



UNIVERSIDAD TECNOLÓGICA EQUINOCCIAL

FACULTAD DE CIENCIAS DE LA INGENIERÍA

**CARRERA DE INGENIERÍA EN INFORMÁTICA Y
CIENCIAS DE LA COMPUTACIÓN**

TEMA:

**“IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA DISPOSITIVOS
MÓVILES CON SISTEMA OPERATIVO ANDROID QUE PERMITA
GENERAR ORDENES DE TRABAJO EN SITIO Y QUE MEDIANTE UNA
CONEXIÓN A INTERNET SINCRONICE INMEDIATAMENTE LA
INFORMACIÓN EN LA BASE DE DATOS DE LA EMPRESA AUVICOM”**

**TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN**

AUTOR: BLADIMIR RAMIRO VALLEJO ALTUNA

DIRECTOR: ING. VÍCTOR HUGO GÁLVEZ

QUITO - ECUADOR

2015

DECLARACIÓN

Yo, **VALLEJO ALTUNA BLADIMIR** declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Tecnológica Equinoccial puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

Bladimir Vallejo Altuna

C.C. 171424254-0

CERTIFICACIÓN

Certifico que el presente TRABAJO DE TITULACIÓN que lleva por título **“IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID QUE PERMITA GENERAR ORDENES DE TRABAJO EN SITIO Y QUE MEDIANTE UNA CONEXIÓN A INTERNET SINCRONICE INMEDIATAMENTE LA INFORMACIÓN EN LA BASE DE DATOS DE LA EMPRESA AUVICOM”**, que, para aspirar al título de **INGENIERO EN INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN** fue desarrollado por **VALLEJO ALTUNA BLADIMIR**, bajo mi dirección y supervisión, en la Facultad de Ciencias de la Ingeniería; y cumple con las condiciones requeridas por el reglamento de Trabajos de Titulación artículos 18 y 25.

Ing. Víctor Hugo Gálvez

Director de Trabajo de Titulación

C.C. 1707796536

DEDICATORIA

El presente trabajo de titulación va dedicado a todas las personas que de una u otra manera hicieron posible que pueda hoy estar finalizando mis estudios y obteniendo mi titulación, a mi familia que fue el pilar importante en mi vida y que me brindo fortaleza en momentos cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón, mi agradecimiento y ganas de seguir superándome cada día.

A la Universidad Tecnológica Equinoccial que supo ser fiel testigo de todos mis esfuerzos por superarme día a día y a sus docentes quienes fueron los guías de la estructuración de este objetivo que se presentó en mi vida.

Este trabajo se los dedico a ustedes.

Bladimir Vallejo Altuna

AGRADECIMIENTOS

Agradezco a Dios y a la vida, por darme la fortaleza y haberme rodeado de personas que siempre estuvieron brindándome su apoyo, conocimiento, paciencia y nunca me dejaron caer para poder seguir adelante a pesar de los inconvenientes encontrados en el camino.

Bladimir Vallejo

RESUMEN

El trabajo que se presenta a continuación está estructurado y dirigido al análisis y automatización de los diferentes procesos que intervienen en la ejecución de una orden de trabajo por parte de la empresa AUVICOM.

Esta aplicación está diseñada para controlar y administrar de una manera mucho más eficiente cada uno de los tratamientos por los cuales pasan los datos para ser ejecutados e ingresados a los registros de la empresa presentes en las bases de datos.

El sistema cuenta con una interfaz amigable para el usuario y que se la puede visualizar en dispositivos móviles con acceso al internet y con sistemas operativos actuales tales como android, Windows mobile, entre otros los cuales con cada una de sus características particulares ayuda a que la aplicación tenga una mejor visualización y funcionamiento al momento de su operatividad.

El trabajo escrito cuenta con capítulos donde se detalla cada una de las etapas que tuvo el sistema para su estructuración y posterior uso, en él se trata todas y cada una de las técnicas de desarrollo que se implementó en la realización de la aplicación con su respectivo análisis para poder obtener una mejor comprensión de lo que en si consta en el sistema.

El análisis previamente realizado ayudó para tener un mejor entendimiento del funcionamiento e implementación de aplicaciones web en dispositivos móviles así como conceptos importantes en lo relacionado a la programación orientada a objetos con lo que se brinda una ventana más clara a la visualización y posterior uso de la aplicación desarrollada para este proyecto de titulación.

ABSTRACT

The work presented below is organized and directed for the analysis and automation of the various processes involved in the execution of a work order by the company Auvicom.

This application is designed to control and manage much more efficiently each of the treatments which pass the data to be executed and entered into the register of the company present in the databases.

The system has a friendly user interface and it can be displayed on mobile devices with access to the internet and current operating systems such as Android, Windows Mobile, among others ... which each of their particular characteristics help that the application has a better display and operation at the time of operation.

Written work has detailed chapters where each stage had the system for structuring and subsequent use, it is about each and every development techniques that are implemented in the implementation of the application with its respective analysis to gain a better understanding of what is indicated on the system.

The analysis performed previously helped to have a better understanding of the functioning and implementation of web applications in mobile devices as well as important concepts related to object-oriented with a clearer view to the window and subsequent use of the programming is provided application developed for this project titration.

ÍNDICE

1.	INTRODUCCIÓN	2
2.	MARCO TEÓRICO	6
2.1	. APLICACIÓN WEB PARA DISPOSITIVOS MÓVILES (ANDROID, WINDOWS MOBILE, ENTRE OTROS) PARA GENERAR ORDENES DE TRABAJO	6
2.2.	APLICACIONES MÓVILES PARA TOMA DE PEDIDOS	9
2.3.	ESTRUCTURA DE LA APLICACIÓN MÓVIL	10
2.3.1.	SEGURIDADES	11
2.3.1.1.	NIVELES DE USUARIOS	11
2.4.	SISTEMAS OPERATIVOS MÓVILES	12
2.4.1.	KERNEL	13
2.4.2.	MIDDLEWARE	14
2.4.3.	ENTORNO DE EJECUCIÓN DE APLICACIONES:	15
2.4.4.	INTERFAZ DE USUARIO	15
2.5.	SISTEMAS OPERATIVOS MÓVILES MÁS UTILIZADOS.....	15
2.5.1.	SYMBIAN	15
2.5.2.	WINDOWS PHONE	15
2.5.3.	BLACKBERRY OS	16
2.5.4.	IOS	16
2.5.5.	ANDROID.....	16
2.6.	PROCESO DE NEGOCIO.....	17
2.6.1.	DEFINICIÓN DE PROCESO.....	19
2.6.2.	GESTIÓN DE PROCESOS DE NEGOCIO	20
2.7.	FLUJO DE TRABAJO.....	22
2.8.	SISTEMAS DE DOCUMENTOS.....	24

2.8.1.	ECM (ADMINISTRACIÓN DE CONTENIDO EMPRESARIAL)	24
2.8.2.	CMS (SISTEMA DE GESTIÓN DE CONTENIDOS)	25
2.9.	TIPOS DE GESTORES DE CONTENIDOS	29
2.9.1.	POR SUS CARACTERÍSTICAS:	29
2.9.2.	POR SU USO Y FUNCIONALIDAD:	29
2.10.	INGENIERÍA DE SOFTWARE	30
2.10.1.	ANÁLISIS DE REQUISITOS DE SOFTWARE	32
2.10.2.	METODOLOGÍA RUP	33
2.11.	PROCESOS DEL DESARROLLO	34
2.12.	CARACTERÍSTICAS DEL RUP	35
2.13.	FASES DEL RUP	35
2.14.	CICLOS DE VIDA DE DESARROLLO	37
2.15.	EL USUARIO DENTRO DEL PROCESO DE DESARROLLO DE SOFTWARE	38
2.16.	METODOLOGÍA DE INVESTIGACIÓN	40
3.	METODOLOGÍA	44
3.1.	FASE 1: INICIO	45
3.1.1.	Levantamiento de Información	45
3.2.	FASE 2: ELABORACIÓN	45
3.2.1.	ARQUITECTURA DEL SISTEMA	46
3.3.	FASE 3: CONSTRUCCIÓN	47
3.4.	FASE 4: TRANSICIÓN	48
3.4.1.	PRUEBAS Y DESPLIEGUE	48
4.	RESULTADOS Y DISCUSIÓN	50
4.1.	FASE INICIAL – ANÁLISIS	50
4.1.1.	INTRODUCCIÓN	50

4.1.2.	DESCRIPCIÓN GENERAL	51
4.1.3.	SUPERVISAR PROYECTOS DE INVESTIGACIÓN.....	53
4.2.	FASE DE ELABORACIÓN	55
4.2.1.	DIAGRAMAS CASOS DE USO.....	56
	Actor No.1 Super usuario	56
	Actor No.2 Administrador	56
	Actor No.3 Operario	56
	Actor No.4 Técnico.....	56
	Actor No.5 Cliente	56
4.3.	FASE DE CONSTRUCCIÓN.....	104
4.3.2.	MAQUETACIÓN.....	105
4.4.	FASE DE TRANSICIÓN	109
4.4.1.	PRUEBAS Y DESPLIEGUE	109
4.4.2.	DIAGRAMA DE DESPLIEGUE	110
4.4.3.	MANUAL DE USUARIO	111
5.	CONCLUSIONES Y RECOMENDACIONES	113
	BIBLIOGRAFÍA.....	118
	ANEXOS.....	125

ÍNDICE DE GRÁFICOS

Gráfico 1. Plantillas visual studio 2010	8
Gráfico 2. Componentes esenciales de teléfonos celulares	13
Gráfico 3. Representación de un Proceso de Negocio	17
Gráfico 4. Ciclo de vida del BPM	22
Gráfico 6.WorkFlow	23
Gráfico 7. Componentes de un CMS	25
Gráfico 8. Flujos de procesos	37
Gráfico 9. Ciclo de vida del software.....	37
Figura 14. Imagen funcionamiento aplicación móvil	46
Gráfico 11. MVC(MODELO VISTA CONTROLADOR)	47
Figura 1. Diagrama Caso de Uso 1 - Actor Super Usuario	57
Figura 2. Diagrama Caso de Uso 2 – Actor Administrador	57
Figura 3. Diagrama Caso de Uso 3 – Actor Operario	58
Figura 4. Diagrama Caso de Uso 4 – Actor Técnico.....	58
Figura 5. Diagrama Caso de Uso 5 – Cliente	59
Figura 6. Diagrama Caso de Uso 6 - Sistema	60
Figura 7. Diagrama Caso de Uso 7 – Ingreso al Sistema Usuario General..	61
Figura 8. Diagrama Caso de Uso 8 – Ingresar Administrador	63
Figura 9.Diagrama de Secuencia 1 – Registro Administradores	66
Figura 10. Diagrama Caso de Uso 9 – Modificar Administrador	66
Figura 11. Diagrama de Secuencia 2 – Modificación Administrador.....	69
Figura 12. Diagrama Caso de Uso 10 – Eliminar Administrador.....	69
Figura 13.Diagrama de Secuencia 3 - Eliminación Administrador	71
Figura 14. Diagrama Caso de Uso 11 – Ingreso Usuarios.....	72
Figura 15. Diagrama de Secuencia 4 - Registro de Usuarios	75

Figura 16. Diagrama Caso de Uso 12 – modificar Usuarios	75
Figura 17. Diagrama de Secuencia 5 - Modificación de Usuarios	78
Figura 18. Diagrama Caso de Uso 13 – Eliminar Usuarios.....	78
Figura 19. Diagrama de Secuencia 6 - Eliminación de Usuarios	81
Figura 20. Diagrama Caso de Uso 14 – Enviar Requerimiento	81
Figura 21. Diagrama de Secuencia 7 - Enviar Requerimiento	83
Figura 22. Diagrama Caso de Uso 15 – Enviar Proforma.....	84
Figura 23. Diagrama de Secuencia 8 - Enviar Proforma.....	86
Figura 24. Diagrama Caso de Uso 16 – Elaboración Orden de Trabajo.....	86
Figura 25. Diagrama de Secuencia 9 – Elaboración de Orden de Trabajo...	89
Figura 26. Diagrama Caso de Uso 17 – Asignación Orden de Trabajo	89
Figura 27. Diagrama de Secuencia 10 - Asignación de Orden de Trabajo...	92
Figura 28. Diagrama Caso de Uso 19 – Registro Abono Orden de Trabajo .	92
Figura 29. Diagrama de Secuencia 11 – Registro de Abono de Orden de Trabajo.....	94
Figura 30. Diagrama Caso de Uso 20 – Finalización Orden de Trabajo.....	95
Figura 31. Diagrama de Secuencia 12 – Finalizar Orden de Trabajo	97
Figura 32. Diagrama Caso de Uso 21 – Verificación Orden de Trabajo	97
Figura 33. Diagrama de Secuencia 13 – Verificación Orden de Trabajo	99
Figura 34. Diagrama Caso de Uso 22 – Visualización Reportes	99
Figura 35. Diagrama de Secuencia 14 – Visualizar Reportes.....	101
Figura 36. Modelo conceptual de base de datos	102
Figura 37. Diagrama de clases	103

ÍNDICE DE TABLAS

Tabla 1 Especificación del caso de uso 7	61
Tabla 2. Flujo Básico Caso de Uso 7.....	62
Tabla 3. Flujo Alternativo Caso de Uso 7.....	62
Tabla 4. Especificación del caso de uso 8	63
Tabla 5. Flujo Básico Caso de Uso 8.....	64
Tabla 6. Flujo Alternativo Caso de Uso 8.....	65
Tabla 7. Especificación del caso de uso 9	67
Tabla 8. Flujo Básico Caso de Uso 9.....	67
Tabla 9. Flujo Alternativo Caso de Uso 9.....	68
Tabla 10. Especificación del caso de uso 10	69
Tabla 11. Flujo Básico 10	70
Tabla 12. Flujo Alternativo Caso de Uso 10.....	71
Tabla 13. Especificación del caso de uso 11	72
Tabla 14. Flujo Básico del caso de uso 11	73
Tabla 15. Flujo Alternativo caso de uso 11	74
Tabla 16. Especificación del caso de uso 12	76
Tabla 17. Flujo Básico caso de uso 12	76
Tabla 18. Flujo Alternativo caso de uso 12.....	77
Tabla 19. Especificación del caso de uso 13	79
Tabla 20. Flujo Básico caso de uso 13	79
Tabla 21. Flujo Alternativo caso de uso 13.....	80
Tabla 22. Especificación del caso de uso 14	81
Tabla 23. Flujo Básico caso de uso 14	82
Tabla 24. Flujo Alternativo caso de uso 14	82
Tabla 25. Especificación del caso de uso 15	84
Tabla 26. Flujo Básico caso de uso 15	85
Tabla 27. Flujo Alternativo caso de uso 15.....	85
Tabla 28. Especificación del caso de uso 16	87
Tabla 29. Flujo Básico caso de uso 16	87
Tabla 30. Flujo Alternativo caso de uso 16.....	88
Tabla 31. Especificación del caso de uso 17	89

Tabla 32. Flujo Básico caso de uso 17	90
Tabla 33. Flujo Alternativo caso de uso 17	91
Tabla 34. Especificación del caso de uso 19	92
Tabla 35. Flujo Básico caso de uso 19	93
Tabla 36. Flujo Alternativo caso de uso 19	94
Tabla 37. Especificación del caso de uso 20	95
Tabla 38. Flujo Básico 20	96
Tabla 39. Especificación del caso de uso 21	97
Tabla 40. Flujo Básico caso de uso 21	98
Tabla 41. Flujo Alternativo 21	98
Tabla 43. Especificación del caso de uso 22	100
Tabla 43. Flujo Básico caso de uso 22	100

INTRODUCCIÓN

1. INTRODUCCIÓN

La empresa AUVICOM es una empresa que se dedica al servicio de soporte técnico, reparación y mantenimiento de equipos. Por motivos legales y por dar formalidad a los acuerdos económicos y morales que se realizan con los clientes la empresa se ve obligada a generar un documento denominado “orden de trabajo”, en dicho documento consta información de suma importancia para la empresa como por ejemplo: que fallo tiene, cual es el requerimiento del cliente, que accesorios adicionales entrega el cliente, cual fue el abono realizado por el cliente, etc. Estas órdenes de trabajo son emitidas ya sea en la oficina o en el lugar que el cliente lo requiera y además son generadas por un representante de la empresa y en presencia del cliente, escritas a mano en formularios pre impresos.

La generación de órdenes de trabajo manuales ha causado en reiteradas ocasiones problemas como información ilegible y perdida del documento generando molestias tanto para la empresa como para los clientes.

Cuando la información es ilegible en la orden de trabajo o se ha extraviado el documento se ha dado casos que no se puede contactar al cliente por ende generando demoras en la entrega de un presupuesto o un informe y esto tiene como consecuencia la mala imagen de la empresa; es por este motivo que se desarrollará una aplicación móvil para ayudar a optimizar el proceso que se realiza de forma manual, se enfocará principalmente en las ordenes de trabajo que tiene la mencionada empresa, ayudando así a su automatización y desarrollo tecnológico.

El sistema al que se hará mención presenta varios módulos de ejecución los cuales se encuentran establecidos de acuerdo a los roles de la persona que se encuentre manipulando los dispositivos móviles donde estén instalados la aplicación.

La parte funcional del sistema permitirá tanto al técnico como al cliente de la empresa tener una mejor experiencia en la ejecución de un trabajo, teniendo

en cuenta la interface amigable con la que cuenta y las diversas funcionalidades que hacen que este proceso sea mucho más eficiente y rápido a comparación con el antiguo método que se utilizaba para la recepción de órdenes de trabajo así como la ejecución de las mismas.

La aplicación está desarrollada para su utilización en dispositivos móviles, teniendo en cuenta su portabilidad y un sinnúmero de ventajas que hacen que la tecnología juegue un papel muy importante en el modelo de negocios que se maneja en la actualidad, con lo cual se hará integrar dichas tecnologías de manera eficiente dentro de la empresa.

Los dispositivos a los que se enfoca nuestra aplicación son aquellos que tienen como interfaz el sistema operativo Android, que fue la plataforma elegida para la utilización en este sistema, por el bajo costo de los dispositivos que cuentan con esta tecnología, teniendo también un ambiente web al cual acceder a través de la aplicación que se encuentra en la nube.

La aplicación será amigable con el usuario, ya sea que este cuente con alto grado de experiencia dentro del área del manejo de aplicaciones informáticas o solo los conocimientos básicos en la materia, además la aplicación se encuentra establecida dentro de los parámetros por los cuales se manejan las diferentes necesidades que los usuarios presentan en este tema y primordialmente al personal externo a la empresa como lo son los clientes los mismos que se encuentran en todo tipo y niveles académicos, los cuales van a manipular la aplicación de una manera empírica, por lo cual está controlando varios errores comunes que se pueden presentar en el proceso de revisar una orden de trabajo por parte de ellos.

Toda empresa busca siempre la evolución dentro del campo en el cual realizan sus operaciones y **AUVICOM** al estar vinculada de forma directa con el mundo de la tecnología a través del servicio que brinda, incorpora de esta manera la aplicación a sus labores cotidianas.

OBJETIVOS DEL PROYECTO

OBJETIVO GENERAL

Desarrollar una aplicación web en el lenguaje de programación Visual Basic.net para dispositivos móviles con sistema operativo Android que permita generar órdenes de trabajo en sitio y pueda interactuar en forma inmediata con la base de datos del sistema de la empresa AUVICOM.

Objetivos Específicos

- Analizar la situación actual de la generación de órdenes de trabajo en sitio y establecer el modelo del negocio para facilitar su proceso.
- Recopilar información suficiente para entender los principios y secciones específicas en el funcionamiento de aplicaciones para dispositivos móviles, metodologías de desarrollo, modelamiento y estándares.
- Utilizar la metodología de desarrollo adecuada que facilite la comprensión del funcionamiento lógico del sistema y agilite el desarrollo del proyecto
- Implementar la aplicación sobre un dispositivo móvil con sistema operativo Android para realizar las pruebas concernientes y verificar los alcances del proyecto.
- Realizar pruebas y determinar el correcto funcionamiento y sincronización de la aplicación para optimizar de correcta forma la utilización de la aplicación.

MARCO TEÓRICO

2. MARCO TEÓRICO

2.1. APLICACIÓN WEB PARA DISPOSITIVOS MÓVILES (ANDROID, WINDOWS MOBILE, ENTRE OTROS) PARA GENERAR ORDENES DE TRABAJO

“RegisterPCXpres”, es una aplicación gestora de órdenes de trabajo la cual tiene un fin, que es optimizar el registro de las operaciones que involucran el brindar soporte técnico a los clientes de la empresa AUVICOM y posterior sincronización de la información a la base de datos que se maneja en la empresa antes mencionada.

Para el desarrollo de este sistema se necesitó:

- Tener una fuerte base de conocimientos sobre programación web ligado al desarrollo en el lenguaje de programación Visual Basic.net y sus diferentes funcionalidades en dispositivos móviles, lo cual involucra ciertos aspectos como:
 - El diseño del sistema se encuentra desarrollado en especial para interfaces táctiles a través de una pantalla capacitiva, “Pantalla capacitiva es una pantalla que mediante un toque directo sobre su superficie permite la entrada de datos y órdenes al dispositivo. A su vez, actúa como periférico de salida, mostrando los resultados introducidos previamente” (Aponte & Davila, 2011), además también permite la utilización del mismo con el empleo de un teclado para el ingreso de datos al sistema.
 - La mayor parte de los dispositivos móviles cuentan con una conexión a internet, que en este caso será una herramienta de notificación que ayudará al registro de pagos o depósitos realizados por el cliente optimizando el registro de transacciones en el sistema de la empresa AUVICOM.

- Las nuevas tecnologías han hecho posible la utilización de una gran diversidad de herramientas que permiten que los dispositivos móviles tengan un avance impresionante en el transcurso de los últimos años y entre estas se tiene lo que es el acelerómetro, “dispositivo que mide la aceleración y la fuerzas inducidas por las gravedad ” (Apple Inc) que en pocas palabras permite que la pantalla de nuestros dispositivos gire de acuerdo a la necesidad del usuario lo que hace que el sistema tenga una adaptabilidad a cualquier vista de cualquier dispositivo en el que se esté ejecutando el sistema (Informática Hoy, 2011) .
- Para el desarrollo de la aplicación se utiliza el lenguaje de programación Visual Basic .NET, que tiene diversas funcionalidades muy imponentes en desarrollo de aplicaciones web, “El visual basic .NET es una tecnología dinámica funcionando del lado del servidor, lo que significa que cuando el usuario solicita un documento Visual Basic .net, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante. La ventaja principal de las tecnologías dependientes del servidor radica en la seguridad que tiene el programador sobre su código, ya que éste se encuentra únicamente en los archivos del servidor que al ser solicitado a través del web, es ejecutado, por lo que los usuario no tienen acceso más que a la página resultante en su navegador.” (maestros del web, 2001), lo que lo hace un sistema ideal para lo referente a dispositivos móviles.
- Se necesitó una interface de desarrollo para el lenguaje de programación Visual Basic .net para lo cual se utilizó el software Microsoft Visual Studio 2010, “es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta múltiples lenguajes de programación tales como C++, C#,

Visual Basic .NET, F#, Java, Phyton, Ruby, php; al igual que entornos de desarrollo web como ASP.NET MVC, Django, etc, adicionándole las nuevas capacidades online bajo Windows Azure en forma del editor Mónico.” (HTCMANIA, 2013) Visual Studio es muy versátil e incorpora novedosas soluciones en sus nuevas plantillas de diseño fluido como se muestra en la gráfica siguiente:

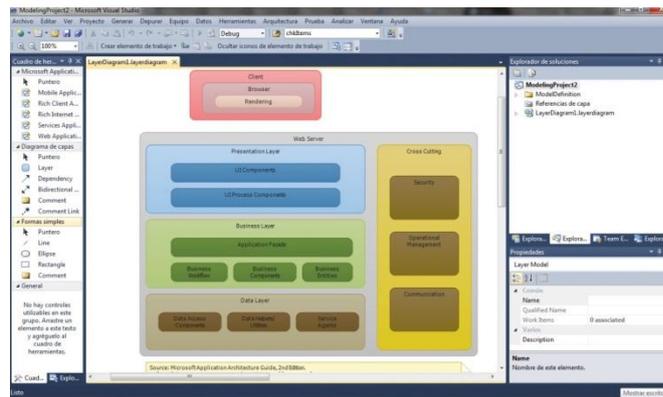


Gráfico 1. Plantillas visual studio 2010

Fuente: <http://highscalability.files.wordpress.com/2010/07/layer.jpg>

- El sistema cuenta con una base de datos, que contiene la información que se genera con el registro de la orden de trabajo y que se la maneja a través del Microsoft SQL Server 2008 que, “es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional” (Fling, 2009) la cual se sincroniza a la base de datos de la empresa AUVICOM.
- La empresa AUVICOM también aportó con la información necesaria para el desarrollo del sistema, teniendo como referente su base de datos con la cual se pudo realizar las pruebas pertinentes tanto en el proceso de desarrollo como en la fase final del proyecto.
- La presentación del sistema requiere un conocimiento en Diseño gráfico en un nivel básico, ya que la estructura del mismo no requiere de grandes efectos ópticos sino más bien de una interfaz

amigable para el operario quien lo va a manipular y que se optimicen los procesos (Weitzenfeld, 2005).

2.2. APLICACIONES MÓVILES PARA TOMA DE PEDIDOS

Existen varios entes que se encuentran involucrados en el proceso de realización de una orden de trabajo.

- **Cliente.**- Los datos que recibe son en forma general haciendo énfasis en los tiempos de duración, precios estimados, fechas de entrega estimadas, datos de contacto e información de su orden de trabajo
 - **Empresa.**- Los datos que recibe engloban todo lo estipulado con el cliente, así como la información técnica; para la futura facturación, almacenamiento, coordinación y despacho de la orden de trabajo
 - **Técnico.**- Los datos que recibe el trabajador se enfocan a fechas límites, datos específicos del tipo de soporte y fallas del hardware para proceder a su evaluación y reparación
- Sincronización con base de datos: una vez almacenada la información en la base de datos interna, es procesada y encriptada para su traslado a la base de datos de la empresa, proceso el cual se hace de forma transparente tanto para el usuario como para los trabajadores, operarios y directivos.

En el sistema para cada usuario se almacena sus datos de contacto referenciados a través de un número único de identificación que en este caso es el número de cédula. Información que es almacenada en la base de datos interna del sistema para el acceso oportuno al mismo. Los

usuarios son totalmente ajenos a la estructura interna de la base de datos, así como al flujo de traslado de información de la misma, los datos filtrados solamente serán visibles desde una terminal móvil con acceso a internet. (Ableson, 2010)

Dicha aplicación fue desarrollada en el entorno de programación Microsoft Visual Studio 2010, con los complementos adecuados y actualizados a la fecha, además, se encuentra el control de almacenamiento interno y sincronización a la base de datos Microsoft SQL Server 2008.

Para el desarrollo de la aplicación se necesitó:

- Realizar la instalación y preparación del entorno de trabajo, aquí se incluyen todos los programas necesarios para el desarrollo del proyecto.
- Realizar la base estructural del diseño de los formularios.
- Realizar el diseño de la aplicación, de manera muy intuitiva y sencilla de manejar para el usuario.
- Realizar el diseño y sincronización del intercambio de datos entre bases.

2.3. ESTRUCTURA DE LA APLICACIÓN MÓVIL

Para el desarrollo de la aplicación se empleó la metodología óptima de programación, ya que es una metodología la cual se puede implementar en diversos lenguajes y ayuda en el entorno global de la programación.

El proyecto tiene un muy amplio alcance en la forma de diseño, desarrollo y mantenimiento de la aplicación, por esto permite minimizar los problemas y errores que existen. (Gramlich, 2010)

2.3.1. SEGURIDADES

Para el manejo oportuno de la seguridad al momento de hacer la respectiva sincronización entre las bases de datos, de la aplicación Android hacia la empresa AUVICOM, se procederá a hacer un traspaso mediante nombre y código de la persona o personas que estén manipulando el sistema, de esta manera es de responsabilidad del operario y políticas internas de la empresa, el detectar y controlar el flujo de información mediante una tabla de auditoria que se implementará de forma paralela en el sistema. (Angoar, 2011)

2.3.1.1. NIVELES DE USUARIOS

Con el fin de optimizar los procesos, se utilizará 5 tipos de usuario para agilizar tiempos. Los usuarios a manejar son los siguientes

- **Administrador.-** Tendrá la potestad de crear nuevos usuarios, darlos de baja, eliminarlos lógicamente y almacenarlos en la base de datos del sistema.
- **Empresa.-** Este ente tiene acceso a una página que contiene un menú con opciones referentes a proformas, órdenes de trabajo y reportes de las mismas para su respectiva administración y control de cuestiones económicas y confirmación de clientes para la empresa.

- **Técnico.-** Tendrá acceso a una página con la información de las órdenes de trabajo y reportes de las mismas, teniendo de este modo toda la información sobre las peticiones nuevas y antiguas teniendo la autoridad para actualizar estados y verificarlas como activas o caducadas.
- **Operario.-** Tendrá acceso al formulario principal del sistema, será capaz de interactuar entre las diferentes páginas que competen a la orden de trabajo y tendrá la opción de sincronizar cada orden individualmente usando seguridades para garantizar la llegada oportuna de los datos.
- **Ciente.-** Este usuario tiene acceso a un formulario de contacto donde para realizar la petición de servicio tendrá que llenar sus datos y enviarlos, además tiene acceso al estado de sus peticiones de trabajo, conociendo de este modo la situación de las mismas y ver si ya concluyó el trabajo que solicitó o si todavía se encuentra en ejecución.
(Puon, 2014)

2.4. SISTEMAS OPERATIVOS MÓVILES

Un sistema operativo móvil, es un sistema operativo el cual gestiona los recursos y provee servicios a los programas de aplicación en un dispositivo móvil, se encuentra orientado más a aplicaciones multimedia para móviles, que brindan diferentes maneras de introducir información en ella, estos sistemas operativos móviles son mucho más simples. (Angoar, 2011)

Un sistema operativo móvil se encuentra conformado por varias capas:

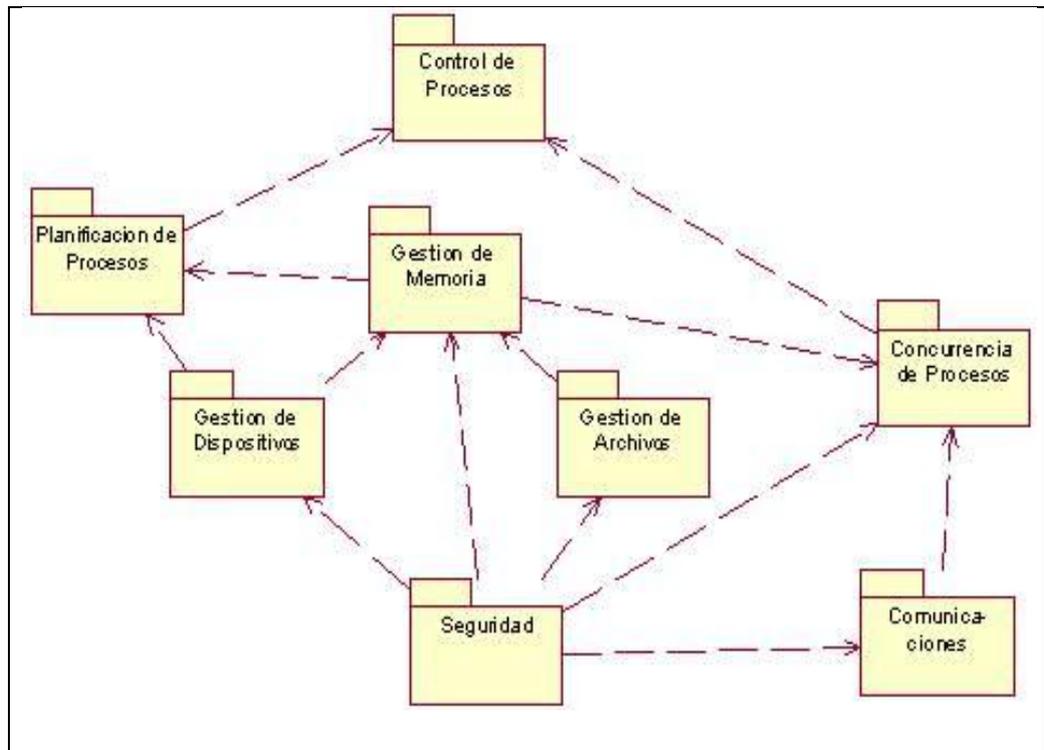


Gráfico 2. Componentes esenciales de teléfonos celulares

Fuente:<http://www.informatica-hoy.com.ar/soluciones-moviles/Componentes-de-un-Sistema-Operativo-Movil.php>

2.4.1. KERNEL

Sea de una pc o de un móvil el Kernel es el núcleo, es una de las piezas más importantes que componen cualquier sistema operativo, permite el acceso a los diferentes elementos de hardware que conforman un móvil. También es el encargado de brindar diferentes servicios a las capas superiores como los controladores de hardware, gestión de procesos, sistemas de archivos, además del acceso y administración de la memoria del sistema. (Haseman, 2008)

2.4.2. MIDDLEWARE

Esta capa es el conjunto de módulos que permite que las aplicaciones diseñadas y escritas para tales plataformas puedan ser ejecutadas, en otras palabras es la capa la cual permite la existencia de aplicaciones para móviles, y dan características muy importantes como (Perumal, 2012):

- Su funcionamiento es totalmente transparente para el usuario.
- Brinda la posibilidad de ejecutar servicios.

Entre los servicios que presta esta capa se puede citar:

- Los motores de comunicaciones y mensajería.
- Funciones de seguridad.
- Servicios para la gestión de diferentes aspectos del móvil.
- Ofrece servicios claves como el motor de mensajería y comunicaciones.
- Codecs multimedia.
- Intérpretes de páginas Web y servicios WAP.
- Soporte para una gran variedad de servicios concernientes al apartado multimedia que es capaz de ejecutar el móvil.

2.4.3. ENTORNO DE EJECUCIÓN DE APLICACIONES:

Esta capa provee de todos los elementos necesarios para la creación y desarrollo de software a los programadores, ya que consiste de un conjunto de interfaces programables abiertas (Lequerica, 2011).

2.4.4. INTERFAZ DE USUARIO

Esta es otra de las capas más importantes, es el elemento de móvil el cual se utilizará para interactuar con él, sin esta capa no sería posible utilizarlo, ya que este presenta todos los elementos necesarios para facilitar las tareas e interacción de un dispositivo como son:

Botones, menús, pantallas y listas, entre otros (Gargenta, 2011).

2.5. SISTEMAS OPERATIVOS MÓVILES MÁS UTILIZADOS

2.5.1. SYMBIAN

Symbian fue producto de la alianza de varias empresas de telefonía móvil, entre las que se encuentran Nokia como la más importante, Sony Ericsson, Samsung, Siemens, Benq, Fujitsu, Lenovo, LG, Motorola esta alianza le permitió en un momento dado ser uno de los pioneros y más usados, pero actualmente está perdiendo vertiginosamente cuota de usuarios aceleradamente; está por la versión 3 (Nadel, 1994).

2.5.2. WINDOWS PHONE

Windows Phone, anteriormente llamado Windows Mobile es un sistema operativo móvil compacto desarrollado por Microsoft, se basa en el núcleo del sistema operativo Windows CE y cuenta con un conjunto de aplicaciones básicas, actualmente va por la versión 7.

Está diseñado para ser similar a las versiones de escritorio de Windows (Informática Hoy, 2011).

2.5.3. BLACKBERRY OS

BlackBerry OS desarrollado por la empresa canadiense RIM (Research In Motion) para sus dispositivos. El sistema permite multitarea y tiene soporte para diferentes métodos exclusivos de RIM particularmente hace referencia a sus trackwheel, trackball, touchpad y pantallas táctiles (Beavis, 2014).

2.5.4. IOS

IOS de los iPhone, anteriormente denominado iPhone OS creado por Apple originalmente para el iPhone. Es un derivado de Mac OS X, se lanzó en el año 2007, aumento el interés con el iPod Touch e iPad que son dispositivos con las capacidades multimedia del iPhone pero sin la capacidad de hacer llamadas telefónicas, en si su principal revolución es una combinación casi perfecta entre hardware y software, el manejo de la pantalla multitáctil (Fling, 2009).

2.5.5. ANDROID

El sistema operativo Android es sin duda el líder del mercado móvil en sistemas operativos, está basado en GNU Linux diseñado originalmente para dispositivos móviles como los teléfonos inteligentes pero después tuvo modificación para ser usado en tablets como es el caso del Galaxy Tab de Samsung, actualmente se encuentra en desarrollo para usarse en notebooks y PCs, el desarrollador de este sistema operativo es Google, fue anunciado en el 2007 y liberado en el 2008; además de la creación de la Open

Handset Alliance, compuesto por 78 compañías de hardware, software y telecomunicaciones dedicadas al desarrollo de estándares abiertos para celulares, esto le ha ayudado mucho a Google a masificar el sistema operativo hasta el punto de ser usado por empresas como HTC, LG, Samsung, Motorola entre otros (Gramlich, 2010).

2.6. PROCESO DE NEGOCIO

Un proceso de negocio se lo define como un conjunto de tareas lógicamente relacionadas, llevadas a cabo con el fin de lograr un resultado de negocio específico (Cepeda, 2013). Cada proceso está compuesto de tres elementos:

- **Entradas:** Las entradas son datos o requisitos sobre los cuales se aplica una función
- **Funciones:** Una función es la rutina que realiza una tarea específica y devuelve un valor o conjunto de valores
- **Salidas:** Las salidas son el resultado, las cuales son la finalidad misma de la ejecución de un proceso.

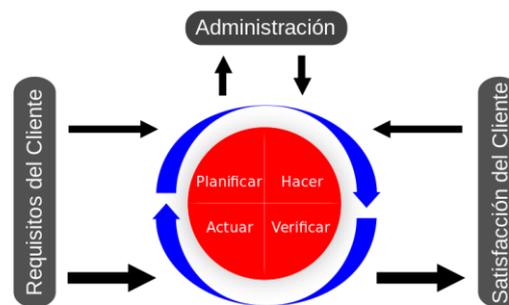


Gráfico 3. Representación de un Proceso de Negocio

Fuente: http://es.wikipedia.org/wiki/Gesti%C3%B3n_de_procesos_de_negocio

Donde los requerimientos del cliente vendrían a ser la entrada de datos, la administración toma las funciones de: planificar, hacer, actuar y verificar los datos e información para dar solución y respuesta a los requerimientos, dando así como resultado la satisfacción del cliente, que vendría a ser la salida de procesos.

Un proceso a su vez representa una secuencia de actividades estructuradas que producen valor a la organización, sus inversionistas, socios, clientes o afiliados. Un proceso de negocio puede ser parte de un proceso de mayor tamaño que lo comprenda a él mismo y a otros subprocesos, o del mismo modo un proceso puede incluir dentro de sí procesos de negocio menores, en éste sentido, un proceso de negocio puede ser visualizado desde distintos niveles jerárquicos, así también, el vínculo existente entre los procesos de negocio y la generación de valor permite visualizarlos como flujos de trabajo, los cuales realizan y representan todas las tareas de una organización (Becker, 2012).

Los procesos tienen las siguientes características:

- Pueden ser medidos.
- Gestión de calidad total.
- Sufren automatización.
- Se orientan al rendimiento.
- Reestructuración organizacional y valores compartidos.
- Generan un resultado concreto.
- Entregan resultados finales a los clientes.
- Se mejoran continuamente.
- Se producen en función de alguna acción o evento específicos.
- Constan con múltiples versiones.
- Las actividades añaden valor a las entradas del proceso.
- Sensibles al cambio.

Los procesos tienen las siguientes etapas:

- Identificación de los procesos estratégicos y operacionales o necesarios.
- Jerarquización del mapa de procesos para su rediseño y determinación de los procesos clave.
- Desarrollo de la visión de los nuevos procesos mejorados.
- Reingeniería de procesos.
- Preparación y prueba de los nuevos procesos.
- Mejora y actualización de procesos.

2.6.1. DEFINICIÓN DE PROCESO

“Conjunto de recursos y actividades interrelacionados que transforman elementos de entrada en elementos de salida. Los recursos pueden incluir personal, finanzas, instalaciones, equipos, técnicas y métodos.” (Carrasco, 2011)

En el libro Gestión de Procesos de Carrasco publicado el 2011, obtenida el 18 de diciembre cita: “La gestión de procesos es una disciplina de gestión que ayuda a la dirección de la empresa a identificar, representar, diseñar, formalizar, controlar, mejorar y hacer más productivos los procesos de la organización para lograr la confianza del cliente. La estrategia de la organización aporta las definiciones necesarias en un contexto de amplia participación de todos sus integrantes, donde los especialistas en procesos son facilitadores” (Cepeda, 2013)

2.6.1.1. Tipos de Procesos

- **Estratégicos**

Estos procesos dirigen el negocio hacia un objetivo determinado. Por ejemplo, "Planificar estrategias", "Establecer metas, logros y objetivos".

- **Centrales**

Estos procesos generan valor en el lado del cliente, constituyen la parte fundamental del negocio. Por ejemplo, "Distribución de productos"

- **De soporte**

Estos procesos sustentan los procesos centrales. Por ejemplo, "Contabilidad", "Servicio técnico", "Pagos".

Los procesos de negocio están constituidos internamente por: subprocesos, decisiones, tareas y actividades, un subproceso es considerado como una parte de un proceso de mayor nivel el cual tiene su propia meta, propietario, datos y resultados, las tareas son parte de los procesos de negocio que no incluyen ninguna toma de decisión ni vale la pena descomponer (aunque fuese posible hacerlo). Por ejemplo, "Contestar el teléfono" o "Elaborar un acta de entrega/recepción", y las actividades son las labores propias que una persona ejecuta para cumplir con una tarea (Casmon, 2011).

2.6.2. GESTIÓN DE PROCESOS DE NEGOCIO

Es la aplicación de metodologías orientadas a optimizar de forma sistemática los resultados de una empresa garantizando procesos

efectivos y eficientes que estén alineados a la estrategia y que aprovechen al máximo los beneficios de la tecnología y de los equipos de trabajo.

“Se llama Gestión de procesos de negocio (Business Process Management o BPM en inglés) a la metodología empresarial cuyo objetivo es mejorar la eficiencia a través de la gestión sistemática de los procesos de negocio, que deben ser modelados, automatizados, integrados, monitoreados y optimizados de forma continua.” (Cepeda, 2013)

Una de las muchas ventajas al usar la automatización de procesos, es la reducción de posibles y futuros errores, ya que según la estructura que se maneja, todos los procesos tendrán una similitud en corriente de trabajo, por ende trabajarán de una manera parecida. Al momento de administrar dichos procesos, se lo realizan de una manera eficiente y segura a la hora de la obtención de los datos que se utilizaran en el sistema (Cepeda, 2013).

Para hacer el seguimiento adecuado de ésta estrategia se necesita contar con herramientas que den el soporte necesario, haciendo cumplir con el ciclo de vida de la gestión de los procesos del negocio, dichas herramientas serán:

- Levantamiento de un sistema robusto (Normas ISO).
- Seguir normas legales (país o localidad).
- Automatización en tiempo real de procesos.
- Documentación de procesos.
- Manuales de procesos.
- Agilizar el levantamiento de tareas.
- Gestionar los cambios de proceso.
- Implementación de herramientas de auditoría.

- Adaptación de los procesos

El Ciclo de vida de los procesos de negocio se gestiona mediante estándares de tiempo y producción, dependiendo de lo extenso o complicado del mismo. Teniendo en cuenta que la buena optimización y el análisis darán como resultado un proceso mucho más sólido.



Gráfico 4. Ciclo de vida del BPM

Fuente: <http://bpmsincronicity.wordpress.com/>

2.7. FLUJO DE TRABAJO

Un flujo de trabajo (en inglés: workflow) es la representación de una secuencia de operaciones que define, crea y gestiona la ejecución de procesos a través del uso del software, siendo una estructura de tareas donde fluye la información de una manera sincronizada, siendo capaz de interpretar el proceso de definición e interactuar con los participantes de flujos de trabajo y, donde se requiere, recurrir al uso de herramientas y aplicaciones. Dentro de las actividades que se puede distinguir son las siguientes (Sommerville I. , 2005):

- Actividades colaborativas: Los usuarios trabajan en grupos, e interactúan sobre un repositorio común para obtener un resultado general.
- Actividades cooperativas: Los usuarios trabajan sobre repositorios independientes, y establecen mecanismos de cooperación entre ellos.
- Actividades de coordinación: Un usuario establece la dinámica de la cooperación entre grupos.

Los objetivos principales de un flujo de trabajo son:

- Automatizar los métodos del sistema informático.
- Establecer el control y seguimiento de los procesos.
- Separar al flujo de trabajo con los métodos
- Hacer transparente para las personas los procesos internos
- Gestionar y analizar de manera automática el intercambio de información entre capas
- Facilitar el desplazamiento de las personas dentro del sistema



Gráfico 6.WorkFlow

Fuente: <http://tech.gaeatimes.com/index.php/archive/top-10-java-workflow-engine/>

2.8. SISTEMAS DE DOCUMENTOS

2.8.1. ECM (ADMINISTRACIÓN DE CONTENIDO EMPRESARIAL)

A diario las empresas generan una serie de recursos informativos de clientes y propios de su administración lo que puede generar un caos generalizado en su manejo, para lo cual se emplea la ECM (Enterprise Content Management) que agiliza el control de la manipulación de estos documentos a través de técnicas y herramientas que se utilizan a lo largo de la vida útil de esta información.

El manejo a través del ECM tiene sus ventajas y beneficios que se describen a continuación:

- Reduce los gastos generales necesarios para almacenamiento de documentos físicos;
- Disminuye la mano de obra requerida para completar procesos críticos con documentación intensiva;
- Amplía la capacidad de colaboración, estimulada por el acceso en tiempo real al contenido;
- Aprovecha el capital intelectual en nivel elevado;
- Elimina atrasos e ineficiencias de los procesos manuales;
- Permite a los auxiliares de servicios de atención al consumidor resolver problemas en la primera llamada, por medio de informaciones precisas e instantáneas;
- Ofrece a los clientes acceso seguro vía Web ("self-service") a los documentos importantes;
- Asegura mejor sustentación a las iniciativas de gobierno, riesgos y compliance [GRC], habilitando acceso seguro al contenido corporativo, estandarizando actividades y aplicando retención de contenido con base en políticas;

- Ofrece prontitud avanzada y riesgo reducido a las cuestiones judiciales.

2.8.2. CMS (SISTEMA DE GESTIÓN DE CONTENIDOS)

El CMS (**Content Manager System**) cumple con la función como su nombre mismo lo indica de gestionar contenidos dirigidos a la web a través de una aplicación que presenta una interfaz administrativa permitiéndole de esta forma al administrador el manejo óptimo de documentos y recursos de la página (Trevejo Alonso, 2007).



Gráfico 7. Componentes de un CMS

Fuente: http://1.bp.blogspot.com/_Lw8Mmn7N_6E/TTYw8mS-J2I/AAAAAAAAAR4/f5e-BTiiVxE/s640/componentes

“Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido

y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo público.” (Nieto Rojas, 2011)

2.8.2.1. Estructura

“Las diferentes funcionalidades ofrecidas por un CMS pueden ser separadas en diferentes categorías. Todas juntas constituyen la estructura de un CMS.

2.8.2.2. Front-end y Back-end.

Un CMS consiste en un Front-end, que es el sitio web (website), que los visitantes y los usuarios registrados pueden ver. Y un Back-end, que contiene la parte de administración, localizado en una URL diferente al sitio web; y es donde se realizan las tareas de configuración, mantenimiento, limpieza, creación de estadísticas, etc.

2.8.2.3. Configuración.

Las decisiones de configuración que se aplican al sitio web son especificadas en “Configuration settings”. Esto incluye el título del sitio web, las palabras para los motores de búsqueda, opciones que permiten o prohíben darse de alta como usuario en el sitio web y muchas otras funciones.

2.8.2.4. Derechos de acceso.

En un CMS, los nombres de usuario son asignados a personas que están involucradas en el desarrollo del sitio web y su administración, y poseen diferentes derechos de acceso. El rango de derechos va desde un usuario simplemente registrado, registrado como autor o editor, hasta la categoría de “super administrador”, que tiene total control sobre el dominio. Según los derechos asignados, el sitio web muestra diferentes contenidos, o el usuario trabaja en diferentes partes de la administración aparte de en el sitio web (Trevejo Alonso, 2007).

2.8.2.5. Contenido.

Texto simple, imágenes, vínculos, música y archivos multimedia en general o una combinación de todos ellos. Para mantener una buena presentación de los contenidos, son incluidos en estructuras jerarquizadas, siendo clasificados según secciones y categorías, que a su vez, deben ser administrados.

2.8.2.6. Plantillas.

Una plantilla define los colores, las fuentes y su tamaño, imágenes de fondo, espaciado y distribución de la página, es decir, todo lo que tiene que ver con la apariencia del sitio web.

2.8.2.7. Extensiones.

Todo sistema debe poder ser ampliable para poder crecer en función de los requerimientos que aparezcan. Las funcionalidades

que pertenecen a determinado contexto son cubiertas por lo que se conoce como componente. Por ejemplo, componentes típicos son una tienda online, un gestor de usuarios, un gestor de listas de correo, un foro, una galería de imágenes, un gestor de descargas.

Los módulos que se necesitan integrar con los componentes son utilizados para insertar contenidos en la parte deseada dentro de la plantilla. Son ejemplos de módulos el que permite ver los usuarios conectados en el momento, el que permite ver las estadísticas del sitio, el que muestra los artículos de contenido más recientes, etc.

Son consideradas extensiones también las plantillas, los paquetes de idiomas, y también los llamados plugins, cuya funcionalidad principal, pero no la única, es interceptar los contenidos y realizar alguna modificación en ellos antes de ser mostrados en el navegador.

2.8.2.8. Rutina de trabajo.

Si hay varias personas trabajando, una buena rutina de trabajo facilitará la gestión de los contenidos. Por ejemplo, si un editor revisa una lista de noticias enviada al sitio para ser corregida, y después de revisarla indica que las noticias ya han sido corregidas, el jefe de redacción podrá directamente decidir si la noticia se publica en el Frontpage (página principal del sitio web) o no.

2.9. TIPOS DE GESTORES DE CONTENIDOS

2.9.1. POR SUS CARACTERÍSTICAS:

- Según el lenguaje de programación y frameworks empleados, como por ejemplo Active Server Pages, Java, PHP, ASP.NET, Ruby OnRails, Python.
- Según la licencia: Código abierto (como MySQL) o Software privativo.

2.9.2. POR SU USO Y FUNCIONALIDAD:

- Blogs: pensados para páginas personales (i.e. Blogger, WordPress, LifeType).
- Foros: pensados para compartir opiniones (i.e. phpBB, Simple Machines Forum, MyBB).
- Wikis: pensados para el desarrollo colaborativo (i.e. Wikipedia, Media Wiki).
- Enseñanza: Sistemas de Gestión de Contenidos para el Aprendizaje o LCMS (i.e. ATutor, Sakai, Moodle).
- Comercio electrónico: plataforma de gestión de usuarios, catálogo, compras y pagos (i.e. Magento, Opencart, Zen cart, osCommerce).
- Publicaciones digitales: i.e. Public Knowledge Project con sus respectivos Open Journal System, Open Conference System, etc.
- Difusión de contenido multimedia.
- Y por último habría que añadir los pensados para realizar portales y sedes Web (i.e. Drupal, Joomla, Xoops, Plone).

A esta clasificación se le podrían añadir otras categorías, como son:

- Según la base de datos que utilizan: MySQL, Access, etc.

- Según el entorno de trabajo: entorno Web o entorno “no Web” (de escritorio).
- Según el estilo de uso:
 - Servicio comercial.
 - Servicio de comunidad (el soporte depende de la comunidad de desarrolladores y usuarios).
 - Hospedado (Software as Service): Microsoft Office Live, Blogger, el mismo WordPress, etc.

2.10. INGENIERÍA DE SOFTWARE

Esta parte del desarrollo de una aplicación comprende prácticamente la vida de un sistema informático ya que se encuentra constituida por etapas que comprenden desde la estructuración de la aplicación así como su respectivo mantenimiento ya cuando esta se encuentra en funcionamiento. La ingeniería de software para aplicar su metodología requiere atravesar por 7 etapas descritas a continuación (Gabheran, 2012):

- a) **Análisis de requerimientos:** Se extraen los requisitos del producto de software. En esta etapa la habilidad y experiencia en la ingeniería del software es crítica para reconocer requisitos incompletos, ambiguos o contradictorios. Usualmente el cliente/usuario tiene una visión incompleta/inexacta de lo que necesita y es necesario ayudarlo para obtener la visión completa de los requerimientos. El contenido de comunicación en esta etapa es muy intenso ya que el objetivo es eliminar la ambigüedad en la medida de lo posible.
- b) **Especificación:** Es la tarea de describir detalladamente el software a ser escrito, de una forma rigurosa. Se describe el comportamiento esperado del software y su interacción con los usuarios y/o otros sistemas.

- c) **Diseño y arquitectura:** Determinar cómo funcionará de forma general sin entrar en detalles incorporando consideraciones de la implementación tecnológica, como el hardware, la red, etc. Consiste en el diseño de los componentes del sistema que dan respuesta a las funcionalidades descritas en la segunda etapa también conocidas como las entidades de negocio. Generalmente se realiza en base a diagramas que permitan describir las interacciones entre las entidades y su secuenciado.
- d) **Programación:** Se traduce el diseño a código. Es la parte más obvia del trabajo de ingeniería de software y la primera en que se obtienen resultados “tangibles”. No necesariamente es la etapa más larga ni la más compleja aunque una especificación o diseño incompletos/ambiguos pueden exigir que, tareas propias de las etapas anteriores tengan que ser realizadas en esta.
- e) **Prueba:** Consiste en comprobar que el software responda/realice correctamente las tareas indicadas en la especificación. Es una buena praxis realizar pruebas a distintos niveles (por ejemplo primero a nivel unitario y después de forma integrada de cada componente) y por equipos diferenciados del de desarrollo (pruebas cruzadas entre los programadores o realizadas por un área de test independiente).
- f) **Documentación:** Realización del manual de usuario, y posiblemente un manual técnico con el propósito de mantenimiento futuro y ampliaciones al sistema. Las tareas de esta etapa se inician ya en la primera fase pero sólo finalizan una vez terminadas las pruebas.
- g) **Mantenimiento:** En esta etapa se realizan un mantenimiento correctivo (resolver errores) y un mantenimiento evolutivo (mejorar la funcionalidades y/o dar respuesta a nuevos requisitos) (El Software, 2013).

2.10.1. ANÁLISIS DE REQUISITOS DE SOFTWARE

El análisis de requisitos es el primer paso del desarrollo de todo sistema software y, por lo tanto, resulta de gran importancia, ya que asienta la base del resto de etapas. Existen muchas formas de especificar los requisitos software y depende del equipo de desarrollo el utilizar una u otra.

2.10.1.1. IEEE 830

El estándar IEEE 830 fue generado por un equipo de trabajo del IEEE, su finalidad es la integración de los requerimientos del sistema desde la perspectiva del usuario, cliente y desarrollador. (IEEE Computer Society, 1998)

Respecto a las recomendaciones que realiza este estándar son:

- **Naturaleza**

La especificación de requisitos software es una especificación para un producto software en particular, programa, o conjunto de programas que realizan ciertas funciones en un entorno específico. Son cuestiones básicas que deben ser tratadas:

- Funcionalidad.
- Interfaces externos.
- Rendimiento.
- Atributos: corrección, mantenimiento, seguridad, transportabilidad, etc.
- Restricciones de diseño impuestas sobre la implementación.

- **Entorno**

La especificación de requisitos software debe:

- Definir correctamente todos los requisitos software. Un requisito software puede existir por la naturaleza de la tarea a resolver o por una característica especial del proyecto.
- No debe describir ningún diseño o detalle de implementación.

- No debe imponer ninguna limitación adicional al software.
- Características
 - Correcto
 - No ambiguo
 - Completo
 - Consistente
 - Clasificado por importancia y/o estabilidad
 - Verificable
 - Modificable
 - Rastreadable
- Evolución

La especificación de requisitos puede evolucionar según progresa el proceso de desarrollo del software.
- Inclusión de diseño.

La especificación de requisitos software no debe incluir generalmente cuestiones de diseño como:

- Partición del software en módulos.
- Asignar funciones a módulos.
- Describir flujos de información o control entre módulos.
- Elección de estructuras de datos.

METODOLOGÍA DE DESARROLLO DEL SOFTWARE

2.10.2. METODOLOGÍA RUP

De las siglas en inglés significa Rational Unified Process (Proceso Unificado de Racional) es un producto del desarrollo de software y junto con el lenguaje unificado de modelado (UML) que proporciona un enfoque disciplinado para asignar tareas y responsabilidades, que constituye la metodología estándar más utilizada dentro de una

organización del desarrollo. Es un sistema con pasos establecidos donde su meta es asegurar la producción del software de alta calidad que ayuda a la resolución de necesidades de los usuarios dentro de un tiempo establecidos (Metodología de Software, 2012).

2.11. PROCESOS DEL DESARROLLO

- **Adaptar el proceso:** Se debe adaptar a las necesidades del usuario o cliente, ya que es el actor más importante y el que interactuará directamente con el sistema.
- **Equilibrar prioridades:** Se debe encontrar un equilibrio entre los procesos prioritarios y los secundarios, sin descuidar ninguno y dando la importancia del caso a cada uno.
- **Demostrar valor iterativamente:** Los proyectos se entregan en etapas, se debe analizar y estabilizar cada una de las etapas para tener una alta calidad del producto final
- **Colaboración entre equipos:** Se debe tener una comunicación fluida entre todos los departamentos, sectores o personas que están involucradas en el desarrollo del sistema.
- **Elaborar el nivel de abstracción:** El principio es usar métodos reutilizables como frameworks. Esto brindará una mayor fluidez al sistema, haciendo que las personas encargadas del desarrollo no vayan directamente a los requisitos o a la codificación del sistema.
- **Enfocarse en la calidad:** El control se realiza al final de cada etapa, para pulir y evitar los errores en el producto final.

2.12. CARACTERÍSTICAS DEL RUP

Casos de Uso. En los procesos se refiere a la utilización de los Casos de Uso para el análisis y desarrollo de los procesos necesarios.

Los casos de uso representan la forma en como el cliente maneja el sistema en desarrollo, además de la forma y el orden de los distintos elementos.

Un diagrama de casos de uso consta de los siguientes elementos:

Actor : Una definición previa, es que un **Actor** es un rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra rol, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.

Como ejemplo a la definición anterior, se tiene el caso de un sistema de ventas en que el rol de Vendedor con respecto al sistema puede ser realizado por un Vendedor o bien por el Jefe de Local.

Casos de Uso: Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso.

Relaciones: son los tipos de invocaciones o llamadas que hacen de un actor a un caso de uso (Pérochon, 2012).

2.13. FASES DEL RUP

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un

mayor o menor hincapié en los distintas actividades (Metodología de Software, 2012).

2.13.1. Inicio

Esta fase tiene como propósito definir y acordar el alcance del proyecto con los patrocinadores, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software y producir el plan de las fases y el de iteraciones posteriores.

2.13.2. Elaboración

En la fase de elaboración se definen los casos de uso que permiten definir la arquitectura base del sistema y se desarrollarán en esta fase, se realiza la especificación de los casos de uso seleccionados y el primer análisis del dominio del problema, se diseña la solución preliminar.

2.13.3. Construcción

El propósito de esta fase es completar la funcionalidad del sistema, para ello se deben clarificar los requisitos pendientes, administrar los cambios de acuerdo a las evaluaciones realizados por los usuarios y se realizan las mejoras para el proyecto.

2.13.4. Transición

El propósito de esta fase es asegurar que el software esté disponible para los usuarios finales, ajustar los errores y defectos encontrados en las pruebas de aceptación, capacitar a los usuarios y proveer el

soporte técnico necesario. Se debe verificar que el producto cumpla con las especificaciones entregadas por las personas involucradas en el proyecto.

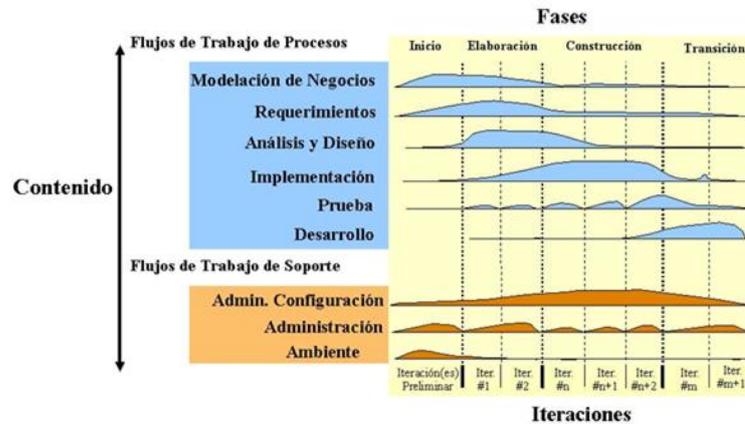


Gráfico 8. Flujos de procesos

Fuente: http://metodologiadesoftware.blogspot.com/2012/11/fases-del-modelo-rup_27.html

2.14. CICLOS DE VIDA DE DESARROLLO

Los ciclos de vida del desarrollo de un sistema informático son todas y cada una de las etapas por las cuales pasa la estructuración de una aplicación, iniciando desde el planteamiento de la idea y el análisis de los diferentes requerimientos para su ejecución, hasta su etapa terminal de uso donde es remplazado por otra aplicación.



Gráfico 9. Ciclo de vida del software

Fuente:<http://grupo3trayecto2trimestre1seccion2.blogspot.com/2013/03/vision-general-del-proceso-de.html>

Análisis: Construye un modelo de los requisitos

Diseño: A partir del modelo de análisis se deducen las estructuras de datos, la estructura en la que se descompone el sistema y la interfaz de usuario.

Codificación: Construye el sistema. La salida de esta fase es código ejecutable.

Pruebas: Se comprueba que se cumplen criterios de corrección y calidad.

Validación: es el proceso de comprobar que lo que se ha especificado es lo que el usuario realmente quería.

Mantenimiento: En esta fase, que tiene lugar después de la entrega se asegura que el sistema siga funcionando y adaptándose a nuevos requisitos.

2.15. EL USUARIO DENTRO DEL PROCESO DE DESARROLLO DE SOFTWARE

El usuario es un ente más que importante en la fase de desarrollo de una aplicación, ya que es quien va a ser el involucrado directo en el correcto funcionamiento de la misma cuando ya se encuentre en ejecución, por lo que su intervención va asegurar el éxito en el uso para el cual fue destinado el sistema (Puon, 2014).

A continuación se presenta algunas matizaciones:

1) El proyecto no se hace sólo, porque incluso existiendo una gran ayuda por parte de los usuarios, si no se consigue interpretar con precisión lo que

quieren y no se dinamiza un feedback continuo de los mismos durante todo el proceso de desarrollo, se incrementarán las posibilidades de que algún requisito funcional no se haya recogido adecuadamente o de que se haya realizado un software con una usabilidad incómoda para los usuarios.

Estas circunstancias son fuente de innumerables problemas en las fases finales del proyecto y provocan retrasos, sobrecostes y grandes dificultades para cerrar el proyecto, además de crear conflictos con el cliente que pueden perjudicar las relaciones futuras con el mismo. Esto hace que sea fundamental el papel que desempeñan tanto el jefe de proyectos, como el equipo de analistas funcionales y analistas programadores.

2) Es importante que entre el grupo de usuarios asignados al proyecto haya usuarios que vayan a estar implicados en el futuro uso del sistema de información, es decir, no es suficiente que el equipo de usuarios esté formado por “ideólogos” o “teóricos” que se nutrirán del resultado del trabajo de la herramienta, sino que es fundamental que participen usuarios que después se tengan que poner el mono de trabajo y vayan a trabajar con el software. Es importante conseguir la combinación de ambos tipos de usuarios (tampoco es positivo que en el grupo de usuarios no participen usuarios directores, ya que pueden existir conflictos entre usuarios que éstos deben solucionar y también es recomendable que el software no sólo se diseñe para el corto plazo, sino que sirva para tareas de gestión, planificación, etc... y esta visión la proporcionan principalmente los usuarios directores), por lo que el jefe de proyectos debe poner en conocimiento del cliente esta necesidad, como es lógico explicando los riesgos de que no se aplique esta estrategia.

3) Los analistas están para ayudar y para colaborar con los usuarios en la especificación y diseño de la solución, pero no están para “dar lecciones” a los usuarios y enseñarle cómo deben hacer su trabajo. Si los usuarios hacen su trabajo de una determinada manera, aunque no sea la más ortodoxa, siempre tendrá una justificación que sólo se entendería si realmente se estuviera haciendo su trabajo durante un tiempo y se observara los

problemas con los que se enfrentan cotidianamente. La clave por tanto está en la colaboración y en el diálogo, es decir, se pueden proponer cosas al usuario, se le pueden dar ideas, pero no se le puede dar una vuelta al calcetín de cómo hacen sus tareas, salvo que ellos mismos lo soliciten y procurando en estos casos y en consenso con los usuarios que los cambios sean tranquilos. (Condesa, 2011)

4) Es fundamental documentar el proyecto, en primer lugar con la documentación que se especifique en las normativas de desarrollo de la organización para la que se realiza el servicio, con las matizaciones que indique el Director del Proyecto, en segundo lugar con la documentación que establezcan las normativas internas de calidad de la organización (no requerirá un sobreesfuerzo, ya que en la mayor parte de los casos coincidirá) y a todo lo anterior sumarle toda la documentación de trabajo que sea necesaria para trabajar con los usuarios, que no tienen por qué entender de modelos de datos, de diagramas de casos de uso, etc..., es más, es un error trabajar con los usuarios utilizando dichas herramientas, ya que estas son de utilidad técnica y no hablan el mismo lenguaje de los usuarios. Este tipo de documentación, por tanto, no tiene por qué tener los formalismos de la técnica y tiene como objetivo que el usuario capte lo que el analista está interpretando y se pueda ir perfilando a partir de esto, tanto requisitos, como casos de uso, interfaces, etc... Es muy importante trabajar todo esto, ya que comenzar demasiado pronto con la construcción, es algo muy arriesgado, ya que los costes de modificar algo en las distintas fases de la construcción pueden ser muy importantes y provocar que se tengan que reconstruir varias veces distintas funcionalidades de la aplicación. (Fling, 2009).

2.16. METODOLOGÍA DE INVESTIGACIÓN

La metodología es el instrumento que enlaza el sujeto con el objeto de la investigación. Sin la metodología es casi imposible llegar a la lógica que conduce al conocimiento científico. (El Software, 2013)

2.16.1. MÉTODO DEDUCTIVO

En este método se desciende de lo general a lo particular, de forma que partiendo de enunciados de carácter universal y utilizando instrumentos científicos, se infieren enunciados particulares, pudiendo ser axiomático-deductivo cuando las premisas de partida la constituyen axiomas (proposiciones no demostrables), o hipotético-deductivo si las premisas de partida son hipótesis contrastables (Sommerville I. , 2005).

2.16.2. MÉTODO SISTÉMICO

Es un proceso mediante el cual se relacionan hechos aparentemente aislados y se formula una teoría que unifica los diversos elementos. Consiste en la reunión racional de varios elementos dispersos en una nueva totalidad, este se presenta más en el planteamiento de la hipótesis. El investigador sintetiza las superaciones en la imaginación para establecer una explicación tentativa que someterá a prueba. También como pensamiento sistemático se encuentra que este método consiste en identificar algunas reglas, algunas series de patrones y sucesos para preparar a la persona de cara al futuro e influir en alguna medida. Está dirigido a modelar el objeto mediante la determinación de sus componentes, así como las relaciones entre ellos. Esas relaciones determinan por un lado la estructura del objeto y por otro su dinámica (Metodología de Software, 2012).

El método sistémico vendría a ser un orden manifestado por reglas, que permite llegar a tener una comprensión sistémica de una situación dada. Quien intente utilizar el método sistémico deberá:

- Conocer los rasgos fundamentales del sistema (o subsistema) bajo estudio: componentes, medio, y estructura, utilizando a tal fin los conceptos y arquetipos básicos brindados por el pensamiento sistémico. En los casos que sean necesarios se contemplará la posibilidad de profundizar el conocimiento de la estructura por medios

matemáticos que aporten las disciplinas vinculadas a la Teoría General de Sistemas.

- Poder diferenciar entre las propiedades del sistema, cuales son resultantes y cuales emergentes y definir cuál es el estado atractor del sistema (si lo tuviera).
- Integrar el hecho particular bajo análisis en el sistema en su conjunto.
- Interpretar el hecho dentro de la estructura y evolución del sistema

METODOLOGÍA

3. METODOLOGÍA

La ejecución del presente trabajo de titulación se lo realizó en las instalaciones de la empresa AUVICOM, donde se desarrolló un exhaustivo análisis sobre todo lo relacionado a la generación de órdenes de trabajo, para realizar la implementación de una aplicación.

La empresa AUVICOM facilita toda la información necesaria sobre el funcionamiento de los procesos que ellos manejan con respecto a la ejecución de órdenes de trabajo, partiendo de una proforma que se la desarrolla previo a la realización de un trabajo determinado en la empresa.

Basándose en el método deductivo se identifican los procesos para describir los actores y actividades involucrados en cada uno de ellos. Además se apoyó en el método sistémico el cual es utilizado en desarrollo de software ya que con este método se involucra elementos aparentemente externos del sistema como por ejemplo la infraestructura tecnológica y el recurso humano.

Para la identificación de los requisitos, y levantamiento de procesos se apoyó en el modelamiento UML (Lenguaje Unificado de Modelado), ya que es representado por una serie de diagramas especiales para modelamiento de sistemas orientado a objetos.

Para el análisis y desarrollo de la aplicación móvil se utilizará la metodología RUP la cual es fácilmente adaptable a cualquier organización y además es estrechamente ligada al modelamiento UML.

Para el desarrollo de la aplicación se utiliza el IDE Visual Estudio 2010 implementando las clases necesarias en lenguaje de programación Visual Basic. Net. Este IDE de desarrollo que habilita el uso compartido de herramientas y hace más sencilla la creación de soluciones en varios lenguajes.

3.1. FASE 1: INICIO

3.1.1. Levantamiento de Información

Se estableció como herramienta para la investigación la entrevista, dirigida especialmente a las personas que eran actores directos en cada uno de los procesos de la empresa, teniendo dos frentes para sacar conclusiones, por un lado se tiene a la empresa AUVICOM, quien con su personal operacional ofrecen el servicio técnico a los clientes y se pudo obtener las diferentes experiencias que habían tenido con respecto al anterior manejo de los procesos para la ejecución de una orden de trabajo. El otro ente que pertenece al proceso es el cliente, a quien se le comentó la necesidad de realizar una aplicación web para automatizar los procesos y de esta manera optimizar los tiempos de respuesta a la petición que él podía realizar a la empresa a través de la aplicación antes mencionada.

3.2. FASE 2: ELABORACIÓN

En la etapa de elaboración se ejecutó un análisis sobre los diversos procesos que intervienen en la ejecución de una orden de trabajo, interpretados a través de UML con la utilización de diagramas donde se interpreta a los elementos de los procesos de la aplicación se puede detallar en gran medida toda la estructura de sistema y la funcionalidad de cada uno de los procesos con los cuales cuenta la aplicación

Para la estructuración se utilizó:

- Diagramas de casos de uso
- Diagramas de Secuencia
- Diseño de Base de Datos
- Diagrama de Clases

3.2.1. ARQUITECTURA DEL SISTEMA

Las aplicaciones de hoy en día han tomado nuevos rumbos con respecto a la forma en la que realizan sus funciones, así como el medio por el cual este llega a convertirse en herramienta de uso por parte del usuario, es por eso que con la utilización masiva del internet se estableció que la arquitectura del mismo esté orientada y dirigida a la web, teniendo como herramienta de ingreso una aplicación android que es la puerta por la cual se va a llegar a la manipulación del sistema con todas sus características de acuerdo a los roles que este le dé a la persona que lo manipule.

Una aplicación web está destinada a ser utilizada en una infinidad de plataformas ya que al encontrarse en la nube, cualquier ordenador o dispositivo móvil con acceso al red va a tener la opción de acceder, y en este caso en particular con el proyecto actual está destinado a dispositivos móviles que van a ser manipulados por los usuarios y los operadores de la empresa AUVICOM, a través de un dispositivo móvil que tenga instalado el sistema operativo android como software base.

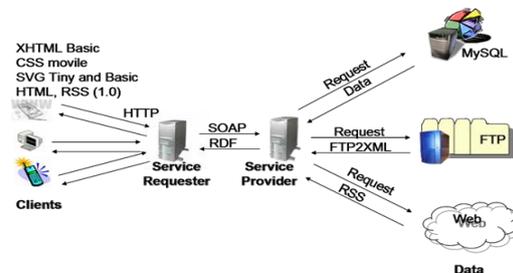


Figura 14. Imagen funcionamiento aplicación móvil

Fuente: http://www.w3c.es/gira/premio/2/concursow3c_archivos/image002.png

3.2.1.1. Patrón de diseño

Para poder realizar la correcta estructuración del sistema, se necesitaba definir un patrón de diseño que se adapte a los

requerimientos necesarios para el desarrollo del proyecto, por lo cual se escogió el llamado **MVC (MODELO – VISTA – CONTROLADOR)**, “Es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El estilo de llamada y retorno MVC, se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.” (Sommerville I. , 2006)

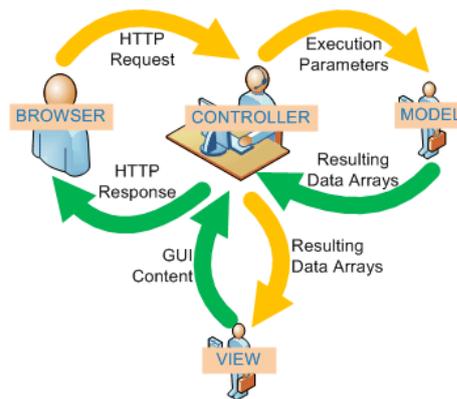


Gráfico 11. MVC(MODELO VISTA CONTROLADOR)

Fuente: <http://bittacorp.files.wordpress.com/2008/12/mvc.gif>

3.3. FASE 3: CONSTRUCCIÓN

Una vez realizado un análisis completo a todos los requerimientos del sistema, partiendo del estudio previamente hecho, se procede a utilizar la arquitectura de desarrollo de software MVC. Para no tener problemas con futuros cambios se dividió los componentes del sistema en tres capas que son: la capa de datos, la capa de interfaz y la capa lógica.

3.3.1. MAQUETACIÓN

La propuesta de la empresa AUVICOM para la elaboración de la interfaz de usuario es que sea algo sencillo con un requerimiento de recursos mínimo y bajo consumo en transferencia de datos y por ende sea veloz al momento de realizar los procesos solicitados por el usuario.

3.4. FASE 4: TRANSICIÓN

3.4.1. PRUEBAS Y DESPLIEGUE

Al finalizar el desarrollo, se procede a la implementación del sistema en dispositivos móviles propios de la empresa para comprobar con datos reales los alcances de la aplicación en su fase BETA.

3.4.1.1. PRUEBAS DE HUMO

Se ejecutará el sistema constantemente de extremo a extremo es decir en la totalidad de su funcionalidad durante varios días intentando encontrar errores que normalmente no se detectan con las pruebas normales y a la vez determinando la aceptación del sistema.

Este tipo de pruebas se basa en la funcionalidad de los casos de uso es decir se ejecuta el sistema en base a las descripciones de los casos de uso descritos en el modelamiento del sistema determinado errores en la funcionalidad y aprobando la usabilidad del sistema

Las pruebas se realizarán en dos dispositivos con sistema operativo Android, el primero será una tablet Marca Netodragon provista de un procesador Dual Core A23 de 1.6Ghz, 2 Gb de memoria Ram y Android 4.4.2. Esta tablet se utilizará en el área técnica de la empresa AUVICOM; el segundo dispositivo será un celular Marca Motorola Modelo XT925 el cual será utilizado por el personal técnico de la empresa cuando se necesite tomar una orden de trabajo en sitio.

RESULTADOS Y DISCUSIÓN

4. RESULTADOS Y DISCUSIÓN

El objetivo primordial del presente trabajo es describir todos y cada uno de los procesos que conllevan la emisión de una orden de trabajo de mantenimiento o arreglo de computadoras por parte de la empresa AUVICOM, cuyos procesos están determinados por:

- Proceso estructuración de la proforma
- Proceso emisión orden de trabajo

En el capítulo anterior se definió a la metodología de desarrollo RUP y a continuación se detalla las fases de su estructuración.

4.1. FASE INICIAL – ANÁLISIS

La fase de análisis es la raíz de toda la estructuración del sistema ya que abarca todos los requisitos y necesidades que el cliente solicita. En esta parte se encajona todas las características con las cuales se dará forma al sistema y se enfocará el desarrollo pensando siempre en su evolución de acuerdo a los requerimientos de cada etapa por las cuales el sistema va a ser la herramienta de trabajo de la empresa, para la especificación de requerimientos del sistema se utilizó el estándar IEEE830, el mismo que tiene como finalidad es la integración de los requerimientos del sistema desde la perspectiva del usuario, cliente y desarrollador.

4.1.1. INTRODUCCIÓN

4.1.1.1. PROPÓSITO

El propósito del presente trabajo de investigación es el de realizar un análisis sobre todas la inquietudes que tengan los entes

relacionados en el proceso de la emisión de una orden de trabajo posterior a la entrega de una proforma para la empresa AUVICOM.

Esta información recabada a los empleados y clientes de la empresa abre una visión global de la estructura del sistema con todas y cada una de sus especificaciones.

4.1.1.2. ÁMBITO DEL SISTEMA

En la actualidad la empresa AUVICOM no cuenta con una aplicación web que esté destinada a dispositivos móviles, por lo cual este documento se convierte en una guía de toda la estructura del sistema y sus diferentes funcionalidades explicadas de forma clara en cada uno de los procesos que componen la aplicación.

4.1.2. DESCRIPCIÓN GENERAL

4.1.2.1. PERSPECTIVA DEL PRODUCTO

Como producto se ha implementado una aplicación web destinada para la utilización a través de dispositivos móviles y su beneficio por parte de los diferentes usuarios sean estos empleados de la empresa o intervengan como la representación de un cliente, logrando de esta forma optimizar los diversos procesos que forman parte de la ejecución de una orden de trabajo por parte de la empresa antes mencionada.

4.1.2.2. FUNCIONES DEL PRODUCTO

El producto consta de diferentes funciones que involucran a cada uno de los usuarios que van a manipular el sistema y que se los describe a continuación:

- Gestiona la recopilación de los datos del cliente para almacenarlos en la base de datos y poseer un registro.
- Elabora una proforma de trabajo referente a los datos proporcionados por el cliente donde se especifica los costos de trabajo y los tiempos estimados de entrega del mismo.
- Recpta confirmaciones de trabajo por parte de los clientes.
- Elabora la orden de trabajo que ya ha sido confirmada y designa al personal quien se va hacer del trabajo.
- Administra los estados de las cuentas, eliminando de esta forma las proformas caducadas y registrando siempre todos esos movimientos.

4.1.2.3. IDENTIFICACIÓN DE USUARIOS PARTICIPANTES

En el punto actual se especifica cada uno de los usuarios que intervienen en el funcionamiento del sistema que son:

- Empresa
- Administrador
- Operario
- Técnico
- Cliente

4.1.2.4. RESTRICCIONES

El presente trabajo tiene como limitantes la implementación del sistema para su uso, ya que no todos los clientes disponen de un

teléfono inteligente con sistema operativo de última tecnología para una correcta visualización y utilización de la aplicación.

4.1.3. SUPERVISAR PROYECTOS DE INVESTIGACIÓN

4.1.3.1. Requerimientos Funcionales

- ✓ El cliente llenará los datos necesarios para enviar la solicitud de una proforma
- ✓ Los administrativos de la empresa revisan los datos enviados por el cliente y proceden a elaborar una proforma con valores y tiempos estimados; dicha proforma se la guarda internamente en el sistema.
- ✓ El cliente puede entrar al sistema y revisar las proformas con datos actualizados.
- ✓ El Operario llena la información de la orden de trabajo con los datos específicos del cliente y tipo de trabajo que se desea realizar, los mismos que son guardados y generan un reporte que es enviado vía correo electrónico tanto al cliente como a la empresa.
- ✓ La empresa revisa las ordenes de trabajo y coordina con sus técnicos para proceder a la iniciación del trabajo
- ✓ Los técnicos actualizan periódicamente los estados de la orden de trabajo

4.1.3.2. REQUISITOS NO FUNCIONALES

4.1.3.2.1. Rendimiento

La información almacenada podrá ser consultada y actualizada permanentemente y simultáneamente, debe estar en capacidad de dar respuesta y acceso a todos los usuarios sin que afecte el tiempo de respuesta aceptable del mismo además todas las peticiones de los usuarios se solventarán en tiempo real.

La información guardada en la base de datos a través del sistema, será capaz de ser consultada y actualizada de forma instantánea y sin tiempo de demora por todos los tipos de usuarios que tendrán acceso al mismo. Las peticiones que se programan de tal forma que la estructura de abasto a la entrada de usuarios en tiempo real.

4.1.3.2.2. Seguridad

El acceso al sistema será restringido por una línea jerárquica de usuarios. Además cada usuario contará con un perfil único anexado a su número de identificación único (Cédula), también contando con una contraseña y nombre de usuario únicos; dichos datos serán almacenados con una llave de encriptación.

4.1.3.2.3. Disponibilidad

El sistema tendrá alta disponibilidad en tiempo real, durante el rango de tiempo 24/7/365, ya que se orientó a un almacenamiento web y contará con un servidor que permanecerá encendido todo el tiempo con energía de respaldo, siendo así capaz de visualizar en cualquier momento a través de un dispositivo móvil.

4.1.3.2.4. Mantenibilidad

El sistema deberá ser diseñado para que tenga un mantenimiento de fácil acceso, y de esta manera pueda ser ampliado y corregido en caso de ser necesario.

A la vez, el sistema deberá estar completamente documentado, con cada uno de los componentes referenciales que hagan falta, tales como:

- Manuales de usuario (ver anexo 2)
- Código Comentado (ver anexo 3)

4.1.3.2.5. Portabilidad

El sistema debe ser portable, para que se pueda ejecutar en diferentes dispositivos sin dificultad, haciendo énfasis y refiriéndose al sistema operativo móvil Android, haciendo que sea de un ágil acceso a las diferentes plataformas como tabletas, celulares, y demás dispositivos que cuenten con una función touch y tengan una señal de internet.

4.2. FASE DE ELABORACIÓN

UML sirve para especificar, visualizar y documentar esquemas de sistemas de software orientado a objetos. UML no es un método de desarrollo, lo que significa que no sirve para determinar qué hacer en primer lugar o cómo diseñar el sistema, sino que simplemente le ayuda a visualizar el diseño y a hacerlo más accesible para otros. UML está controlado por el grupo de administración de objetos (OMG) y es el estándar de descripción de esquemas de software.

UML está diseñado para su uso con software orientado a objetos, y tiene un uso limitado en otro tipo de cuestiones de programación.

4.2.1. DIAGRAMAS CASOS DE USO

DEFINICIÓN DE ACTORES

Actor No.1 Super usuario



Actor No.2 Administrador



El administrador del sistema, es la persona encargada de ingresar a todas las funciones de configuración, como también la actualización de la información de esta.

Actor No.3 Operario



Actor No.4 Técnico



Actor No.5 Cliente



El Cliente, es la persona que realiza el recorrido e interactúa con el sistema.

Caso de Uso Usuario Super Usuario

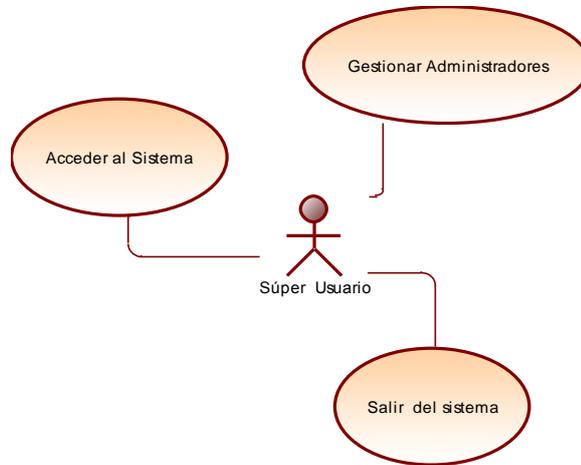


Figura 1. Diagrama Caso de Uso 1 - Actor Super Usuario

Caso de Uso Usuario Administrador

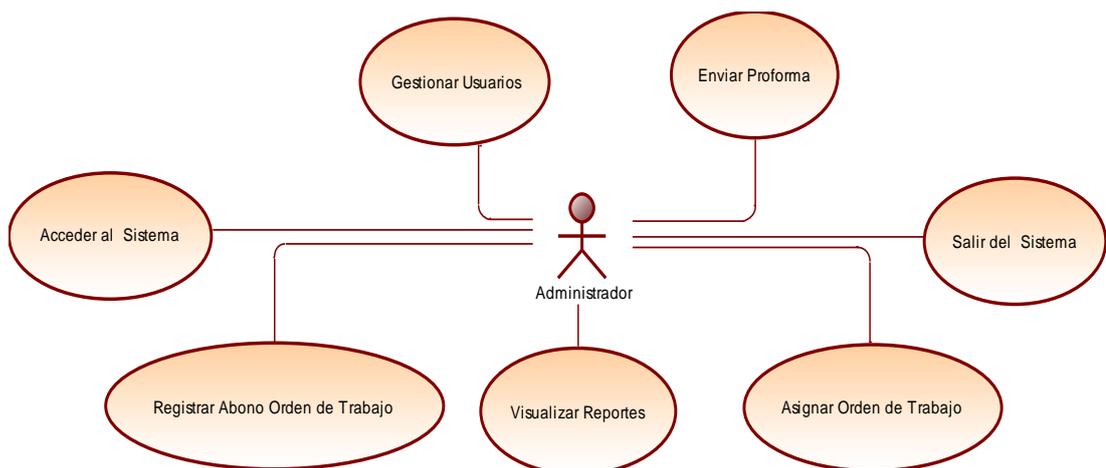


Figura 2. Diagrama Caso de Uso 2 – Actor Administrador

Caso de Uso Usuario Operario

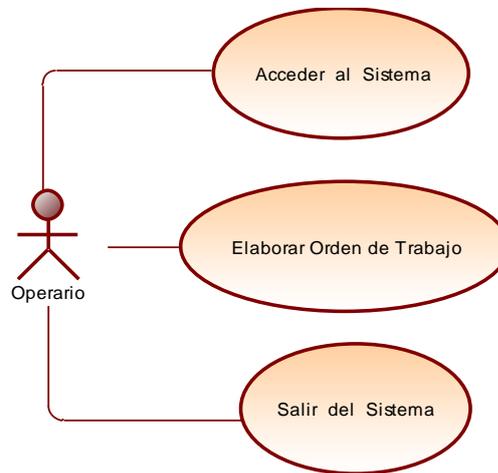


Figura 3. Diagrama Caso de Uso 3 – Actor Operario

Caso de Uso Usuario Técnico

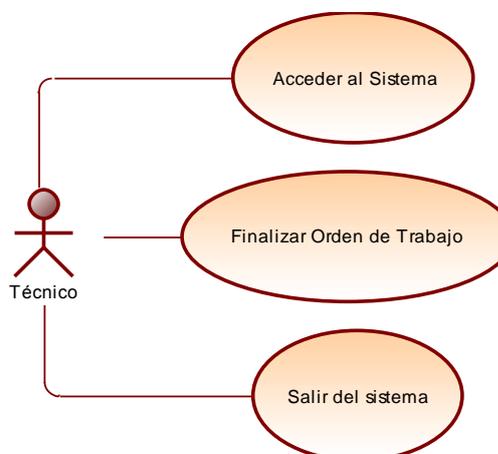


Figura 4. Diagrama Caso de Uso 4 – Actor Técnico

Caso de Uso Usuario Cliente

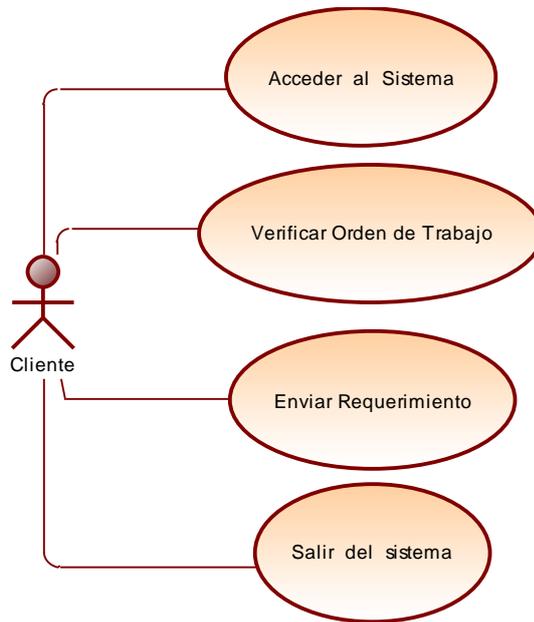


Figura 5. Diagrama Caso de Uso 5 – Cliente

Caso de Uso Sistema

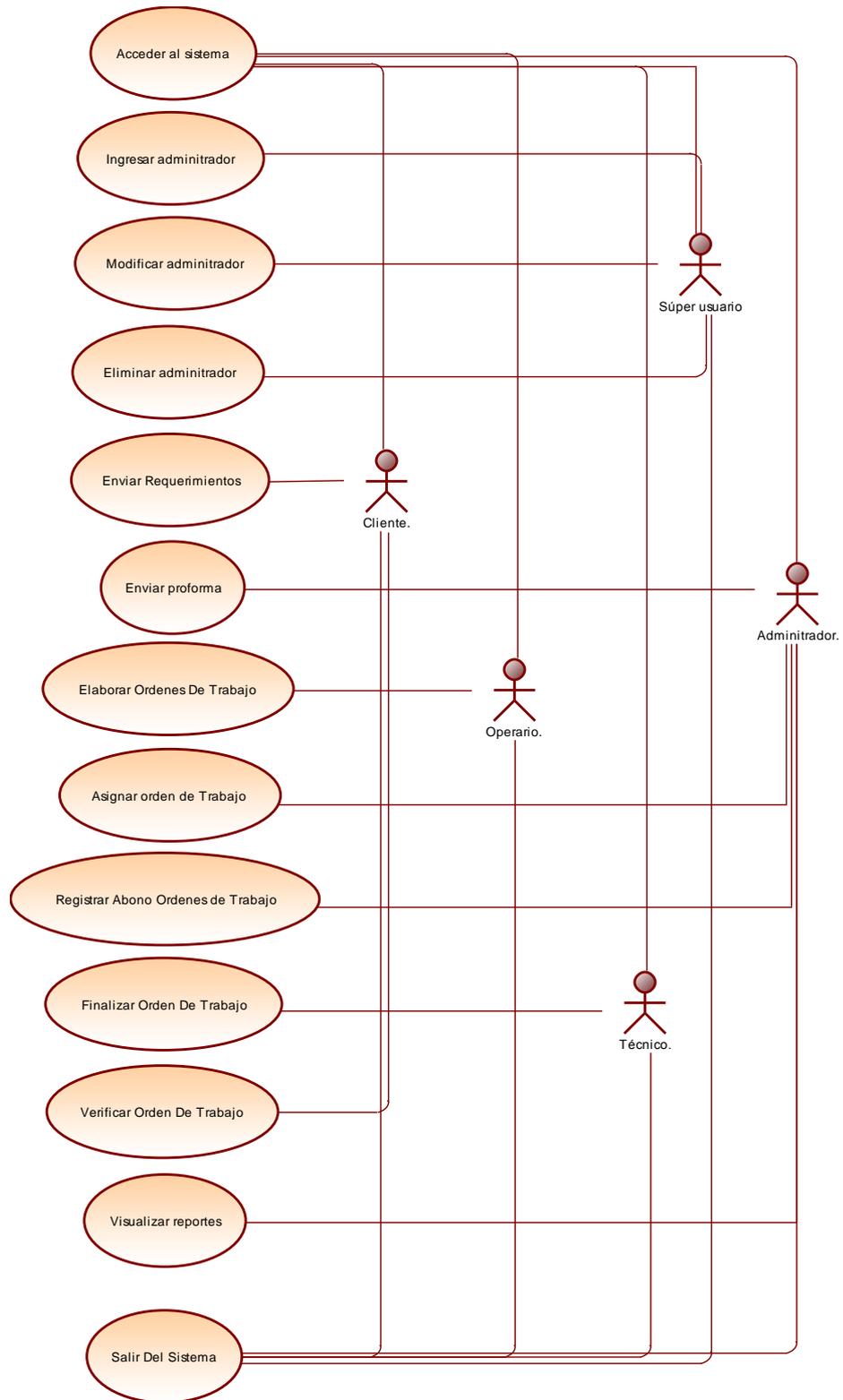


Figura 6. Diagrama Caso de Uso 6 - Sistema

CASOS DE USO

Especificación de Caso de Uso: Ingreso al sistema

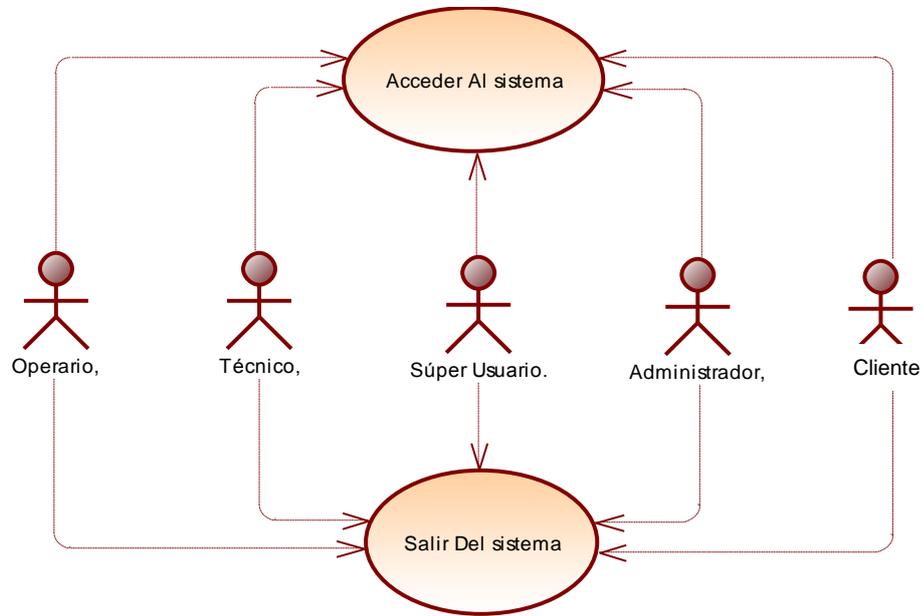


Figura 7. Diagrama Caso de Uso 7 – Ingreso al Sistema Usuario General

Tabla 1 Especificación del caso de uso 7

Especificación del caso de uso			
Nombre del caso de uso:	Ingreso al sistema	Versión No:	1.0
Descripción:	Posibilita al actor para que acceda al sistema a través de sus datos previo a un registro.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario en general		

Tabla 2. Flujo Básico Caso de Uso 7

Flujo Básico		
Paso	Actor	Sistema
1	El usuario ingresará al sistema a través de una aplicación android.	El sistema muestra en pantalla las opciones disponibles para ingresar los datos de acceso (Login y Password).
2	El usuario ingresará sus datos de identificación.	El sistema validará los datos ingresados y le asignará el nivel de acceso que este va a disponer.
3	El usuario accede a la pantalla principal donde se encuentra el menú otorgado.	El sistema muestra las opciones a las cuales puede acceder el usuario de acuerdo al nivel que el sistema le haya asignado.

Tabla 3. Flujo Alternativo Caso de Uso 7

Flujo Alternativo		
Paso	Actor	Sistema
1	El usuario ingresa sus datos de verificación de forma errónea.	El sistema le muestra en pantalla un mensaje notificándole el error cometido para que ingrese correctamente sus datos y los campos se enceran.
2	El usuario no está registrado en el	El sistema en su pantalla principal va a

	sistema	presentar un mensaje notificando al usuario que no se encuentra registrado en el sistema, para lo cual debe dirigirse al administrador y realizar la petición de registro.
--	---------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Especificación de Caso de Uso: Ingreso usuario /administradores

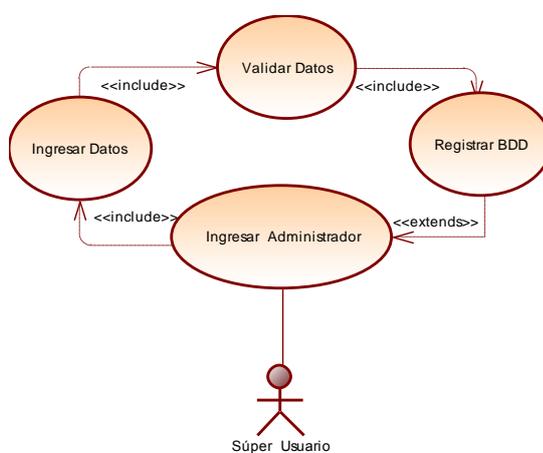


Figura 8. Diagrama Caso de Uso 8 – Ingresar Administrador

Tabla 4. Especificación del caso de uso 8

Especificación del caso de uso			
Nombre del caso de uso:	Ingreso usuario / clientes	Versión No:	1.0
Descripción:	Posibilita al actor para que realice el registro de usuarios al sistema.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Super usuario		

Archivos:

- Wfrm_usuarios_ingreso.aspx

- Wfrm_usuarios_ingreso.aspx.vb
- Cls_usuarios.vb -Cls_Telefonos.vb
- Style.css
- Base de datos : BDDRegister

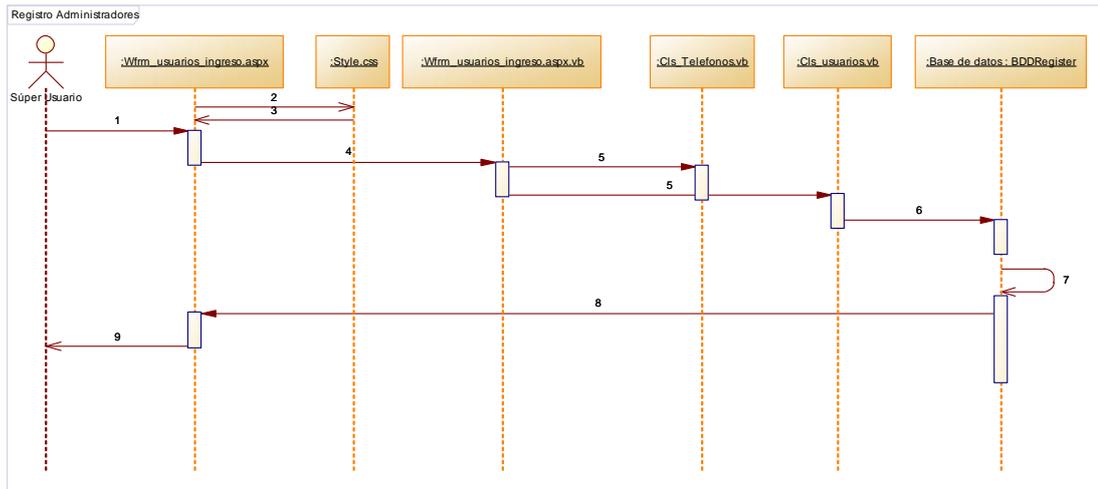
Tabla 5. Flujo Básico Caso de Uso 8

Flujo Básico		
Paso	Actor	Sistema
1	El super usuario ingresará al sistema a través de sus credenciales.	El sistema despliega en pantalla todas las opciones disponibles para la manipulación del sistema.
2	El super usuario realiza el registro de usuarios administradores.	En pantalla se despliega un formulario con diferentes campos para el ingreso de la información de los usuarios.
3	El super usuario realiza la selección del rango de administrador	El sistema carga por default los 4 tipos de usuarios para su selección.
4	El super usuario llena los campos	El sistema valida campo por campo para controlar las restricciones internas (números, no se permiten caracteres especiales, cédula, correo, solo letras y campos requeridos).
5	El super usuario presiona en el botón ingresar.	El sistema envía toda la información de los campos a la base de datos y se presenta un mensaje en pantalla notificando el ingreso, teniendo el super usuario una opción en el mensaje donde acepta la acción que se ejecutó en el sistema.

6	El super usuario realiza la consulta de los datos ingresados en el sistema.	El sistema muestra en pantalla un filtro para realizar la consulta de datos a través de la cédula del usuario en el registro.

Tabla 6. Flujo Alternativo Caso de Uso 8

Flujo Alternativo		
Paso	Actor	Sistema
1	El super usuario ingresa datos de registro de usuarios de manera errónea	El sistema a través sus restricciones inhabilita el botón de ingresar, hasta que el super usuario digite los datos correctos.
2	El super usuario no ingresa registro alguno.	El sistema a través sus restricciones inhabilita el botón de ingresar, hasta que el super usuario digite los datos correctos.



Notación:

- 1: Abrir formulario
- 4: Ingresar campos
- 5: Validar restricciones
- 6: Cargar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 9. Diagrama de Secuencia 1 – Registro Administradores

Especificación de Caso de Uso: Modificación De Administradores

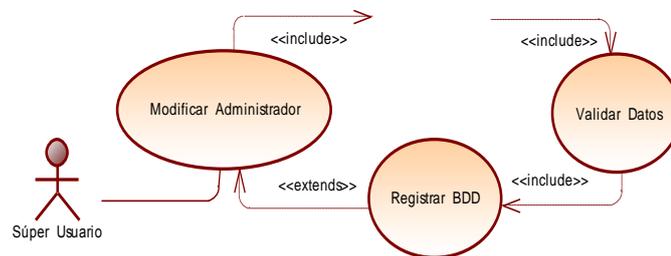


Figura 10. Diagrama Caso de Uso 9 – Modificar Administrador

Tabla 7. Especificación del caso de uso 9

Especificación del caso de uso			
Nombre del caso de uso:	Modificación de administradores	Versión No:	1.0
Descripción:	Posibilita al actor para que acceda al sistema a través de sus datos y realice la modificación de la información del usuario administrador.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Super usuario		

Archivos:

- Wfrm_usuarios_ingreso.aspx
- Wfrm_usuarios_ingreso.aspx.vb
- Cls_Utilitarios.vb -Cls_Telefonos.vb
- Style.css
- Base de datos : BDDRegister

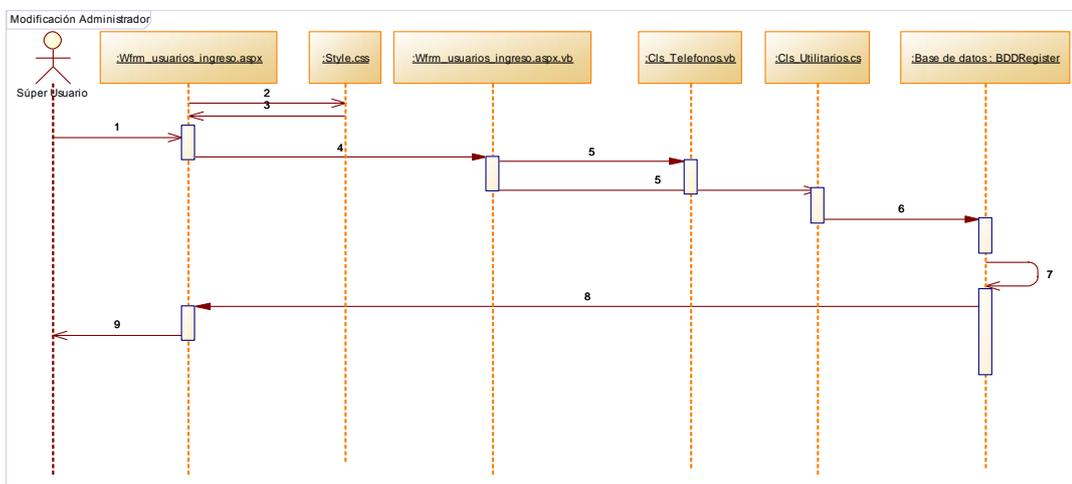
Tabla 8. Flujo Básico Caso de Uso 9

Flujo Básico		
Paso	Actor	Sistema
1	El super usuario realiza la consulta del usuario administrador en el sistema.	El sistema muestra en pantalla la información referente al usuario administrador que se encuentre registrado con esa cédula.
2	El super usuario realiza la modificación de los datos del usuario administrador.	El sistema habilita todos los campos referentes a la información del sistema para su respectiva modificación, y

		respectivo almacenamiento de datos en la base a través del botón modificar que se encuentra habilitado para realizar dicha acción.
--	--	------------------------------------------------------------------------------------------------------------------------------------

Tabla 9. Flujo Alternativo Caso de Uso 9

Flujo Alternativo		
Paso	Actor	Sistema
1	El Super usuario no encuentra registrado al usuario administrador en el sistema.	El sistema despliega en pantalla un mensaje informándole al super usuario la inexistencia del registro perteneciente al usuario administrador a través de la búsqueda que realizó previamente.



Notación:

- 1: Abrir formulario
- 4: Modificar Información
- 5: Validar restricciones
- 6: Cargar BDD
- 7: Actualizar BDD
- 8: Consultar Información
- 9: Salir

Figura 11. Diagrama de Secuencia 2 – Modificación Administrador

Especificación de Caso de Uso: Eliminación de administradores

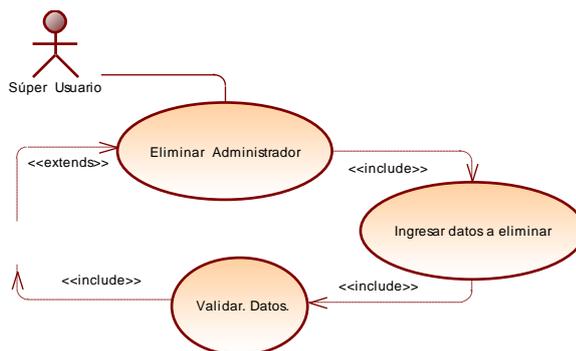


Figura 12. Diagrama Caso de Uso 10 – Eliminar Administrador

Tabla 10. Especificación del caso de uso 10

Especificación del caso de uso			
Nombre del caso de uso:	Eliminación de administradores	Versión No:	1.0
Descripción:	Posibilita al actor para que acceda al sistema a través de sus datos y realice la eliminación de la información del usuario administrador.		

Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Super usuario		

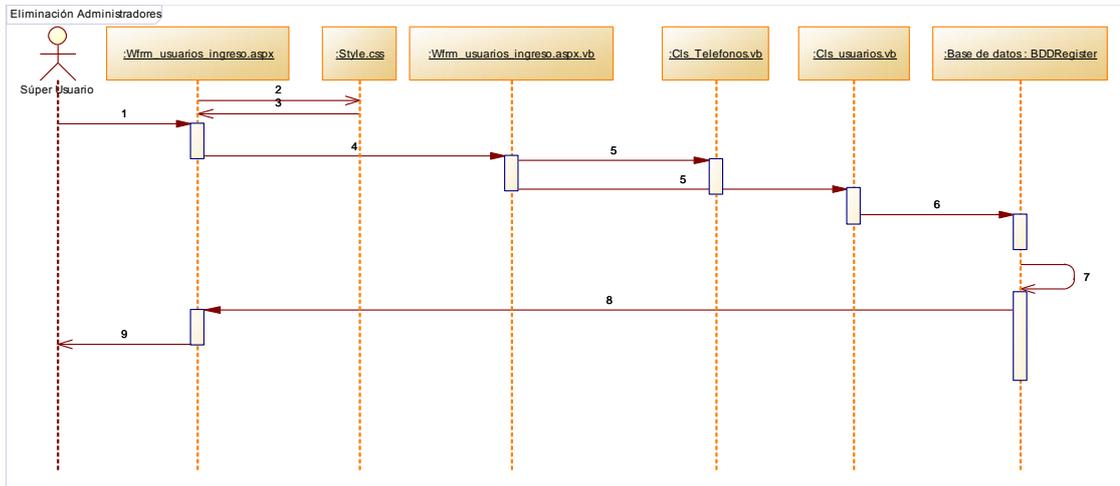
- Wfrm_usuarios_ingreso.aspx
- Wfrm_usuarios_ingreso.aspx.vb
- Cls_usuarios.vb- Cls_Telefonos.vb
- Style.css
- Base de datos : BDDRegister

Tabla 11. Flujo Básico 10

Flujo Básico		
Paso	Actor	Sistema
1	El super usuario realiza la consulta del usuario administrador en el sistema.	El sistema muestra en pantalla la información referente al usuario administrador que se encuentre registrado con esa cédula.
2	El super usuario realiza la eliminación de los datos del usuario administrador.	El sistema habilita la opción de eliminar, para que el usuario administrador quede inhabilitado en el sistema, a través de un botón que al momento de presionarlo despliega un mensaje donde pregunta si realmente se quiere eliminar ese registro.

Tabla 12. Flujo Alternativo Caso de Uso 10

Flujo Alternativo		
Paso	Actor	Sistema
1	El Super usuario no encuentra registrado al usuario administrador en el sistema.	El sistema despliega en pantalla un mensaje informándole al super usuario la inexistencia del registro perteneciente al usuario administrador a través de la búsqueda que realizó previamente.



Notación:

- 1: Abrir formulario
- 4: Eliminar Usuario
- 5: Validar restricciones
- 6: Cargar BDD
- 7: Eliminar de BDD
- 8: Consultar Información
- 9: Salir

Figura 13. Diagrama de Secuencia 3 - Eliminación Administrador

Especificación de Caso de Uso: Ingreso de usuarios

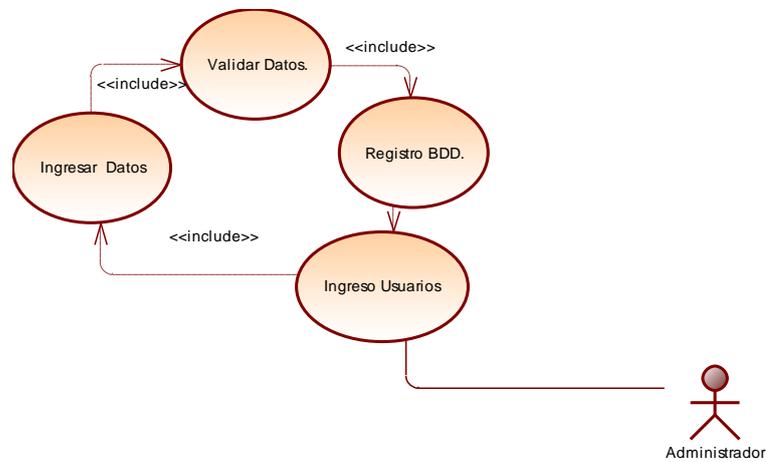


Figura 14. Diagrama Caso de Uso 11 – Ingreso Usuarios

Tabla 13. Especificación del caso de uso 11

Especificación del caso de uso			
Nombre del caso de uso:	Ingreso de usuarios	Versión No:	1.0
Descripción:	Posibilita al actor para que realice el ingreso de sus datos al sistema a través de un formulario de datos presente en la aplicación web.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario administrador		

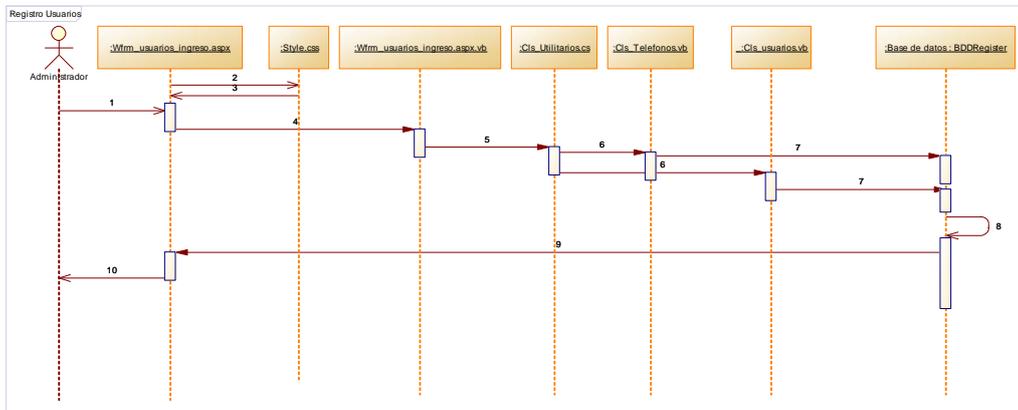
- Wfrm_usuarios_ingreso.aspx
- Wfrm_usuarios_ingreso.aspx.vb
- Cls_usuarios.vb – Cls_Utilitarios.vb -Cls_Telefonos.vb
- Style.css
- Base de datos : BDDRegister

Tabla 14. Flujo Básico del caso de uso 11

Flujo Básico		
Paso	Actor	Sistema
1	El usuario administrador ingresará al sistema a través de sus credenciales.	El sistema despliega en pantalla todas las opciones disponibles para la manipulación del sistema.
2	El usuario administrador realiza el registro de usuarios administradores.	En pantalla se despliega un formulario con diferentes campos para el ingreso de la información de los usuarios.
3	El usuario administrador realiza la selección del rango de usuario	El sistema carga por default los 4 tipos de usuarios para su selección.
4	El usuario administrador llena los campos	El sistema valida campo por campo para controlar las restricciones internas (números, no se permiten caracteres especiales, cédula, correo, solo letras y campos requeridos).
5	El usuario administrador presiona en el botón ingresar.	El sistema envía toda la información de los campos a la base de datos y se presenta un mensaje en pantalla notificando el ingreso, teniendo el usuario administrador una opción en el mensaje donde acepta la acción que se ejecutó en el sistema.
6	El usuario administrador realiza la consulta de los datos ingresados en el sistema.	El sistema muestra en pantalla un filtro para realizar la consulta de datos a través de la cédula del usuario.

Tabla 15. Flujo Alternativo caso de uso 11

Flujo Alternativo		
Paso	Actor	Sistema
1	El usuario administrador ingresa datos de registro de usuarios de manera errónea	El sistema a través sus restricciones inhabilita el botón de ingresar, hasta que el usuario administrador digite los datos correctos.
2	El usuario administrador no ingresa registro alguno.	El sistema a través de sus restricciones inhabilita el botón de ingresar, hasta que el usuario administrador digite los datos correctos.



Notación:

1: Abrir formulario

4: Ingresar campos

5: Función insertar

6: Validar restricciones

7: Cargar BDD

8: Guardar BDD

9: Consultar Información

10: Salir

Figura 15. Diagrama de Secuencia 4 - Registro de Usuarios

Especificación de Caso de Uso: Modificación De Usuario

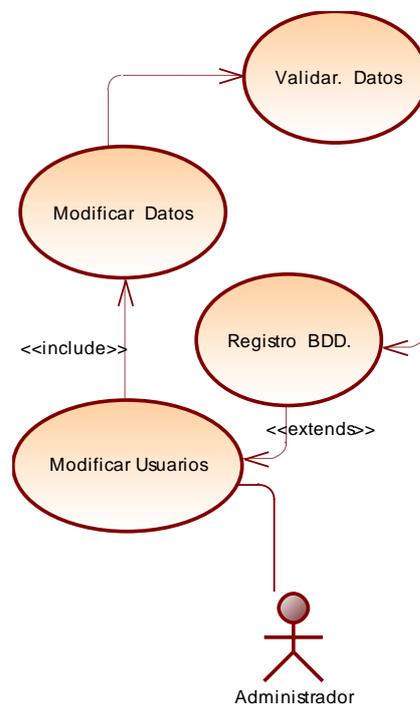


Figura 16. Diagrama Caso de Uso 12 – modificar Usuarios

Tabla 16. Especificación del caso de uso 12

Especificación del caso de uso			
Nombre del caso de uso:	Modificación de Usuarios	Versión No:	1.0
Descripción:	Posibilita al actor para que acceda al sistema a través de sus datos y realice la modificación de la información de los usuarios.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Administrador		

- Wfrm_usuarios_ingreso.aspx
- Wfrm_usuarios_ingreso.aspx.vb
- Cls_usuarios.vb – Cls_Utilitarios.vb -Cls_Telefonos.vb
- Style.css
- Base de datos : BDDRegister

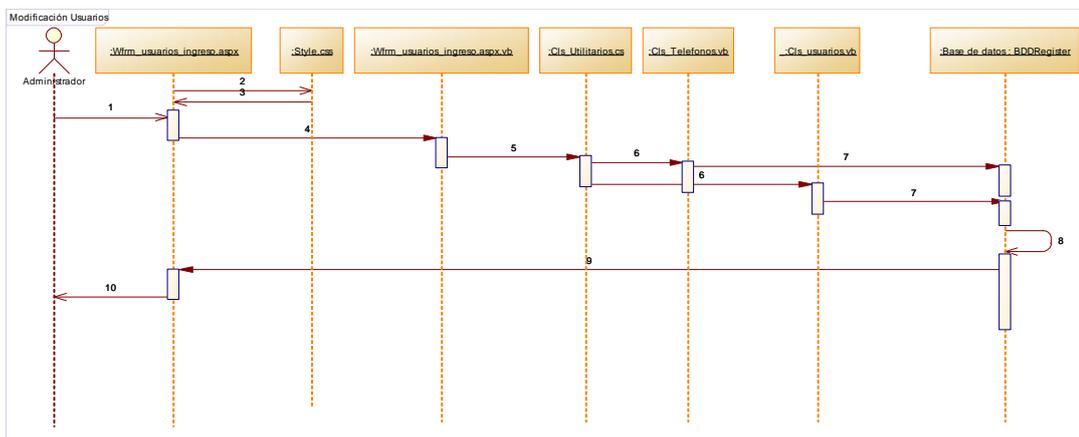
Tabla 17. Flujo Básico caso de uso 12

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Administrador realiza la consulta del usuario administrador en el sistema.	El sistema muestra en pantalla la información referente al usuario administrador que se encuentre registrado con esa cédula.
2	El Usuario Administrador realiza la modificación de los datos del usuario administrador.	El sistema habilita todos los campos referentes a la información del sistema para su respectiva modificación, y

		respectivo almacenamiento de datos en la base a través del botón modificar que se encuentra habilitado para realizar dicha acción.
--	--	------------------------------------------------------------------------------------------------------------------------------------

Tabla 18. Flujo Alternativo caso de uso 12

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Administrador no encuentra registrado al usuario administrador en el sistema.	El sistema despliega en pantalla un mensaje informándole al super usuario la inexistencia del registro perteneciente al usuario a través de la búsqueda que realizó previamente.



Notación:

1: Abrir formulario

4: Modificar campos

5: Función modificar

6: Validar restricciones

7: Actualizar BDD

8: Guardar BDD

9: Consultar Información

10: Salir

Figura 17. Diagrama de Secuencia 5 - Modificación de Usuarios

Especificación de Caso de Uso: Eliminación de Usuarios

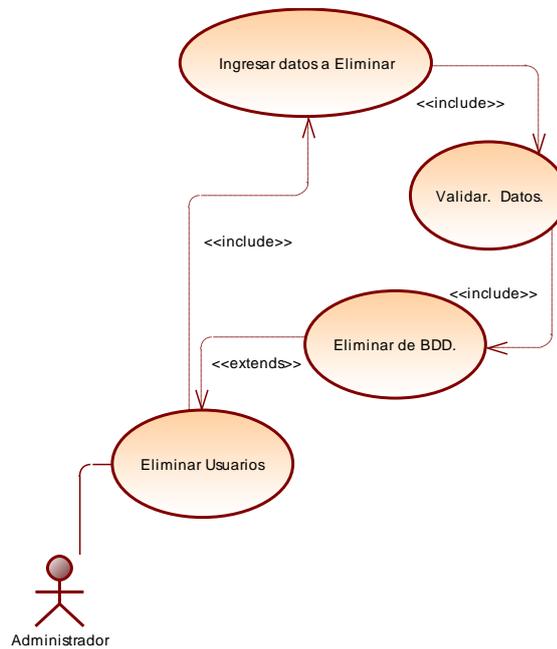


Figura 18. Diagrama Caso de Uso 13 – Eliminar Usuarios

Tabla 19. Especificación del caso de uso 13

Especificación del caso de uso			
Nombre del caso de uso:	Eliminación de Usuarios	Versión No:	1.0
Descripción:	Posibilita al actor para que acceda al sistema a través de sus datos y realice la eliminación de la información de los usuarios.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Administrador		

- Wfrm_usuarios_ingreso.aspx
- Wfrm_usuarios_ingreso.aspx.vb
- Cls_usuarios.vb – Cls_Utilitarios.vb -Cls_Telefonos.vb
- Style.css
- Base de datos : BDDRegister

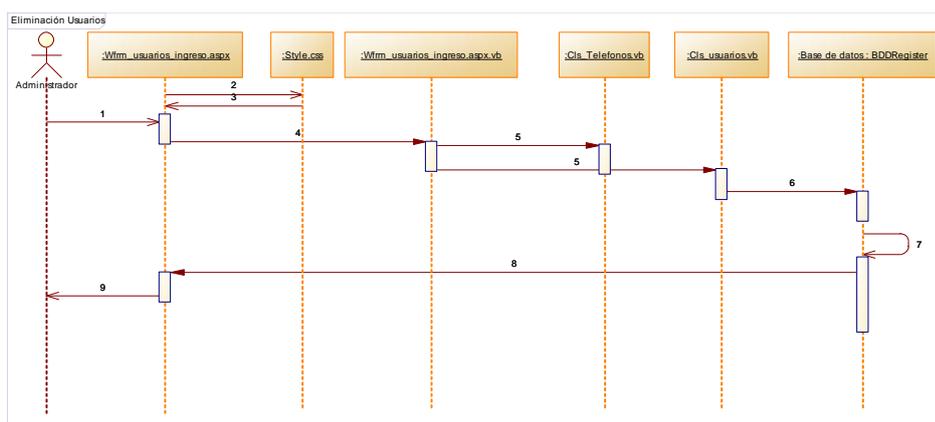
Tabla 20. Flujo Básico caso de uso 13

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Administrador realiza la consulta del usuario administrador en el sistema.	El sistema muestra en pantalla la información referente al usuario administrador que se encuentre registrado con esa cédula.
2	El Usuario Administrador realiza la eliminación de los datos del usuario.	El sistema habilita la opción de eliminar, para que el usuario quede inhabilitado en el sistema, a través de un botón que

		al momento de presionarlo despliega un mensaje donde se pregunta si realmente se quiere eliminar ese registro.
--	--	----------------------------------------------------------------------------------------------------------------

Tabla 21. Flujo Alternativo caso de uso 13

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Administrador no encuentra registrado al usuario en el sistema.	El sistema despliega en pantalla un mensaje informándole al usuario administrador la inexistencia del registro perteneciente al usuario a través de la búsqueda que realizó previamente.



Notación:

- 1: Abrir formulario
- 4: Eliminar Usuario
- 5: Función eliminar
- 6: Validar restricciones
- 7: Eliminar de BDD
- 8: Guardar BDD
- 9: Consultar Información
- 10: Salir

Figura 19. Diagrama de Secuencia 6 - Eliminación de Usuarios

Especificación de Caso de Uso: Enviar requerimiento

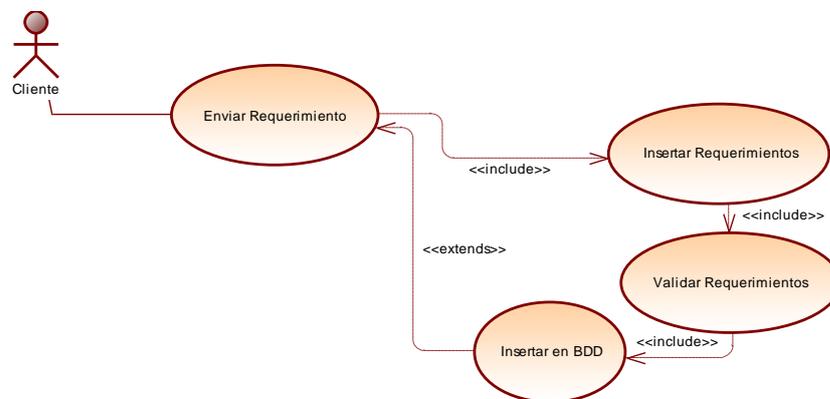


Figura 20. Diagrama Caso de Uso 14 – Enviar Requerimiento

Tabla 22. Especificación del caso de uso 14

Especificación del caso de uso			
Nombre del caso de uso:	Enviar Requerimiento	Versión No:	1.0
Descripción:	Posibilita al actor para que realice el envío de los datos		

Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Cliente		

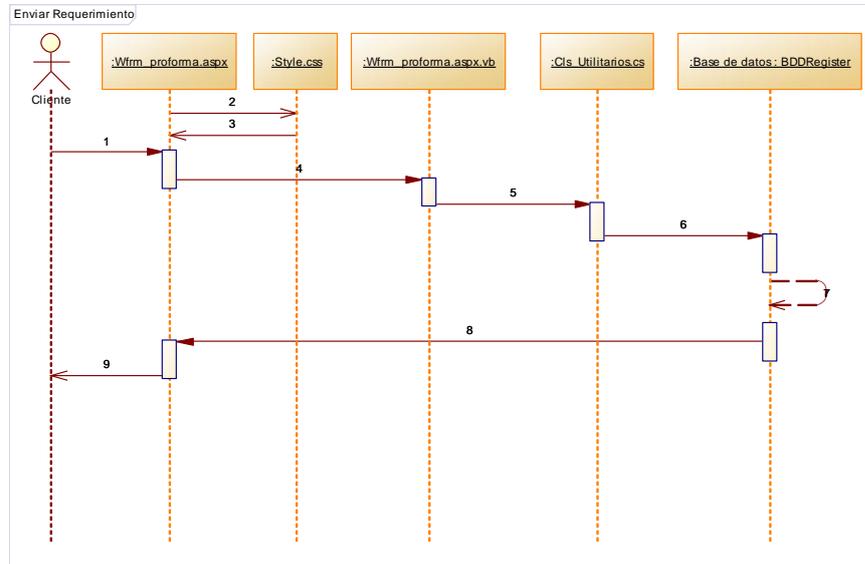
- Wfrm_proforma.aspx
- Wfrm_proforma.aspx.vb
- Cls_Proforma.vb– Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 23. Flujo Básico caso de uso 14

Flujo Básico		
Paso	Actor	Sistema
1	El cliente llena todos datos y envía la información.	El sistema muestra en pantalla las opciones disponibles para ingresar los datos con un botón de ingreso que al presionarlo los datos se almacenan en la base del sistema.

Tabla 24. Flujo Alternativo caso de uso 14

Flujo Alternativo		
Paso	Actor	Sistema
1	El cliente no llena todos datos y desea realizar el envío.	El sistema le muestra en pantalla un mensaje notificándole al cliente que hay campos sin llenar y que son requeridos para poder ser enviados a la base.



Notación:

- 1: Abrir formulario
- 4: Ingresar campos
- 5: Función insertar
- 6: Cargar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 21. Diagrama de Secuencia 7 - Enviar Requerimiento

Especificación de Caso de Uso: Enviar Proforma

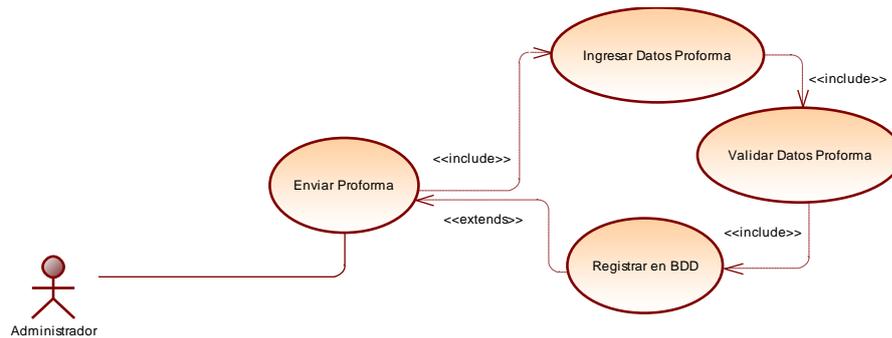


Figura 22. Diagrama Caso de Uso 15 – Enviar Proforma

Tabla 25. Especificación del caso de uso 15

Especificación del caso de uso			
Nombre del caso de uso:	Enviar Proforma	Versión No:	1.0
Descripción:	Posibilita al actor para que realice el envío de la proforma informativa al cliente.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Administrador		

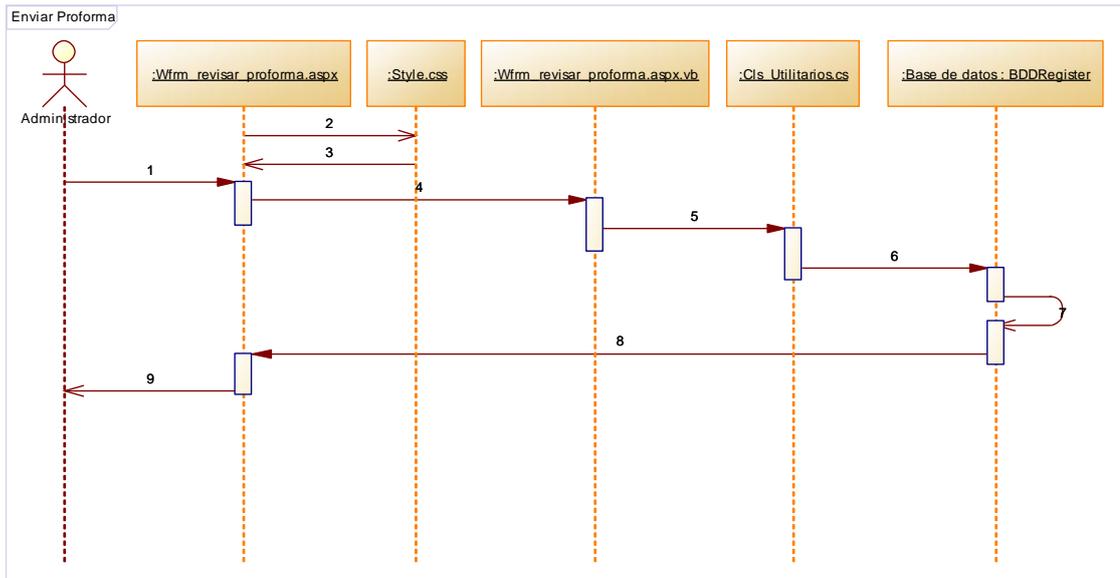
- Wfrm_revisar_proforma.aspx
- Wfrm_revisar_proforma.aspx.vb
- Cls_Proforma.vb– Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 26. Flujo Básico caso de uso 15

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Administrador realiza la consulta de las solicitudes de proformas.	El sistema muestra en pantalla las opciones disponibles para ingresar los datos.
2	El Usuario Administrador llena la proforma de trabajo.	El sistema muestra en pantalla una opción para la creación de proformas las mismas que partiendo de los datos del cliente y su petición crea la proforma de trabajo con los datos pertinentes.
3	El Usuario Administrador envía las proformas a los clientes.	El sistema verifica que la proforma esté completa y a través de una opción presente en la aplicación realiza el envío al cliente para su confirmación.

Tabla 27. Flujo Alternativo caso de uso 15

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Administrador no llena todos los campos de la proforma de trabajo para el envío.	El sistema le muestra en pantalla un mensaje notificándole el error cometido para que complete la información



Notación:

- 1: Abrir formulario
- 4: Revisar Información
- 5: Función enviar
- 6: Cargar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 23. Diagrama de Secuencia 8 - Enviar Proforma

Especificación de Caso de Uso: Elaboración de la orden de trabajo

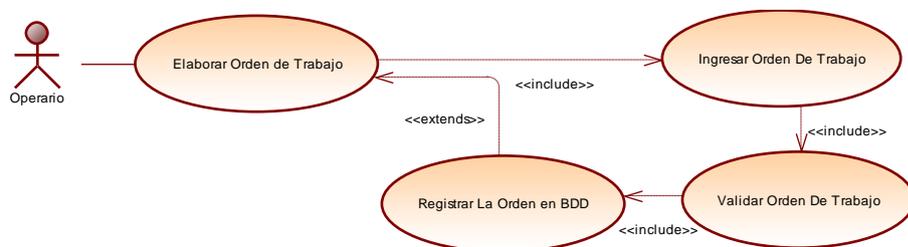


Figura 24. Diagrama Caso de Uso 16 – Elaboración Orden de Trabajo

Tabla 28. Especificación del caso de uso 16

Especificación del caso de uso			
Nombre del caso de uso:	Elaboración de la orden de trabajo	Versión No:	1.0
Descripción:	Posibilita al actor la emisión de una orden de trabajo.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Operario		

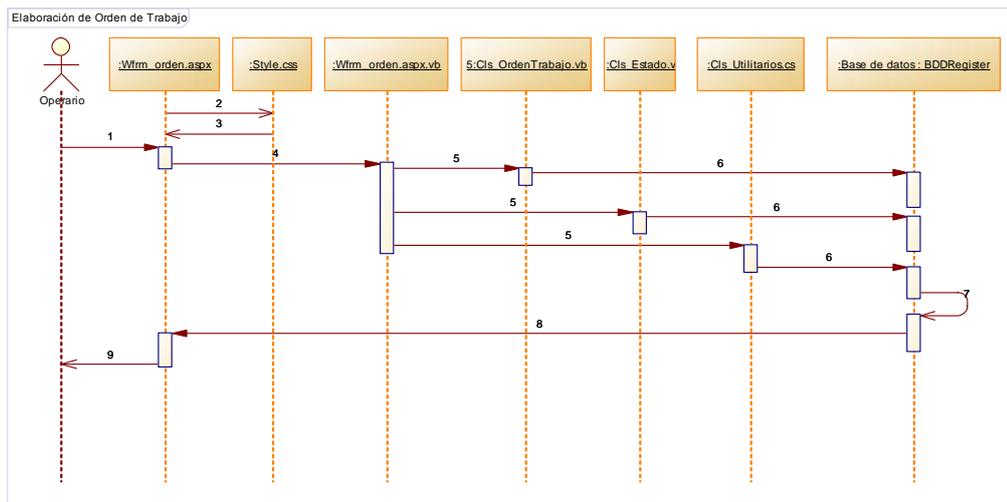
- Wfrm_orden.aspx
- Wfrm_orden.aspx.vb
- Cls_OrdenTrabajo.vb - Cls_Estado.vb - Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 29. Flujo Básico caso de uso 16

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Operario elabora la orden de trabajo.	El sistema muestra en pantalla un formulario que contiene la información a base de la proforma entregada previamente al cliente y con las opciones necesarias para que el técnico elabore fechas de entrega y costos por los servicios que presta la empresa.
2		El sistema hace un envío de la orden de trabajo vía correo electrónico tanto al cliente como al administrador del sistema con fines informativos.

Tabla 30. Flujo Alternativo caso de uso 16

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Operario realizó una mala elaboración de la orden de trabajo.	El sistema le muestra en pantalla un mensaje notificándole que el error en la elaboración de la orden de trabajo para su respectiva corrección.



Notación:

- 1: Abrir formulario
- 4: Ingresar campos
- 5: Validar restricciones
- 6: Cargar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 25. Diagrama de Secuencia 9 – Elaboración de Orden de Trabajo

Especificación de Caso de Uso: Asignación de orden de trabajo

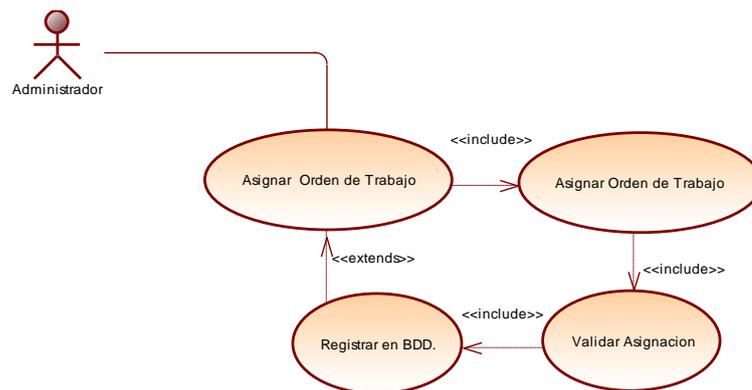


Figura 26. Diagrama Caso de Uso 17 – Asignación Orden de Trabajo

Tabla 31. Especificación del caso de uso 17

Especificación del caso de uso			
Nombre del caso de uso:	Asignación de orden trabajo	Versión No:	1.0
Descripción:	Posibilita al actor la asignación de una orden de trabajo a uno o varios técnicos.		

Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Administrador		

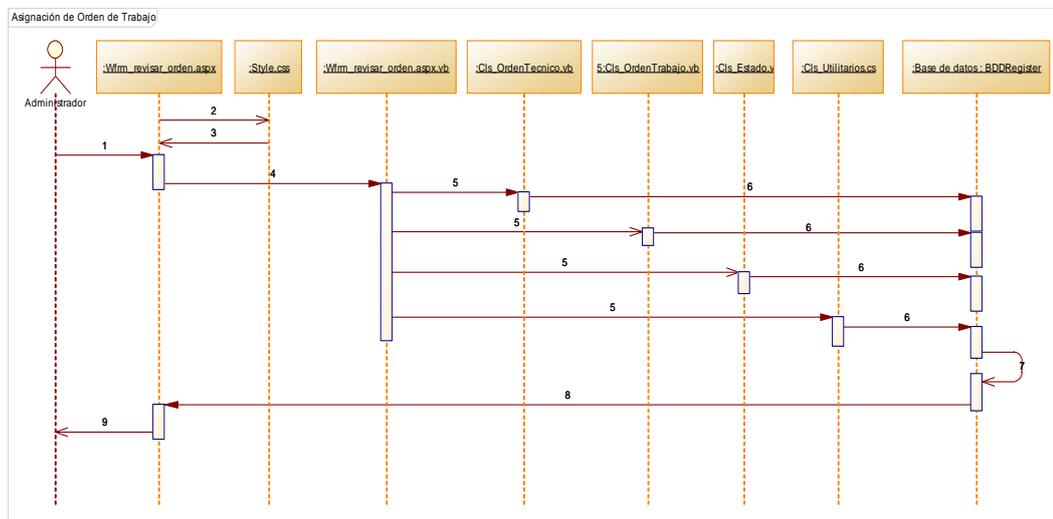
- Wfrm_revisar_orden.aspx
- Wfrm_revisar_orden.aspx.vb
- Cls_OrdenTrabajo.vb - Cls_Estado.vb - Cls_OrdenTecnico.vb - Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 32. Flujo Básico caso de uso 17

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Administrador ingresa a la página y selecciona un cliente.	El sistema despliega en pantalla todas las órdenes de trabajo del cliente seleccionado.
2	El Usuario Administrador selecciona una orden de trabajo específica.	El sistema despliega en pantalla la información referente a la orden de trabajo seleccionada.
3	El Usuario Administrador selecciona a los técnicos para la orden de trabajo.	El sistema anexa a los técnicos seleccionados para la ejecución orden de trabajo seleccionada.

Tabla 33. Flujo Alternativo caso de uso 17

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Administrador guarda la orden de trabajo sin anexar técnico.	El sistema le muestra en pantalla un mensaje notificándole el error cometido para que el administrador seleccione a un técnico y pueda almacenar esa orden de trabajo.
2	El Usuario Administrador seleccione al mismo técnico repetidamente	El sistema le muestra en pantalla un mensaje notificándole al administrador, que el técnico que está seleccionando ya está asignado a una orden en particular por lo que no puede ser seleccionado nuevamente.



Notación:

- 1: Abrir formulario
- 4: Revisar campos
- 5: Validar Restricciones
- 6: Cargar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 27. Diagrama de Secuencia 10 - Asignación de Orden de Trabajo

Especificación de Caso de Uso: Registro De Abono A La Orden De Trabajo

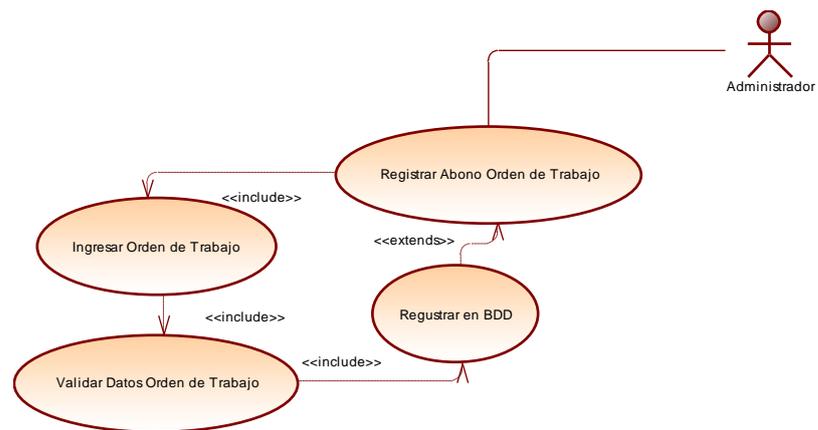


Figura 28. Diagrama Caso de Uso 18 – Registro Abono Orden de Trabajo

Tabla 34. Especificación del caso de uso 19

Especificación del caso de uso			
Nombre del caso de uso:	Registro de abono a la orden de trabajo.	Versión No:	1.0
Descripción:	Posibilita al actor para que realice el registro de un abono realizado por el cliente a una orden de trabajo.		

Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Administrador		

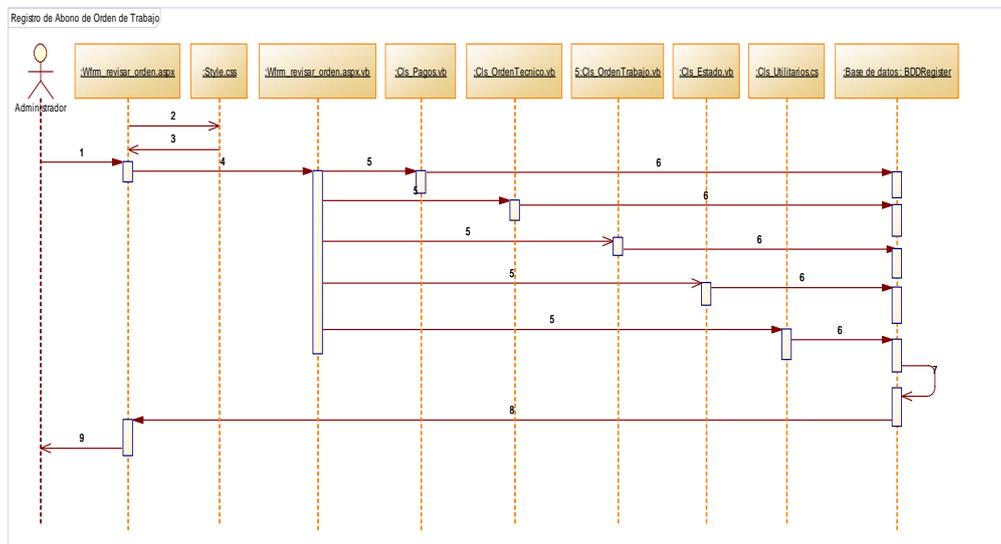
- Wfrm_revisar_orden.aspx
- Wfrm_revisar_orden.aspx.vb
- Cls_OrdenTrabajo.vb - Cls_Pagos.vb - Cls_OrdenTecnico.vb - Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 35. Flujo Básico caso de uso 19

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Administrador ingresa a la página y selecciona un cliente.	El sistema despliega en pantalla todas las órdenes de trabajo del cliente seleccionado.
2	El Usuario Administrador selecciona una orden de trabajo específica.	El sistema despliega en pantalla la información referente a la orden de trabajo seleccionada.
3	El Usuario Administrador registra un abono a una orden de trabajo.	El sistema almacena el abono ingresado por el administrador a través de una opción en pantalla, llamada guardar abono que al seleccionarlo ejecuta dicha acción.

Tabla 36. Flujo Alternativo caso de uso 19

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Administrador no ingresa abono y desea guardar en el sistema.	El sistema le muestra en pantalla un mensaje notificándole el error cometido para que ingrese un valor y pueda realizar el respectivo registro del mismo.



Notación:

- 1: Abrir formulario
- 4: Ingresar Abono
- 5: Validar Restricciones
- 6: Cargar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 29. Diagrama de Secuencia 11 – Registro de Abono de Orden de Trabajo

Especificación de Caso de Uso: Finalizar Orden De Trabajo

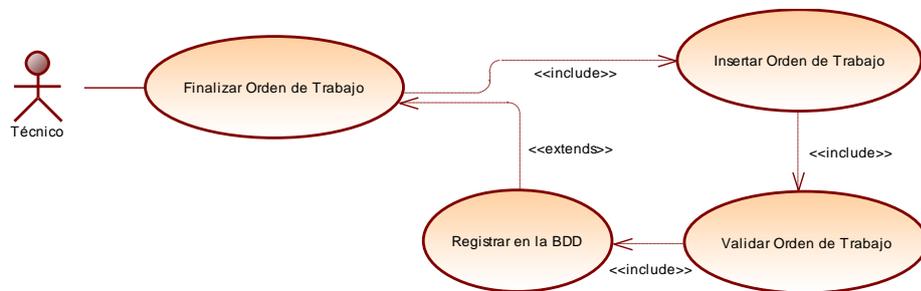


Figura 30. Diagrama Caso de Uso 19 – Finalización Orden de Trabajo

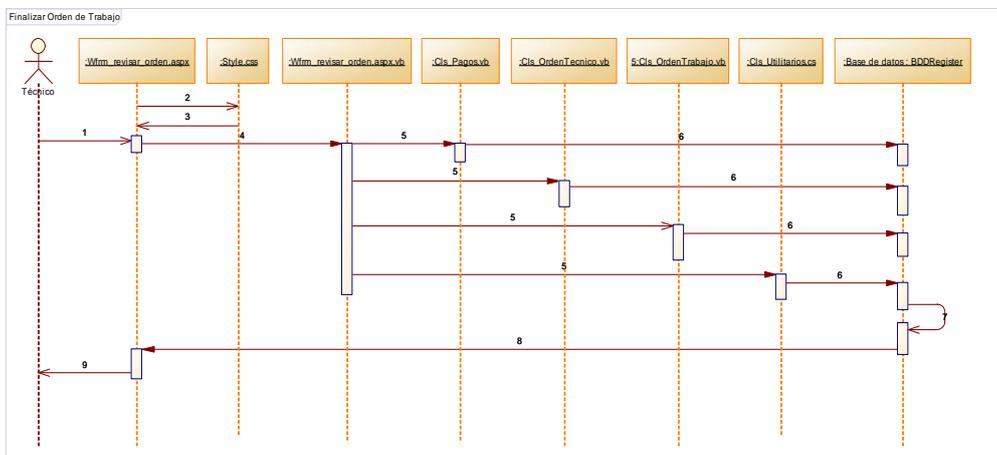
Tabla 37. Especificación del caso de uso 20

Especificación del caso de uso			
Nombre del caso de uso:	Finalizar orden de trabajo	Versión No:	1.0
Descripción:	Posibilita al actor para que realice la finalización de la orden una vez realizado el trabajo.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Técnico		

- Wfrm_revisar_orden.aspx
- Wfrm_revisar_orden.aspx.vb
- Cls_OrdenTrabajo.vb - Cls_Pagos.vb - Cls_OrdenTecnico.vb - Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 38. Flujo Básico 20

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Técnico ingresa a la página de órdenes de trabajo.	El sistema muestra en pantalla un listado de todas las órdenes de trabajo que se encuentran registradas en el sistema.
2	El Usuario Técnico selecciona una orden de trabajo para su finalización.	El sistema despliega en pantalla la orden de trabajo seleccionada con una opción, que al presionarla muestra un mensaje de notificación preguntando si desea finalizar dicha orden.
3	El Usuario Técnico ingresa una observación de la orden de trabajo.	El sistema muestra en pantalla un sector de la página, donde el técnico puede ingresar una observación de la orden de trabajo para futuras consultas.



Notación:

- 1: Abrir formulario
- 4: Finalizar orden
- 5: Validar Restricciones
- 6: Actualizar BDD
- 7: Guardar BDD
- 8: Consultar Información
- 9: Salir

Figura 31. Diagrama de Secuencia 12 – Finalizar Orden de Trabajo

Especificación de Caso de Uso: Verificación Orden De Trabajo Por El Cliente

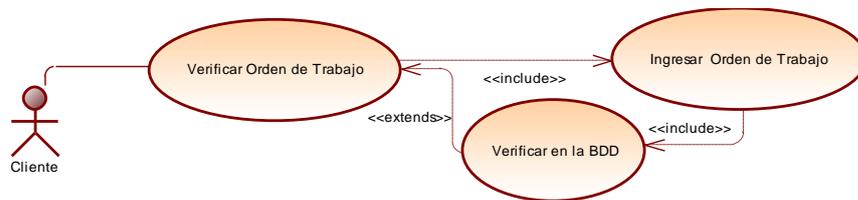


Figura 32. Diagrama Caso de Uso 20 – Verificación Orden de Trabajo

Tabla 39. Especificación del caso de uso 21

Especificación del caso de uso			
Nombre del caso de uso:	Verificación de la orden de trabajo por el cliente	Versión No:	1.0
Descripción:	Posibilita al actor para que realice la consulta del estado en el que se encuentra su orden de trabajo.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Cliente		

- Wfrm_revisar_estados.aspx
- Wfrm_revisar_estados.aspx.vb
- Cls_OrdenTrabajo.vb - Cls_Pagos.vb - Cls_OrdenTecnico.vb - Cls_Utilitarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 40. Flujo Básico caso de uso 21

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Cliente ingresa a su cuenta para verificar una orden de trabajo.	El sistema muestra en pantalla un reporte con la información correspondiente a la orden de trabajo que el cliente registró en la empresa, teniendo de este modo la información sobre si su pedido está siendo revisado o ya está listo para la entrega.

Tabla 41. Flujo Alternativo 21

Flujo Alternativo		
Paso	Actor	Sistema
1	El Usuario Cliente ingresa al sistema y digita mal el código de la orden de trabajo que registró.	El sistema le muestra en pantalla un mensaje notificándole el error cometido para que ingrese correctamente el código de la orden de trabajo y poder ofrecer la información correspondiente a la misma.

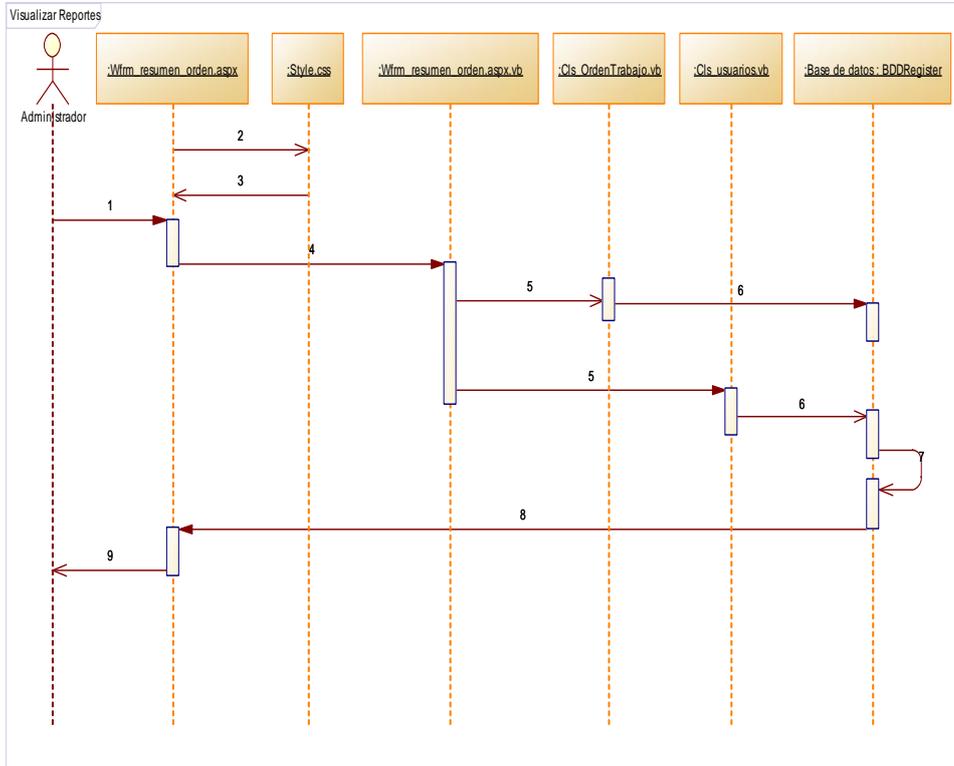
Tabla 322. Especificación del caso de uso 22

Especificación del caso de uso			
Nombre del caso de uso:	Visualizar Reportes.	Versión No:	1.0
Descripción:	Posibilita al actor visualizar un reporte completo de una orden de trabajo específica.		
Autor:	Bladimir Vallejo Altuna	Fecha:	13/12/2013
Usuario/Actor:	Usuario Administrador		

- Wfrm_resumen_orden.aspx
- Wfrm_resumen_orden.aspx.vb
- Cls_OrdenTrabajo.vb - Cls_Usuarios.vb
- Style.css
- Base de datos : BDDRegister

Tabla 43. Flujo Básico caso de uso 22

Flujo Básico		
Paso	Actor	Sistema
1	El Usuario Administrador ingresa a la página y selecciona un cliente.	El sistema despliega en pantalla todas las órdenes de trabajo del cliente seleccionado.
2	El Usuario Administrador selecciona una orden de trabajo específica.	El sistema despliega en pantalla la información referente a la orden de trabajo seleccionada a través de un reporte.



Notación:

- 1: Abrir formulario
- 4: Seleccionar Reporte
- 5: Consultar
- 6: Cargar BDD
- 7: Seleccionar BDD
- 8: Mostrar Información
- 9: Salir

Figura 35. Diagrama de Secuencia 14 – Visualizar Reportes

DIAGRAMA DE BASE DE DATOS

MODELO CONCEPTUAL

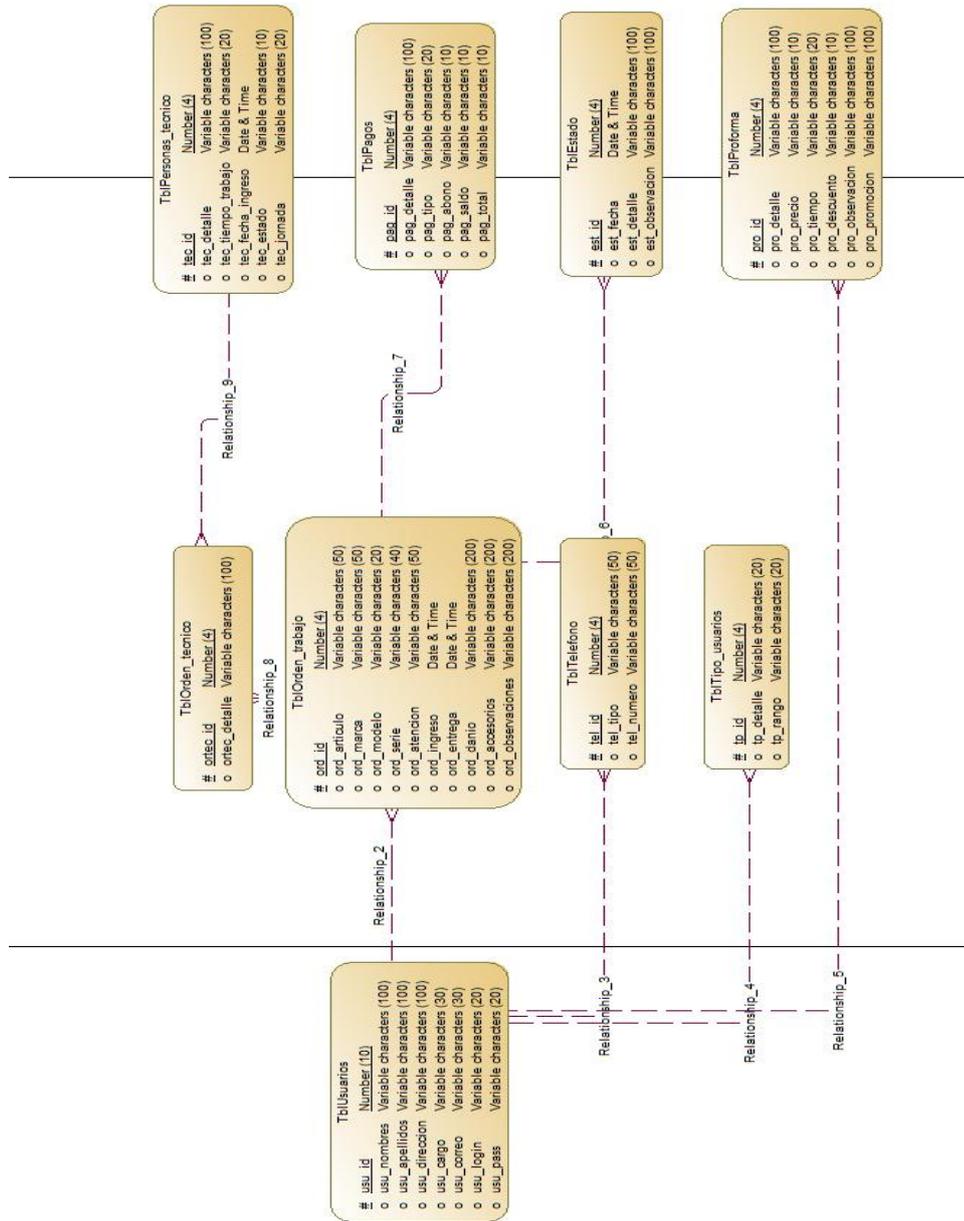


Figura 36. Modelo conceptual de base de datos

DIAGRAMA DE CLASES

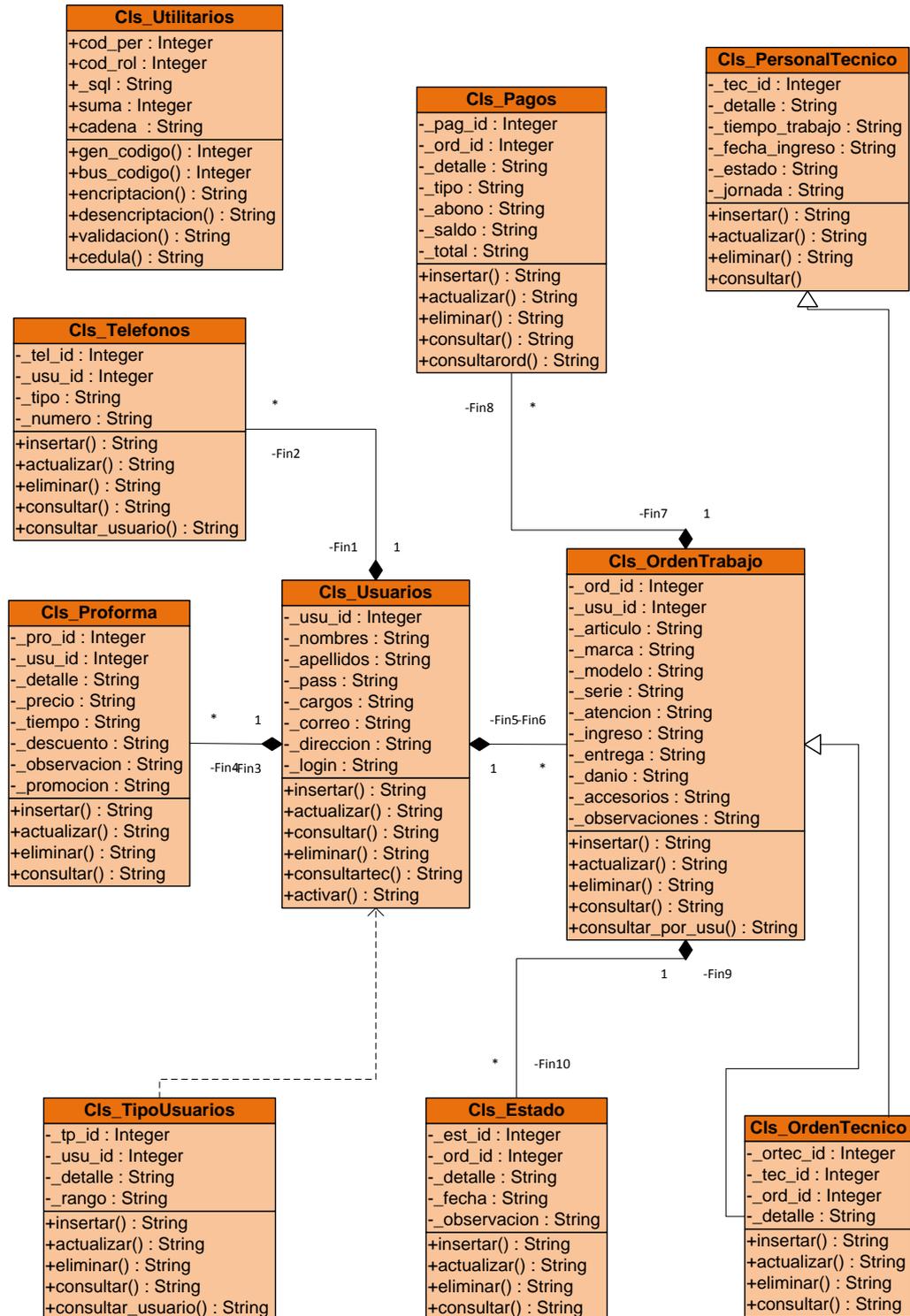
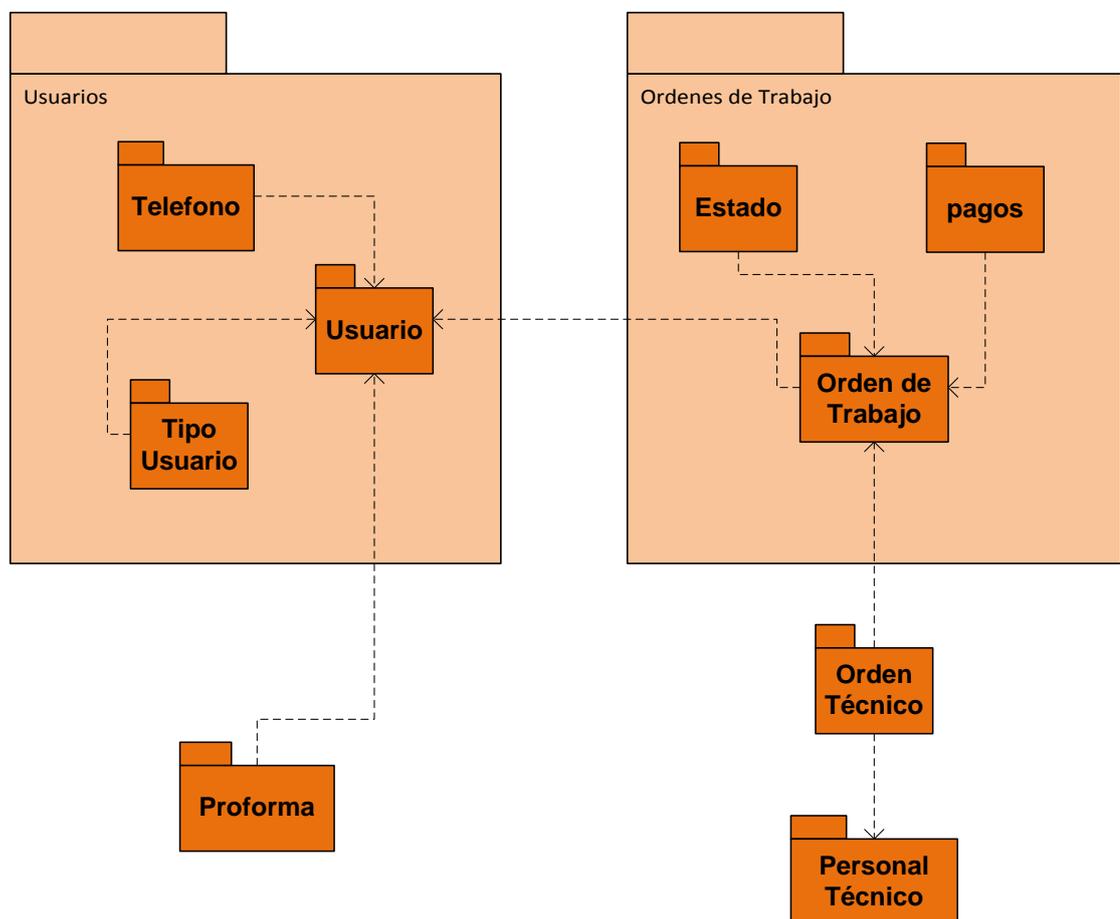


Figura 37. Diagrama de clases

4.3. FASE DE CONSTRUCCIÓN

DIAGRAMA DE PAQUETES

En el siguiente diagrama se muestra cómo el sistema está dividido en agrupaciones lógicas mostrando las dependencias entre ellas.



4.3.2. MAQUETACIÓN

En esta fase se realiza la estructuración de cada uno de los elementos que conforman la aplicación, como son labels, cuadros de texto, botones y demás objetos que contienen cada uno de los formularios y que estarán programados con la codificación en ASP en base VISUAL BASIC que realizarán las diferentes funcionalidades para las cuales estará estructurada la aplicación.

1. Inicio de sesión

Login	
Pass	
Limpiar	Insertar

```
// crea la conexion pasandole el servidor , usuario y clave
```

```
$connect= mysql_connect($conexion['server'],$conexion['user'],$conexion['pass']);
```

```
$this->con = mysqli_connect($conexion['server'],$conexion['user'], "", $conexion['base']);
```

2. Menú Usuario

Titulo : Menú (Correspondiente al usuario)
Opciones Padre del menú
Opciones hijas menú

3. Administrar usuario

Titulo de la pagina		
Cedula	(campo texto)	Buscar (Boton)
Nombre	(campo texto)	
Apellido	(campo texto)	
Direccion	(campo texto)	
Cargo	(campo texto)	
Correo	(campo texto)	
Telefono	(campo texto)	
Rango	(campo texto)	
Login	(campo texto)	
Contraseña	(campo texto)	
		Limpiar (Boton)
		Insertar (Boton)
	Regresar (Boton)	

```
$sql="INSERT INTO tblusuarios VALUES (".${this->_id}",".${this->_nombre}",".${this->_apellido}",".${this->_direccion}",".${this->_cargo}",".${this->_correo}",".${this->_login}",".${this->_pass}");  
echo $sql;
```

```
$sql="select * from tblusuarios where usu_id=".${this->_id};
```

```
$sql="UPDATE tblusuarios set usu_nombres=".${this->_nombre}","usu_apellidos=".${this->_apellido}","usu_direccion=".${this->_direccion}","usu_cargo=".${this->_cargo}","usu_correo=".${this->_correo}","usu_login=".${this->_login}","usu_pass=".${this->_pass}" where usu_id=".${this->_id};  
echo $sql;
```

```
$sql="delete from tblusuarios where usu_id =".".${this->_id}";  
echo $sql;
```

4. Creación orden de trabajo

Titulo orden de trabajo			Logo empresa
			#orden de trabajo
id cliente	(campo texto)	Consultar(Boton)	Nuevo Usuario(Boton)
Nombres	(campo texto)	Apellidos	(campo texto)
Direccion			
Correo	(campo texto)	Telefono	(campo texto)
Articulo	(campo texto)	Marca	(campo texto)
Modelo	(campo texto)	Serie	(campo texto)
Tipo de Atencion	(campo texto)		
Fecha de ingreso	(campo texto)	Fecha de salida	(campo texto)
Daño	(campo texto)		
Accesorios	(campo texto)		
Abono	(campo texto)	Total	(campo texto)
Saldo	(campo texto)		
Detalle Pago	(campo texto)		
Tipo de pago	(campo texto)		
Observaciones	(campo texto)		
	Regresar (Boton)		

```

$sql="INSERT INTO tblproforma VALUES (".$this->_id.", ".$this->_idusu.", ".$this->_detalle.", ".$this->_precio.", ".$this->_tiempo.", ".$this->_descuento.", ".$this->_observacion.", ".$this->_promocion.");
echo $sql;

```

5. Orden de trabajo

Título del proceso			
cedula	(campo de texto)	Buscar (Boton)	Limpiar (Boton)
	Grilla datos filtro		
	Grilla ordenes de trabajo		

```
$sql="select * from tblproforma where pro_id=".$this->_id;
```

```
$sql="UPDATE          tblproforma          set          usu_id=".$this->_idusu.",pro_detalle=".$this->_detalle.",pro_precio=".$this->_precio.",pro_tiempo=".$this->_tiempo.",pro_descuento=".$this->_descuento.",pro_observacion=".$this->_observacion.",pro_promocion=".$this->_promocion." where pro_id=".$this->_id;
echo $sql;
```

```
$sql= "delete from tblproforma where pro_id ="."".$this->_id.""";
echo $sql;
```

6. Proforma

Precio	
Tiempo	
descuento	
Promocion	
observaciones	
	enviar (boton) limpiar (Boton)
	regresar (boton)

```
$sql="INSERT INTO tblproforma VALUES (".$this->_id.",".$this->_idusu.",".$this->_detalle.",".$this->_precio.",".$this->_tiempo.",".$this->_descuento.",".$this->_observacion.",".$this->_promocion."");
echo $sql;
```

```
$sql="select * from tblproforma where pro_id=".$this->_id;
```

4.4. FASE DE TRANSICIÓN

Al culminar el desarrollo del sistema, se realizaron las etapas pertinentes dentro de los parámetros que la empresa dictaminó. Se encontraron ciertos errores en la aplicación y se corrigieron al instante para así obtener un sistema móvil beta que contenga el mínimo de errores.

4.4.1. PRUEBAS Y DESPLIEGUE

La aplicación en la versión Beta que se está desarrollando no estará disponible para descarga desde Google Play debido al costo por publicación por cada versión, por lo que para la descarga de la aplicación se tendrá una dirección web donde estará publicada la aplicación.

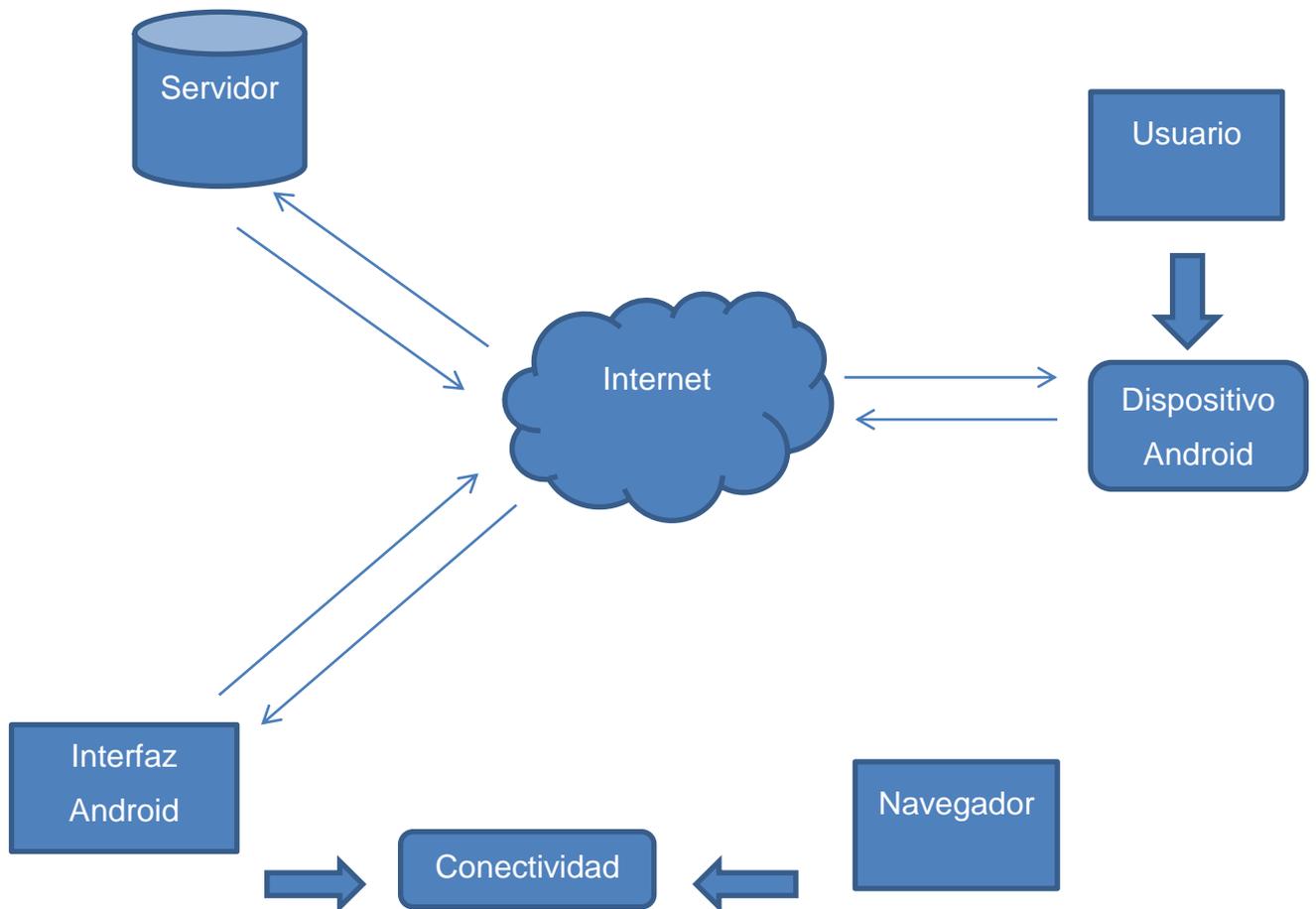
El link en el que estará temporalmente disponible la aplicación para la descarga del archivo APK es el siguiente:

<http://webregister.azurewebsites.net/index.aspx>

Para las pruebas de Humo se puso en funcionamiento en paralelo al proceso manual por el lapso de tiempo de una semana logrando detectar errores los cuales ya fueron corregidos, además se realizaron cambios en la interfaz por sugerencias del gerente, técnicos y clientes logrando así la aceptación del sistema. Ver Anexo 1.

4.4.2. DIAGRAMA DE DESPLIEGUE

En el siguiente diagrama se indica la topología sobre la cual está funcionando el sistema.



4.4.3. MANUAL DE USUARIO

Para una mejor comprensión del funcionamiento del sistema se elaboró un manual en el cual está detallado como debe operarse cada módulo del sistema para cada nivel de usuario. Ver Anexo 2.

CONCLUSIONES Y RECOMENDACIONES

5. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- Con el desarrollo del presente trabajo se inició el proceso de actualización de la empresa en la implementación de nuevas tendencias tecnológicas e informáticas como el uso de dispositivos móviles y documentos electrónicos mostrando así una imagen positiva de la empresa.
- La adaptación de los usuarios al uso de la aplicación fue relativamente sencilla ya que la mayoría de usuarios tienen experiencia en el uso de este tipo de aplicaciones debido al uso cotidiano de Smartphones o dispositivos móviles.
- Las nuevas tecnologías vinculadas a la portabilidad de los datos han sido un ente que ha dado el empuje a la reestructuración de la forma como se ejecutan los procesos, accediendo de una forma más eficiente a la información y realizando tareas que en el caso de la aplicación desarrollada han ayudado a mejorar la atención a los clientes y optimizar procesos de la empresa.
- La utilización de la metodología de desarrollo RUP la cual está estrechamente ligada al modelamiento UML ayudó a tener de manera muy clara y óptima como estará conformado el sistema notando elementos que al parecer eran externos al sistema o ayudarían de mejor manera a la optimización del proceso de elaboración de órdenes de trabajo.
- La utilización de dispositivos móviles para la generación de órdenes de trabajo por parte de la empresa AUVICOM ha optimizado el proceso que se realizaba anteriormente, ya que los dispositivos son una gran herramienta que han beneficiado de manera positiva en la ejecución de procesos, su portabilidad ofrecen disponibilidad de ciertas funciones que sin ellos llevaría más tiempo realizar por lo que genera un beneficio en el manejo de tiempos.

- La forma como se ejecutan los procesos actualmente en la empresa AUVICOM con respecto a la interacción con el cliente se han visto beneficiadas con los cambios que la automatización de procesos trajo a la empresa, teniendo que en la realización de una orden de trabajo se ven avances notorios, por la utilización de dispositivos móviles que a través de las diferentes opciones que ofrecen ayudan a la ejecución y agiliza el proceso de datos en cada uno de los procesos que se ven inmersos en la orden de trabajo.
- Los dispositivos móviles que tienen como software base a Android en cualquiera de sus versiones, ofrece mucha funcionalidad y accesibilidad ya que a través de las aplicaciones móviles permiten que un mayor número de usuarios puedan manipular de forma completamente interactiva y dinámica y sobretodo amigable para su uso, a una diversidad de funcionalidades y gracias a su potencialidad los procesos de la orden de trabajo se realizan con total movilidad y facilidad.
- Las nuevas tecnologías vinculadas a la portabilidad de los datos han sido un ente que ha dado el empuje a la reestructuración de la forma como se ejecutan los procesos, accediendo de una forma más eficiente a la información y realizando procesos que en el caso del presente sistema han ayudado a mejorar la atención a los clientes de la empresa AUVICOM brindándoles mejores prestaciones en los procesos y manejo de una orden de trabajo y ofreciéndoles mayor información sobre el proceso en el cual se encuentra su pedido.

5.2. RECOMENDACIONES

- Para un rápido y eficiente, diseño e implementación de un sistema, es de gran importancia tener clara los requerimientos globales para dar una solución que se ajuste a las necesidades exactas de los problemas y puntos críticos del cliente.
- Tener en claro las herramientas precisas para un buen desarrollo, que se enfoque a una implementación fija, que contenga todos los módulos convenientes para una rápida y eficiente programación de los todos los aspectos que involucran el sistema.
- Al momento del enfoque del sistema a un dispositivo android, es de gran ayuda tener los textos apropiados y toda la información pertinente a la mano, ya que todas las funciones y complicaciones que se encuentren, se las resolverán en gran medida por la rapidez en la que se encuentre las soluciones indicadas.
- Al diseñar un sistema que será visualizado en sistemas android, se debe tomar en cuenta las dimensiones de la pantalla, así como el dimensionamiento de los elementos que comprenden nuestra aplicación, como son: botones, cuadros de textos, tamaño y fuente de letra, mensajes, títulos e imágenes en general.
- Se debe mantener una comunicación constante con el cliente e ir informando de forma continua y presentando los avances apropiados de la solución que se esta desarrollado, esto hará que todos los errores y requerimientos sean pulidos de manera rápida.
- En el momento de la finalización de la solución y la puesta en marcha en tiempo real, se tiene que dar un periodo coherente de pruebas, ya que los sistemas que están direccionados a dispositivos móviles, tienen la tendencia de generar errores con datos a gran escala o con la conectividad de sus señales inalámbricas, así como la pronto respuesta de sus bases de datos.

- Se recomienda para un futuro la complementación del sistema con el módulo de facturación electrónica y así poder estar sobre la tendencia actual de la facturación electrónica y cero papeles.

BIBLIOGRAFÍA

BIBLIOGRAFÍA

Maestros del Web. (abril de 2001). Recuperado el 2013, de Que es ASP:
<http://www.maestrosdelweb.com/editorial/aspintro/>

Informática Hoy. (octubre de 2011). Obtenido de Componentes de un Sistema Operativo Movil: <http://www.informatica-hoy.com.ar/soluciones-moviles/Componentes-de-un-Sistema-Operativo-Movil.php>

Metodología de Software. (Noviembre de 2012). Recuperado el 2014, de Metodología RUP:
http://metodologiadesoftware.blogspot.com/2012/11/fases-del-modelo-rup_27.html

El Software. (marzo de 2013). Recuperado el 2014, de Visión General del Proceso de Desarrollo de Software:
<http://grupo3trayecto2trimestre1seccion2.blogspot.com/2013/03/vision-general-del-proceso-de.html>

HTCMANIA. (26 de enero de 2013). Obtenido de http://www.htcmania.com/mediawiki/index.php/Programaci%C3%B3n_de_aplicaciones_para_m%C3%B3viles_Android_-_Unidad_1

Ableson, F. (2010). *A developer's Guide*. Saddle River: Addison-Wesley.

Angoar, M. (2011). *Introducción Android*. Madrid: Complutense.

Aponte, S., & Davila, C. (2011). *Sistemas Operativos Móviles: Funcionalidades, Efectividad y Aplicaciones Útiles en Colombia*. IOS.

Apple Inc. (s.f.). *iOS 8*. Obtenido de <https://www.apple.com/es/ios/what-is/>

AppThwack. (s.f.). *Test your Android, iOS, and web apps in real devices*. Obtenido de <https://appthwack.com/>

Beavis, G. (17 de abril de 2014). *Techradar*. Obtenido de The best smartphones compared: what phone is best for you?:

<http://www.techradar.com/news/phone-and-communications/mobile-phones/20-best-mobile-phones-in-the-world-today-645440/10#articleContent>

Becker, F. (05 de 02 de 2012). *Andes 2K - Gestión Documental*. Recuperado el 12 de 05 de 2012, de Andes 2K - Gestión Documental: http://www.andes2000.cl/index.php?option=com_content&view=article&id=133&Itemid=180

Carrasco. (2011). *Gestión de Procesos*. Colombia: Evolución.

Casmon, J. (14 de 06 de 2011). *Wikipedia - Gestión Documental*. Recuperado el 02 de 05 de 2012, de Wikipedia - Gestión Documental: http://es.wikipedia.org/wiki/Gesti%C3%B3n_documental

Cepeda, L. A. (2013). *BPM el futuro de los procesos de negocio*. Madrid: Gardnet Group.

Condesa. (4 de julio de 2011). *Androideity*. Obtenido de <http://androideity.com/2011/07/04/arquitectura-de-android/>

Debate. (11 de septiembre de 2012). *10 mejores celulares del 2012*. Obtenido de El Debate: <http://www.debate.com.mx/eldebate/noticias/default.asp?IdArt=12467906&IdCat=17585>

Eclipse Foundarion. (s.f.). *About us*. Obtenido de <https://eclipse.org/org/>

El Comercio, P. G. (7 de diciembre de 2013). *El Comercio*. Obtenido de Los 10 mejores smartphones del 2013: http://www.elcomercio.com.ec/tecnologia/smartphones-tecnologia-celulares-Galaxy-Iphone-LG-Xperia-Nexus-blackberry_0_1043295846.html

Fling, B. (2009). *Mobile Desing and Development*.

- Floría, A. (27 de 11 de 2011). *Sidar - Prototipado*. Recuperado el 19 de 06 de 2012, de Sidar - Prototipado: <http://www.sidar.org/recur/desdi/traduc/es/visitable/tecnicas/Prototyping.htm>
- Gabheran. (14 de diciembre de 2012). *Android*. Obtenido de Universidad Politécnica de Valencia: <http://histinf.blogs.upv.es/2012/12/14/android/>
- Gálvez, S. (s.f.). *Java Platform Micro Edition (Java ME)*. Obtenido de <http://www.sicuma.uma.es/es/formacion/doctoradoargentina/independientes/argentina08/Gaona-Perez/inicio.html>
- Gargenta, M. (2011). *Learning Android*. España: O'Reilly Media.
- Google. (2014). *Registro de Programadores*. Obtenido de <https://support.google.com/googleplay/android-developer/answer/113468?hl=es-419>
- Gramlich, N. (2010). *Andbook*. Inglaterra: Anddev Org.
- Haseman, C. (2008). *Android Essentials*. Estados Unidos: Apress.
- Hildenbrand, J. (23 de enero de 2012). *Android A to Z: What is a kernel?* Obtenido de <http://www.androidcentral.com/android-z-what-kernel>
- IEEE Computer Society, I. (22 de 10 de 1998). *standards ieee*. Recuperado el 15 de 07 de 2012, de standards ieee: <http://standards.ieee.org/findstds/standard/830-1998.html>
- InformáticaHoy. (s.f.). *Componentes de un Sistema Operativo Móvil*. Obtenido de <http://www.informatica-hoy.com.ar/soluciones-moviles/Componentes-de-un-Sistema-Operativo-Movil.php>
- losdroides. (septiembre de 2012). *La historia tras los nombres de postres para Android*. Obtenido de <http://iosdroides.blogspot.com/2012/09/la-historia-tras-los-nombres-de-postres.html>

- Lequerica, J. R. (2011). *Desarrollo de aplicaciones para Android*. México: Anaya Multimedia.
- Marcianosmx. (30 de diciembre de 2012). *Los celulares más baratos con Android*. Obtenido de <http://marcianosmx.com/celulares-baratos-con-android/>
- masadelante. (s.f.). *¿Qué es un Sistema Operativo?* Obtenido de <http://www.masadelante.com/faqs/sistema-operativo>
- Motorola Solutions. (2014). *RhoMobile Suite*. Obtenido de http://www.motorolasolutions.com/US-EN/Business+Product+and+Services/Software+and+Applications/RhoMobile+Suite?WT.mc_id=RhoMobile+Suite
- Nadel, B. (25 de octubre de 1994). Simon: More than a Phone, Less than a PC. *PC Magazine*, 46.
- Nieto Rojas, D. (28 de 09 de 2011). *Los CMS*. Recuperado el 15 de 06 de 2012, de Los CMS: <http://www.deividart.com/blog/?p=106>
- Nvidia Themes Symbian Theme. (s.f.). Obtenido de <http://megahdwall.com/high-resolution-wallpaper/nvidia-themes-symbian-theme-1377893.html>
- OhMyGeek. (2012 de agosto de 13). *WhatsApp está de vuelta en Windows Phone*. Obtenido de <http://www.ohmygeek.net/2014/06/01/volvio-whatsapp-para-windows-phone/>
- Oracle. (s.f.). *JAVA PLATFORM, MICRO EDITION (JAVA ME)*. Obtenido de <http://www.oracle.com/technetwork/java/embedded/javame/index.html>
- Oracle. (s.f.). *The Java EE 6 Tutorial*. Obtenido de <http://docs.oracle.com/javasee/6/tutorial/doc/gijvh.html>
- P, D. (21 de noviembre de 2013). *How to bring the KitKat interface to any Android with Apex Launcher 2.2, transparency included*. Obtenido de

http://www.phonearena.com/news/How-to-bring-the-KitKat-interface-to-any-Android-with-Apex-Launcher-2.2-transparency-included_id49546

Pérochon, S. (2012). *GUIA DE DESARROLLO DE APLICACIONES PARA SMARTPHONES*. Ediciones ENI.

Perumal, T. (2012). *Middleware Technology*. Obtenido de <http://hometoys.com/emagazine/2008/04/middleware-technology-for-digital-home-services/1030>

Perumal, T. (s.f.). *Middleware Technology*. Obtenido de <http://hometoys.com/emagazine/2008/04/middleware-technology-for-digital-home-services/1030>

Puon, L. (31 de marzo de 2014). *Los 5 tipos de usuarios móviles*. Obtenido de Merca20: <http://www.merca20.com/los-5-tipos-de-usuarios-moviles/>

Readthedocs. (s.f.). *Android OS*. Obtenido de <http://androidos.readthedocs.org/en/latest/data/historia/>

Softonic. (28 de enero de 2013). *BlackBerry: la evolución de su sistema operativo en imágenes*. Obtenido de <http://articulos.softonic.com/evolucion-sistema-operativo-blackberry-imagenes>

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

Sommerville, I. (2005). *Ingeniería del Software*. Top Printer Plus. S. L. L.

Sommerville, I. (2006). *Ingeniería del software* (Séptima Edición ed.). Madrid.

Tedeschi, N. (s.f.). *MSDN*. Obtenido de Web Services , un ejemplo práctico: <http://msdn.microsoft.com/es-es/library/bb972248.aspx>

Trejejo Alonso, J. A. (2007). *Aprendiendo a Crear y Mantener Sitios Web*. Asturias: joomlaos.

udtools. (s.f.). *Modelo Vista Controlador*. Obtenido de <https://foro.udtools.net/showthread.php?12382-Modelo-Vista-Controlador>

Universidad de Oriente. (s.f.). *Metodologías para el desarrollo de software*. Obtenido de Modelo en Cascada : http://wiki.monagas.udo.edu.ve/index.php/Metodolog%C3%ADas_para_el_desarrollo_de_software#Metodolog.C3.ADas_Vs._Ciclo_de_vida

Velez, J. (05 de Diciembre de 2013). *Top 10: Posibles avances en smartphones para 2014*. Obtenido de ALTONIVEL.COM.MX: <http://www.altonivel.com.mx/39594-top-10-posibles-avances-en-smartphones-para-2014.html>

Weitzenfeld, A. (2005). *Ingeniería de Software Orientada a Objetos Con Java E Internet*. Mexico: Thomson.

ANEXOS

ANEXOS

ANEXO 1. "CERTIFICADO DIGITALIZADO DE PRUEBAS Y ACEPTACIÓN DEL SISTEMA"



Quito, 16 de febrero del 2015

CERTIFICADO

A quien interese:

Por medio de la presente, CERTIFICO, que la aplicación para elaborar ordenes de trabajo en dispositivos móviles desarrollada por el Sr. Bladimir Ramiro Vallejo Altuna, fue concluida y puesta a prueba por el lapso de una semana laboral, del 2 al 6 de febrero del año en curso; Además certifico que después de las pruebas realizadas en este periodo, la aplicación cumple con las expectativas y aceptación por parte de la empresa y de los clientes.

Es todo cuanto puedo certificar y autorizo al Sr. Bladimir Vallejo Altuna hacer uso del presente Certificado, como mejor creyera conveniente.

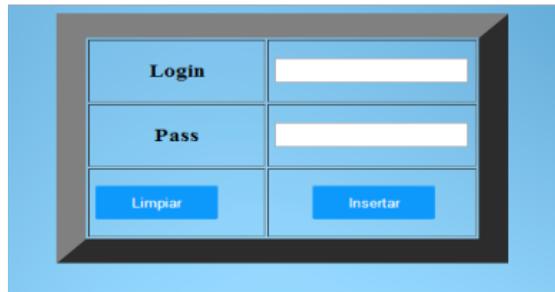
Atentamente,

Nelson Sarango
GERENTE GENERAL
AUVICOM

ANEXO 2 “MANUAL DE USUARIO”

MENÚ ADMINISTRADOR

Para iniciar el uso del sistema, indicará una ventana de inicio de sesión donde el usuario tiene que digitar sus datos de verificación para poder acceder al mismo con el rol al que le fue asignado.

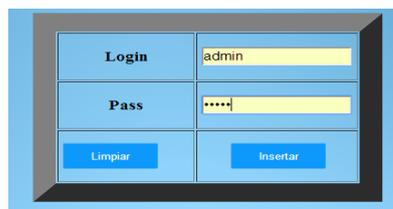


Formulario de inicio de sesión con los siguientes elementos:

Login	<input type="text"/>
Pass	<input type="password"/>
Limpiar	Insertar

Ingreso al sistema por parte del Administrador (Super Usuario)

- La página de inicio es la misma en todos los casos, la modificación se ve cuando el usuario ingresa al sistema con sus datos de verificación y le aparecen las diferentes opciones a las cuales tiene acceso.



Formulario de inicio de sesión con los siguientes elementos:

Login	admin
Pass
Limpiar	Insertar

- En esta ocasión se está documentando el caso del ingreso al sistema por parte del administrador o dentro de la estructura del sistema es conocido como **super usuario** el cual tiene acceso a todas las funcionalidades del sistema y registros en la base de los diferentes usuarios que están dentro de la interacción de emitir una orden de trabajo por parte de la empresa AUVICOM



- **Ingresar nuevos usuarios al sistema**

El administrador del sistema tiene la posibilidad de ingresar a nuevos usuarios a la base, así también está habilitado para poder designar el rango que tendrá este usuario y así brindarle un rol con el cual va a tener acceso a las diferentes funcionalidades que posee la aplicación de acuerdo al estándar sobre el cual está trabajando.

A screenshot of a web form titled "Mantenimiento de Usuarios". The form contains several input fields: "Codigo", "Nombre", "Apellido", "Direccion", "Cargo", "Correo", "Telefono", "Rango" (a dropdown menu), "Login", and "Contraseña". There are three buttons: "Buscar" (next to the Codigo field), "Guardar" (at the bottom right), and "Insertar" (at the bottom right).A close-up screenshot of the "Rango" dropdown menu. The dropdown is open, showing the following options: "Administrador", "Tecnico", "Cliente", and "Operario". The "Administrador" option is currently selected and highlighted in blue.

Una vez ingresados los datos del nuevo usuario a la aplicación, en la parte inferior del formulario hay una opción llamada insertar con la cual se completa el registro de los datos del usuario en la base de datos y lo posibilita para realizar el ingreso al sistema cuando así lo requiera.

Al momento de asignar un rango al usuario, si la selección realizada por el administrador es la de **Técnico** en la parte inferior aparecen nuevos campos en los cuales se realiza el registro de datos adicionales como detalle del técnico que se está registrando en el sistema, además del tiempo de trabajo que lleva laborando, fecha de ingreso y tipo de jornada de trabajo con el cual él se desarrolla en la empresa.

Correo: marco@hotmail.com
Telefono: 0987564132
Rango: Técnico
Login: gmarco
Contraseña: 001

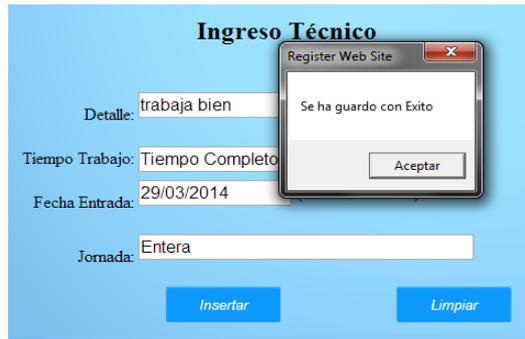
Limpiar
Insertar

Ingreso Técnico

Detalle:
Tiempo Trabajo:
Fecha Entrada: (DD/MM/YYYY)
Jornada:

Insertar
Limpiar

Al aceptar la petición de ingreso aparece una ventana de diálogo donde confirma la petición de ingresar esos datos en la base.



El administrador esta posibilitado para realizar la modificación de registros de usuarios que se encuentran en el sistema, para lo cual previamente realiza la búsqueda el usuario a través de su número de cédula. Si el usuario se encuentra en el sistema aparece el nombre del mismo en la parte superior de la ventana con la opción de **seleccionar**, para que se carguen los datos del registro que se busca y poder realizar las actualizaciones que crea convenientes el administrador.



Una vez que el administrador haya realizado las modificaciones pertinentes tiene la opción de modificar para que se almacenen los registros actualizados y pueda ser visualizado en el sistema.

Selección 1703820488 Marco Gualoto

Cédula: 1703820488

Nombre: Marco

Apellido: Gualoto

Dirección: La Comuna

Cargo: Empleado Público

Correo: marco@hotmail.com

Teléfono: 0987564132

Rango: Técnico

Login: gmarco

Contraseña: 001

Cuando escogió la opción de modificar le aparece una pantalla de diálogo donde le pregunta si está seguro de realizar el cambio de esos registros con las opciones de si y no donde puede confirmar o denegar la acción.

Dentro de lo concerniente al mantenimiento de los datos, también existe la opción de eliminar, donde se posibilita al administrador a realizar la eliminación del registro de un usuario cuando estime conveniente, cuando ya se realizó el proceso aparece una ventana de diálogo donde se indica si se está seguro de realizar la operación de eliminado del registro con las opciones de si y no, para poder aceptar o denegar la petición en el sistema.

Cuando se realiza la búsqueda de registros en el sistema y estos no están almacenados en la base de datos, la aplicación le muestra en pantalla un cuadro de diálogo notificándole que no se ha encontrado el registro para que el administrador pueda ingresarlo al sistema o corregir la información, que en este caso es el número de cédula por alguna equivocación que pudiera existir en el ingreso del mismo a la búsqueda del registro.

Mantenimiento de Usuarios

Cedula: Bus

Nombre:

Apellido:

Dirección:

Cargo:

Register Web Site

No se encontraron datos

Aceptar

La opción que posee el menú del administrador en el sistema, denominada **orden de trabajo** le posibilita al super usuario a crear y llevar un control de las órdenes de trabajo que se ingresen en el sistema.

MENU ADMINISTRADOR

- **Crear Orden de trabajo**

Esta opción del menú del administrador lleva a la creación de la orden de trabajo, donde se encuentran descritos todos los datos informativos sobre la cual se crea la orden con un cliente.

TOSHIBA <i>Leading Innovation</i> <i>Servicio Técnico Garantizado</i>		AUVICOM N° ORDEN DE TRABAJO 101	
ID CLEINTE:	<input type="text"/>	<input type="button" value="CONSULTAR"/>	<input type="text"/>
Nombre:	<input type="text"/>	Apellidos:	<input type="text"/>
Dirección:	<input type="text"/>		
Correo:	<input type="text"/>	Teléfono:	<input type="text"/>
Artículo:	<input type="text"/>	Marca:	<input type="text"/>
Modelo N°:	<input type="text"/>	Serie N°:	<input type="text"/>
Tipo de Atención:	<input type="text"/>		
Fecha de Ingreso:	<input type="text" value="29/3/2014"/>		
Fecha de Entrega:	<input type="text"/>		

En esta sección de la orden de trabajo se encuentra descrito las características de la orden, como son el daño del equipo, accesorios que el cliente deja con el equipo que se desea reparar y otras opciones referentes a pagos y observaciones que se encuentren en la recepción de trabajo.

The image shows a screenshot of a service order form. It contains several input fields and labels, including:

- Accesorios (Accessories)
- Daño (Damage)
- Observaciones (Observations)
- Fecha de pago (Payment date)
- Tipo de pago (Payment type)
- Accesorios (Accessories)
- Daño (Damage)
- Observaciones (Observations)

La aplicación se encuentra validada en toda su estructura por la cual, campos como por ejemplo la cédula que son indispensables para el registro de la orden en el sistema se lo deja vacío, este les genera una advertencia donde les notifica que ese campo es requerido y que no se puede realizar ningún proceso si no es proporcionado ese dato.

The image shows a screenshot of a customer search form. It includes the following elements:

- Logos for TOSHIBA (Leading Innovation, Servicio Técnico Garantizado) and AUVICOM (N° ORDEN DE TRABAJO).
- A search field for 'ID CLIENTE' with a red asterisk indicating a required field.
- A 'CONSULTAR' button.
- Fields for 'Nombres', 'Apellidos', 'Dirección', 'Correo', and 'Teléfono'.
- A red error message at the bottom: 'Se Presentaron Los Siguietes Errores: * Cédula Requerido'.

Para poder generar la orden de trabajo el cliente se debe encontrar registrado en el sistema por lo cual, a través del número de cédula se realiza la búsqueda y automáticamente se toma la información del cliente de la base de datos y se llenan los campos de contacto de la orden de trabajo.

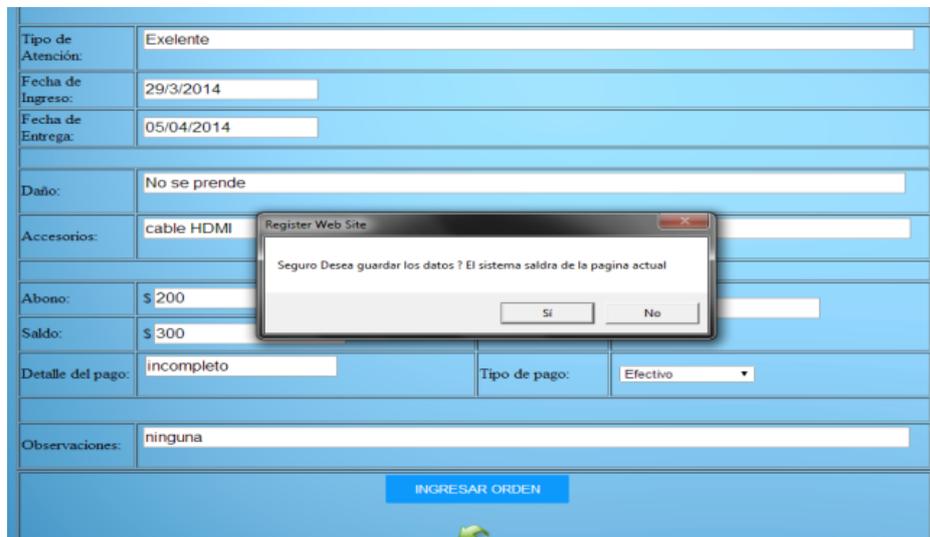
TOSHIBA Leading Innovation Servicio Técnico Garantizado		AUVICOM N° ORDEN DE TRABAJO 102	
ID CLEINTE:	1707018576	CONSULTAR	
Nombres:	Karen	Apellidos:	Cedeño
Dirección:	Quitumbe		
Correo:	karenc@hotmail.com	Teléfono:	0999856231

Ya con los datos de contacto registrados en la parte superior de la orden de trabajo, se procede a realizar el llenado de los demás campos referentes al tipo de artículo, marca, modelo, serie, entre otros con los cuales se generará la orden de trabajo.

Servicio Técnico Garantizado		N° ORDEN DE TRABAJO 102	
ID CLEINTE:	1707018576	CONSULTAR	
Nombres:	Karen	Apellidos:	Cedeño
Dirección:	Quitumbe		
Correo:	karenc@hotmail.com	Teléfono:	0999856231
Artículo:	ps4	Marca:	Sony
Modelo N°:	11414	Serie N°:	00012112
Tipo de Atención:	Excelente		
Fecha de Ingreso:	29/3/2014		
Fecha de Entrega:	05/04/2014		
Daño:	No se prende		

Una vez registrados todos los datos de la orden de trabajo, se procede a ingresarla al sistema, a través de la opción **Ingresar Orden** donde al momento de seleccionar esta opción aparecerá un cuadro de **diálogo** que notificará sobre si se debe guardar ese registro de esa orden con

las opciones de si y no, para poder confirmar o cancelar la petición en el sistema.



The image shows a web form for entering an order. The form has several fields: 'Tipo de Atención' (Exelente), 'Fecha de Ingreso' (29/3/2014), 'Fecha de Entrega' (05/04/2014), 'Daño' (No se prende), 'Accesorios' (cable HDMI), 'Abono' (\$ 200), 'Saldo' (\$ 300), 'Detalle del pago' (incompleto), and 'Tipo de pago' (Efectivo). There is an 'Observaciones' field with 'ninguna'. A blue button labeled 'INGRESAR ORDEN' is at the bottom. A dialog box titled 'Register Web Site' is overlaid on the form, asking 'Seguro Desea guardar los datos ? El sistema saldra de la pagina actual' with 'Si' and 'No' buttons.

Al momento de seleccionar SI en la aplicación esta mostrará una ventana de diálogo donde notificará sobre la correcta ejecución del proceso de ingreso de la orden al sistema y con la opción de aceptar para finalizar el proceso.

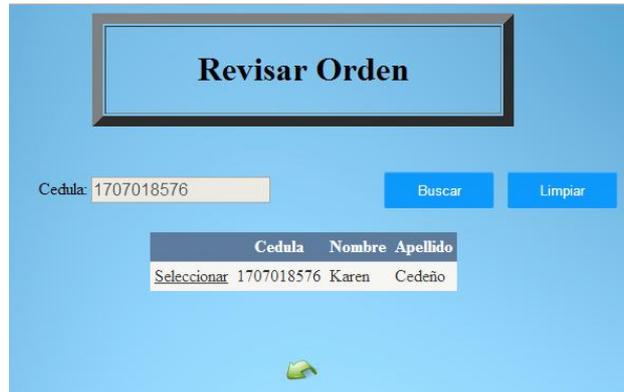
- **Revisar y asignar orden**

La siguiente opción en el menú del administrador es la de revisar y asignar orden con la cual a través del número de cédula se realiza la búsqueda de la misma en el sistema para verificar y asignar trabajos a los técnicos.



The image shows a form titled 'Revisar Orden'. It has a search field labeled 'Cedula:' and two buttons: 'Buscar' and 'Limpiar'.

Al momento de realizar la búsqueda aparece los resultados de la misma en la parte inferior de la ventana con la opción de seleccionar, para cargar la información referente a ese registro.



The screenshot shows a web interface titled "Revisar Orden". At the top, there is a search input field labeled "Cedula:" containing the value "1707018576". To the right of the input field are two buttons: "Buscar" and "Limpiar". Below the search area, there is a table with the following data:

	Cedula	Nombre	Apellido
Seleccionar	1707018576	Karen	Cedeño

Cuando se ingresa a la información del registro encontrado, despliega las órdenes que se encuentran registradas con ese nombre, con su respectiva descripción donde se visualiza en número de orden, el tipo de artículo y fecha de entrega, teniendo cada uno de estos la opción de seleccionar, para ingresar a la información de cada artículo y su respectiva descripción en la orden de trabajo destinada para ello.



The screenshot shows the same "Revisar Orden" web interface. The search input field still contains "1707018576". Below the search area, there is a table with the following data:

	Cedula	Nombre	Apellido
Seleccionar	1707018576	Karen	Cedeño

Below this table, there is another table showing a list of orders:

	Numero Orden	Articulo	Fecha de Entrega
Seleccionar	101	Ps3	05/04/2014
Seleccionar	102	ps4	05/04/2014
Seleccionar	103	Portatil	10/04/2014

Al seleccionar el registro de un artículo determinado para el cual se generó la orden de trabajo se puede visualizar la información del daño, observaciones, pagos, entre otros, con los cuales el administrador está posibilitado a designar al o los técnicos que van a estar inmersos en una orden de trabajo específica.

Seleccionar 1707018576 Karen Cedeño

	Numero Orden	Artículo	Fecha de Entrega
Seleccionar	101	Ps3	05/04/2014
Seleccionar	102	ps4	05/04/2014
Seleccionar	103	Portatil	10/04/2014

El Numero de Orden de Trabajo es : 101
 Artículo : Ps3, Daño : No se prende, Observaciones : ninguna

Abono:	\$90
Saldo:	\$200
Total:	\$200
Tipo de pago:	Efectivo
Detalle:	incompleto
TÉCNICOS	
Principal:	Ninguno
Técnico 2:	Ninguno
Auxiliar:	Ninguno

TÉCNICOS	
Principal:	Ninguno
Técnico 2:	Ninguno Santiago Martínez
Auxiliar:	Elisa Cozar

Una vez seleccionados los técnicos hay una opción en el sistema llamada enviar, donde se le asigna la orden al técnico para que de inmediato este sea notificado y pueda realizar la operación.

En el menú administrador se encuentra la opción de crear proforma donde lleva a una nueva ventana de reportes donde se encontrara la información de las órdenes registradas en el sistema y de las cuales se

obtendrá la información concerniente a una de ellas, que se la localizará a través del número de cédula con que se ingresó el cliente en el sistema.



- **Selección de reportes**

Esta sección ayuda en la búsqueda de los registros en el sistema para lo cual se emplea el número de cédula del usuario para su localización.



Una vez realizada la búsqueda y encontrado el registro se procede a su selección para obtener la información referente a las órdenes de trabajo registradas con ese número.



En esta sección muestra el resultado de la búsqueda donde en cada uno de los registros hay la opción seleccionar y con la cual se puede acceder a su respectivo reporte.

Selección de Reporte

Cédula:

	Cédula	Nombre	Apellido
Seleccionar	1707018576	Karen	Cedeño

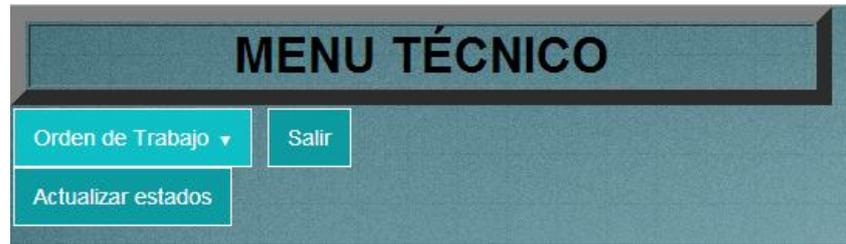
	Numero Orden	Artículo	Fecha de Entrega
Seleccionar	101	Ps3	05/04/2014
Seleccionar	102	ps4	05/04/2014
Seleccionar	103	Portatil	10/04/2014

TOSHIBA <i>Leading Innovation</i> Servicio Técnico Garantizado		AUVICOM N° ORDEN DE TRABAJO 102	
ID CLIENTE:	1707018576		
Nombre:	Karen	Apellidos:	Cedeño
Dirección:	Quitumbe		
Correo:	karenc@hotmail.com	Teléfono:	0999856231
Artículo:	ps4	Marca:	Sony
Modelo N°:	Sony	Serie N°:	00012112
Tipo de Atención:	Ejecente		
Fecha de Ingreso:	29/3/2014		
Fecha de Entrega:	05/04/2014		
Daño:	No se prende		
Accesorios:	cable HDMI		

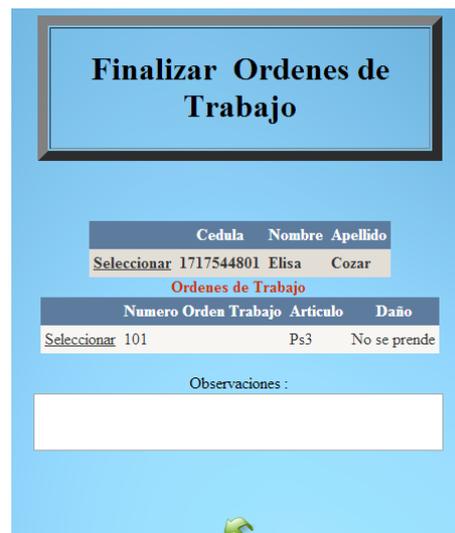
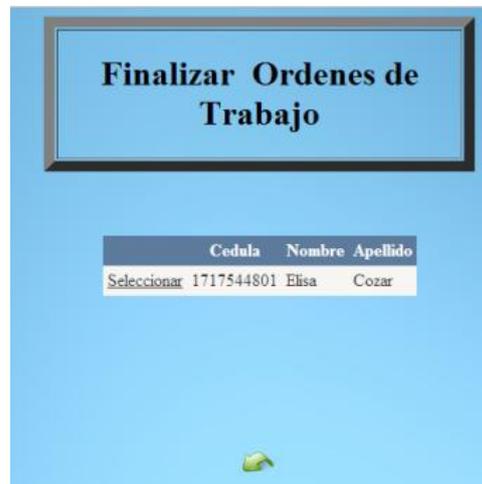
Abono:	\$200	TOTAL:	\$300
Saldo:	\$300		
Detalle del pago:	incompleto	Tipo de pago:	Efectivo
Observaciones:	ninguna		

MENÚ TÉCNICO

El técnico es otro elemento que está involucrado en la estructuración del sistema y como el administrador también tiene su rol y por ende su menú.

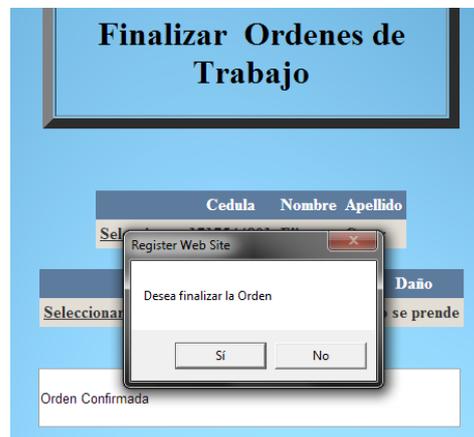


La opción actualizar estados lleva a una nueva ventana denominada finalizar órdenes de trabajo, donde se despliega la lista de órdenes registradas en la empresa para lo cual el técnico puede acceder y realizar modificaciones.



El sistema se encuentra controlando todas las actividades que realice el usuario por lo cual si no realiza algún paso en el proceso, el sistema le despliega un mensaje notificándole lo acontecido para que verifique y realice la corrección respectiva y se complete el proceso.

Al momento de llenar todos los campos de la orden, y tener listo para procesar se finaliza la orden y el sistema muestra en pantalla un cuadro de diálogo con las opciones de si y no para aceptar o cancelar la operación.



Al seleccionar si y aprobar el proceso el sistema muestra en pantalla un mensaje notificándoles de esta forma la finalización de la orden con una opción de aceptar finalizar el proceso.

MENÚ OPERARIO

El usuario Operario tiene las opciones de crear Órdenes de trabajo, así como de un reporteador el cual le permite ver la información de las órdenes de trabajo que ya han sido creadas



- Creación de orden de trabajo

Para la creación de una nueva orden de trabajo, dentro del menú se encuentra el botón "Crear"



Se presenta una página para el ingreso de datos

A screenshot of a data entry form titled "Servicio Técnico Garantizado". The form is blue and contains various input fields for customer information, service details, and payment information. The fields are organized into a grid-like structure. The "Fecha de Ingreso" field is pre-filled with "2/4/2014". The "Tipo de pago" field is set to "Efectivo".

Al llenar el campo de la ID del cliente y posteriormente hacer una consulta, los datos del mismo serán enviados del registro de datos hasta la aplicación para hacer más fácil la asignación de una orden de trabajo

TOSHIBA Leading Innovation Servicio Técnico Garantizado		AUVICOM N° ORDEN DE TRABAJO	
ID CLEINTE:	900540782	CONSULTAR	
Nombre:	martin	Apellidos:	valverde
Dirección:	martin		
Correo:	mf@hotmail.com	Teléfono:	5558744125
Artículo:		Marca:	

Se llenan todos los datos necesarios y al presionar en el botón “ingresar orden” la aplicación mostrará un mensaje donde se debe aceptar la creación de una nueva orden de trabajo, posteriormente se guardará toda la información en los registros internos del sistema y la página se cerrará para evitar datos duplicados

ID CLEINTE:	900540782	CONSULTAR	
Nombre:	martin	Apellidos:	valverde
Dirección:	martin		
Correo:	mf@hotmail.com	Teléfono:	5558744125
Artículo:	Laptop	Marca:	Toshiba
Modelo N°:	serie 1	Serie N°:	1112254
Tipo de Atención:	Buena		
Fecha de Ingreso:	2/4/2014		
Fecha de Entrega:	2/10/2014		
Dato:	no se prende		
Accesorios:	mouse y enfriador		
Abono:	\$ 20	TOTAL:	\$ 30
Saldo:	\$ 10		
Detalle del pago:	sin novedad	Tipo de pago:	Efectivo
Observaciones:	se presenta una orden de pago		
INGRESAR ORDEN			

Registrar Web Site

Seguro: ¿Desea guardar los datos ? El sistema saldrá de la página actual.

- **Vista de Reporte**

Para visualizar las órdenes de trabajo que ya han sido creadas, dentro del menú se encuentra el botón “Reportes”



Al presionar en el botón anteriormente mencionado, direccionará a una página donde se realizará la respectiva búsqueda



Se introduce dentro del campo, la cédula del cliente el cual este asignado la orden que se quiere reportear. Al dar clic en el botón “Buscar” presentará una serie de clientes, en donde se escogerá uno dentro de la lista; posteriormente mostrará otra lista con todas las órdenes de trabajo que han sido asignadas a dicho cliente



Al seleccionar una orden de trabajo dentro de la lista la aplicación devolverá el reporte completo

TOSHIBA <i>Leading Innovation</i> Servicio Técnico Garantizado		AUVICOM N° ORDEN DE TRABAJO	
		101	
ID CLEINTE:	900540782		
Nombre:	martin	Apellidos:	valverde
Dirección:	maria		
Correo:	jef@hotmail.com	Teléfono:	5558744125
Artículo:	Laptop	Marca:	Toshiba
Modelo N°:	Toshiba	Serie N°:	1112254
Tipo de Atención:	Buena		
Fecha de Ingreso:	2/4/2014		
Fecha de Entrega:	2/10/2014		
Daño:	no se prende		
Accesorios:	mouse y enfriador		
Abono:	\$20	TOTAL:	\$ 30
Saldo:	\$ 10		
Detalle del pago:	sin novedad	Tipo de pago:	Efectivo
Observaciones:	se presenta una orden de pago		

MENÚ CLIENTE

El usuario Cliente tiene las opciones:

- Contactos: la cual le permite solicitar una proforma a la empresa
- Ordenes de trabajo: en la cual se listan las órdenes que han sido ingresadas, así como los estados y pagos de las mismas



- **Enviar Requerimiento**

Para la creación de un nuevo requerimiento (envío de solicitud a la empresa), dentro del menú se encuentra el botón "Enviar Requerimiento"



La aplicación muestra una página donde se podrá llenar todos los datos respectivos a cuanto una proforma se refiere.

The screenshot shows a web interface with a blue background. At the top, there is a title box containing the word "Contactos". Below the title, a red message reads "Bienvenido martin valverde". The main part of the form consists of four input fields labeled "Articulos:", "Marca:", "Daño:", and "Detalle:". Each field is currently empty. At the bottom of the form, there are two blue buttons labeled "Enviar" and "Limpiar", with a small green mouse cursor icon positioned between them.

Se llenarán los campos con datos pertinentes para la creación de una proforma

This screenshot shows the same web form as the previous one, but now the input fields are filled with text: "Articulos:" contains "2 Computadoras de escritorio", "Marca:" contains "Dell", "Daño:" contains "no se prende", and "Detalle:" contains "incluye fuente de poder y mouse". The "Enviar" and "Limpiar" buttons are still present. A small dialog box titled "Register" is open in the foreground, displaying the question "Seguro que desea enviar la informacion?" and two buttons labeled "Si" and "No".

Al dar clic en el botón "Enviar" la aplicación registrará la solicitud para que sea procesada y revisada por la empresa

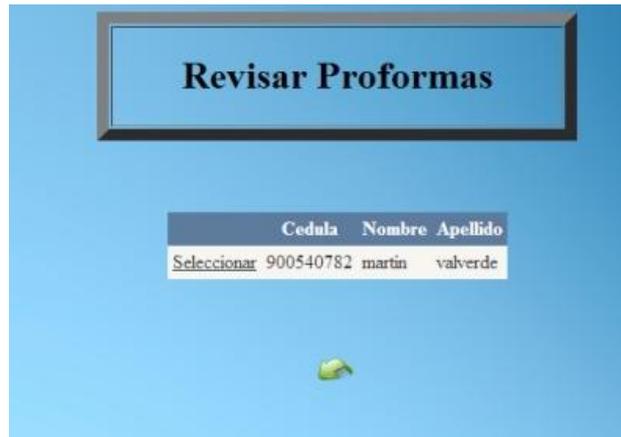


- **Revisar Proforma**

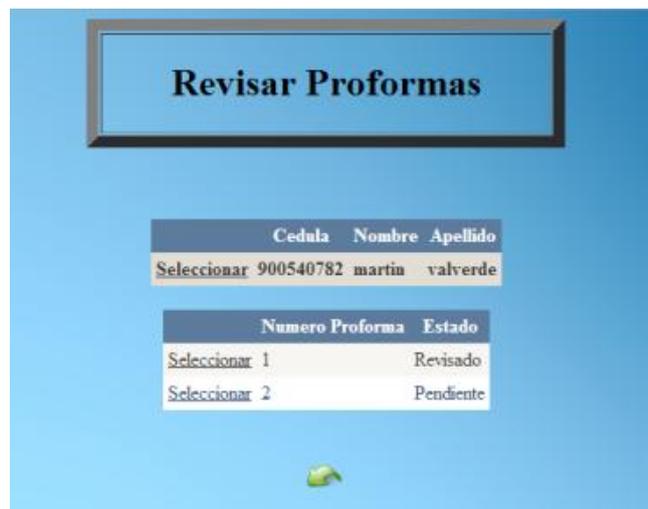
Para revisar las proformas, dentro del menú existe una opción llamada “Revisar Proforma” que mostrará los datos solicitados en cuanto a proformas se refiere



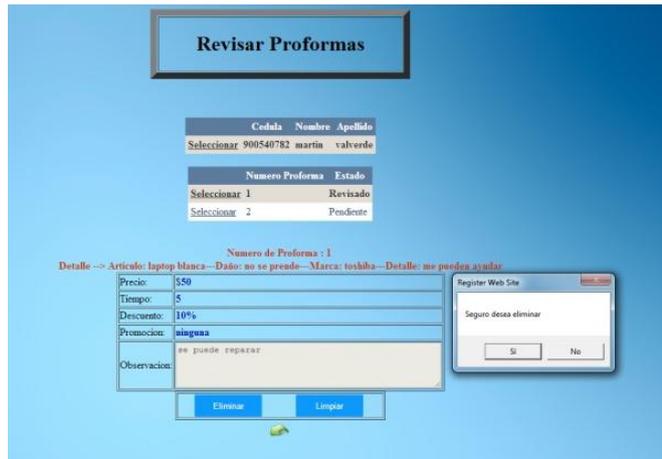
Al seleccionar la opción antes mencionada, presentará una página donde se confirmara los datos



Al hacer la selección de nuestra información se presentará las proformas que han sido revisadas y las cuales aún están en estado “pendiente”



Se selecciona una de las proformas listas, se presentará el administrador de proformas. Donde se podrá consultar los datos y eliminarlos si es que ya no son de utilidad para el cliente

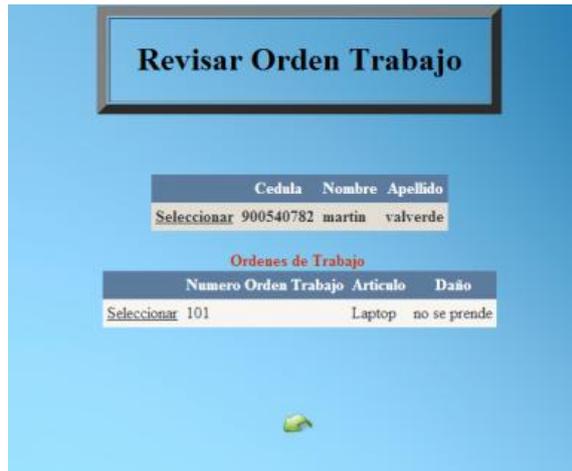


- **Revisar Estados**

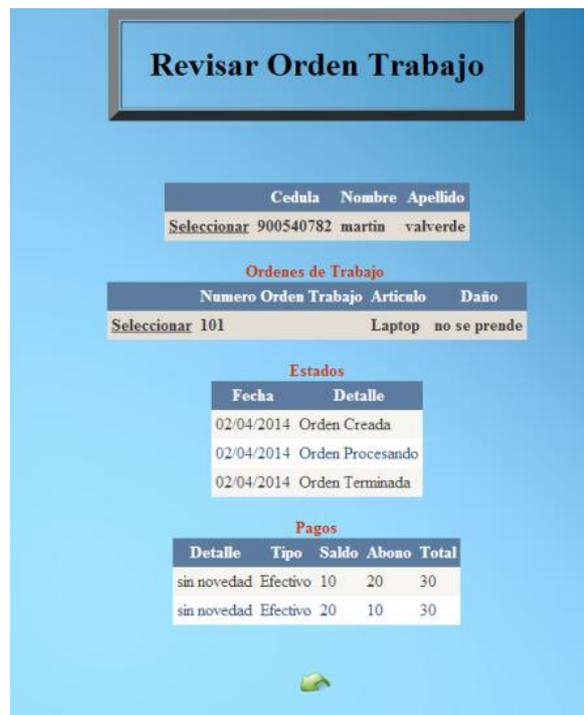
Para revisar los estados y pagos de las órdenes de trabajo de un cliente, dentro del menú existe una opción llamada “Revisar Estados”



La aplicación muestra una pantalla donde se confirmará los datos y presentará una lista de las órdenes de trabajo que el cliente tiene



De la lista que presentará, se selecciona una orden de trabajo y la aplicación presentará todos los datos relacionados con la misma, que son: estados y pagos.



MENÚ EMPRESA

El usuario Empresa tiene las opciones:

- Proforma: la cual le permite revisar las proformas enviadas por parte del cliente y adjuntar una respuesta
- Ordenes de trabajo: en la cual se listan las órdenes que han sido ingresadas y posteriormente se asigna los técnicos que trabajarán para cada una de las mismas



- **Crear Proforma**

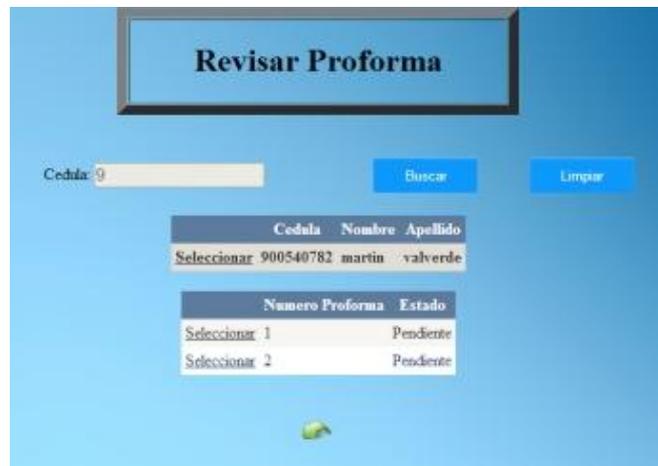
Para revisar y crear una proforma y enviarle a un cliente, dentro del menú existe una opción llamada "Crear Proformas"



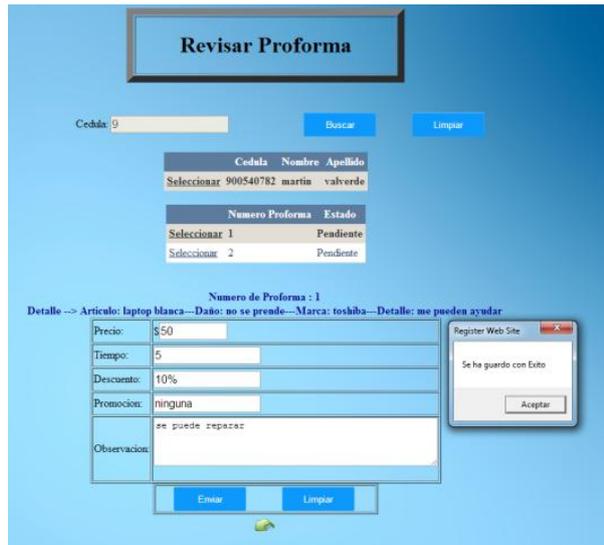
Se selecciona la opción y el sistema enviará a una ventana de administración de proformas



Se realizará una búsqueda por cédula y aparecerán las proformas de cada cliente en una lista



Al seleccionar una de las proformas enviadas, se presentará los campos para su posterior llenado. Una que vez que se haya verificado y comprobado la información, se procede a dar click en el botón "Enviar", donde la aplicación guardará en sus registros la información de la proforma anexada al cliente antes seleccionado



Revisar y asignar orden

La siguiente opción en el menú del administrador es la de revisar y asignar orden con la cual a través del número de cédula se realizará la búsqueda de la misma en el sistema para verificar y asignar trabajos a los técnicos.



Al momento de realizar la búsqueda aparecen los resultados de la misma en la parte inferior de la ventana con la opción de seleccionar, para cargar la información referente a ese registro.



The screenshot shows a web interface titled "Revisar Orden". At the top, there is a search input field labeled "Cedula:" containing the value "1707018576". To the right of the input field are two buttons: "Buscar" and "Limpiar". Below the search area, there is a table with the following data:

	Cedula	Nombre	Apellido
Seleccionar	1707018576	Karen	Cedeño

Cuando se ingresa a la información del registro encontrado, se desplegarán las órdenes que se encuentran registradas con ese nombre, con su respectiva descripción donde se visualiza en número de orden, el tipo de artículo y fecha de entrega, teniendo cada uno de estos la opción de seleccionar, para ingresar a la información de cada artículo y su respectiva descripción en la orden de trabajo destinada para ello.



The screenshot shows the same "Revisar Orden" web interface. The search input field still contains "1707018576". The table below the search area now displays a list of orders:

	Numero Orden	Artículo	Fecha de Entrega
Seleccionar	101	Ps3	05/04/2014
Seleccionar	102	ps4	05/04/2014
Seleccionar	103	Portatil	10/04/2014

Al seleccionar el registro de un artículo determinado para el cual se generó la orden de trabajo se puede visualizar la información del daño, observaciones, pagos, entre otros, con los cuales el administrador está posibilitado a designar al o los técnicos que van a estar inmersos en una orden de trabajo específica.

Seleccionar 1707018576 Karen Cedeño

	Numero Orden	Artículo	Fecha de Entrega
Seleccionar	101	Ps3	05/04/2014
Seleccionar	102	ps4	05/04/2014
Seleccionar	103	Portatil	10/04/2014

El Numero de Orden de Trabajo es : 101
Artículo : Ps3, Daño : No se prende, Observaciones : ninguna

Abono:	\$ 90
Saldo:	\$ 200
Total:	\$ 200
Tipo de pago:	Efectivo
Detalle:	incompleto
TÉCNICOS	
Principal:	Ninguno
Técnico 2:	Ninguno
Auxiliar:	Ninguno

TÉCNICOS	
Principal:	Elisa Cozar
Técnico 2:	Santiago Martín
Auxiliar:	Ninguno

En el menú administrador se encuentra la opción de crear proforma donde lleva a una nueva ventana de reportes donde se encontrara la información de la órdenes registradas en el sistema y de la cuales se obtendra la información concerniente a una de ellas, que se localizará a través del número de cédula con que se ingresó el cliente en el sistema.



Selección de reportes

Esta sección ayuda en la búsqueda de los registros en el sistema para lo cual se emplea el número de cédula del usuario para su localización.



Una vez realizada la búsqueda y encontrado el registro se procede a su selección para obtener la información referente a las órdenes de trabajo registradas con ese número.



Selección de Reporte

Cedula:

Buscar

Limpiar

Cedula	Nombre	Apellido
Seleccionar 1707018576	Karen	Cedeño

Numero Orden	Articulo	Fecha de Entrega
Seleccionar 101	Ps3	05/04/2014
Seleccionar 102	ps4	05/04/2014
Seleccionar 103	Portatil	10/04/2014

TOSHIBA <i>Leading Innovation</i> Servicio Técnico Garantizado		AUVICOM N° ORDEN DE TRABAJO 102	
ID CLEINTE:	1707018576		
Nombres:	Karen	Apellidos:	Cedeño
Dirección:	Quitumbe		
Correo:	karencv@hotmail.com	Teléfono:	0999856231
Artículo:	ps4	Marca:	Sony
Modelo N°:	Sony	Serie N°:	00012112
Tipo de Atención:	Exelente		
Fecha de Ingreso:	29/3/2014		
Fecha de Entrega:	05/04/2014		
Daño:	No se prende		
Accesorios:	cable HDMI		

Abono:	\$200	TOTAL:	\$ 300
Saldo:	\$ 300		
Detalle del pago:	incompleto	Tipo de pago:	Efectivo
Observaciones:	ninguna		

ANEXO 3 “CÓDIGO COMENTADO”

Código Wfrm_usuarios_ingreso

Partial Class Paginas_Wfrm_usuarios_ingreso

```
Sub llenar_campos()
    'Se llenan los campos para la tabla TblUsuarios
    obj_usuario.Nombres = Me.txtnombre.Text
    obj_usuario.Apellidos = Me.txtapellido.Text
    obj_usuario.Direccion = Me.txtdireccion.Text
    obj_usuario.Usu_id = Me.txtcedula.Text
    obj_usuario.Login = Me.txtlog.Text
    obj_usuario.Pass = Me.txtcont.Text
    obj_usuario.Correo = Me.TxtCorreo.Text
    obj_usuario.Cargo = Me.TxtCargo.Text

    'Se llenan los campos para la tabla TblTipo_usuarios
    obj_tipo.Usu_Id = Me.txtcedula.Text
    obj_tipo.Rango = Me.DdlRango.Text
    obj_tipo.Detalle = "permiso otorgado"

    'Se llenan los campos para la tabla TblPersonal_tecnico
    obj_tecnico.Tec_id = Me.txtcedula.Text
    obj_tecnico.Detalle = Me.TxtDetalle.Text
    obj_tecnico.Estado = "Activo"
    obj_tecnico.Jornada = Me.TxtJornada.Text
    obj_tecnico.Tiempo_trabajo = TxtTiempo.Text
    obj_tecnico.Fecha_ingreso = TxtFecha.Text

    'Se llenan los campos para la tabla TblTelefono
    obj_telefono.Usu_Id = Me.txtcedula.Text
    obj_telefono.Tipo = "Predefinido"
    obj_telefono.Numero = Me.Txtnumero.Text
End Sub

Protected Sub BtnInsertar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnInsertar.Click

    'Llamada de la Función
    llenar_campos()

    'Validación de la cedula
    Dim ced As String
    ced = obj_extra.cedula(Me.txtcedula.Text)
    If ced = "1" Then
        obj_tipo.insertar()
        obj_telefono.insertar()
        Me.TxtMensaje.Text = obj_usuario.insertar()
        If Me.DdlRango.Text = "Tecnico" Then
            Me.txtcedula.Enabled = False
            Me.BtnModftecnico.Visible = False
            Me.div_trabajador.Visible = True
            Me.BtnInserTecnico.Visible = True
            Me.BtnInserTecnico.Enabled = True
        End If
    Else
        Me.TxtMensaje.Text = "La Cedula es incorrecta. Profavor Verifique"
    End If
End If
```

End Sub

Protected Sub Limpiar_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Limpiar.Click

```
'Limpia los cuadros de texto
Me.txtnombre.Text = ""
Me.TxtMensaje.Text = ""
Me.txtapellido.Text = ""
Me.txtcedula.Text = ""
Me.txtcont.Text = ""
Me.txtdireccion.Text = ""
Me.txtlog.Text = ""
Me.txtcedula.Text = ""
Me.Txtnumero.Text = ""
Me.TxtCargo.Text = ""
Me.TxtCorreo.Text = ""
```

```
Me.Modificar.Visible = False
Me.Eliminar.Visible = False
'Se activa el botón BtnBuscar
Me.BtnBuscar.Enabled = True
'Se activa el cuadro de texto txtcedula
Me.txtcedula.Enabled = True
'Limpia el DataGridView
Me.GVPersona.DataSource = Nothing
Me.GVPersona.DataBind()
```

End Sub

Protected Sub Modificar_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Modificar.Click

```
'Se llenan los campos de los cuadros de texto a las clases seleccionadas
obj_usuario.Usu_id = Me.txtcedula.Text
obj_tipo.Usu_Id = Me.txtcedula.Text
```

```
'Se realiza una consulta a la tabla TblUsuarios
obj_usuario.consultar()
'Se realiza una consulta por usuario a la tabla TblTipo_usuarios
obj_tipo.consultar_usuario()
```

```
'Llamada de la Función
llenar_campos()
```

```
'Actualiza la tabla TblUsuarios
Me.TxtMensaje.Text = obj_usuario.actualizar
```

```
'Actualiza la tabla TblTipo_usuarios
obj_tipo.actualizar()
```

```
'Actualiza la tabla TblTelefono
obj_telefono.actualizar()
```

```
'Se filtra para ver si el usuario es de tipo Tecnico
If Me.DdlRango.Text = "Tecnico" Then
    obj_tecnico.Tec_id = Me.txtcedula.Text
    Me.BtnModftecnico.Enabled = True
    Me.BtnModftecnico.Visible = True
    Me.BtnInserTecnico.Visible = False
```

```
'Se realiza una consulta a la tabla TblPersonal_tecnico
If obj_tecnico.consultar = "Consulta Exitosa" Then
```

```

        'Se guarda la información extraída por la consulta la tabla
TblPersonal_tecnico en los correspondientes cuadros de texto
        Me.TxtDetalle.Text = obj_tecnico.Detalle
        Me.TxtJornada.Text = obj_tecnico.Jornada
        Me.TxtTiempo.Text = obj_tecnico.Tiempo_trabajo
        Me.TxtFecha.Text = obj_tecnico.Fecha_ingreso

        Me.div_trabajador.Visible = True
        'Desactiva el Botón BtnInserTecnico
        Me.BtnInserTecnico.Enabled = False
        'Activa el Botón BtnModftecnico
        Me.BtnModftecnico.Enabled = True

    Else
        Me.TxtMensaje.Text = "No se encontraron datos"
    End If
End If

End Sub

Protected Sub Eliminar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Eliminar.Click

    'Elimina un usuario de la tabla TblUsuarios
    obj_usuario.Usu_id = Me.txtcedula.Text
    Me.TxtMensaje.Text=obj_usuario.eliminar()

End Sub

Protected Sub GVPersona_SelectedIndexChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles GVPersona.SelectedIndexChanged

    'Se realiza una consulta a la tabla TblUsuarios
    obj_usuario.Usu_id = Me.GVPersona.SelectedRow.Cells(1).Text
    If obj_usuario.consultar = "Consulta Exitosa" Then

        'Se guarda la información extraída por la consulta la tabla TblUsuarios
en los correspondientes cuadros de texto
        Me.txtnombre.Text = obj_usuario.Nombres
        Me.txtapellido.Text = obj_usuario.Apellidos
        Me.txtcedula.Text = obj_usuario.Usu_id

        'Se realiza una consulta a la tabla TblTipo_usuarios
        obj_tipo.Usu_Id = obj_usuario.Usu_id
        obj_tipo.consultar_usuario()

        'Se guarda la información extraída por la consulta la tabla
TblTipo_usuarios
        Me.DdlRango.Text = obj_tipo.Rango

        Me.txtcont.Text = obj_usuario.Pass
        Me.txtdireccion.Text = obj_usuario.Direccion
        Me.txtlog.Text = obj_usuario.Login
        Me.TxtCargo.Text = obj_usuario.Cargo
        Me.TxtCorreo.Text = obj_usuario.Correo

        'Se realiza una consulta a la tabla TblTelefono
        obj_telefono.Usu_Id = Me.txtcedula.Text
        obj_telefono.consultar_usuario()

```

```

        'Se guarda la información extraída por la consulta la tabla TblTelefono
        Me.Txtnumero.Text = obj_telefono.Numero

        'Desactiva el botón BtnBuscar y el cuadro de texto txtcedula
        Me.BtnBuscar.Enabled = False
        Me.txtcedula.Enabled = False

        Me.Modificar.Enabled = True
        Me.Eliminar.Enabled = True
        Me.Modificar.Visible = True
        Me.Eliminar.Visible = True

    Else
        Me.TxtMensaje.Text = "No se encontraron datos"
    End If
End Sub

Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
    'Re direcciona al menú del Administrador
    If Session("var_operario") = "1" Then
        Response.Redirect("Wfrm_orden.aspx")
    Else
        Response.Redirect("Wfrm_menu_administrador.aspx")
    End If
End Sub

Protected Sub BtnInserTecnico_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnInserTecnico.Click

    'Llamada de la Función
    llenar_campos()

    'Validación de la cedula
    Dim ced As String
    ced = obj_extra.cedula(Me.txtcedula.Text)

    If ced = "1" Then
        Me.TxtMensaje.Text = obj_tecnico.insertar()
        Me.BtnInserTecnico.Enabled = False
    Else
        Me.TxtMensaje.Text = "La Cedula es incorrecta. Profavor Verifique"
    End If

End Sub

Protected Sub BtnModftecnico_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnModftecnico.Click

    'Se realiza una consulta a la tabla TblPersonal_tecnico
    obj_tecnico.Tec_id = Me.txtcedula.Text
    obj_tecnico.consultar()

    'Llamada de la Función
    llenar_campos()

    'Actualiza la tabla TblPersonal_tecnico
    TxtMensaje.Text = obj_tecnico.actualizar()

End Sub

```

```
Protected Sub BtnLimTecnico_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnLimTecnico.Click
```

```
'Limpia los cuadros de texto
Me.TxtDetalle.Text = ""
Me.TxtJornada.Text = ""
Me.TxtTiempo.Text = ""
Me.TxtMensaje.Text = ""
Me.TxtFecha.Text = ""
```

```
Me.div_trabajador.Visible = False
'Se desactiva el botón BtnModftecnico
Me.BtnModftecnico.Enabled = False
'Se activa el botón BtnInserTecnico
Me.BtnInserTecnico.Enabled = True
```

```
End Sub
```

```
Sub MessageBox(ByVal message As String)
```

```
Dim sb As New StringBuilder
```

```
sb.Append("<script language='JavaScript'>")
sb.Append("alert('').Append(message).Append('');")
sb.Append("</script>")
```

```
ScriptManager.RegisterStartupScript(Me, GetType(Page), "alerta",
sb.ToString(), False)
```

```
End Sub
```

Código Wfrm_orden

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
```

```
If Not Me.IsPostBack Then
```

```
'Se obtiene el número de orden y la fecha del sistema
```

```
Me.Txtnume_orden.Text = obj_utilitarios.gen_codigo("Tblorden_trabajo",
"ord_id", 101)
```

```
Me.Txtfecha_ingreso.Text = CStr(System.DateTime.Now.Day) + "/" +
CStr(System.DateTime.Now.Month) + "/" + CStr(System.DateTime.Now.Year)
```

```
men = nwLatHidden.Value
```

```
Session("var_operario") = 0
```

```
End If
```

```
End Sub
```

```
Protected Sub Btnconsultar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Btnconsultar.Click
```

```
obj_usuario.Usu_id = CStr(Me.Txtcedula.Text)
```

```
'Se realiza una consulta a la tabla TblUsuarios
```

```
If obj_usuario.consultar = "Consulta Exitosa" Then
```

```
'Se guarda la información extraída por la consulta la tabla TblUsuarios
en los correspondientes cuadros de texto
```

```
Me.Txtnombres.Text = obj_usuario.Nombres
```

```
Me.Txtapellidos.Text = obj_usuario.Apellidos
```

```
Me.Txtdireccion.Text = obj_usuario.Direccion
```

```
Me.Txtcorreo.Text = obj_usuario.Correo
```

```
'Consulta el teléfono del Usuario
```

```

obj_telefono.Usu_Id = Me.Txtcedula.Text
obj_telefono.consultar_usuario()
Me.Txttelefono.Text = obj_telefono.Numero

'Se activa el Botón Btningresar_orden
Me.Btningresar_orden.Enabled = True
Me.Btningresar_orden.Visible = True

End If
End Sub

Protected Sub Btningresar_orden_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Btningresar_orden.Click

'ORDEN DE TRABAJO
'Se llenan los campos para la tabla TblOrden_trabajo
obj_orden.Ord_id = Me.Txtnume_orden.Text
obj_orden.Usu_id = Me.Txtcedula.Text
obj_orden.Articulo = Me.Txtarticulo.Text
obj_orden.Marca = Me.Txtmarca.Text
obj_orden.Modelo = Me.Txtmodelo.Text
obj_orden.Serie = Me.Txtserie.Text
obj_orden.Atencion = Me.Txttipo_atencion.Text
obj_orden.Ingreso = Me.Txtfecha_ingreso.Text
obj_orden.Entrega = Me.Txtfecha_entrega.Text
obj_orden.Danio = Me.Txtdanio.Text
obj_orden.Accesorios = Me.Txtaccesorios.Text
obj_orden.Observaciones = Me.Txtobservaciones.Text

'PAGOS
'Se llenan los campos para la tabla TblPagos
obj_pagos.Ord_id = Me.Txtnume_orden.Text
obj_pagos.Detalle = Me.Txtdetalle_pago.Text
obj_pagos.Tipo = Me.Ddltipo_pago.Text
obj_pagos.Abono = Me.Txtabono.Text
obj_pagos.Saldo = Me.Txtsaldo.Text
obj_pagos.Total = Me.Txttotal.Text

If obj_orden.insertar() = "Se ha guardo con Exito" And obj_pagos.insertar()
= "Se ha guardo con Exito" Then
'ESTADO
'Se llenan los campos para la tabla TblEstado
obj_estado.Ord_id = Me.Txtnume_orden.Text
Dim var_fecha As String
var_fecha = System.DateTime.Now.Date
obj_estado.Fecha = var_fecha
obj_estado.Detalle = "Orden Creada"
obj_estado.Observacion = "1"
obj_estado.insertar()
'obj_utilitarios.enviar_correo(Me.Txtcorreo.Text, "Orden de trabajo",
"Con un saludo le informacion que la orden de trabajo ha sido enviada con éxito")
'obj_utilitarios.enviar_correo(Session("correo"), "Orden de trabajo",
"El cliente : " + Me.Txtnombres.Text + " " + Me.Txtapellidos.Text + ", con cedula :
" + Me.Txtcedula.Text + ", ha echo una orden de trabajo en la fecha : " +
Me.Txtfecha_entrega.Text)

End If

'Re direcciona al menú
If Session("rango") = "Super" Then
Response.Redirect("Wfrm_menu_administrador.aspx")
Else
Response.Redirect("Wfrm_menu_operario.aspx")

```

```

End If

End Sub

Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
    'Re direcciona al menú
    If Session("rango") = "Super" Then
        Response.Redirect("Wfrm_menu_administrador.aspx")
    Else
        Response.Redirect("Wfrm_menu_operario.aspx")
    End If
End Sub
Sub MessageBox(ByVal message As String)

    Dim sb As New StringBuilder

    sb.Append("<script language='JavaScript'>")
    sb.Append("alert('').Append(message).Append('');")
    sb.Append("</script>")
    ScriptManager.RegisterStartupScript(Me, GetType(Page), "alerta",
sb.ToString(), False)

End Sub

Protected Sub Btnconsultar0_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Btnconsultar0.Click
    Session("var_operario") = "1"
    Response.Redirect("Wfrm_usuarios_ingreso.aspx")
End Sub

Protected Sub Calendar1_SelectionChanged(sender As Object, e As EventArgs)
Handles Calendar1.SelectionChanged
    Me.Txtfecha_entrega.Text = Calendar1.SelectedDate.Day.ToString + "/" +
Calendar1.SelectedDate.Month.ToString + "/" + Calendar1.SelectedDate.Year.ToString
End Sub

Protected Sub Btnconsultar1_Click(sender As Object, e As EventArgs) Handles
Btnconsultar1.Click
    Dim saldo, total, abono As Double

    abono = Me.Txtabono.Text
    total = Me.Txttotal.Text

    saldo = total - abono

    Me.Txtsaldo.Text = CStr(saldo)
End Sub

```

Código Wfrm_proforma

```

Protected Sub BtnEnviar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnEnviar.Click

    'Se guarda la información para ingresar a la tabla TblProforma
    obj_proforma.Usu_id = CStr(Session("codigo_usuario"))

```

```

        obj_proforma.Detalle = "Articulo: " + Me.TxtArticulos.Text + "---Daño:
" + Me.TxtDanio.Text + "---Marca: " + Me.TxtMarca.Text + "---Detalle: " +
Me.TxtDetalle.Text
        obj_proforma.Descuento = "0"
        obj_proforma.Observacion = "0"
        obj_proforma.Precio = "0"
        obj_proforma.Promocion = "0"
        obj_proforma.Tiempo = "0"

        'Se insertan los datos en la tabla TblProforma
        Me.TxtMensaje.Text = obj_proforma.insertar()
        Me.BtnEnviar.Enabled = False

    End Sub

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load

        If Not Me.IsPostBack Then
            'Se obtiene el código de usuario y se muestra una bienvenida al
mismo
            obj_usuario.Usu_id = Session("codigo_usuario")
            obj_usuario.consultar()

            Me.TxtInformacion.Text = "Bienvenido " + obj_usuario.Nombres + " "
+ obj_usuario.Apellidos
            men = nwLatHidden.Value
        End If
    End Sub

    Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
        'Re direcciona al menú del Cliente
        Response.Redirect("Wfrm_menu_cliente.aspx")
    End Sub

    Protected Sub BtnLimpiar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnLimpiar.Click
        'Se limpian los cuadros de texto
        Me.TxtArticulos.Text = ""
        Me.TxtDanio.Text = ""
        Me.TxtMarca.Text = ""
        Me.TxtDetalle.Text = ""
        'Se activa el Botón BtnEnviar
        Me.BtnEnviar.Enabled = True
    End Sub
    Sub MessageBox(ByVal message As String)

        Dim sb As New StringBuilder

        sb.Append("<script language='JavaScript'>")
        sb.Append("alert('').Append(message).Append('');")
        sb.Append("</script>")
        ScriptManager.RegisterStartupScript(Me, GetType(Page), "alerta",
sb.ToString(), False)

    End Sub

```

Código Wfrm_resumen_orden

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
```

```
    If Not Me.IsPostBack Then
        men = nwLatHidden.Value

        obj_usuario.Usu_id = Session("codigo_usuario")

        If obj_usuario.consultar = "Consulta Exitosa" Then
            Me.LbCedula.Text = Session("codigo_usuario")
            Me.LbNombres.Text = obj_usuario.Nombres
            Me.LbApellidos.Text = obj_usuario.Apellidos
            Me.LbDireccion.Text = obj_usuario.Direccion
            Me.LbCorreo.Text = obj_usuario.Correo
        End If
        obj_telefono.Usu_Id = Session("codigo_usuario")
        If obj_telefono.consultar_usuario = "Consulta Exitosa" Then
            Me.LbTelefono.Text = obj_telefono.Numero
        End If

        obj_orden.Ord_id = Session("codigo_orden")
        If obj_orden.consultar = "Consulta Exitosa" Then
            Me.Txtnume_orden.Text = obj_orden.Ord_id
            Me.LbArticulo.Text = obj_orden.Articulo
            Me.LbMarca.Text = obj_orden.Marca
            Me.LbModelo.Text = obj_orden.Modelo
            Me.LbSerie.Text = obj_orden.Serie
            Me.LbTipo_atencion.Text = obj_orden.Atencion
            Me.LbFecha_ingreso.Text = obj_orden.Ingreso
            Me.LbFecha_entrega.Text = obj_orden.Entrega
            Me.LbDanio.Text = obj_orden.Danio
            Me.LbAccesorios.Text = obj_orden.Accesorios
            Me.LbObservaciones.Text = obj_orden.Observaciones
        End If
        obj_pagos.Ord_id = Me.Txtnume_orden.Text
        If obj_pagos.consultarord = "Consulta Exitosa" Then
            Me.LbDetalle_pago.Text = obj_pagos.Detalle
            Me.LbTipo_pago.Text = obj_pagos.Tipo
            Me.LbAbono.Text = obj_pagos.Abono
            Me.LbSaldo.Text = obj_pagos.Saldo
            Me.LbTotal.Text = obj_pagos.Total
        End If
    End If
End Sub
```

```
Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
    Response.Redirect("Wfrm_seleccion_reporte.aspx")
End Sub
```

```
Sub MessageBox(ByVal message As String)

    Dim sb As New StringBuilder

    sb.Append("<script language='JavaScript'>")
    sb.Append("alert('').Append(message).Append('');")
    sb.Append("</script>")
    ScriptManager.RegisterStartupScript(Me, GetType(Page), "alerta",
sb.ToString(), False)
```

End Sub

Código Wfrm_revisar_estados

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
    If Not Me.IsPostBack Then
        men = nwLatHidden.Value
        Me.GVPersona.DataSource = Nothing
        Me.GVPersona.DataBind()
        Dim tabla As New DataTable 'Se crea la tabla en la cual se cargara
información referente al usuario

        'Se añaden las columnas para la tabla
        tabla.Columns.Add("Cedula")
        tabla.Columns.Add("Nombre")
        tabla.Columns.Add("Apellido")

        'Se realiza una consulta a la tabla TblUsuarios
        obj_extra.SQL = "select * from TblUsuarios where usu_id =" +
CStr(Session("codigo_usuario"))
        obj_extra.sentencia()

        'Se extrae la información de TblUsuarios y se añade a la tabla creada
en esta página
        If obj_extra.oregi.RecordCount > 0 Then
            While Not obj_extra.oregi.EOF
                Dim fila1 As DataRow = tabla.NewRow

                fila1("Cedula") = obj_extra.oregi.Fields(0).Value
                fila1("Nombre") = obj_extra.oregi.Fields(1).Value
                fila1("Apellido") = obj_extra.oregi.Fields(2).Value

                tabla.Rows.Add(fila1)
                obj_extra.oregi.MoveNext()
            End While
            Me.GVPersona.DataSource = tabla
            Me.GVPersona.DataBind()
        Else
            Me.TxtMensaje.Text = "No se encontraron datos"
        End If
        obj_extra.ocone.Close()
    End If
End Sub

Protected Sub GVProformas_SelectedIndexChanged(ByVal sender As Object, ByVal e
As System.EventArgs) Handles GVOrden.SelectedIndexChanged
    Me.GVEstados.DataSource = Nothing
    Me.GVEstados.DataBind()
    Me.GVPagos.DataSource = Nothing
    Me.GVPagos.DataBind()
    Dim tabla As New DataTable 'Se crea la tabla en la cual se cargara
información referente al Estado

    'Se añaden las columnas para la tabla
    tabla.Columns.Add("Fecha")
    tabla.Columns.Add("Detalle")
```

```

'Se realiza una consulta a la tabla TblEstado
obj_extra.SQL = "select * from TblEstado where ord_id=" +
Me.GVOrden.SelectedRow.Cells(1).Text
obj_extra.sentencia()

'Se extrae la información de TblEstado y se añade a la tabla creada en
esta página
If obj_extra.oregi.RecordCount > 0 Then
    While Not obj_extra.oregi.EOF
        Dim fila1 As DataRow = tabla.NewRow

        fila1("Fecha") = obj_extra.oregi.Fields(2).Value
        fila1("Detalle") = obj_extra.oregi.Fields(3).Value

        tabla.Rows.Add(fila1)
        obj_extra.oregi.MoveNext()
    End While
    Me.GVEstados.DataSource = tabla
    Me.GVEstados.DataBind()
End If

obj_extra.ocone.Close()

'Se añaden las columnas para la tabla
Dim tabla1 As New DataTable
tabla1.Columns.Add("Detalle")
tabla1.Columns.Add("Tipo")
tabla1.Columns.Add("Saldo")
tabla1.Columns.Add("Abono")
tabla1.Columns.Add("Total")

'Se realiza una consulta a la tabla TblPagos
obj_extra.SQL = "select * from TblPagos where ord_id=" +
Me.GVOrden.SelectedRow.Cells(1).Text
obj_extra.sentencia()

'Se extrae la información de TblPagos y se añade a la tabla creada en esta
página
If obj_extra.oregi.RecordCount > 0 Then
    While Not obj_extra.oregi.EOF
        Dim fila1 As DataRow = tabla1.NewRow

        fila1("Detalle") = obj_extra.oregi.Fields(2).Value
        fila1("Tipo") = obj_extra.oregi.Fields(3).Value
        fila1("Saldo") = obj_extra.oregi.Fields(5).Value
        fila1("Abono") = obj_extra.oregi.Fields(4).Value
        fila1("Total") = obj_extra.oregi.Fields(6).Value

        tabla1.Rows.Add(fila1)
        obj_extra.oregi.MoveNext()
    End While
    Me.GVPagos.DataSource = tabla1
    Me.GVPagos.DataBind()
End If
obj_extra.ocone.Close()

Me.TxtTitulo2.Visible = True
Me.TxtTitulo3.Visible = True
End Sub

```

```

Protected Sub GVPersona_SelectedIndexChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles GVPersona.SelectedIndexChanged
    Me.GVOrden.DataSource = Nothing
    Me.GVOrden.DataBind()
    Dim tabla As New DataTable 'Se crea la tabla en la cual se cargara
información referente al Estado

    'Se añaden las columnas para la tabla
    tabla.Columns.Add("Numero Orden Trabajo")
    tabla.Columns.Add("Articulo")
    tabla.Columns.Add("Daño")

    'Se realiza una consulta a la tabla TblOrden_trabajo
    obj_extra.SQL = "select * from TblOrden_trabajo where usu_id=" +
Me.GVPersona.SelectedRow.Cells(1).Text
    obj_extra.sentencia()

    'Se extrae la información de TblOrden_trabajo y se añade a la tabla creada
en esta página
    If obj_extra.oregi.RecordCount > 0 Then
        While Not obj_extra.oregi.EOF
            Dim fila1 As DataRow = tabla.NewRow

            fila1("Numero Orden Trabajo") = obj_extra.oregi.Fields(0).Value
            fila1("Articulo") = obj_extra.oregi.Fields(2).Value
            fila1("Daño") = obj_extra.oregi.Fields(9).Value

            tabla.Rows.Add(fila1)
            obj_extra.oregi.MoveNext()
        End While

        Me.GVOrden.DataSource = tabla
        Me.GVOrden.DataBind()
        Me.TxtTitulo1.Visible = True

    Else
        Me.TxtMensaje.Text = "No existen ordenes nuevas"
    End If
    obj_extra.ocone.Close()

End Sub

Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
    'Re direcciona al menú del Cliente
    Me.Response.Redirect("Wfrm_menu_cliente.aspx")
End Sub

Sub MessageBox(ByVal message As String)

    Dim sb As New StringBuilder

    sb.Append("<script language='JavaScript'>")
    sb.Append("alert('').Append(message).Append('');")
    sb.Append("</script>")
    ScriptManager.RegisterStartupScript(Me, GetType(Page), "alerta",
sb.ToString(), False)

End Sub

```

Código Wfrm_revisar_orden

```
Protected Sub BtnBuscar_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles BtnBuscar.Click
    Me.GVPersona.DataSource = Nothing
    Me.GVPersona.DataBind()
    Dim tabla As New DataTable 'Se crea la tabla en la cual se cargara
información referente al usuario

    'Se añaden las columnas para la tabla
    tabla.Columns.Add("Cedula")
    tabla.Columns.Add("Nombre")
    tabla.Columns.Add("Apellido")

    'Se realiza una consulta a la tabla TblUsuarios
    obj_extra.SQL = "select * from TblUsuarios a, TblTipo_usuarios b where
b.tp_rango='Cliente' and a.usu_id=b.usu_id and a.usu_id like '%" +
Me.txtcedula.Text + "%'"
    obj_extra.sentencia()

    'Se extrae la información de TblUsuarios y se añade a la tabla creada en
esta página
    If obj_extra.oregi.RecordCount > 0 Then
        While Not obj_extra.oregi.EOF
            Dim fila1 As DataRow = tabla.NewRow

            fila1("Cedula") = obj_extra.oregi.Fields(0).Value
            fila1("Nombre") = obj_extra.oregi.Fields(1).Value
            fila1("Apellido") = obj_extra.oregi.Fields(2).Value

            tabla.Rows.Add(fila1)
            obj_extra.oregi.MoveNext()
        End While
        Me.GVPersona.DataSource = tabla
        Me.GVPersona.DataBind()
        Me.txtcedula.Enabled = False
    Else
        Me.TxtMensaje.Text = "No se encontraron datos"
    End If
End Sub

Protected Sub GVPersona_SelectedIndexChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles GVPersona.SelectedIndexChanged
    Me.GVOrden.DataSource = Nothing
    Me.GVOrden.DataBind()
    Dim tabla As New DataTable 'Se crea la tabla en la cual se cargara
información referente a la orden de trabajo

    'Se añaden las columnas para la tabla
    tabla.Columns.Add("Numero Orden")
    tabla.Columns.Add("Artículo")
    tabla.Columns.Add("Fecha de Entrega")

    'Se realiza una consulta a la tabla TblOrden_trabajo
    obj_extra.SQL = "select * from TblOrden_trabajo where usu_id=" +
Me.GVPersona.SelectedRow.Cells(1).Text
    obj_extra.sentencia()

    'Se extrae la información de TblOrden_trabajo y se añade a la tabla creada
en esta página
    If obj_extra.oregi.RecordCount > 0 Then
```

```

While Not obj_extra.oregi.EOF
    Dim fila1 As DataRow = tabla.NewRow

    fila1("Numero Orden") = obj_extra.oregi.Fields(0).Value
    fila1("Artículo") = obj_extra.oregi.Fields(2).Value
    fila1("Fecha de Entrega") = obj_extra.oregi.Fields(8).Value

    tabla.Rows.Add(fila1)
    obj_extra.oregi.MoveNext()
End While
Me.GVOrden.DataSource = tabla
Me.GVOrden.DataBind()
Else
    Me.TxtMensaje.Text = "No existen Ordenes nuevas"
End If
End Sub

Protected Sub GVProformas_SelectedIndexChanged(ByVal sender As Object, ByVal e
As System.EventArgs) Handles GVOrden.SelectedIndexChanged
    obj_pagos.Ord_id = Me.GVOrden.SelectedRow.Cells(1).Text
    'Se realiza una consulta a la tabla TblPagos
    If obj_pagos.consultarord = "Consulta Exitosa" Then
        obj_orden.Ord_id = Me.GVOrden.SelectedRow.Cells(1).Text
        obj_orden.consultar()
        'Se muestra en el label TxtId_orden el número de orden de trabajo
        Me.TxtId_orden.Text = "El Numero de Orden de Trabajo es : " +
CStr(Me.obj_orden.Ord_id)
        'Se muestra en el label TxtDetalle_orden el articulo
        Me.TxtDetalle_orden.Text = "Articulo : " + obj_orden.Articulo + ", Daño
: " + Me.obj_orden.Danio + ", Observaciones : " + obj_orden.Observaciones
        'Se guarda la información extraída por la consulta la tabla TblPagos en
los correspondientes cuadros de texto
        Me.TxtAbono.Text = obj_pagos.Abono
        Me.TxtSaldo.Text = obj_pagos.Saldo
        Me.TxtTotal.Text = obj_pagos.Total
        Me.DdlTipo_pago.Text = obj_pagos.Tipo
        Me.TxtDetalle.Text = obj_pagos.Detalle

        Me.DivIngreso.Visible = True
    Else
        Me.TxtMensaje.Text = "No existen Ordenes"
    End If

End Sub

Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load

    'Se realiza una consulta a la tabla TblUsuarios
    If Not Me.IsPostBack Then
        men = nwLatHidden.Value
        obj_extra.SQL = "Select * From TblUsuarios a, TblTipo_usuarios b where
a.usu_id = b.usu_id and b.tp_rango = 'Tecnico'"
        obj_extra.sentencia()

        'Se agrega los técnicos al DropDownList
        While Not obj_extra.oregi.EOF

```

```

        Me.DdlTecnico1.Items.Add(obj_extra.oregi.Fields(1).Value + " " +
obj_extra.oregi.Fields(2).Value)
        Me.DdlTecnico2.Items.Add(obj_extra.oregi.Fields(1).Value + " " +
obj_extra.oregi.Fields(2).Value)
        Me.DdlTecnico3.Items.Add(obj_extra.oregi.Fields(1).Value + " " +
obj_extra.oregi.Fields(2).Value)
        obj_extra.oregi.MoveNext()
    End While
    obj_extra.ocone.Close()

End If
End Sub

Protected Sub BtnEnviar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnEnviar.Click

    If Not DdlTecnico1.Text = "Ninguno" Then
        'Se realiza una consulta a la tabla TblUsuarios
        obj_extra.SQL = "Select * From TblUsuarios a where a.usu_nombres + ' '
+ a.usu_apellidos = '" + DdlTecnico1.Text + "'"
        obj_extra.sentencia()
        cod_tec1 = obj_extra.oregi.Fields(0).Value
        obj_extra.ocone.Close()
        'Se guarda la información para ingresar a la tabla TblOrden_tecnico
        obj_orden_tec.Tec_Id = cod_tec1
        obj_orden_tec.Ord_Id = Me.GVOrden.SelectedRow.Cells(1).Text
        obj_orden_tec.Detalle = "Principal"

        Dim errores As String
        'Se insertan los datos en la tabla TblOrden_tecnico
        errores = obj_orden_tec.insertar()

        If errores = "Se ha guardado con Exito" Then
            Me.TxtMensaje.Text = "Se ha enviado con exito al técnico " +
DdlTecnico1.Text
            ingreso_estado()
        Else
            Me.TxtMensaje.Text = "error al enviar al técnico " + errores
        End If
    End If

    If Not DdlTecnico2.Text = "Ninguno" Then
        'Se realiza una consulta a la tabla TblUsuarios
        obj_extra.SQL = "Select * From TblUsuarios a where a.usu_nombres + ' '
+ a.usu_apellidos = '" + DdlTecnico2.Text + "'"
        obj_extra.sentencia()
        cod_tec2 = obj_extra.oregi.Fields(0).Value
        obj_extra.ocone.Close()

        'Se guarda la información para ingresar a la tabla TblOrden_tecnico
        obj_orden_tec.Tec_Id = cod_tec2
        obj_orden_tec.Ord_Id = Me.GVOrden.SelectedRow.Cells(1).Text
        obj_orden_tec.Detalle = "Técnico 2"

        Dim errores As String
        'Se insertan los datos en la tabla TblOrden_tecnico
        errores = obj_orden_tec.insertar()

        If errores = "Se ha guardado con Exito" Then
            Me.TxtMensaje.Text = "Se ha enviado con exito al técnico " +
DdlTecnico2.Text
            ingreso_estado()
        Else

```

```

        Me.TxtMensaje.Text = "error al enviar al técnico " + errores
    End If
End If

If Not DdlTecnico3.Text = "Ninguno" Then
    'Se realiza una consulta a la tabla TblUsuarios
    obj_extra.SQL = "Select * From TblUsuarios a where a.usu_nombres + ' '
+ a.usu_apellidos = '" + DdlTecnico3.Text + ""
    obj_extra.sentencia()
    cod_tec3 = obj_extra.oregi.Fields(0).Value
    obj_extra.ocone.Close()

    'Se guarda la información para ingresar a la tabla TblOrden_tecnico
    obj_orden_tec.Tec_Id = cod_tec3
    obj_orden_tec.Ord_Id = Me.GVOrden.SelectedRow.Cells(1).Text
    obj_orden_tec.Detalle = "Auxiliar"
    Dim errores As String

    'Se insertan los datos en la tabla TblOrden_tecnico
    errores = obj_orden_tec.insertar()
    If errores = "Se ha guardo con Exito" Then

        Me.TxtMensaje.Text = "Se ha enviado con exito al técnico " +
DdlTecnico3.Text
        ingreso_estado()
    Else
        Me.TxtMensaje.Text = "error al enviar al técnico " + errores
    End If
End If
End Sub

Function ingreso_estado() As String
    'Se guarda la información para ingresar en la tabla TblEstados
    obj_estado.Ord_id = Me.GVOrden.SelectedRow.Cells(1).Text
    Dim var_fecha As String
    var_fecha = System.DateTime.Now.Date
    obj_estado.Fecha = var_fecha
    obj_estado.Detalle = "Orden Procesando"
    obj_estado.Observacion = "2"
    'Se desactiva el Botón BtnEnviar
    Me.BtnEnviar.Enabled = False
    'Se insertan los datos en la tabla TblEstados
    Return obj_estado.insertar()
End Function

Protected Sub Btn_actualizar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Btn_actualizar.Click
    'Se guarda la información de los cuadros de texto en los objetos para
actualizar la tabla TblPagos
    obj_pagos.Ord_id = Me.GVOrden.SelectedRow.Cells(1).Text
    obj_pagos.Abono = Me.TxtAbono.Text
    obj_pagos.Saldo = Me.TxtSaldo.Text
    obj_pagos.Total = Me.TxtTotal.Text
    obj_pagos.Tipo = Me.DdlTipo_pago.Text
    obj_pagos.Detalle = Me.TxtDetalle.Text

    'Se insertan los datos en la tabla TblPagos
    If obj_pagos.insertar = "Se ha guardo con Exito" Then

        TxtMensaje.Text = "Se ha guardo con Exito"

    Else
        TxtMensaje.Text = "No se pudo actualizar"
    End If
End Sub

```

```

        End If
    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnLimpiar1.Click
        'Se desactiva el Botón BtnEnviar
        Me.BtnEnviar.Enabled = True
        'Limpia DataGridView
        Me.DivIngreso.Visible = False
        Me.TxtMensaje.Text = ""
        Me.GVOrden.DataSource = Nothing
        Me.GVOrden.DataBind()
    End Sub

    Protected Sub BtnLimpiar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnLimpiar.Click
        Me.TxtMensaje.Text = ""
        'Se activa el Botón BtnEnviar
        Me.BtnEnviar.Enabled = True
        Me.txtcedula.Text = ""
        'Limpia DataGridView
        Me.DivIngreso.Visible = False
        Me.GVOrden.DataSource = Nothing
        Me.GVOrden.DataBind()
        Me.GVPersona.DataSource = Nothing
        Me.GVPersona.DataBind()
        'Se activa el cuadro de texto txtcedula
        Me.txtcedula.Enabled = True
    End Sub

    Protected Sub TxtAbono_TextChanged(ByVal sender As Object, ByVal e As
System.EventArgs) Handles TxtAbono.TextChanged
        'Realiza una resta para sacar el Total
        Me.TxtSaldo.Text = Me.TxtTotal.Text - Me.TxtSaldo.Text
    End Sub

    Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
        'Re direcciona al menú
        If Session("rango") = "Super" Then
            Response.Redirect("Wfrm_menu_administrador.aspx")
        Else
            Response.Redirect("Wfrm_menu_empresa.aspx")
        End If
    End Sub

    Sub MessageBox(ByVal message As String)

        Dim sb As New StringBuilder

        sb.Append("<script language='JavaScript'>")
        sb.Append("alert('').Append(message).Append('');")
        sb.Append("</script>")
        ScriptManager.RegisterStartupScript(Me, GetType(Page), "alerta",
sb.ToString(), False)

    End Sub

    Protected Sub Btnconsultar1_Click(sender As Object, e As EventArgs) Handles
Btnconsultar1.Click
        Dim saldo, total, abono As Double

        abono = Me.TxtAbono.Text
        total = Me.TxtTotal.Text

```

```

        saldo = total - abono

        Me.TxtSaldo.Text = CStr(saldo)
    End Sub

```

Código Cls_Utilitarios

```

Function gen_codigo(ByVal tabla As String, ByVal campo As String, ByVal max_cod As Integer) As Integer
    conexion()

```

```

        _sql = "select " + campo + " from " + tabla + " order by " + campo
        oregi = ocone.Execute(_sql)
        Dim var As Integer

        If oregi.EOF Then
            ocone.Close()
            Return max_cod
        Else
            oregi.MoveLast()
            var = oregi.Fields(0).Value + 1
            ocone.Close()
            Return var
        End If

```

```

    End Function

```

```

Function bus_codigo(ByVal tabla As String, ByVal campo As String, ByVal valor As String) As Integer
    conexion()

```

```

        _sql = "select * from " + tabla + " where " + campo + "=" + valor + ""
        oregi = ocone.Execute(_sql)

        Return oregi.Fields(0).Value

```

```

    End Function

```

```

Sub conexion()
    ocone.CursorLocation = ADODB.CursorLocationEnum.adUseClient
    ocone.ConnectionString = cadena
    ocone.Open()
End Sub

```

```

Sub sentencia()
    conexion()
    oregi = ocone.Execute(_sql)
End Sub

```

```

Public Function encriptacion(ByVal cadena As String) As String

```

```

    Dim longitud, cont, valor As Integer
    Dim res As String

    longitud = Len(cadena)
    res = ""

```

```

    For cont = 1 To longitud
        valor = Asc(Mid(cadena, cont, 1))
        valor += 10
        If valor = 39 Then
            valor = 172
        End If
        res += Chr(valor)
    Next

    Return res

End Function

Public Function desencripcion(ByVal cadena As String) As String

    Dim longitud, cont, valor As Integer
    Dim res As String

    longitud = Len(cadena)
    res = ""

    For cont = 1 To longitud
        valor = Asc(Mid(cadena, cont, 1))
        valor -= 10
        If valor = 162 Then
            valor = 49
        End If
        res += Chr(valor)
    Next

    Return res
End Function

Public Function validacion(ByVal login As String, ByVal pass As String) As
String
    Try
        conexion()
        _sql = "Select count(1) as existe from Tblusuarios where usu_login='" +
login + "'"
        oregi = ocone.Execute(_sql)

        If Not oregi.Fields("existe").Value = "0" Then
            _sql = "Select * from Tblusuarios where usu_pass='" + pass + "' and
usu_login='" + login + "'"
            oregi = ocone.Execute(_sql)
            If Not oregi.RecordCount = "0" Then
                cod_per = oregi.Fields(0).Value
                Return ("Usuario Valido")
            Else
                Return ("Contraseña incorrecta")
            End If
        Else
            Return ("Usuario no registrado")
        End If
        ocone.Close()

    Catch ex As Exception
        Return ex.Message
    End Try

End Function

```

```
Public Function cedula(ByVal cadena As String) As String
```

```
    Dim vec(10), cont, sum, dec As Integer
```

```
    sum = 0
```

```
    For cont = 1 To 10
```

```
        vec(cont) = Val(Mid(cadena, cont, 1))
```

```
    Next
```

```
    For cont = 2 To 8 Step 2
```

```
        sum += vec(cont)
```

```
    Next
```

```
    For cont = 1 To 9 Step 2
```

```
        If vec(cont) * 2 >= 10 Then
```

```
            sum += ((vec(cont) * 2) - 9)
```

```
        Else
```

```
            sum += (vec(cont) * 2)
```

```
        End If
```

```
    Next
```

```
    dec = sum Mod 10
```

```
    If dec <> 0 Then
```

```
        dec = 10 - dec
```

```
    End If
```

```
    If dec = vec(10) Then
```

```
        Return "1"
```

```
    Else
```

```
        Return "0"
```

```
    End If
```

```
End Function
```

```
Public Function procedimiento_comilla(ByVal cadena As String) As String
```

```
    Dim lon, cont As Integer
```

```
    Dim res As String = ""
```

```
    lon = Len(cadena)
```

```
    For cont = 1 To lon
```

```
        If Mid(cadena, cont, 1) = "" Then
```

```
            res += "'+char(39)+'"
```

```
        Else
```

```
            res += Mid(cadena, cont, 1)
```

```
        End If
```

```
    Next
```

```
    Return res
```

```
End Function
```

```
Public Function enviar_correo(ByVal correo As String, ByVal asunto As String,  
ByVal cuerpo As String) As String
```

```
    Dim _Message As New System.Net.Mail.MailMessage()
```

```
    Dim _SMTP As New System.Net.Mail.SmtpClient
```

```
    'CONFIGURACIÓN DEL STMP
```

```

        _SMTP.Credentials = New
System.Net.NetworkCredential("registersite92@gmail.com", "4dmin1234")
        _SMTP.Host = "smtp.gmail.com"
        _SMTP.Port = 587
        _SMTP.EnableSsl = True

        ' CONFIGURACION DEL MENSAJE
        _Message.[To].Add(correo) 'Cuenta de Correo al que se le quiere enviar el
e-mail
        _Message.From = New System.Net.Mail.MailAddress("registersite92@gmail.com",
"Register Web", System.Text.Encoding.UTF8) 'Quien lo envía
        _Message.Subject = asunto 'Sujeto del e-mail
        _Message.SubjectEncoding = System.Text.Encoding.UTF8 'Codificacion
        _Message.Body = cuerpo 'contenido del mail
        _Message.BodyEncoding = System.Text.Encoding.UTF8
        _Message.Priority = System.Net.Mail.MailPriority.Normal
        _Message.IsBodyHtml = False
        'ENVIO

        Try
            _SMTP.Send(_Message)
            Return "Mensaje enviado correctamene"
        Catch ex As System.Net.Mail.SmtpException
            Return ex.ToString + "Error!"
        End Try

    End Function

```

#End Region

Código Cls_Usuarios

```

Function insertar() As String
    Try
        coneccion()

        _sql = "insert into
TblUsuarios(usu_id,usu_nombres,usu_apellidos,usu_direccion,usu_cargo,usu_correo,usu
_login,usu_pass) "
        _sql += "values(" + CStr(_usu_id) + "," + _nombres + "," + _
obj_extra.procedimiento_comilla(_apellidos) + "," +
        _sql += _direccion + "," + _cargo + "," + _correo + "," + _login
+ "," + _pass + ")"

        oregi = ocone.Execute(_sql)
        ocone.Close()

        Return "Se ha guardo con Exito"
    Catch ex As Exception
        Return ex.Message
    End Try
End Function

Function actualizar() As String
    Try
        coneccion()

```

```

        _sql = "update TblUsuarios set usu_nombres='" + _nombres +
        "',usu_apellidos='" + obj_extra.procedimiento_comilla(_apellidos)
        _sql += "',usu_pass='" + _pass + "',usu_cargo='" + _cargo +
        "',usu_direccion='" + _direccion
        _sql += "',usu_login='" + _login + "',usu_correo='" + _correo + "'
where usu_id=" + CStr(_usu_id)

        oregi = ocone.Execute(_sql)
        ocone.Close()

        Return "Se ha guardo con Exito"
    Catch ex As Exception
        Return ex.Message
    End Try
End Function

Function eliminar() As String
    Try
        coneccion()

        _sql = "delete from TblUsuarios where usu_id=" + CStr(_usu_id)
        oregi = ocone.Execute(_sql)
        ocone.Close()

        Return "Se ha Eliminado con Exito"
    Catch ex As Exception
        Return ex.Message
    End Try
End Function

Function consultar() As String
    Try
        coneccion()

        _sql = "select * from TblUsuarios where usu_id=" + CStr(_usu_id)
        oregi = ocone.Execute(_sql)

        _cargo = oregi.Fields("usu_cargo").Value
        _nombres = oregi.Fields("usu_nombres").Value
        _apellidos = oregi.Fields("usu_apellidos").Value
        _pass = oregi.Fields("usu_pass").Value
        _direccion = oregi.Fields("usu_direccion").Value
        _correo = oregi.Fields("usu_correo").Value
        _login = oregi.Fields("usu_login").Value

        ocone.Close()

        Return "Consulta Exitosa"
    Catch ex As Exception
        Return ex.Message
    End Try
End Function

Function consultartec() As String
    Try
        coneccion()

        _sql = "select * from TblUsuarios where usu_cargo = 'Técnico'"
        oregi = ocone.Execute(_sql)

        _cargo = oregi.Fields("usu_cargo").Value
        _nombres = oregi.Fields("usu_nombres").Value

```

```

        _apellidos = oregi.Fields("usu_apellidos").Value
        _pass = oregi.Fields("usu_pass").Value
        _direccion = oregi.Fields("usu_direccion").Value
        _correo = oregi.Fields("usu_correo").Value
        _login = oregi.Fields("usu_login").Value

        ocone.Close()

        Return "Consulta Exitosa"
    Catch ex As Exception
        Return ex.Message
    End Try
End Function

Function activar() As String
    Try
        conexion()

        _sql = "update TblUsuario set usu_estado='activo' where usu_id=" +
CStr(_usu_id)

        oregi = ocone.Execute(_sql)
        ocone.Close()

        Return "Se ha Activado con Exito"
    Catch ex As Exception
        Return ex.Message
    End Try
End Function

#End Region

```