familia que siempre me dieron fuerzas y consejos para culminar mis estudios.


Ricardo Márquez Enríquez

TRIBUNAL DE SUSTENTACIÓN



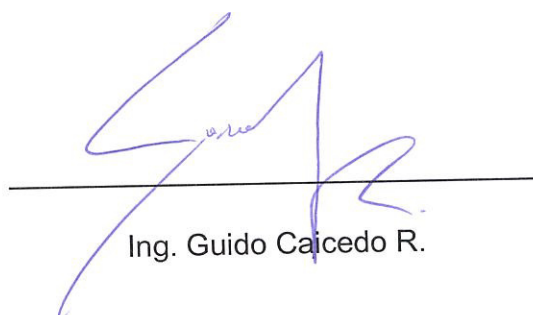
Ing. Boris Vintimilla

SUBDECANO DE LA FIEC



Ing. Xavier Ochoa Ch.

DIRECTOR DE PROYECTO DE
GRADUACIÓN



Ing. Guido Caicedo R.

MIEMBRO PRINCIPAL



DECLARACIÓN EXPRESA

La responsabilidad del contenido de este Informe, me
corresponde exclusivamente; y el patrimonio intelectual
de la misma a la Escuela Superior Politécnica del
Litoral



Ricardo Antonio Márquez Enríquez



RESUMEN

Actualmente las personas tienen que ir directamente a un local de computación para poder cotizar y comprar sus suministros de computación. Otra opción es la de poder llamar por teléfono a un número de la misma para saber si existe un producto y cotizarlo, para poder después comprarlo.

Pero estas dos opciones implican que el cliente tenga que moverse del lugar que se encuentra (el cual genera una incomodidad del mismo), no adhiere una forma de publicidad al local y tampoco da una forma interactiva y actual de acceso al negocio de computación.

Este Proyecto se basa en resolver estos problemas y para ello está dividido en cinco partes las cuales son:

Levantamiento de requerimientos, los cuales nos facilitará el entender las necesidades de nuestros clientes y así poder armar de forma eficiente el proyecto y mostrárselo a nuestra empresa cliente.

Análisis y diseño, tras haber recibido los requerimientos, se procede a hacer un diseño físico y lógico de nuestro programa el cual nos ayudará a realizarlo de una forma más rápida y entendible para nosotros y nuestro cliente.

Implementación, aquí se hará efectivo nuestro proyecto y se logrará que ya sea visual para nuestro cliente, esta es la parte más importante del proyecto ya que es toda la programación.

Pruebas, se hará uso de una cantidad de personas para que nos ayuden a ver si es de utilidad y hemos hecho bien nuestra interfaz, dándonos como resultado de las mismas una retroalimentación para poder mejorar nuestro programa.

Por último se realizará el respectivo capítulo de conclusiones y recomendaciones, el cual dejará impreso toda la vida del proyecto y consejos

a futuros programadores para que no caigan en problemas que se vayan a presentar en este proyecto.

ÍNDICE GENERAL

RESUMEN	VI
ÍNDICE GENERAL	IX
ÍNDICE DE FIGURAS	XII
ÍNDICE DE TABLAS	XIV
INTRODUCCIÓN	XV
CAPÍTULO I. ANÁLISIS DEL PROBLEMA	1
1.1. Descripción del problema	1
1.2. Justificación	3
1.3. Soluciones existentes	4
1.4. Objetivos	7
1.4.1. Objetivo principal	7
1.4.2. Objetivos específicos	8

1.5. Herramientas a utilizarse	9
CAPÍTULO II. ANÁLISIS DE LA SOLUCIÓN	15
2.1. Levantamiento de Requerimientos	15
2.2. Requerimientos funcionales	18
2.3. Requerimientos no funcionales	20
2.4. Casos de uso	21
2.4.1. Cliente	23
2.4.2. Usuario	28
2.5. Arquitectura de la solución	31
2.6. Alcance	32
CAPÍTULO III. DISEÑO	34
3.1. Diseño lógico	34
3.2. Diseño físico	41
3.3. Diseño de pruebas	44
3.3.1. Métricas	45
3.3.2. Diseño de cuestionario	46

CAPÍTULO IV.	IMPLEMENTACIÓN	52
4.1	Plataformas	52
4.2	Problemas al implementar	56
CAPÍTULO V.	PRUEBAS	60
5.1.	Pruebas de funcionalidad	60
5.2.	Pruebas de usabilidad	62
5.3.	Pruebas de bajo nivel	64
5.4.	Pruebas de alto nivel	68
5.5.	Análisis de resultados	73
CONCLUSIONES Y RECOMENDACIONES		74
BIBLIOGRAFÍA		77

ÍNDICE DE FIGURAS

Figura 1.1	Computers High Tech Store	5
Figura 1.2	Compu tienda Internacional	6
Figura 1.3	Computer Store	6
Figura 2.1	Casos de uso cliente	23
Figura 2.2	Casos de uso usuario	28
Figura 2.3	Arquitectura de la solución	31
Figura 3.1	Diagrama de secuencias de Cotización	35
Figura 3.2	Diagrama de secuencias de Productos	36
Figura 3.3	Diagrama de secuencias de Lista de cotización	38
Figura 3.4	Diagrama de secuencias de Servicios	39
Figura 3.5	Diagrama de entidad relación de solicitud de cotizaciones	40
Figura 3.6	Diseño físico computadores separados	43
Figura 3.7	Diseño físico un solo computador	44

Figura 5.1 Pruebas de bajo nivel64

ÍNDICE DE TABLAS

Tabla #1.	Pruebas de funcionalidad	61
Tabla #2.	Pruebas de bajo nivel	66
Tabla #3.	Pruebas de alto nivel	70

INTRODUCCIÓN

Una de los requerimientos actuales del ser humano es tener los suministros para sus computadoras. Este requerimiento se suple a través de la compra de las mismas en locales de computación.

Tiempo atrás las personas tenían que ir a las empresas computacionales de forma presencial para poder realizar el pedido de sus suministros. Pasó el tiempo y se integró la opción de poder enviar los productos a las casas o empresas de las personas gracias a las llamadas telefónicas que estos realizaban a la empresa.

Otra opción para poder realizar la cotización y pedido de suministros es por vía web, pero esta opción no le ofrece al cliente una forma de movilidad al no hacerlo de forma fácil en un dispositivo móvil por el tamaño del mismo.

Mi interés en realizar una aplicación de solicitud de cotización para empresas de suministros de computación, es el de:

1. Ayudar a personas a ver el inventario existente de la empresa.

2. Ayudar a personas que no pueden trasladarse, ni tienen los medios para ir físicamente a la empresa a hacer sus cotizaciones desde sus móviles.
3. Ayudar a las personas a encontrar a la empresa.
4. Mejorar la comunicación entre la empresa computacional y sus clientes.

Los puntos más importantes a destacarse sobre la aplicación son:

- Envío de solicitud de cotización de suministros de computación.
- Ver los productos existentes en la empresa, las cantidades y descuentos.
- La aplicación se realizará sobre plataformas móviles para facilitar el acceso a la misma.

CAPÍTULO 1

ANÁLISIS DEL PROBLEMA

1.1. Descripción del problema

Muchas veces cuando hemos necesitado un suministro de computación en concreto nos ha tocado hasta hace poco el ir directamente a la empresa. Mucha gente se siente cansada, ofuscada, enojada por el simple hecho de tener que acudir a la empresa, a lo cual no tienen otra opción ya que si no

tienen determinado producto como una tinta no podrán trabajar así que tendrán que ir.

Claro está, actualmente las personas tienen que ir directamente a un local de computación para poder buscar y comprar sus suministros computacionales pero existe otra opción la cual es la de poder llamar por teléfono a un número de la misma, para que uno de los motorizados le envíe el producto hacia el lugar donde se encuentre, pero esta solución solo la tienen ciertas empresas, no todas y a veces no le prestan el servicio por no estar cercana a ellos.

Pero estas dos opciones implican que el cliente tenga que moverse del lugar que se encuentra (el cual genera una incomodidad del mismo), no adhiere una forma de publicidad al local y tampoco da una forma interactiva y actual de acceso al negocio de computación.

1.2. Justificación

Para resolver los problemas descritos en la sección anterior, crearemos una aplicación que ofrezca una opción diferente y de forma gratuita para clientes con dispositivos móviles, que le permita realizar una solicitud de cotización desde la comodidad del lugar donde se encuentre y permitiéndole además poder ver el stock de la empresa y si este cuenta con algún descuento.

De esta manera lograremos tener más precisión al momento de realizar los pedidos y tener un acercamiento más sólido entre usuario y empresa, dando como resultado una ganancia para ambas partes, logrando así una aceleración del proceso de pedido sin tener que hacer una llamada para consultar si hay stock y al realizar el pedido explícito reducirá el tiempo que necesite el usuario para hacer la transacción.

La aplicación contará con una interfaz amigable o de fácil uso, con un uso correcto de la interacción Hombre - Máquina, divertida, que impulse a utilizarla de nuevo. Podrá ver la existencia actual del producto y si existe algún descuento disponible al comprarlo en mayor cantidad. Podrá pedir la cotización de los productos que necesita desde el lugar donde esté. Podrá al

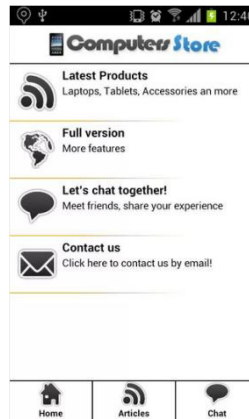
ver el producto agregarlo al carrito, para al momento de pedir cotización no tenga que poner el producto que necesita. Podrá ver una lista de las últimas cotizaciones y verlas. Podrá ver que servicios y en que horarios tiene la empresa para ofrecer a sus clientes. Por último podrá ver la ubicación de la empresa en un mapa.

1.3. Soluciones existentes

En la actualidad, recientemente han aparecido en el mercado aplicaciones móviles para empresas computacionales, pero en su mayoría han sido mas como tiendas virtuales, mostrando que productos podría comprar el cliente, no se han encontrado aplicaciones que resuelvan todos los problemas descritos en las secciones anteriores por eso aquí les traigo las únicas opciones que he encontrado en el play store de Android y que han sido afines al proyecto.

- Computers | High Tech Store, la cual es una aplicación que permite ver los productos que ofrece la empresa computacional, compartir la aplicación y contactarse con la empresa.

Figura 1.1 Computers | High Tech Store



Fuente: Google Play

Elaborado por: Tech Dev

- Computienda Internacional, es una aplicación que permite buscar productos mediante código de barras, catalogo en línea, cotización y pedido vía twitter, herramienta para armar la computadora al gusto y ubicar a la empresa.

Figura 1.2 Computiendia Internacional



Fuente: Google Play

Elaborado por: Diego Amador Guerrero

- Computer Store, permite buscar, comprar, comparar precios de productos desde el dispositivo móvil.

Figura 1.3 Computer Store



Fuente: Google Play

Elaborado por: kingjushiung

Como hemos visto la mayoría solo tiene el servicio para ver los productos que ofrecen las empresas. Si bien Computienda Internacional se acerca a la solución, no presenta las opciones de servicios, ver las últimas cotizaciones y poder compartir la aplicación.

Cada uno de nosotros ha aportado con ideas sustanciales para la creación de nuestro proyecto, algunas hasta con ideas para las metáforas, colores, nos ha servido de ejemplo algunas de ellas ya que sirven mucho para mejorar la interacción hombre máquina y no gastar la visión de los usuarios al usar nuestra aplicación.

1.4. Objetivos

1.4.1. Objetivo principal

Crear una aplicación móvil que acerque los servicios que ofrece la empresa de suministros de computación a los clientes de la misma que tengan dispositivos móviles inteligentes con sistema operativo Android.

1.4.2. Objetivos específicos

- Programar un módulo para la aplicación que pueda pedir la cotización de productos (tintas, impresoras, etc.) seleccionando los productos que necesite y que sea enviado a un sistema.
- Programar un módulo para la aplicación que pueda buscar los productos y ver si hay existencia del mismo en la empresa y sus descuentos, agregar el producto al carrito de compras para que salga en la solicitud de cotización.
- Programar un módulo para la aplicación que pueda buscar las cotizaciones que le han realizado al cliente y este pueda ver las mismas.
- Programar un módulo para la aplicación en donde se pueda ver los servicios que actualmente tiene la empresa para ofrecer a sus clientes.

- Programar un módulo para la aplicación que pueda dar una vía de contacto directo por medio de celular y mail de parte de la empresa para los clientes y permita ver la ubicación de la empresa en un mapa.
- Programar un módulo para la aplicación que pueda compartirse con familia y amigos de los clientes.

1.5. Herramientas a utilizarse

En el mercado se encuentran diferentes tipos de plataformas en las que se puede hacer este proyecto las cuales son Blackberry, IOS, Windows Phone, Symbian y Android, cada uno tiene su pedazo de mercado, pero los más utilizados son Blackberry, IOS y Android. La herramienta principal que se ha decidido usar es la Plataforma Android ya que esta plataforma es gratuita y está en la mayoría de dispositivos móviles que se encuentran en el mundo entero según el artículo escrito por la PcWorld el 14 de febrero de este año.¹

¹ *PcWorld*, Android e IOS se disputan el mercado de smartphones este año,

Fecha de acceso el 11 de septiembre de 2013,

Además, cuenta con las opciones necesarias en su SDK para poder cumplir a cabalidad con la aplicación propuesta aquí.

Java²

Es un lenguaje de alto nivel orientado a objetos el cual fue desarrollado por James Gosling en 1995 y toma mucho de su sintaxis de C, Cobol y Visual Basic. Cuenta con un modelo de objetos más sencillo y elimina las herramientas de bajo nivel, con lo cual ayuda a que no hayan tantos errores. La memoria es manipulada por medio de un "recolector de basura".

<http://www.pcworld.com.ec/android-e-ios-se-disputan-el-mercado-de-smartphones-este-ano/>

² *Wikipedia*, Java (lenguaje de programación), Fecha de acceso el 12 de septiembre de 2013,

[http://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)](http://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))

El diseño de Java, su robustez, su fácil portabilidad y el respaldo de la industria han logrado que Java tenga un crecimiento casi exponencial y sea usado cada vez en más campos de la Informática.

PHP³

Fue diseñado para el desarrollo de páginas web de contenido dinámico en 1995 por Rasmus Lerdorf. Fue uno de los primeros lenguajes de programación del lado del servidor que se podía integrar directamente al HTML sin tener que llamar a un archivo externo para que procese los datos.

³ *Wikipedia*, PHP, Fecha de acceso el 12 de septiembre de 2013,

<http://es.wikipedia.org/wiki/PHP>

Eclipse⁴

Es un entorno de desarrollo Software de código abierto y es multiplataforma para poder realizar lo que se conoce como "Aplicaciones de Cliente Enriquecido". Típicamente es utilizado como IDE (entorno de desarrollo integrados) conjuntamente con el JDT (Java Development Toolkit) y el compilador ECJ los cuales vienen con Eclipse aunque también se puede usar para BitTorrent o Azureus que son otros tipos de aplicaciones clientes.

Sql⁵

Es un lenguaje declarativo con acceso a bases de datos relacionadas que permite especificar varias operaciones en dichas bases de datos, tales como Creación, Modificación, Consulta y Eliminación de tablas o datos. Es un

⁴ *Wikipedia*, Eclipse (software), Fecha de acceso el 12 de septiembre de 2013, [http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software))

⁵ *Wikipedia*, SQL, Fecha de acceso el 12 de septiembre de 2013, <http://es.wikipedia.org/wiki/SQL>, *Wikipedia*, SQL Server, Fecha de acceso el 12 de septiembre de 2013, http://es.wikipedia.org/wiki/SQL_Server

lenguaje que logra explotar la flexibilidad y la potencia de los sistemas relacionales y con pequeñas sentencias logra conseguir cambios que con un lenguaje de bajo nivel se necesitarían varios programas.

Conjuntamente con SQL Server que es un sistema de gestión de base de datos basado en el modelo relacional, se logra tener una sólida base de datos con mucha estabilidad y opciones para el manejo de bases de datos.

Android SDK⁶

Es un conjunto de librerías API y herramientas de desarrollo necesarias para construir, probar y compilar aplicaciones para Android. Herramienta fundamental para comenzar a crear aplicaciones móviles para todos los dispositivos con sistema operativo Android.

⁶ *Android developer*, Fecha de acceso el 12 de septiembre de 2013,

<http://developer.android.com>

Android Emulator⁷

En el Android SDK incluye el emulador de dispositivo móvil el cual permite hacer pruebas de las aplicaciones que se ha creado en el IDE sin tener que utilizar un dispositivo real.

Esta herramienta, nos permite también ajustar al tamaño de dispositivo que uno quiere, entre toda la gama de tamaños de pantalla, facilita de gran manera la programación para dispositivos móviles Android.

⁷ *Android developer*, Fecha de acceso el 12 de septiembre de 2013,

<http://developer.android.com>

CAPÍTULO 2

ANÁLISIS DE LA SOLUCIÓN

2.1. Levantamiento de requerimientos⁸

Para cualquier proyecto de programación siempre es necesario realizar un levantamiento de requerimientos, para esto es necesario entablar conversaciones tanto con la empresa que nos contrata como con los clientes de la misma, para saber las necesidades de ambos lados y así poder ofrecer

⁸ Ian Sommerville. Ingeniería del software (7ma. edición). Madrid, España: Pearson. (2005). Capítulo 6, pág. 109-115.

una solución adecuada para que mejore el servicio de la empresa para con sus clientes.

Lo primero fue hablar con la empresa, los cuales nos dijeron que querían tener un contacto más cercano con sus clientes, que puedan saber que productos tenían en stock y si los mismos tenían descuentos, que puedan encontrar su local y que puedan realizar solicitud de cotizaciones remotas, ya que así aumentarían sus ventas y una conexión más filial con ellos.

Se hicieron varias reuniones que a la final nos llevaron a tener un diseño base de lo que podría contener nuestra aplicación. Cada paso que realizábamos nos tocó cambiar cosas que ya habíamos decidido, para que sea más útil para el cliente. El fin era llegar a tener una buena aplicación con varias utilidades y que logre que el cliente decida utilizarla de nuevo.

La parte de hablar con la empresa es la más difícil de todas ya que es un tiempo de desacuerdos e inconsistencias, como se sabe la empresa siempre pide cosas que no se pueden realizar y nos toca a los programadores explicarles que límites se tiene para la creación de la aplicación, otro factor que es muy importante al decidir que se hace o no, es cuánto va a costar

todo lo que la empresa nos pide, lo cual todo el tiempo hace que la empresa desista en hacer una cosa u otra.

Cuando se tuvo una aplicación base, se decidió acercarse a los clientes de la empresa, cuya opinión era de suma importancia para nosotros ya que por más que la empresa fuera la que nos contrató y nos paga, ellos serían los que iban a usarla, por ende hicimos varias encuestas mostrándoles el modelo base y preguntándoles a ver si les parecía útil y si lo usarían constantemente.

La toma de requerimientos es una parte fundamental de todo proyecto de programación y la misma requiere que sea constante mientras dure el proyecto, como la implementación que voy a usar es una tecnología ágil se va a hacer todo antes de volver a la empresa y así volver a enfatizar más en la cuestión de toma de requerimientos para que no se demore tanto en esta parte solamente y así poder avanzar más rápidamente.

2.2. Requerimientos funcionales

Estos requerimientos se refieren a cómo debe comportarse la aplicación ante diferentes entradas y como debe funcionar en situaciones específicas y dar un servicio determinado. En ciertos casos estos requerimientos también son conocidos por mostrar que es lo que el programa no debería hacer.

Los requerimientos funcionales son en sí cómo funcionará nuestro programa, como piensa nuestro cliente que debería hacerlo, pero esto es un tanto abstracto ya que es como funciona, entradas y salidas, excepciones, etc. depende mucho de que programa se va a hacer y para qué tipo de personas está orientado.

Los requerimientos funcionales identificados para este proyecto son:

1. El cliente debe poder tener un mapa adecuado para buscar la ubicación de la empresa.

2. Podrá visualizar los productos que hay en stock y ver si tienen descuentos.
3. En el caso de no poder cargar una imagen de un producto, mostrar un mensaje que sirva de retroalimentación al cliente.
4. Al hacer la configuración inicial el cliente solo va a poder ingresar datos validos en los campos, por ejemplo, si es numérico solo podrá poner números.
5. Solo se podrá llenar la parte de los productos en las solicitudes de cotización, con los productos vistos y añadidos al carrito de productos.
6. El sistema cada vez que cargue datos de la base de datos, que se encuentra en la nube como los productos o cotizaciones, debe mostrar un mensaje de que está cargando.

2.3. Requerimientos no funcionales⁹

Estos requerimientos como su propio nombre lo dice no se refieren a las funciones del programa en sí. Más se enfocan en la parte de fiabilidad, velocidad de respuesta o almacenamiento del programa, lo cual hace que estos requerimientos sean más críticos para la usabilidad del programa.

En nuestro caso se tomaron ciertos requerimientos no funcionales, los cuales no son muchos ya que no es un sistema crítico el que estamos haciendo aquí, y se pidieron estos requerimientos mas es para darle un poco de confort al usuario.

1. El tiempo de la primera pantalla de carga no debe superar los 3 segundos.

⁹ Bernd Bruegge, Allen H. Dutoit. Object Oriented Software Engineering (3ra. edición). Estados Unidos: Prentice Hall. (2000). Capítulo 4, pág. 100–106, 118-119.

2. En la parte de contáctenos, el sistema debe de inmediato direccionarle a la opción que haya elegido el cliente, sea vía teléfono o mail sin demora alguna.

3. Debe haber un Standard en los botones de todas las pantallas.

4. Debe haber un Standard en los mensajes de errores y notificaciones en todo el sistema.

Los requerimientos no funcionales surgen gracias a las necesidades de los usuarios en una respuesta pronta del programa, a la necesidad de interoperabilidad entre software y hardware del dispositivo. Como se dijo con anterioridad al no ser un sistema crítico no fueron necesarios muchos requerimientos no funcionales

2.4. Casos de uso

Los casos de uso nos sirven para poder especificar la comunicación y la interacción que hay en el sistema con los usuarios u otros sistemas. Para

poder entender mejor como realizar nuestra aplicación debemos realizar estos casos de uso para basarnos en ellos y crear una aplicación más fuerte.

Los casos de uso son creados con la ayuda de los requerimientos ya que estos muestran lo que nos han pedido ahí. Muestran cómo reacciona a eventos el sistema, como realizaría las cosas para poder satisfacer los requerimientos.¹⁰

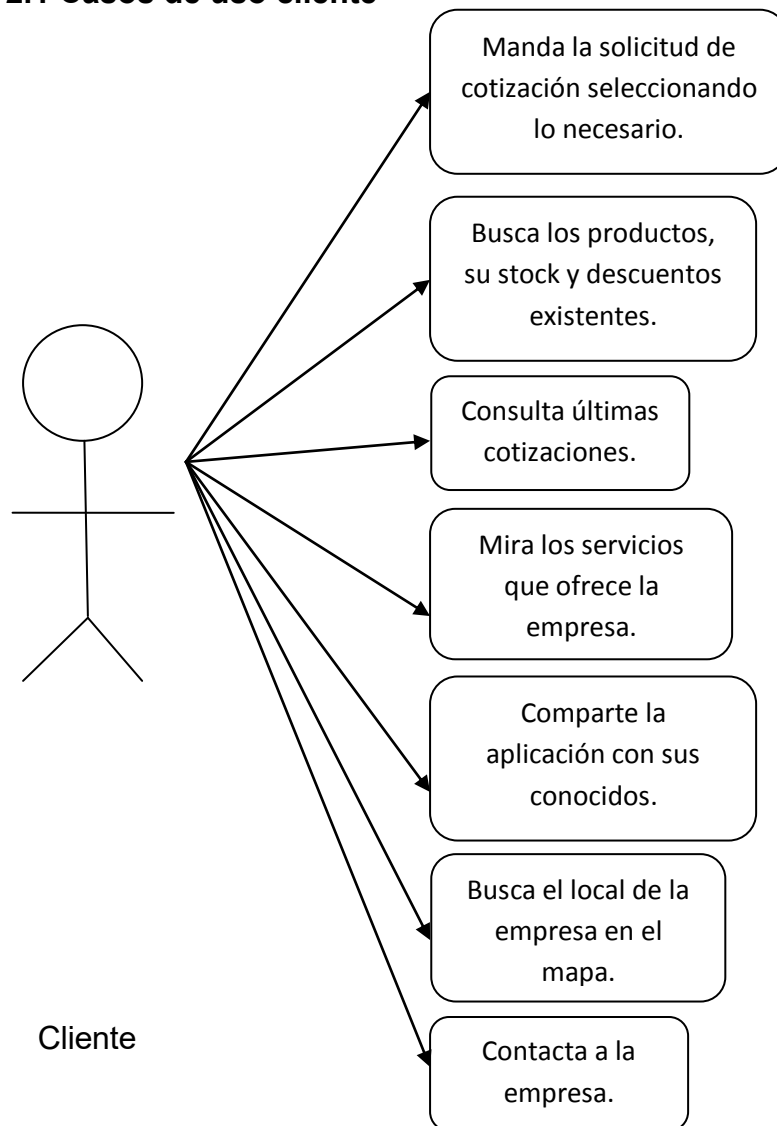
En el sistema se tiene dos actores: cliente y usuario. Puede parecer un tanto confuso el uso de estos dos términos pero aquí no se ve a los actores desde el punto de vista del desarrollador sino el de nuestro cliente que es la empresa computacional. El cliente es el que va a utilizar la aplicación móvil que la empresa ofrece, el que hará las solicitudes de cotización o buscará las cotizaciones que ha recibido, mientras que el usuario es la empresa computacional que usará esta aplicación desde la nube para atender las solicitudes de sus clientes y cambiar los datos que se encuentren en la base de datos.

¹⁰ *Wikipedia*, Caso de uso, Fecha de acceso el 15 de abril de 2013,

http://es.wikipedia.org/wiki/Caso_de_uso

2.4.1. Cliente

Figura 2.1 Casos de uso cliente



Fuente: Propio

Elaborado por: Ricardo Márquez

Caso de uso: Manda la solicitud de cotización seleccionando lo necesario.

El cliente va a poder hacer una solicitud de cotización hacia la empresa enviándole a su sistema. El cliente va a tener que poner su número de cédula de identidad o RUC y poner un correo electrónico al cual se lo pueda contactar para que así la empresa pueda enviarle la cotización de lo solicitado. Además de esto, para que se pueda enviar la solicitud de cotización el cliente va a tener que seleccionar los productos u objetos varios que formarán parte de la cotización.

Caso de uso: Busca los productos, su stock y descuentos existentes.

Aquí el cliente va a poder ver primero las marcas de los productos que tiene la empresa para ofrecer.

Al escoger una de las marcas, aparecerán todos los tipos de productos de esa marca. Posteriormente, aparecerán la lista de productos de esa marca y ese tipo seleccionado.

Al escoger uno de los productos, se le presentará al cliente una pantalla donde podrá ver la cantidad en el stock del mismo y si tiene algún descuento. También podrá ver una imagen del producto, para la confirmación visual. Por último, podrá añadir al carrito de productos al producto visualizado para la solicitud de cotización.

Caso de uso: Consulta últimas cotizaciones.

Aquí el cliente va a poder ver una lista de las últimas cotizaciones que la empresa le ha hecho. El cliente va a tener que digitar su número de cédula o RUC para acceder al listado. Al finalizar, el cliente podrá visualizar dichas solicitudes y cotizaciones.

Caso de uso: Mira los servicios que ofrece la empresa.

Al cliente se le presentará los servicios existentes en la empresa computacional, después de lo cual al seleccionar una de ellas, le permitirá al

cliente observar toda la información exacta sobre el servicio que ofrece la empresa.

Caso de uso: Comparte la aplicación con sus conocidos.

El cliente va a poder enviar vía mail u otro programa que tenga instalado que le permita el envío de mensajes la invitación a todos sus conocidos, amigos y familiares para que puedan acceder a la aplicación con un link que los direcciona al instalador de la aplicación y así también que ellos se beneficien con este programa.

Caso de uso: Busca el local de la empresa en el mapa.

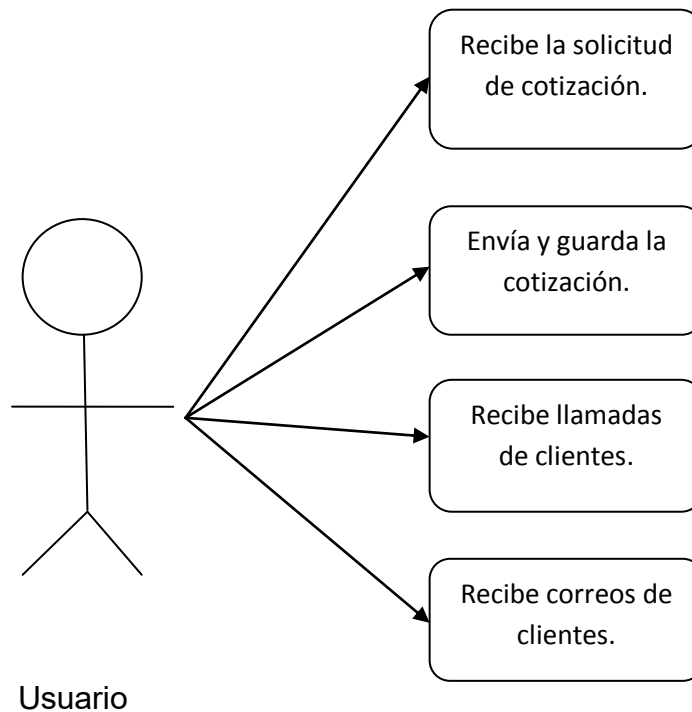
Aquí el cliente va a poder ver la ubicación exacta de la empresa computacional por medio de un mapa. El cliente encontrará esta opción en contáctenos y podrá manipular el mapa para ubicarse de mejor manera.

Caso de uso: Contacta a la empresa.

El cliente va a poder contactarse con la empresa computacional y así estar más cerca de la misma y poder hacer las preguntas más frecuentes que suelen tener. Las formas para poderse contactar con la empresa computacional será por vía mail, call center o por medio de uno de los celulares que están a disposición del servicio al cliente de la empresa computacional.

2.4.2. Usuario

Figura 2.2 Casos de uso usuario



Fuente: Propio

Elaborado por: Ricardo Márquez

Caso de uso: Recibe la solicitud de cotización.

El usuario en la empresa, va a poder recibir la solicitud de cotización que ha sido detallada con los productos por el mismo cliente y teniendo de datos extras como la cédula o RUC y un correo electrónico al que el usuario se

pueda comunicar con el cliente para enviar la cotización, recibe todo esto a través de un sistema.

Caso de uso: Envía y guarda la cotización.

El usuario en la empresa, va a poder enviar la cotización solicitada por el cliente y guardarla en el servidor para que el cliente pueda consultar sus últimas cotizaciones desde su dispositivo móvil.

Caso de uso: Recibe llamadas de clientes.

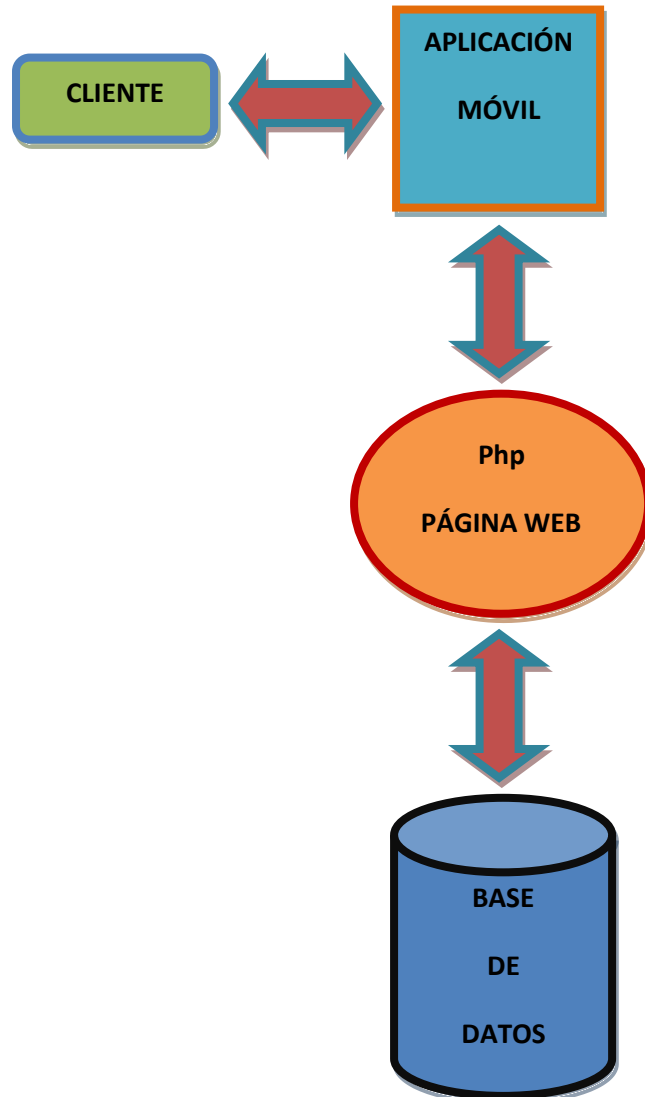
El usuario va a poder recibir en una de sus líneas las llamadas de sus clientes para poder dar información a sus clientes de forma más detallada y directa en base a las inquietudes que tengan los mismos, por el momento se cuenta con un número celular y uno estacional. Así de esta manera los clientes tendrán un acercamiento más directo con la empresa computacional y se logrará que el cliente sea más filial a la empresa computacional.

Caso de uso: Recibe correos de clientes.

El usuario va a poder recibir mails de los clientes de forma directa sobre cualquier inquietud o pregunta que tenga sobre la empresa computacional o sus servicios.

2.5. Arquitectura de la solución

Figura 2.3 Arquitectura de la solución



Fuente: Propia

Elaborado por: Ricardo Márquez

El cliente va a utilizar la aplicación móvil que se encuentra instalada debidamente en un dispositivo móvil Android y al momento de hacer uso de los productos o el listado de cotizaciones, la aplicación se conectará con la página web en formato php que se encuentra en la nube. Por último esta página web se conectará a la base de datos que se encuentra en el computador del usuario que en este caso es la empresa computacional, para después hacer el envío de los datos en sentido contrario, enviando a la web el dato requerido, para que aparezca en la aplicación móvil y así el usuario lo pueda ver.

2.6. Alcance

El alcance es todo lo que se va a realizar, hasta donde va a llegar nuestra aplicación, que va a tener y se puede poner hasta lo que no va a tener. El alcance para nuestro proyecto es:

- Envía una solicitud de cotización seleccionando los productos que necesita viéndolos en la sección de productos a los productos y añadiéndolos a un carrito de productos hacia un sistema.

- Muestra la ubicación exacta del local de la empresa computacional Compuexpress.
- Muestra los productos que la empresa tiene para que el cliente pueda ver si hay en stock y si tiene algún descuento.
- Permite compartir con otras personas la aplicación, enviando un mensaje con el link o url al instalador de la aplicación.
- Contacta a los clientes con la empresa computacional vía telefónica y vía mail, para que puedan informarse de cualquier cosa referente a la misma.
- Guarda los datos necesarios, como los productos y cotizaciones, para que la aplicación funcione correctamente en una base de datos.

CAPÍTULO 3

DISEÑO

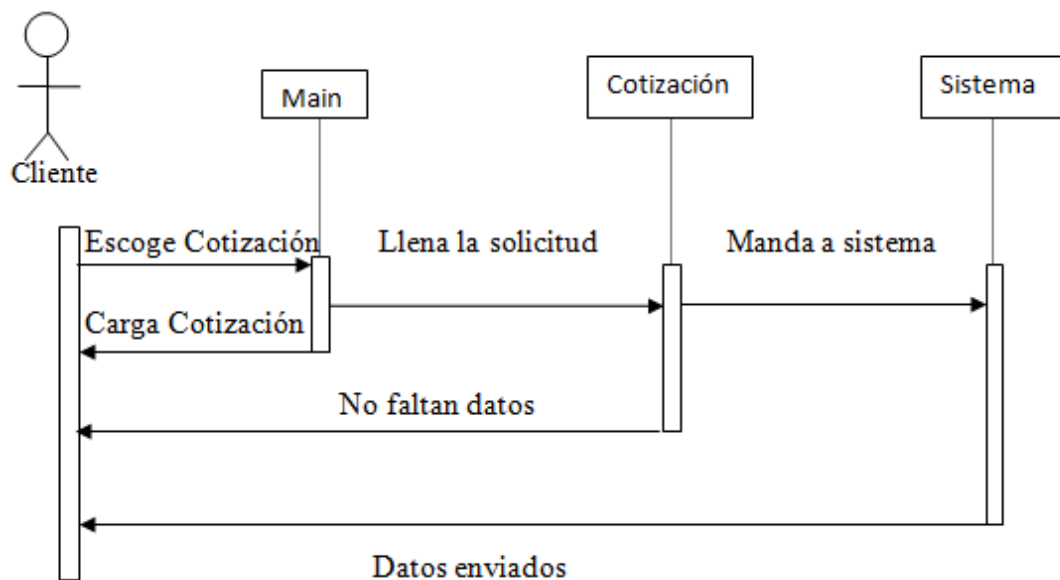
3.1. Diseño lógico¹¹

El diseño lógico nos sirve para poder armar la forma en que va a funcionar nuestro programa, nos ayuda a entender que datos se van a pasar de módulo a módulo, como se va a comportar dicho módulo y que nos va a mostrar como resultado. De esta forma podemos ayudarnos a programar con

¹¹ Ian Sommerville. Ingeniería del software (7ma. edición). Madrid, España: Pearson. (2005). Capítulos 11 - 16, pág. 217-354.

más facilidad, ya que así logramos entender cada módulo de una manera menos abstracta.

Figura 3.1 Diagrama de secuencias de Cotización



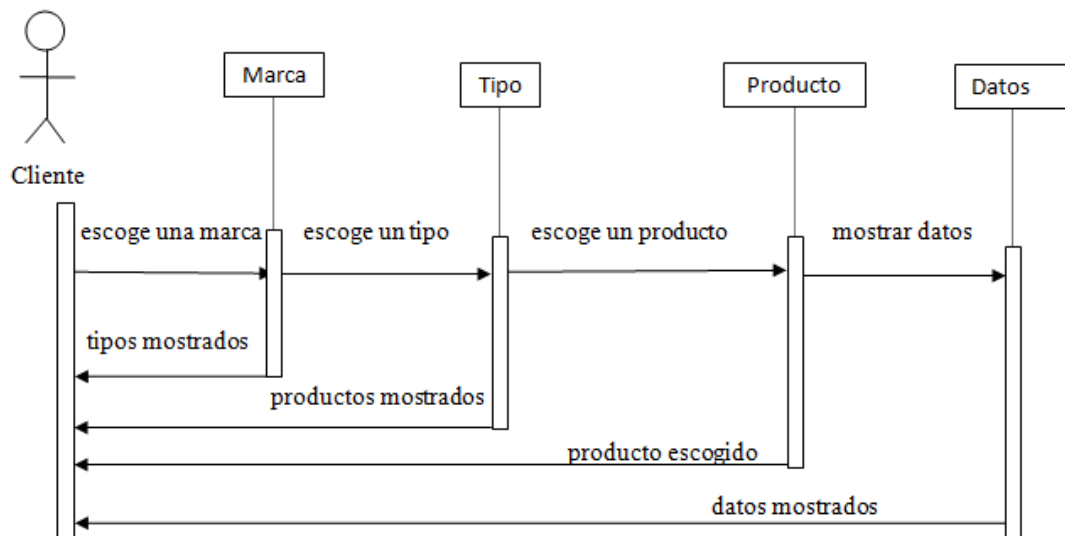
Fuente: Propio

Elaborado por: Ricardo Márquez

El cliente en el Main escoge la opción de Cotización y el sistema cargará la pantalla solicitada. Desde el Main se mandará una solicitud para que se abra el módulo de Cotización.

Después, el usuario llena la solicitud de cotización con sus campos y el sistema comprobará si no faltan datos. Esta validación se dará al intentar mandar la solicitud de cotización. Por último, se manda vía web los datos y la solicitud requerida llega a la empresa.

Figura 3.2 Diagrama de secuencias de Productos



Fuente: Propio

Elaborado por: Ricardo Márquez

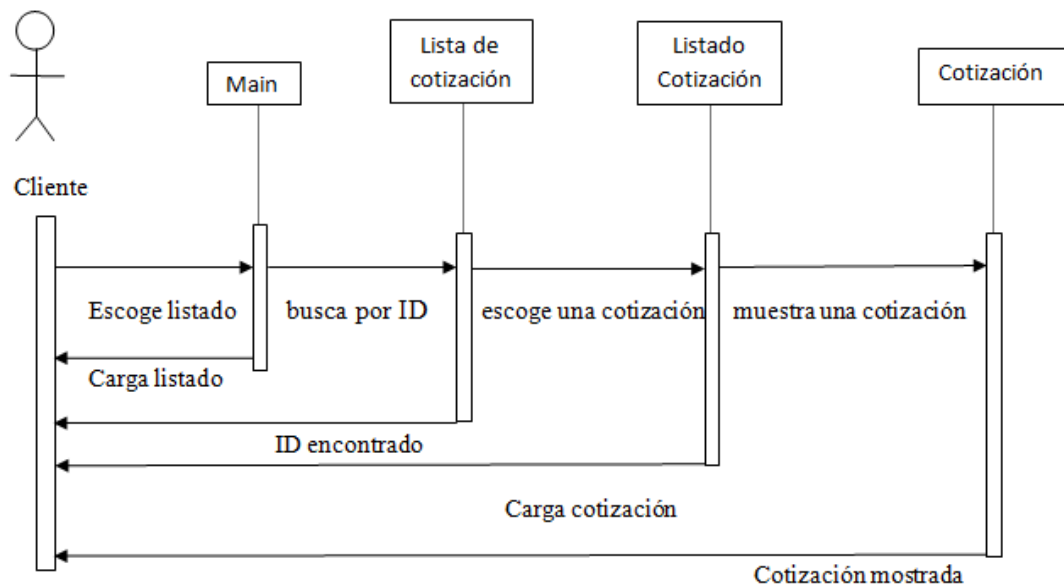
El cliente en el Main escoge la opción de Productos y el sistema cargará la pantalla solicitada. Desde el Main se mandará una solicitud para que se abra el módulo de Productos cargando desde la base las marcas existentes.

Después, el cliente escogerá una marca y le aparecerán los tipos de productos de esa marca que existan. Por último, el cliente escogerá un tipo y le aparecerán todos los productos asociados con esa marca y ese tipo de producto.

Para finalizar, este módulo manda la solicitud para que se abra el módulo de datos y el sistema cargará pantalla solicitada. El cliente podrá ver en esta pantalla la cantidad en stock del producto y si es que tiene descuentos, también tendrá una imagen mostrando el producto.

El cliente podrá agregar el producto al carrito de compras para después ser mandado a la pantalla de cotización.

Figura 3.3 Diagrama de secuencias de Lista de cotización

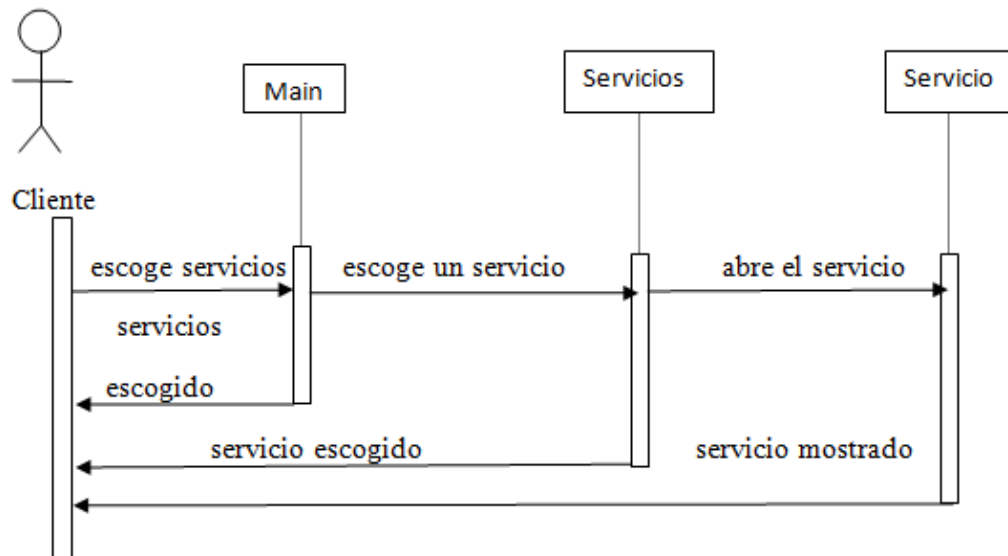


Fuente: Propio

Elaborado por: Ricardo Márquez

El cliente desde Main escogerá la opción de Lista de cotización, desde el Main se mandará una solicitud para que se abra el módulo de Lista de cotización. Ya estando en el módulo de listado el sistema buscará en la base de datos mandando a la nube la dirección url y esta le devolverá un String con todos los datos de la base requeridos en formato JSON y así llenará la lista de últimas cotizaciones. En caso de no existir el número de cédula o RUC, se enviará un mensaje de error al cliente.

Figura 3.4 Diagrama de secuencias de Servicios



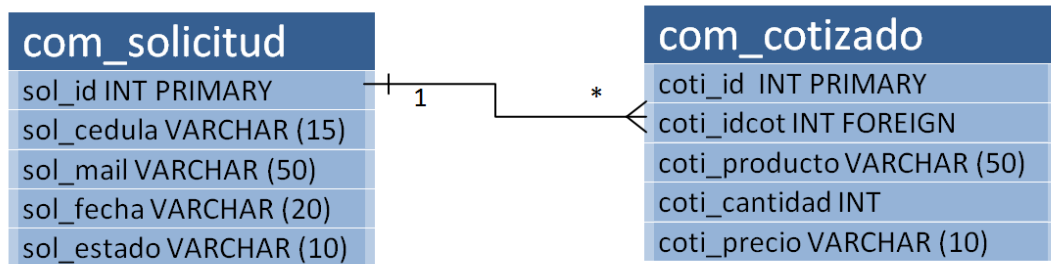
Fuente: Propio

Elaborado por: Ricardo Márquez

El cliente desde Main escoge la opción Servicios, desde el Main se mandará una solicitud para que se abra el módulo de Servicios. Ya estando en el módulo de Servicios se mostrará la lista de todos los servicios que la empresa ofrece.

Después de esto, el cliente escoge uno de los servicios y el sistema carga servicio deseado en una nueva pantalla.

Figura 3.5 Diagrama de entidad relación de solicitud de cotizaciones



Fuente: Propio

Elaborado por: Ricardo Márquez

Al momento de enviar una solicitud de cotización, se afecta a estas dos tablas en la base de datos. la solicitud tendrá a una clave primaria que se auto incrementa, cédula, mail y fecha, como default se pone el estado como Pendiente. Lo cotizado tendrá una clave primaria que se auto incrementa, clave que referencia a que solicitud pertenece, producto, cantidad y precio que por default es NULL hasta que la empresa le cotice.

La entidad relación nos muestra que en una solicitud puede haber uno o muchos productos cotizados y que cada producto cotizado solo tendrá una solicitud.

3.2. Diseño físico

Al momento de diseñar cualquier proyecto de software es necesario delimitar la parte física del proyecto, esto es, decidir que módulos van instalados en qué equipo, cuáles son las configuraciones necesarias para la utilización óptima de nuestro producto.

Vamos a contar de 3 partes fundamentales en el proyecto, estas son aplicación móvil, base de datos y el unificador de las otras dos. Las configuraciones para la utilización óptima de cada una son las siguientes:

1. Aplicación móvil, será necesario tan solo un celular inteligente o dispositivo móvil que lleve como sistema operativo mínimo Android 2.2 Froyo si el sistema operativo del celular fuera más antiguo que el especificado podría dar error ciertas funciones de la aplicación, ya que se lo ha realizado para el 2.2 en adelante. Se decidió realizarlo a partir de Froyo ya que la mayoría de la gente en el Ecuador todavía tiene celulares o dispositivos móviles con este sistema operativo y de esta manera se podrá abarcar una mayor cantidad de gente y prestarles el servicio que nuestro producto ofrece.

2. La base de datos y el unificador, tendrán que estar en un mismo computador o en una misma LAN ya que tienen que conectarse, en caso de que sea en el mismo computador, este debería ser lo suficientemente fuerte para soportar un gran flujo de búsquedas casi al mismo tiempo y obviamente tener acceso a internet. En este caso se debe contar con un server que pueda subir información a internet para que gente de todo el territorio ecuatoriano tenga acceso a la información de la base de datos, se recomienda que el computador sea un servidor con no menos de 8Gb de RAM, porque aunque no sea una base de datos extensa contiene una gran carga de posibles accesos.

3. En caso de que sean en computadores separados la base de datos y el unificador, las dos maquinas deben estar conectados en LAN, el unificador estará muy bien en una Core i5 con 4 Gb de RAM con acceso a internet y permiso de subir información a internet y a la base de datos con una configuración razonable para sostener un posible acceso de varias personas a la vez mientras que la base de datos debe estar en un computador Core i7 con 8Gb de RAM con conexión vía LAN al computador con la nube o unificador.

Estas son las configuraciones necesarias para un trabajo óptimo, así como también en qué equipos van instalados que partes de nuestro proyecto, de esta forma trazamos el diseño físico de donde va a ir cada cosa y qué se necesita para mantenerlo trabajando bien.

Figura 3.6 Diseño físico computadores separados



Fuente: Propio

Elaborado por: Ricardo Márquez

Figura 3.7 Diseño físico un solo computador



Fuente: Propio

Elaborado por: Ricardo Márquez

3.3. Diseño de pruebas

Este diseño se centra en las pruebas que se realizarán en el sistema para verificar entradas y salidas esperadas, las cuales darán como resultado el comprobar que el sistema cumple con los requerimientos levantados. Además, también se harán las pruebas que servirán para validar la interfaz y que esta sea amigable para el usuario.

Es necesario separar las pruebas de interfaz y las de funcionamiento ya que están orientadas a diferentes propósitos. Para las pruebas de funcionamiento hemos decidido realizar unas determinadas pruebas y para las pruebas de interfaz las pruebas de usabilidad, de alto y bajo nivel.

Estas pruebas nos ayudarán a conseguir un producto de calidad, una aplicación que sirva correctamente para satisfacer las necesidades tanto de nuestro usuario como el de los clientes.

En el momento de calificación se deberá ser estricto ya que cualquier falla podría hacer que el cliente deje de utilizar nuestra aplicación, ya que lo encontraría tedioso de utilizar por los fallos que se presenten seguidamente.

3.3.1. Métricas

Todo lo que es físico es susceptible a ser medido, sin embargo, un proyecto de software es algo intelectual, algo cuasi intangible por ese motivo es que muchas de las métricas del software se asemejan.

Las métricas de software se pueden separar en dos objetivos: valorar y estimar. El principio de valorar se encierra en calidad, fiabilidad y productividad. La estimación se centra en valores históricos de esfuerzo y tiempo que se utilizará para poder construir el proyecto y las características que se puedan dar como resultado final.

Se realizará una métrica de fiabilidad de tasa de ocurrencia de fallos (ROCOF), ya que habrán transacciones regulares del servicio y será necesario que funcione correctamente cuando se lo solicite, ya que de no hacerlo el sistema no funcionará correctamete y se espera lograr tener un ROCOF de 2/100, esto quiere decir que de 100 unidades de tiempo de funcionamiento 2 serán las que se puede aceptar que fallen en envíos de datos.

3.3.2. Diseño de cuestionario

Para realizar las pruebas que hemos puesto anteriormente se requiere hacer un cuestionario para poder ir anotando las irregularidades que se van encontrando en el camino de la evaluación.

Para las pruebas de funcionamiento, que nos sirve para verificar si funciona nuestra aplicación, usaremos el siguiente cuestionario:

1. Iniciar una búsqueda de las últimas cotizaciones recibidas por el cliente de la empresa.
2. Seleccionar una cotización de la lista y que muestre al mismo.
3. Seleccionar la opción de productos y que muestre todas las marcas que existan en la empresa al cliente.
4. Seleccionar una marca dada y que muestre los tipos de productos de esa marca.
5. Seleccionar un tipo dado y que muestre los productos de ese determinado tipo.

6. Seleccionar un producto dado y que me muestre la figura con los datos correspondientes al mismo.
7. Agregar el producto al carrito de productos y que el nombre del producto salga en la solicitud de cotización.
8. Agregar varios productos al carrito de productos y que los nombres de los productos salgan en la solicitud de cotización.
9. Hacer uso de solicitud de cotización y que llegue al sistema con los datos correctos.
10. Seleccionar la opción de Servicios y que muestre todos los servicios que ofrece la empresa.
11. Seleccionar uno de los servicios dados y que muestre los datos correspondientes al mismo.

12. Seleccionar la opción de compartir y que permita compartir el link de descarga correctamente.

13. Seleccionar la opción de Contáctenos y que funcione las opciones de llamar correctamente.

14. Seleccionar la opción de Contáctenos y que funcione la opción de correo electrónico correctamente.

15. Seleccionar la opción de Contáctenos y que funcione la opción de búsqueda de la dirección en el mapa, mostrando la misma en el mapa con los datos correspondientes.

Para las pruebas de usabilidad solo tomaremos en cuenta cuantas veces se ha equivocado el usuario en seguir los pasos que haremos en las pruebas de bajo y alto nivel, por ende no será necesario realizar un cuestionario para esta prueba.

Para las pruebas de interfaz de bajo y alto nivel, utilizaremos el mismo cuestionario, ya que se debe seguir el mismo camino y se prueba lo mismo solo que en diferente calidad. Por eso se utilizará el siguiente cuestionario:

1. Ir a configuraciones.
2. Configurar sus datos.
3. Ir a Solicitud de cotización.
4. Enviar una solicitud de cotización.
5. Regresar a Main.
6. Ir a Productos.
7. Seleccionar una marca.
8. Seleccionar un tipo.
9. Seleccionar un producto.
10. Mostrar el producto.
11. Agregar al carrito de productos.
12. Pedir cotización del producto.
13. Regresar a Main.

14. Ir a Listado de cotizaciones.
15. Escoger una cotización.
16. Cancelar una cotización.
17. Regresar a Main.
18. Ir a Servicios.
19. Escoger un servicio.
20. Regresar a Main.
21. Compartir la aplicación.
22. Ir a Contáctenos.
23. Realizar una llamada.
24. Escribir un mail.
25. Ver ubicación en el mapa.
26. Regresar a Main.

Aunque parezca más extenso el cuestionario de la interfaz, tiene más provecho el cuestionario de funcionamiento. El de la interfaz nos muestra un camino que deberán seguir los que vayan a probar la interfaz, pero el de

funcionamiento nos ayuda a que no se presenten fallas, provocando que en un futuro haga desistir de su uso al cliente.

CAPÍTULO 4

IMPLEMENTACIÓN

4.1. Plataformas

Como todo proyecto informático es siempre importante el poder establecer que parte será programada en que plataforma y para esto voy a dividir cada una de las partes que conforma el proyecto.

Aplicación móvil

Esta es la parte fundamental en el proyecto, es la que nos permite interactuar con los clientes, los cuales al momento de usarlo deberían sentirse llamados a volver a usarlo no solo cuando necesiten realizar una solicitud de cotización, sino cada cierto tiempo como para ver productos nuevos o simplemente compartir la aplicación.

Esta parte debe de tener formas, metáforas, apariencia llamativa y que tenga una excelente interacción hombre - máquina. De por sí, el SDK de Android nos ofrece varias formas interesantes, pero esto no nos quita la oportunidad que hagamos cosas tales como botones, banners, fondos, etc. más interesantes, que a nuestro gusto serán los que llamen la atención de nuestros clientes.

Para tener una mejor presentación visual en nuestra aplicación se hará uso del conocido Adobe Illustrator, el cual tiene muchas opciones para poder crear de manera más efectivas las partes visuales de la aplicación que en este caso más serán usadas en botones, fondos y banners.

Al tener la parte visual organizada, listos nuestros botones y demás objetos que formarán parte de nuestra aplicación, debemos enfocarnos ahora en la parte de la programación de nuestra aplicación. La cual como ya dijimos es la parte fundamental de este proyecto.

Se ha tomado muy en cuenta varias características que nos dan mayores ventajas al poder programar para Android y se ha decidido utilizar el IDE Eclipse ya que nos permite mayor movilidad y nos aligera el paquete completo del programa, además que cuenta con una excelente integración con el SDK de Android.

En esta parte, crearemos pantalla por pantalla que sean necesarias para el perfecto funcionamiento de nuestra aplicación, armando los enlaces debidos entre cada una de ellas y cualquier enlace a la base de datos remota que pueda haber, entre ellas: consultar las últimas cotizaciones, los productos existentes y demás interacciones posibles con la base de datos.

Base de datos

Para esta parte serán necesarias dos cosas, la misma base de datos y alguna página o programa que la controle y permita a la aplicación tomar control alguno de nuestra base de datos.

Como primer punto nuestra base de datos será creada en MySQL, por lo mismo necesitamos MySQL Server, el cual es una excelente opción de base de datos, muy flexible y potente. Aunque se podría hacer en cualquier base de datos existente, tales como Oracle o hasta con el mismo Access de Microsoft, escogimos al MySQL Server por motivo de que en un futuro esta base pueda terminar teniendo una ampliación o mejora y así facilitar el trabajo futuro de las personas encargadas del mantenimiento del sistema.

Ahora, para poder interconectar nuestra base de datos con la aplicación es necesario crear, como ya dijimos, una página o programa. La creación de esta parte se va a realizar en un Servidor Web, ya que nos facilita la creación del Php.

Lo que haremos en esta ocasión es crear páginas con extensión PHP para que nuestra aplicación tenga una forma más fácil de comunicarse con la base de datos, las cuales terminarán subidas en la nube para poder ser utilizadas vía internet.

4.2. Problemas al implementar

Ya habiendo pasado por todo el proceso de implementación, después de haber creado la aplicación completa, es momento de repasar y explicar cuáles fueron los problemas que se tuvo en el momento de implementación.

No solo hubo problemas sino también hubo grandes retos al querer implementar ciertas partes de lo que ahora es nuestra aplicación, retos que tuvimos que pasarlos investigando, consultando a gente con más experiencia y haciendo el famoso "prueba y error".

Sin duda alguna los problemas comenzaron en el mismo punto del diseño de la aplicación, no supimos bien que metáforas usar, como debería verse para tener mayor aceptación en los clientes de la empresa computacional. Para

solucionar esto se hizo una pequeña encuesta entre los dueños de la empresa y varios de los clientes, ofreciéndoles varias opciones para que puedan elegir, logrando así escoger las metáforas más votadas para ponerlas en nuestra parte visual.

Como en todo proyecto de programación, los problemas seguían yendo y viniendo, aunque más que problemas o trabas parecían más unos desafíos, el siguiente apareció al momento de poder mostrar el mapa. El utilizar la opción de Open Street Map que es gratuito fue de hecho un tanto complicado, ya que existen pocos ejemplos que sirven, así que nos tocó realizar una investigación a fondo para poder entender cómo utilizar el mapa y las opciones que nos ofrecía Open Street Map.

En el caso de las notificaciones, fue un desafío bien grande, ya que esta parte nunca habíamos utilizado y nos tocó hacer muchas pruebas para que quedara perfecto, pero aun así se logró realizar con éxito las funciones requeridas.

Este último problema fue solucionado con mucha paciencia y haciendo búsquedas a conciencia por varios sitios para ver cuál de todos nos brindaba la solución idónea para nuestro caso.

Otro problema se hizo presente al momento de crear el sistema, que iba a controlar las cotizaciones después de que el cliente enviara la solicitud, por más que el sistema se veía fácil y pequeño, resulto ser complejo al no conocer tan bien la programación web requerida para solucionar esta parte.

Al ver que no se conseguía el objetivo del sistema, decidimos consultar con gente más experimentada en el campo, para que nos diera una idea más sólida. Nos mostraron las opciones que podíamos utilizar, así mismo lo implementamos y solucionamos el problema.

Como punto final, el contacto constante semanal con la empresa formaba ciertos problemas ya que una semana pedía un detalle y a la siguiente la cambiaba, esto es un problema constante y normal en todo proyecto. Las indecisiones que siempre tienen nuestros clientes jamás van a faltar y en este caso no iba a ser la excepción, cambiaban detalles gráficos, detalles de funcionalidades, de formas de poder buscar un producto dado, etc. Un

ejemplo de aquello es que antes teníamos solo Combobox para poder buscar marcas, tipo y productos, pero ellos fueron los que decidieron hacerlo de forma de lista.

CAPÍTULO 5

PRUEBAS

5.1. Pruebas de funcionalidad

Al momento de realizar las pruebas de funcionalidad durante el proceso de implementación, pudimos ver que nos faltaban retroalimentaciones cuando la aplicación se comunicaba con la base de datos.

Tabla #1. Pruebas de funcionalidad

	Funciona	No Funciona
Iniciar una búsqueda de las últimas cotizaciones recibidas por el cliente de la empresa.	X	
Seleccionar una cotización de la lista y que muestre al mismo.	X	
Seleccionar la opción de productos y que muestre todas las marcas que existan en la empresa al cliente.	X	
Seleccionar una marca dada y que muestre los tipos de productos de esa marca.	X	
Seleccionar un tipo dado y que muestre los productos de ese determinado tipo.	X	
Seleccionar un producto dado y que me muestre la figura con los datos correspondientes al mismo.	X	
Agregar el producto al carrito de productos y que el nombre del producto salga en la solicitud de cotización.	X	
Agregar varios productos al carrito de productos y que los nombres de los productos salgan en la solicitud de cotización.	X	
Hacer uso de solicitud de cotización y que llegue al sistema con los datos correctos.	X	
Seleccionar la opción de Servicios y que muestre todos los servicios que ofrece la empresa.	X	
Seleccionar uno de los servicios dados y que muestre los datos correspondientes al mismo.	X	
Seleccionar la opción de compartir y que permita compartir el link de descarga correctamente.	X	
Seleccionar la opción de Contáctenos y que funcione las opciones de llamar correctamente.	X	

Seleccionar la opción de Contáctenos y que funcione la opción de correo electrónico correctamente.	X	
Seleccionar la opción de Contáctenos y que funcione la opción de búsqueda de la dirección en el mapa, mostrando la misma en el mapa con los datos correspondientes.	X	

Fuente: Propio

Elaborado por: Ricardo Márquez

Al momento de terminar las pruebas de funcionalidad, logramos ver que todas las pruebas que estaban planeadas de realizarse se lograron pasar con éxito. Claro está, estos resultados salieron en la última verificación o en las últimas pruebas, ya que en las anteriores que se realizaban mientras la implementación del proyecto seguía, no lograban pasar varias cosas porque todavía no habían sido creadas.

5.2. Pruebas de usabilidad

Las pruebas de usabilidad no son más que pruebas que ayudan a los programadores a ver si el producto que están realizando es fácil de usar e

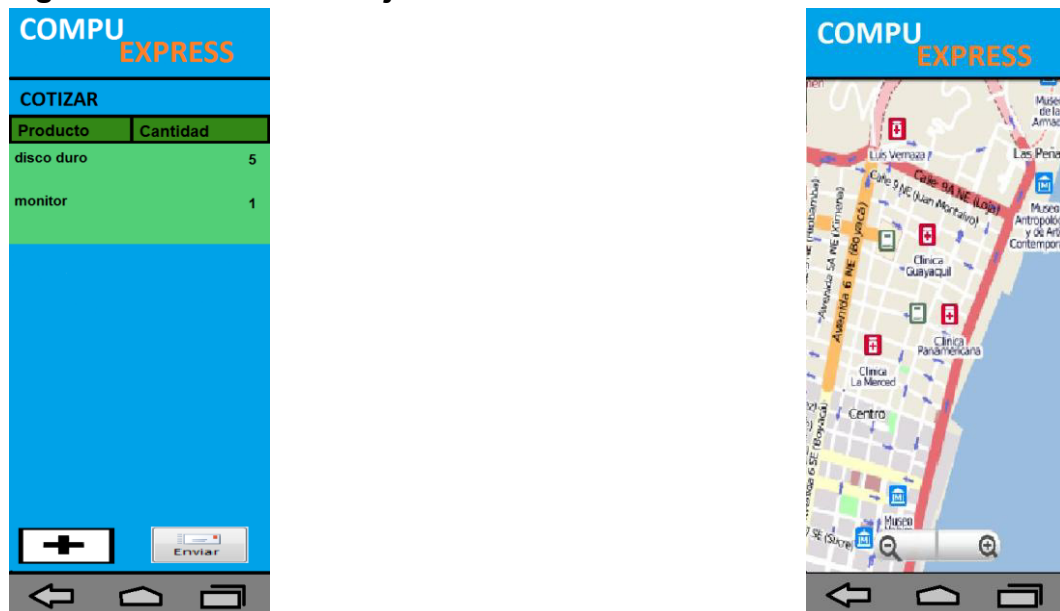
intuitivo. Son normalmente usadas para garantizar calidad y involucramiento de parte de su programa para el cliente, para esto se valen de la interfaz del mismo, poniendo botones, imágenes y demás objetos en diferente orden para saber cuál de estas posiciones le facilita al cliente o usuario el poder manejar nuestro programa y aprenderse donde están cada cosa rápidamente.

Estas pruebas son un elemento poderoso no solo al momento de acabar el proyecto sino también al momento de presentárselo a nuestro cliente para ir progresivamente cambiando cada pantalla, cambiándoles la interfaz a cada una para tener un mejor producto.

Las pruebas de usabilidad nos mostraron falencias pequeñas en el mapa. No supieron bien como hacer aparecer la información de la empresa y en ciertos casos como alejar/acercar al mapa. En general, al cliente se le hizo fácil el utilizar el programa desde las primeras pruebas ya que es una aplicación muy intuitiva.

5.3. Pruebas de bajo nivel

Figura 5.1 Pruebas de bajo nivel



Fuente: Propio

Elaborado por: Ricardo Márquez

Estas pruebas son sumamente importante para un proyecto de software. En síntesis lo que permiten estas pruebas es que los involucrados en las interfaces puedan ver si están usando los elementos correctos.

Para esto se valen de las interfaces dibujadas a groso modo en papel. Se hace una lista de pasos que van a tener que realizar la gente que nos ayudará con las pruebas, pasos que sirven de guía para realizar una u otra acción dentro del programa, los cuales son la navegación entre pantallas, significado de botones y utilización de opciones.

Si las personas logran hacer todas las acciones que la lista define, entonces las interfaces utilizadas son las correctas, en cambio, si la mayoría de las personas cometen muchos errores en los mismos puntos, quiere decir que se deben tomar acciones necesarias en esas interfaces.

Mientras se realiza esta prueba a la persona se debe ir anotando en que parte fallaron, para posteriormente preguntarle a la persona cual fue el motivo del error o duda, estas perspectivas de las personas que ayudan en las pruebas, sirven mucho para poder realizar cambios necesarios para entregar un mejor producto a nuestros clientes. Se debería cambiar las partes de la interfaz sí y solo sí hay varias personas que se equivocan en el mismo punto, porque si solo uno o dos se equivocan ahí, simplemente no es razón suficiente para hacer un cambio.

Estas pruebas se pueden realizar cada vez que tengamos una interfaz nueva, así se podría lograr llegar a una interfaz más fuerte y llamativa para el cliente. Pueden resultar un tanto tediosas estas pruebas al realizarlas más seguido, pero a la final terminan siendo muy necesarias.

Tabla #2. Pruebas de bajo nivel

	Persona 1	Persona 2	Persona 3
Ir a configuraciones.	1	1	1
Configurar sus datos.	1	1	1
Ir a Solicitud de cotización.	1	1	1
Enviar una solicitud de cotización.	1	1	1
Regresar a main.	1	1	1
Ir a Productos.	1	1	1
Seleccionar una marca.	1	1	1
Seleccionar un tipo.	1	1	1
Seleccionar un producto.	1	1	1
Mostrar el producto.	1	1	1
Agregar al carrito de productos.	1	1	1
Pedir cotización del producto.	1	1	1
Regresar a Main.	1	1	1
Ir a Listado de cotizaciones.	1	1	1

Escoger una cotización.	1	1	1
Cancelar una cotización.	3	1	1
Regresar a Main.	1	1	1
Ir a Servicios.	1	1	1
Escoger un servicio.	1	1	1
Regresar a Main.	1	1	1
Compartir la aplicación.	1	1	1
Ir a Contáctenos.	1	1	1
Realizar una llamada.	1	1	1
Escribir un mail.	1	1	1
Ver ubicación en el mapa.	1	1	1
Regresar a main.	1	1	1

1) bien
2) mal
3) dudo pero lo hizo bien

Fuente: Propio

Elaborado por: Ricardo Márquez

Las pruebas de bajo nivel lograron resaltar varias cosas que nos sirvieron de retroalimentación, primero el orden en que se presentan las opciones en el

menú, varios coincidieron en poner primero la opción de solicitar cotización ya que es la función principal de la aplicación.

Al momento de seleccionar una cotización, una de las personas no supo bien como cancelar el documento pero esto se debió a que esta persona no era tan acostumbrada con la tecnología, pero supo hacerlo después de manera correcta.

5.4. Pruebas de alto nivel

Las pruebas de alto nivel se asemejan muchísimo a las pruebas de bajo nivel, pero en este caso ya no se hacen las pruebas en dibujos en papel. Estas pruebas se las hace cuando uno ya está a punto de finalizar el proyecto, ya que se las realiza en la computadora o en el dispositivo donde puede ser instalado.

De la misma forma que en las pruebas de bajo nivel, se debe enlistar los pasos que seguirán las personas que nos van a ayudar. Estas personas deben de seguir los pasos, sin ayuda nuestra y si se llegan a equivocar o se

traban en alguna parte se debe anotar en qué punto ha sido, hay que recordar que estos pasos deben ser la navegación entre pantallas, significado de botones y utilización de opciones.

Cabe recalcar que en estas pruebas por razones de que ya ha sido cambiada la interfaz gracias a las pruebas de bajo nivel, no se presentarán tantos errores sino que va a haber una fluidez mayor a las pruebas anteriores, pero esto no quita que sigan habiendo problemas ya que habrán uso de comandos que se pueden no haber tomado en cuenta y que los probadores intenten hacerlo y lo sugieran.

Al finalizar la lista de opciones que cada persona debía de seguir, se le hará la misma pregunta del porque o cual fue la razón que no pudo hacer una de las opciones enlistadas. Así se logrará tener la retroalimentación que es necesaria en esta prueba y así mejorar en mínimas cosas la interfaz, hay que recordar que se debería cambiar sí y solo sí hay varias personas que se quedan en el mismo punto, porque si solo uno o dos se quedan ahí, simplemente no es razón suficiente para hacer un cambio.

En caso de que nadie se equivoque o sean mínimo el número de personas que se han equivocado en cada paso, podremos decir que nuestra interfaz ha llegado a un punto de éxito. Siempre se puede volver a realizar otra vez estas pruebas si es que lo amerita necesario el programador, pero tiene que tomar en cuenta de que tienen un costo de tiempo/dinero, el cual nunca se puede contar con mucho del mismo.

El realizar o no una de las retroalimentaciones de las personas lo decide uno como programador gracias a su experticia que se lo va ganando, ya que muchos pueden pedir cambios que no es conveniente de forma global hacerlo sino solo para una minoría y eso puede resultar siendo un desastre para el éxito del programa.

Tabla #3. Pruebas de alto nivel

	Persona 1	Persona 2	Persona 3
Ir a configuraciones.	1	1	1
Configurar sus datos.	1	1	1
Ir a Solicitud de cotización.	1	1	1
Enviar una solicitud de cotización.	1	1	1
Regresar a main.	1	1	1
Ir a Productos.	1	1	1

Seleccionar una marca.	1	1	1
Seleccionar un tipo.	1	1	1
Seleccionar un producto.	1	1	1
Mostrar el producto.	1	1	1
Agregar al carrito de productos.	1	1	1
Pedir cotización del producto.	1	1	3
Regresar a Main.	1	1	1
Ir a Listado de cotizaciones.	1	1	1
Escoger una cotización.	1	1	1
Cancelar una cotización.	1	1	1
Regresar a Main.	1	1	1
Ir a Servicios.	1	1	1
Escoger un servicio.	1	1	1
Regresar a Main.	1	1	1
Compartir la aplicación.	1	1	1
Ir a Contáctenos.	1	1	1
Realizar una llamada.	1	1	1
Escribir un mail.	1	1	1
Ver ubicación en el mapa.	1	1	1
Regresar a main.	1	1	1

1) bien
2) mal
3) dudo pero lo hizo bien

Fuente: Propio

Elaborado por: Ricardo Márquez

En estas pruebas saltó el problema de pedir cotización del producto otra vez, pero aquí el probador no se equivocó sino que dio una observación de otra metáfora que podíamos utilizar para que todo el mundo entendiera. Se tomó en cuenta esta observación y se decidió usarla al ver que las siguientes personas que probaron la aplicación estaban de acuerdo con ello.

No se presentaron más problemas ya que se les hizo muy intuitivo el manejo y uso de la aplicación, mostrándonos que habíamos hecho un buen trabajo al hacer los arreglos que vimos que la aplicación necesitaba en las pruebas de bajo nivel.

5.5. Análisis de los resultados

Al realizar nuestras pruebas logramos percatarnos de fallas que existían en cuanto a metáforas se hablan, especialmente en las pruebas de bajo nivel, estas de aquí nos ayudaron bastante para poder posicionar de una mejor manera las opciones según la prioridad que tenían los clientes con respecto a sus necesidades. También se presentó el problema de no entender bien como cancelar correctamente una de las cotizaciones. Este error que es de manipulación por experticia del cliente no se podían corregir pero lo tomamos en cuenta como para hacer una guía de uso del programa en caso de que nuestro cliente, el usuario, requiera que sea adherido para uso del cliente vía web.

En las pruebas de alto nivel, se logró sacar problemas de metáforas en el botón de pedir cotización en la opción de productos. Pero no se puso como error sino como consejo y observación.

Los resultados de las pruebas de bajo nivel, conjuntamente con las pruebas de usabilidad, nos dieron de forma exitosa una retroalimentación, que logró

que las pruebas de alto nivel se hicieran de forma fluida y que sea usado con facilidad la aplicación.

Al momento de haber terminado las pruebas de alto nivel, quedamos satisfechos ya que el producto estaba relativamente perfecto, ya que no se suscitaron errores de interfaz o problemas en el sistema de alguna forma, quedando de resultado que se pueda concluir que el sistema estaba listo para ser entregado al cliente.

CONCLUSIONES Y RECOMENDACIONES

1. El realizar un proyecto informático, por más pequeño que parezca, abarca una secuencia grande de pasos a seguir como programas necesarios para crearlo, funciones que jamás se pensaba al inicio que se las iba a usar. Y es que, realizando un proyecto de esta magnitud por fuerza propia, se da cuenta que se necesita la ayuda de varias personas para realizar un proyecto, ya que así se demoraría al menos la mitad del tiempo.
2. Los momentos cruciales de todo proyecto son las toma de requerimientos con nuestro cliente o el usuario, los cuales nunca faltan a lo largo del

proyecto y en los cuales siempre pedirán cambios imposibles y otros un tanto realizables, aquí es cuando se debe de explicar a nuestro cliente cuales son los límites de nuestro conocimiento y ofrecerle una solución para que queden satisfechos.

3. Los diseños lógicos y físicos nos muestran el esqueleto de todo nuestro proyecto, sirven para ver cómo debe de estar estructurado y podemos valer de ellos para futuros problemas que se presenten al programar, aunque muy pocos programadores se percaten de tal utilidad.
4. El implementar nuestro proyecto conlleva una investigación constante, para poder solucionar cada problema que no dejan de llegar, son momentos en que se debe tener suma paciencia y buscar momentos de iluminación de cierta forma, ya que a veces son errores de lo más obvios, pero por el estrés se lo ve de un tamaño más grande de lo que realmente es.
5. El último paso del proyecto fueron las pruebas, el cual para muchos programadores es tan tedioso como el documentar el código de los proyectos, aduciendo que eso no sirve de nada, pero ambas cosas son

realmente importantes para entregar un producto de calidad a nuestro cliente y se deberían hacer ley en nuestros procesos.

Con todo lo mostrado se logró realizar en su totalidad lo estipulado en los objetivos específicos al igual que obviamente el objetivo general de este proyecto. Todos nuestros objetivos específicos fueron en su medida un reto a elaborar pero se pudo realizar con éxito.

1. Se recomienda a los estimados lectores de este documento que realicen cada uno de los pasos de un proyecto de software sin ningún pero, ya que por más que en el momento pareciera que es una pérdida de tiempo, le ahorrará mucho tiempo y dinero a futuro.
2. Al momento de mostrar un documento que solo es para ser visualizado mas no editado, no complicarse con formas complicadas de visualización. Ver siempre la opción idónea, la cual muchas veces es la opción más sencilla, en este caso usamos `TableLayout`.

BIBLIOGRAFÍA

1. *Wikipedia*, Java (lenguaje de programación), Fecha de acceso el 12 de septiembre de 2013,
[http://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)](http://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))
2. *Wikipedia*, PHP, Fecha de acceso el 12 de septiembre de 2013, [http://](http://es.wikipedia.org/wiki/PHP)
<http://es.wikipedia.org/wiki/PHP>
3. *Wikipedia*, Eclipse (software), Fecha de acceso el 12 de septiembre de 2013, [http://](http://es.wikipedia.org/wiki/Eclipse_(software))
[http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software))
4. *Wikipedia*, SQL, Fecha de acceso el 12 de septiembre de 2013,
[http://](http://es.wikipedia.org/wiki/SQL)
<http://es.wikipedia.org/wiki/SQL>
5. *Wikipedia*, SQL Server, Fecha de acceso el 12 de septiembre de 2013, [http://](http://es.wikipedia.org/wiki/SQL_Server)
http://es.wikipedia.org/wiki/SQL_Server
6. *Android developer*, Fecha de acceso el 12 de septiembre de 2013,
[http://](http://developer.android.com)
<http://developer.android.com>
7. *Wikipedia*, Caso de uso, Fecha de acceso el 12 de septiembre de 2013, [http://](http://es.wikipedia.org/wiki/Caso_de_uso)
http://es.wikipedia.org/wiki/Caso_de_uso

8. Bernd Bruegge, Allen H. Dutoit. *Object Oriented Software Engineering* (3ra. edición). Estados Unidos: Prentice Hall. (2000). Capítulo 4, pág. 100–106, 118-119.
9. Ian Sommerville. *Ingeniería del software* (7ma. edición). Madrid, España: Pearson. (2005). Capítulo 6, pág. 109-115.
10. Ian Sommerville. *Ingeniería del software* (7ma. edición). Madrid, España: Pearson. (2005). Capítulos 11 - 16, pág. 217-354.
11. *PcWorld*, Android e ios se disputan el mercado de smartphones este año, Fecha de acceso el 11 de septiembre de 2013,
<http://www.pcworld.com.ec/android-e-ios-se-disputan-el-mercado-de-smartphones-este-ano/>