

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
ESCUELA DE SISTEMAS



DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL
DE ALQUILER DE MENAJE, INVENTARIOS Y FACTURACIÓN DE LA
EMPRESA “AZAHARES”

RUEDA OJEDA FERNANDO DAVID

TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO
EN SISTEMAS

Quito, 14 de septiembre, 2012

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA

HOJA DE APROBACIÓN DE DISERTACIÓN DE GRADO

DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL
DE ALQUILER DE MENAJE, INVENTARIOS Y FACTURACIÓN DE LA
EMPRESA “AZAHARES”

David Rueda O.
ESTUDIANTE QUE REALIZA LA DISERTACIÓN DE GRADO

Ingeniero Alfredo Calderón
DIRECTOR DE DISERTACIÓN DE GRADO

Ingeniero Oswaldo Luna
MIEMBRO DEL COMITÉ DE DISERTACIÓN DE GRADO

Ingeniero Eddy Sánchez
MIEMBRO DEL COMITÉ DE DISERTACIÓN DE GRADO

DEDICATORIA

A mis amados mamá y papá, quienes me demostraron lo grande que puede llegar a ser una persona por sí misma...

A mi segunda madre, Doña Carmen Torres, su mano siempre estuvo para sacarme de los abismos de la adversidad...

A mi adorada abuelita Conchita, tarde tarde, pero con esto espero pagarle las canas verdes abuelita linda... gracias.

AGRADECIMIENTOS

Estaremos eternamente agradecidos a:

- Ingeniero Alfredo Calderón
- David Koon
- Ingeniero Oswaldo Luna
- Ingeniero Xavier Rueda
- Ingeniero Eddy Sánchez

Por su gran contribución y apoyo al presente trabajo...

RESUMEN

La automatización es definitivamente una necesidad de las empresas que se encuentran en desarrollo, no sólo porque representa una ventaja competitiva, sino también por sus grandes beneficios en reducción de costos, además de tiempos de atención y respuesta con los clientes. Esta es una realidad que no se puede ocultar, y mucho menos con el apogeo de sistemas cada vez más eficientes, que cumplen papeles muy importantes en el comercio en general, tan importantes que simplemente no pueden ser reemplazados por el trabajo humano simplemente.

Es debido a esta situación, que la empresa “AZAHARES”, dedicada al catering desde hace más de 25 años; ha decidido a través de su Gerente General Xavier Rueda; enfrentar los nuevos desafíos comerciales mediante el desarrollo e implantación de un **sistema de control de alquiler de menaje, inventarios y facturación**; el mismo que fue propuesto como una solución que beneficia a ambas partes, tanto como al propietario como para el estudiante que realizará el desarrollo, y que utilizará el mismo como **DISERTACIÓN DE GRADO** para lograr obtener el título de Ingeniero en Sistemas.

El presente trabajo consta de un estudio del Marco Legal y Teórico concerniente; luego del cual se describen las posibles herramientas a utilizar con una evaluación de las mismas para una adecuada selección.

Finalmente la implementación práctica del Sistema en sí en basado en los puntos descritos anteriormente.

TABLA DE CONTENIDOS

GRUPO I: Marco Legal

1. Legislación	7
1.1. Definición legal de Compañía.....	8
1.1.1. Definición de la Ley de Compañías del Ecuador.....	8
1.1.2. Otras definiciones	10
1.2. Tipos de PYMES.....	12
1.3. Facturación.....	13
1.3.1. Aspectos importantes de los comprobantes de venta.....	13
1.3.2. Tipos de comprobantes de venta y de retención autorizados.....	16
1.3.3. Emisión de los comprobantes de venta.....	17
1.3.3.1. Facturas.....	18
1.3.3.2. Notas o boletas de venta.....	18
1.3.3.3. Tickets o vales emitidos por máquinas registradoras.....	18
1.3.4. Información pre-impresa e información no impresa.....	19
1.3.4.1. Información pre-impresa (Todos los casos)	19
1.3.4.2. Información no pre-impresa.....	22
1.3.4.2.1. Información no impresa para el caso de facturas.....	22
1.3.4.2.2. Información no impresa para el caso de notas o boletas de venta.....	23
1.3.4.2.3. Información sobre la transacción en liquidaciones de compras de bienes o prestación de servicios	24
1.3.4.2.4. Información en tiquetes o vales emitidos por máquinas registradoras.....	25
1.4. Infracciones al reglamento de facturación.....	26
1.5. Facultad de la Superintendencia de Compañías.....	27
1.6. Marcas, patentes y Ley de Propiedad Intelectual.....	28

GRUPO II: Marco Teórico

Conceptos Generales: Negocio.....	30
2. Empresas	31
2.1. Definición.....	31
2.2. Características.....	32
2.2.1. Elementos.....	32
2.2.1.1. Factores activos.....	32
2.2.1.2. Factores pasivos.....	33
2.2.1.3. Organización.....	33
2.3. Buen gobierno empresarial.....	33
2.4. Clasificación.....	34
2.4.1. Por sectores económicos.....	35
2.4.2. Por tamaño.....	35
2.4.3. Por el origen del capital.....	36
2.4.4. Por la explotación y conformación de su capital.....	37
2.4.5. Por el pago de impuestos.....	37
2.4.6. Por la función social.....	38
2.5. Clasificación de la empresa Azahares según los puntos documentados.....	39

3. Metodologías de desarrollo	40
3.1. Definición.....	40
3.2. Tipos.....	40
3.2.1. Metodologías estructuradas.....	41
3.2.1.1. Modelo en cascada.....	43
3.2.1.2. Modelo en espiral.....	45
3.2.1.3. Modelo en V.....	47
3.2.1.4. Modelo interactivo.....	48
3.2.1.5. Modelo incremental.....	49
3.2.2. Metodologías Ágiles.....	50
3.2.2.1. Programación extrema.....	54
3.2.2.2. Scrum.....	56
3.2.2.3. Modelamiento ágil.....	58
3.2.2.4. Método de desarrollo de sistemas dinámicos.....	60
3.2.2.5. Entorno de trabajo para soluciones Microsoft.....	62
3.3. Evaluación y elección de la metodología.....	64
4. Herramientas	65
4.1. Base de Datos.....	65
4.1.1. Sistema Gestores de Bases de Datos (SGBD) o Motores de Bases de Datos.....	65
4.1.1.1. Sistemas Gestores de Bases de Datos utilizados en la actualidad.....	67
Sistemas Libres.....	68
4.1.1.1.1. PostgreSQL.....	68
4.1.1.1.2. MySQL.....	69
4.1.1.1.3. FireBird.....	70
Sistemas Comerciales.....	72
4.1.1.1.4. Microsoft Access.....	72
4.1.1.1.5. Microsoft SQL Server.....	74
4.1.1.1.6. MySQL.....	75
4.1.1.1.7. Oracle.....	76
4.1.1.1.8. IBM DB2 Universal Database.....	77
4.1.2. Análisis y Selección del motor de bases de datos.....	81
4.2. Herramientas de Desarrollo.....	82
Lenguajes de programación.....	82
4.2.1. C++.....	83
4.2.2. C#.....	84
4.2.3. Delphi.....	85
4.2.4. Java.....	86
4.2.5. JavaScript.....	87
4.2.6. PHP.....	88
4.2.7. PowerBuilder.....	90
4.2.8. Visual Basic.....	92
4.2.9. Visual Basic.NET.....	94
4.2.10. Visual Fox Pro.....	95
4.3. Análisis y Selección de la herramienta.....	97

5. Desarrollo	98
5.1. Análisis y descripción de la empresa.....	98
5.1.1.Diagrama SIPOC de la empresa antes del sistema.....	101
5.1.2.Diagrama SIPOC de la empresa antes del sistema.....	103
6. Conclusiones	104
7. Recomendaciones	105
8. Medios de consulta / Bibliografía	106

GRUPO I
MARCO LEGAL

CAPÍTULO 1: LEGISLACIÓN

Antes de tener una definición apegada a la Ley, se debe tener en cuenta que una empresa puede dedicarse a actividades diversas, entre las cuales se pueden mencionar:

- Agricultura, caza, silvicultura y pesca.
- Explotación de minas y canteras.
- Industrias manufactureras.
- Construcción
- Comercio al por mayor y al por menor, restaurantes y hoteles
- Transporte, almacenamiento, y comunicaciones
- Establecimientos financieros, seguros, bienes inmuebles y servicios a empresas
- Servicios comunales, sociales y personales¹

Sin embargo de ello, la mayor parte de información específica disponible se ocupa de las "Industrias Manufactureras", por lo que se debe tomar en cuenta las leyes y reglamentos de: Compañías, Tributarios (incluidos los del RUC, Facturación), Propiedad Intelectual, Código del Trabajo, etc., es decir como cualquier empresa, las compañías se rigen por la Constitución del País y todas las Leyes y Reglamentos existentes, **“con la salvedad de que las Pequeñas y Medianas Empresas independientemente del sector económico en que se desarrollen, deben regirse por parámetros específicos como límites en el monto de Capital o Patrimonio, número de empleados, afiliación a la Cámara de la producción que este relacionada con su actividad, porque de ello se derivan beneficios específicos para facilitar el desarrollo de su actividad”**.²

¹⁻² <http://www.pymes-ecuador.com/nuevo-sitio/inicio.htm>

A continuación se intentará en base a lo mencionado anteriormente extraer partes fundamentales de las leyes y reglamentos que de alguna manera rigen en el País para las empresas consideradas como Pequeñas y Medianas.

1.1 Definición legal de Compañía

1.1.1 Definición de La Ley de Compañías del Ecuador

De acuerdo a la **Ley de Compañías** en el País, en la mayoría de casos pueden hacerse contratos para establecer compañías (empresas), esto es lo que dicha ley contiene en su texto, en sus partes más relevantes para nosotros:

“LEY DE COMPAÑIAS DEL ECUADOR

H. CONGRESO NACIONAL

SECCION I

Disposiciones Generales

Art.1.-

Contrato de compañía es aquél por el cual dos o más personas unen sus capitales o industrias, para emprender en operaciones mercantiles y participar de sus utilidades. Este contrato se rige por las disposiciones de esta Ley, por las del Código de Comercio, por los convenios de las partes y por las disposiciones del Código Civil.¹

¹ LEY DE COMPAÑIAS, Codificación 000, Registro Oficial 312 de 5 de Noviembre de 1999

Art.2.-

Hay cinco especies de compañías de comercio, a saber:

- La compañía en nombre colectivo;
- La compañía en comandita simple y dividida por acciones;
- La compañía de responsabilidad limitada;
- La compañía anónima; y,
- La compañía de economía mixta.

Estas cinco especies de compañías constituyen personas jurídicas. La ley reconoce, además, la compañía accidental o cuentas en participación.

A excepción de las compañías que se establecen legalmente, existen las "Compañías Accidentales, Asociaciones o Cuentas en Participación", de las cuales la Ley de Compañías, en su SECCIÓN XV, DE LA ASOCIACIÓN O CUENTAS EN PARTICIPACIÓN Y DE LA COMPAÑÍA HOLDING O TENEDORA DE ACCIONES, Inciso 1, Art. 423.nos dice:

“La asociación en participación es aquella en la que un comerciante da a una o más personas participación en las utilidades o pérdidas de una o más operaciones o de todo su comercio.

Puede también tener lugar en operaciones mercantiles hechas por no comerciantes.”¹

La asociación o compañía accidental se rige por las convenciones de las partes y está exenta de las formalidades establecidas para las compañías con personalidad jurídica.

¹ LEY DE COMPAÑÍAS, Codificación 000, Registro Oficial 312 de 5 de Noviembre de 1999

Adicionalmente la Ley de Compañías nos dice:

“**Art.428.-** Estas asociaciones están exentas de las formalidades establecidas para las compañías. A falta de contrato por escritura pública, se pueden probar por los demás medios admitidos por la ley mercantil. Pero la prueba testimonial no es admisible cuando se trate de un negocio cuyo valor pase de doscientos mil sucres (ocho dólares), si no hay principio de prueba por escrito.”¹

Sin embargo de lo mencionado anteriormente sobre las "Cuentas en Participación", la compañía debe tener su Registro Único de Contribuyentes (RUC) y cumplir con todo lo que establecen las Leyes de Régimen Tributario, para los demás tipos de compañías.

1.1.2 Otras Definiciones.-

Se debe añadir que la **Ley de Fomento de la Pequeña Industria** define lo siguiente:

- **Empresa:** a la unidad de producción, sea persona natural o jurídica, que se dedique a actividades industriales.
- **Empresa nueva:** aquella que no hubiere iniciado su producción efectiva a la fecha de presentación de la solicitud de clasificación.
- **Empresa existente:** aquella que hubiere iniciado su producción efectiva antes de la fecha de presentación de la solicitud de clasificación.

¹ LEY DE COMPAÑÍAS, Codificación 000, Registro Oficial 312 de 5 de Noviembre de 1999

Y en la información consignada en el **Ministerio de Comercio Exterior, Industrialización y Pesca (MICIP)**, se encuentran las siguientes definiciones:

- **Pequeña Industria:** Se la define como aquella empresa con predominio de la operación de la maquinaria sobre la manual, que se dedique a actividades de transformación, inclusive la forma, de materia prima o de productos semielaborados, en artículos finales o intermedios y siempre su activo fijo, excluyéndose terrenos y edificios, no sea mayor del valor que el Comité Interministerial de Fomento de la Pequeña Industria, fije anualmente, es decir no será mayor de US \$ 112.000
- **Pequeña Empresa:** Para los asesores de la Cámara de la Pequeña Industria de Guayas (CAPIG) Pequeña Empresa es una unidad de producción que tiene de 5 a 40 y un máximo de 50 empleados, su capital no tiene piso pero su patrimonio tiene un techo de \$ 150.000.
- **Mediana Empresa:** Para que a una empresa se le considere como Mediana Empresa, se tiene en cuenta el número de empleados entre 50 y 100, según el criterio de los asesores de la CAPIG.
- **Microempresa:** El MICIP, también considera como Microempresa a una unidad económica productiva y puede ser de producción, comercio o servicios, cuyas características básicas son:
 - El manejo operacional y administrativo lo realiza generalmente una persona.
 - Sus activos no sobrepasan los US \$ 20.000. El número de trabajadores es máximo de 10 personas incluido el propietario.

- **Artesanía:** La Artesanía es la actividad que se ejerce en forma individual o colectiva en la transformación de materia prima destinada a la producción de bienes, servicios o artística con predominio de la labor manual, con auxilio o no de máquinas, equipos y herramientas.

En vista de que no existe una definición estrictamente legal, de "Pequeñas y Medianas Empresas (PYMES)", se incluyen las de "Microempresa" y "Artesanía", utilizadas por el MICIP.

Es de ésta definición de donde sale el término **PYMES**, siendo el mismo un acrónimo de: **Pequeña y Mediana Empresa.**

1.2 TIPOS DE PYMES

- Las PYMES se pueden clasificar según el sector productivo o de servicios al que pertenezcan. Ej.: agrícola, comercial, turístico, financiero, etc.
- PYMES establecidas con fines públicos o privados. Ej.: Empresas de servicios municipales o provinciales: agua, luz; las privadas como: zapaterías, mueblerías, etc.;
- PYMES según el monto de capital o patrimonio.
- PYMES según la cantidad de empleados.
- PYMES para la producción del mercado nacional o internacional o los dos a la vez.¹

¹ <http://www.pymes-ecuador.com/nuevo-sitio/inicio.htm>

1.3 FACTURACIÓN

Para facturar productos o servicios es necesario tener el Registro Único de Contribuyente (RUC), luego de su obtención el contribuyente debe solicitar a través del Sistema de Autorización de Impresión de Comprobantes de Venta a una de las imprentas autorizadas por el SRI, la impresión de sus facturas, boletas de venta, notas de crédito y débito y liquidaciones de compras de bienes o prestación de servicios.

En estos documentos constará la razón social del contribuyente y los demás requisitos previstos en el Reglamento de Facturación emitido por el SERVICIO DE RENTAS INTERNAS. Dicho Reglamento tiene por finalidad transparentar las actividades económicas en el país, a través de "Comprobantes de Venta", que registren todas las transacciones que se realicen. De esta manera, se crea un sistema de control a través del cruce de información y se asegura que el Impuesto al Valor Agregado (IVA) pagado por los consumidores en sus compras, sea transferido por los vendedores al Fisco.¹

1.3.1 Aspectos importantes de los comprobantes de venta

El Servicio de Rentas Internas presenta las siguientes características esenciales que deben tener los comprobantes de venta:

- Acreditan la transferencia de bienes o la prestación de servicios.
- Acreditan el traslado de mercadería cuando es realizada por consumidores finales.
- Sirven para sustentar costos y gastos a efectos de determinar y liquidar el Impuesto a la Renta.

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

- Los comprobantes de venta deberán ser impresos por establecimientos gráficos autorizados por el SRI.
- Podrán ser llenados en forma manual, mecánica o por sistemas computarizados. Deberán llenarse en forma conjunta y simultánea con sus copias, mediante el empleo de papel carbón, carbonado o autocopiativo químico, sin embargo las copias de los comprobantes de venta que se emitan mediante sistemas de computación podrán ser emitidas en forma consecutiva a la emisión del original siempre y cuando la copia sea idéntica al original.
- No deberán presentar borrones enmiendas o tachaduras.
- Deberán ser totalizados y cerrados individualmente.
- Cuando se anulen por cualquier motivo, deberá archivarse el documento original con todas sus copias con el sello "anulado".
- La falta de entrega de comprobantes de venta y retención o la entrega de aquellos que no cumplen los requisitos de ley, se sanciona con la clausura temporal de los establecimientos de los infractores.
- El transporte de mercadería no sustentada con comprobante de venta o guía de remisión, se sanciona con la incautación de la mercadería.
- Los fedatarios del Servicio de Rentas Internas controlarán el cumplimiento del Reglamento de Facturación y aplicarán las sanciones de clausura y comiso a los infractores.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

Éste será un punto muy importante a revisar en una parte de la elaboración de ésta disertación de grado, debido a lo delicado del tema FACTURACIÓN. Sobre todo tomando en cuenta que existen en el proceso de alquiler dos tipos de documentos: el CONTRATO DE ALQUILER, y LA FACTURA.

La existencia de estos dos documentos se justifica en el SEXTO inciso de los aspectos que los comprobantes de venta deben cumplir para con el SRI detallados anteriormente, y que dice:

“Los comprobantes de venta no deberán presentar borrones enmiendas o tachaduras.”¹

Observando esta característica, el propietario del local comercial “AZAHARES” ha decidido operar con dos documentos para así entregar una FACTURA nítida y satisfactoria al SRI, mientras que los controles y correcciones, existen muy a menudo, y en ocasiones son numerosos; se harán exclusivamente en el CONTRATO DE ALQUILER.

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.2 Tipos de comprobantes de venta y de retención autorizados

El nuevo **Reglamento de Facturación del Servicio de Rentas Internas** autoriza los siguientes tipos de "Comprobantes de Venta":

“Art. 1.- Comprobantes de venta.- Son comprobantes de venta los siguientes documentos que acreditan la transferencia de bienes o la prestación de servicios:

- a) Facturas
- b) Notas o boletas de venta
- c) Tiquetes emitidos por máquinas registradoras
- d) Liquidación de compras de bienes y prestación de servicios
- e) Boletos o entradas a espectáculos públicos

Son documentos complementarios a los comprobantes de venta, los siguientes:

- a) Notas de crédito;
- b) Notas de débito; y,
- c) Guías de remisión.¹

El Servicio de Rentas internas autorizará la impresión de los comprobantes de venta a través de los establecimientos gráficos autorizados, en los términos y bajo las condiciones del presente Reglamento.

De igual forma el Servicio de Rentas Internas podrá limitar o restringir la impresión de los comprobantes de venta, en función del grado de cumplimiento de las obligaciones del contribuyente que haya solicitado su impresión.²

¹⁻² http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.3 Emisión de los comprobantes de venta.-

Este capítulo está muy bien definido por el SRI en su Reglamento de Facturación, que dice:

“Art. 2.- (Reformado por el Art. 1 del D.E. 1072, R.O. 236-S, 19-VII-99).- Obligación de emisión de comprobantes de venta.-

Están obligados a emitir comprobantes de venta todos los sujetos pasivos de los impuestos a la renta, al valor agregado o a los consumos especiales, sean sociedades o personas naturales, incluyendo las sucesiones indivisas, obligados o no a llevar contabilidad, en los términos establecidos por la Ley de Régimen Tributario Interno.

Dicha obligación nace con ocasión de la transferencia de bienes o de la prestación de servicios de cualquier naturaleza, aún cuando las transferencias o prestaciones se realicen a título gratuito o no se encuentren gravadas con impuestos. Se exceptúa de esta obligación a las transacciones que se realicen por montos inferiores a cien mil sucres. Tampoco se emitirán comprobantes de venta por remuneraciones percibidas por los trabajadores en relación de dependencia.”¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.3.1 Art. 9.- Facturas.-

Se emitirán facturas en los siguientes casos:

- Cuando las operaciones se realicen para transferir bienes o prestar servicios a sociedades o personas naturales que tengan derecho al uso de crédito tributario; y,
- En operaciones de exportación.”¹

1.3.3.2 Art. 10.- Notas o boletas de venta.-

Se emitirán notas o boletas de venta únicamente en operaciones con consumidores o usuarios finales. El emisor identificará al comprador exclusivamente cuando éste requiera sustentar costos y gastos o cuando el monto de la transacción supere al valor establecido en el Art. 19 de este reglamento.”²

1.3.3.3 Art. 12.- Tickets o vales emitidos por máquinas registradoras.-

Se emitirán tiquetes o vales emitidos por máquinas registradoras en operaciones con consumidores o usuarios finales.”³

¹⁻²⁻³ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.4 Información pre-impresa e información no impresa

1.3.4.1 Información pre-impresa.-

Al respecto, el Reglamento de Facturación del SRI, en su Capítulo III, DE LOS REQUISITOS Y CARACTERÍSTICAS DE LOS COMPROBANTES DE VENTA, NOTAS DE CRÉDITO, NOTAS DE DÉBITO; tenemos lo siguiente:

“Art. 17.- Requisitos impresos para las facturas, notas de venta, liquidaciones de compras de bienes y prestación de servicios, notas de crédito y notas de débito.-

Estos documentos deberán contener los siguientes requisitos preimpresos:

1. Número de autorización de impresión del comprobante de venta, otorgado por el Servicio de Rentas Internas;
2. Número del registro único de contribuyentes del emisor;
3. Apellidos y nombres, denominación o razón social del emisor, en forma completa o abreviada conforme conste en el RUC. Adicionalmente podrá incluirse el nombre comercial o de fantasía, si lo hubiere;
4. Denominación del documento, de acuerdo al artículo 1 del presente reglamento;
5. Numeración de trece dígitos, que se distribuirá de la siguiente manera:¹
 - a) Los tres primeros dígitos corresponden al código del establecimiento conforme consta en el registro único de contribuyentes;
 - b) Separados por un guión (-) los siguientes tres dígitos corresponden al código asignado por el contribuyente a cada punto de emisión dentro de un mismo establecimiento; y,

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

- c) Separado también por un guión (-) constará el número secuencial de siete dígitos. Podrán omitirse la impresión de los ceros a la izquierda del número secuencial, pero deberán completarse los siete dígitos antes de iniciar la nueva numeración;
6. Dirección de la matriz y del establecimiento emisor cuando corresponda;
7. Fecha de caducidad del documento, expresada en mes y año, según la autorización del Servicio de Rentas Internas:
8. Datos de la imprenta o del establecimiento gráfico que efectuó la impresión:
- a) Número de autorización de la imprenta o establecimiento gráfico, otorgado por el Servicio de Rentas Internas;
 - b) Número de registro único de contribuyentes;
 - c) Nombres y apellidos, denominación o razón social, en forma completa o abreviada, según conste en el RUC. Adicionalmente podrá incluirse el nombre comercial o de fantasía;
9. Los destinatarios de los ejemplares. El original del documento se entregará al comprador, debiendo constar la indicación "ADQUIRENTE", "COMPRADOR", "USUARIO" o cualquier leyenda que haga referencia al comprador. La copia la conservará el emisor o vendedor, debiendo constar la identificación "EMISOR", "VENDEDOR" o cualquier leyenda que haga referencia al emisor. Se permitirá consignar en todos los ejemplares de los comprobantes de venta la leyenda: original-adquirente / copia-emisor, siempre y cuando el original se diferencie claramente de la copia.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

Para el caso de los comprobantes de venta que permiten sustentar crédito tributario, a partir de la segunda copia que impriman por necesidad del emisor deberá consignarse, además, la leyenda "copia sin derecho a crédito tributario"; y,

10. Los contribuyentes designados por el SRI como especiales deberán imprimir en los comprobantes de venta las palabras: "Contribuyente Especial" y el número de la resolución con la que fueron calificados. En el caso de contribuyentes especiales que a la fecha de su designación tuviesen comprobantes de venta vigentes en inventario podrán imprimir la leyenda de "Contribuyente Especial" mediante sello o cualquier otra forma de impresión.”¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.4.2 Información no pre-impresa

1.3.4.2.1 Art. 18.- Información no impresa para el caso de facturas.-

Las facturas contendrán la siguiente información no impresa sobre la transacción:

- Apellidos y nombres, o denominación o razón social del adquirente o usuario;
- Número de Registro Único de Contribuyentes o cédula del adquirente o usuario, excepto en las operaciones de exportación;
- Descripción del bien transferido o del servicio prestado, indicando la cantidad, unidad de medida y los códigos o numeración en los casos pertinentes, como en el caso de automotores, maquinaria y equipos;
- Precios unitarios de los bienes o servicios;
- Valor total de la transacción, sin incluir los impuestos que afectan la operación ni otros cargos, si los hubiere;
- Descuentos o bonificaciones;
- Impuesto al Valor Agregado y otros cargos adicionales.
- Se deberá consignar por separado indicando el nombre del impuesto o cargo y la tasa correspondiente;
- Importe total de la venta de bienes o de los servicios prestados, incluyendo los descuentos realizados;
- Lugar y fecha de emisión; y,
- Número de las guías de remisión o de cualquier otro documento relacionado con la operación que se factura, en los casos que corresponda. Cada factura debe ser totalizada y cerrada individualmente.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.4.2.2 Art. 19.- Información no impresa para el caso de notas o boletas de venta.-

Las notas, o boletas de venta contendrán la siguiente información no impresa sobre la transacción:

- Lugar y fecha de emisión;
- Concepto del bien transferido o del servicio prestado, indicando la cantidad, unidad de medida, códigos o numeración en los casos pertinentes, tales como equipos electrodomésticos, de computación y otros similares;
- Importe total de la venta o, en general, de la transferencia de bienes o la prestación de servicios, incluyendo los descuentos realizados y el impuesto al valor agregado; y,
- Cuando el importe de la venta, transferencia o servicio prestado, supere la suma de un millón de sucres por operación, será necesario consignar el número de Registro Único de Contribuyentes o cédula de identidad, los apellidos, nombres y dirección del adquirente o usuario.
- Cada boleta de venta debe ser totalizada y cerrada individualmente.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.4.2.3 Art. 20.- Información sobre la transacción en liquidaciones de compras de bienes o prestación de servicios.-

En el caso de liquidaciones de compras de bienes o prestación de servicios, deberá incluirse la siguiente información sobre la transacción:

- Datos de identificación del vendedor, esto es, sus apellidos y nombres, número de cédula de identidad, domicilio y lugar donde se realizó la operación, consignando la ciudad y provincia a las cuales pertenece y otros datos referenciales que permitan su ubicación;
- Descripción del servicio prestado o del bien vendido, indicando la cantidad y unidad de medida;
- Precios unitarios;
- Valor de venta de los productos comprados o de los servicios prestados;
- Monto discriminado del impuesto al valor agregado y otros cargos que gravan la operación;
- Importe total de la compra o la prestación del servicio, expresado en números y letras; y,
- Lugar y fecha de emisión.

Cada liquidación de compra o prestación de servicios debe ser cerrada o totalizada individualmente.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.3.4.2.4 Art. 21.- Información en tiquetes o vales emitidos por máquinas registradoras.-

Los tiquetes, vales o cintas emitidos por máquinas registradoras deberán contener la siguiente información:

- Identificación del emisor, indicando el número del Registro Único de Contribuyentes;
- Apellidos y nombres, denominación o razón social, según consta en el Registro Único de Contribuyentes y el nombre comercial, si lo hubiere;
- Dirección del establecimiento en el cual se emite el tiquete, vale o cinta;
- Número secuencial y número autogenerado por la máquina registradora;
- Marca y número de serie de fabricación de la máquina registradora;
- Descripción del bien vendido o del servicio prestado;
- Importe de la venta o del servicio prestado;
- Lugar, fecha y hora de emisión; y,
- Destino de los ejemplares: original para el adquirente o usuario y copia (optativa), para el emisor.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.4 Infracciones al reglamento de facturación

Clausura.- Por falta de emisión y entrega de comprobantes de venta, o por la emisión y entrega de comprobantes de venta que no cumplen los requisitos.

Incautación.- Por falta de sustento en comprobantes de venta y guías de remisión de la mercadería en transporte o almacenes.

Advertencia.- La administración tributaria procederá en estos casos según las facultades que le permite el actual marco legal. Los sellos de clausura instalados en los establecimientos de los infractores, son inviolables. Las mercaderías incautadas estarán en los depósitos del S.R.I., hasta que se acredite legalmente su propiedad o hasta el momento de su remate.¹

¹ http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html

1.5 Art. 9.- Facultad de la Superintendencia de Compañías.-

De conformidad con la Resolución No. 3179 publicada en el Registro Oficial 132 de 18 de agosto de 1997, la Superintendencia de Compañías, una vez terminado el proceso de constitución de aquellas personas jurídicas que están bajo su control, deberá otorgar el Registro Único de Contribuyentes o actualizarlo cuando se den los cambios mencionados en el Art. 6 de esta Resolución.

No se debe olvidar por lo tanto que todas las empresas naturales o jurídicas que se desarrollan en el País deben cumplir con las Leyes de Régimen Tributario y en el caso que nos ocupa retener el IVA (impuesto sobre el valor agregado) y entregar facturas o documentos justificativos de las operaciones, obligatoriamente cuando el valor de sus compras es de US \$ 4.00, aunque sería mejor que den siempre y que se haga ya una costumbre, porque eso ayuda a lograr una cultura tributaria y mejora la economía del País, a partir del 1 de septiembre del año en curso el porcentaje establecido en la Ley respectiva es de 12%.¹

¹ LEY DE COMPAÑÍAS, Codificación 000, Registro Oficial 312 de 5 de Noviembre de 1999

1.6 Marcas, patentes y Ley de Propiedad Intelectual

De conformidad a la Ley de Propiedad Intelectual, “se protege la Propiedad Intelectual, marcas de fábrica y de comercio, de servicios y lemas comerciales y se legisla respecto a los tramites para la inscripción de patentes de los descubrimientos y creaciones realizados, para ello es necesario tomar en consideración no solo los hechos, si no las consecuencias y responsabilidades por lo que cada cosa que se va a patentar puede ocasionar, no solo se trata de calificar en el sentido material de las cosas si no también desde la ética, la moral y las buenas costumbres, tan venidas a menos”.¹

¹ Ley de Propiedad Intelectual, Ley No. 83, R. O. No.320 de 19 de mayo de 1998

GRUPO II
MARCO TEÓRICO

CONCEPTOS GENERALES

Antes de iniciar el capítulo de Empresas, surge la necesidad de definir correctamente el concepto de negocio, al ser tratado así comúnmente las empresas sobretodo de origen familiar.

Negocio

Negocio es la consecuencia de la correcta administración de los recursos con un resultado económicamente positivo para las partes, es importante señalar que no solamente puede ser dinero sino relaciones de poder.

Por la extensión, la palabra se hizo sinónima "de una empresa individual comercial" (tan recientemente como el siglo XVIII) y tomaba también a veces el sentido "del nexo de actividades comerciales" o "de los representantes de la actividad comercial".

Específicamente, **negocio** puede referirse a entidades individuales de la economía. En algunas jurisdicciones legales, tales entidades son reguladas por la ley para conducir las operaciones a favor de empresarios.¹

¹ <http://es.wikipedia.org/wiki/Negocio>

CAPÍTULO 2: EMPRESAS

2.1 Definición

Persona jurídica, cualquiera que sea su régimen jurídico, y cuya actividad principal consista en la producción de bienes y servicios destinados al mercado. Los empresarios individuales se considerarán equiparados a esta categoría.¹

Se conoce también como el ejercicio profesional de una actividad económica planificada, con la finalidad o el objetivo de intermediar en el mercado de bienes o servicios, y con una unidad económica organizada en la cual ejerce su actividad profesional el empresario por sí mismo o por medio de sus representantes.

Es por lo tanto un grupo social en el que, a través de la administración del capital y el trabajo, se producen bienes y/o servicios tendientes a la satisfacción de las necesidades de la comunidad.

Las empresas son las máximas responsables de la producción, ya que solo ellas son capaces de obtener las ventajas de la producción en masa. Únicamente las empresas pueden reunir las grandes cantidades de recursos financieros y físicos que se precisan para construir las fábricas que en la actualidad se requieren. De igual forma, sólo las empresas tienen capacidad para organizar los complejos procesos de producción y de distribución que las sociedades modernas exigen.

¹ medioambiente.geoscopio.com/medioambiente/temas/profit/terminos.php

2.2 Características de una empresa

2.2.1 Elementos

Una empresa combina tres factores que son:

- Factores activos: empleados, propietarios, sindicatos, bancos,
- Factores pasivos: materias primas, tecnología, conocimiento, contratos financieros,
- Organización: coordinación y orden entre todos los factores y áreas¹

2.2.1.1 Factores activos

Personas físicas o/y jurídicas (otras entidades mercantiles, Cooperativa, fundaciones, etc.) constituyen una empresa realizando, entre otras, aportación de capital (sea puramente dinerario sea de tipo intelectual, patentes, etc.). Estas "personas" se convierten en accionistas de la empresa. Por otra parte, una empresa también se compone de **factores pasivos**, como la tecnología, las materias primas utilizadas, los contratos financieros de los que dispone, etc.

Participan, en sentido amplio, en el desarrollo de la empresa:

- Administradores
- Clientes
- Colaboradores y *partners*
- Fuentes financieras
- Socios
- Suministradores y proveedores
- Trabajadores²

¹⁻² <http://es.wikipedia.org/wiki/Empresa>

2.2.1.2 Factores pasivos

Todos los que son usados por los elementos activos y ayudan a conseguir los objetivos de la empresa.¹

2.2.1.3 Organización

Proporciona eficiencia dividiendo el trabajo en áreas especializadas, coordinándolas y dando los procedimientos estándar a seguir. La organización debe adaptarse a los objetivos de la empresa, y por tanto puede ir cambiando con el tiempo para adaptarse.²

2.3 Buen Gobierno empresarial

Las prácticas de Buen Gobierno empresarial varían enormemente en cuanto a su detalle y aplicación país a país. Básicamente su objetivo es generar confianza ante accionistas, empleados, actores económicos y sociedad en general.

Elementos esenciales del Buen Gobierno empresarial son:

- Transparencia informativa
- Informes y Auditoría de cuentas
- Códigos éticos
- Gestión del Riesgo
- Protección del Patrimonio
- Planificación Estratégica³

¹ <http://es.wikipedia.org/wiki/Empresa>

² medioambiente.geoscopio.com/medioambiente/temas/profit/terminos.php

³ <http://es.wikipedia.org/wiki/Empresa>

Dentro de estos aspectos deben contemplarse como integrantes:

- 4 el Buen Gobierno de los recursos humanos
- 5 el Buen Gobierno de la Calidad
- 6 el Buen Gobierno de los Sistemas de Información y las Comunicaciones
- 7 el Buen Gobierno medioambiental¹

2.4 Clasificación

Las empresas se pueden clasificar de la siguiente manera:

- Sectores Económicos
- El origen de su capital.
- Su Tamaño
- Conformación de su capital
- El pago de impuestos
- El número de propietarios
- La función social
- La forma de explotación²

¹ <http://es.wikipedia.org/wiki/Empresa>

² <http://www.monografias.com/trabajos21/clasificacion-empresas/clasificacion-empresas.shtml#>

2.4.1 Por sectores económicos

TIPO	DESCRIPCION	EJEMPLO
Extractivas	Dedicadas a explotar recursos naturales	Hidalgo & Hidalgo, Cemento Chimborazo, Petroecuador
Servicios	Entregan sus servicios o la prestación de estos a la comunidad	Clínicas, salones de belleza, Azahares
Comercial	Desarrolla la venta de los productos terminados en la fábrica	Almacenes Japón, Supermaxi
Agropecuaria	Explotación del campo y sus recursos	Bananas Bonita, agroindustria
Industrial	la materia prima en un producto terminado	Pinto, Cervecería Nacional

Tabla 1.1: Clasificación de empresas por sector económico

Elaborado por: David Rueda

2.4.2 Por su tamaño

TIPO	DESCRIPCION	EJEMPLO
Grande	Su constitución se soporta en grandes cantidades de capital, un gran número de trabajadores y el volumen de ingresos al año, su número de trabajadores excede a 100 personas	Hidalgo & Hidalgo, Cemento Chimborazo, Petroecuador
Mediana	Su capital, el número de trabajadores y el volumen de ingresos son limitados y muy regulares, número de trabajadores superior a 20 personas e inferior a 100	Micro mercados, hoteles medianos, talleres textiles
Pequeña	Su capital, número de trabajadores y sus ingresos son muy reducidos, el número de trabajadores no excede de 20 personas	Azahares, Tiendas, Almacenes pequeños

Tabla 1.2: Clasificación de empresas por tamaño

Elaborado por: David Rueda

A su vez, las pequeñas empresas se clasifican así:

TIPO	DESCRIPCION	EJEMPLO
Pequeña	El número de trabajadores no excede de 20 personas, pero los ingresos siguen siendo considerados cuantías empresariales.	Micro-mercados, agencias de turismo
Micro-Empresa	Su capital, número de trabajadores y sus ingresos solo se establecen en cuantías muy personales, el número de trabajadores no excede de 10 (trabajadores y empleados)	Almacenes pequeños, mecánicas
Fami-Empresa	Es un nuevo tipo de explotación en donde la familia es el motor del negocio convirtiéndose en una unidad productiva	Azahares, tiendas, licorerías

Tabla 1.3: Sub-Clasificación de empresas pequeñas por tamaño Elaborado por: David Rueda

2.4.3 Por el origen del capital

TIPO	DESCRIPCION	EJEMPLO
Publico	Su capital proviene del Estado o Gobierno	Sistema de transporte
Privado	Aquellas en que el capital proviene de particulares	Bancos, Almacenes
Economía mixta	El capital proviene una parte del estado y la otra de particulares	Seguro Social

Tabla 1.4: Clasificación de empresas por origen del capital Elaborado por: David Rueda

2.4.4 Por la explotación y conformación de su capital

TIPO	DESCRIPCION	EJEMPLO
Multinacionales	En su gran mayoría el capital es extranjero y explotan la actividad en diferentes países del mundo (globalización)	American Airlines, Telefonica
Grupos económicos	Estas empresas explotan uno o varios sectores pero pertenecen al mismo grupo de personas o dueños	LG: división de industria química
Nacionales	El radio de atención es dentro del país normalmente tienen su principal en una ciudad y sucursales en otras	Supertaxi
Locales	Son aquellas en que su radio de atención es dentro de la misma localidad	Almacenes, tiendas

Tabla 1.5: Clasificación de empresas por explotación y conformación de su capital
Elaborado por: David Rueda

2.4.5 Por el pago de impuestos

TIPO	DESCRIPCION	EJEMPLO
Personas naturales	El empresario como Persona Natural es aquel individuo que profesionalmente se ocupa de algunas de las actividades mercantiles, la Persona Natural se inscribe en la Cámara de Comercio, igualmente se debe hacer con la Matrícula del Establecimiento Comercial	Azahares, almacenes, mecánicas
Régimen común	Empresas legalmente constituidas y sobrepasan las limitaciones del régimen simplificado, deben llevar organizadamente su contabilidad	Almacenes Japón, Importadoras de repuestos automotrices
Gran contribuyente	Agrupar el mayor número de empresas con capitales e ingresos compuestos en cuantías superiores a los de régimen común. Son las mas grandes del país	Bananas Bonita, La Cemento Nacional

Tabla 1.6: Clasificación de empresas por pago de impuestos
Elaborado por: David Rueda

2.4.6 Por la función social

TIPO	DESCRIPCION	EJEMPLO
Con Ánimo de Lucro	Se constituye la empresa con el propósito de explotar y ganar más dinero	Azahares, almacenes, mecánicas
Trabajo Asociado	Grupo organizado como empresa para beneficio de los integrantes	Asociaciones de talleres artesanales para exportación
Sin Ánimo de Lucro	Aparentemente son empresas que lo más importante para ellas es el factor social de ayuda y apoyo a la comunidad	INNFA, SOLCA
Economía Solidaria	En este grupo pertenecen todas las cooperativas sin importar a que actividad se dedican lo más importante es el bienestar de los asociados y su familia	Cooperativas de taxis, cooperativas de ahorro

Tabla 1.7: Clasificación de empresas por función social

Elaborado por: David Rueda

2.5 Clasificación de la empresa Azahares según los puntos documentados

Según los puntos anteriormente descritos, la empresa en estudio en este trabajo encaja en las siguientes características:

GRUPO	CLASIFICACIÓN
Sector económico	SERVICIOS
Tamaño	MICRO EMPRESA- FAMIEMPRESA
Origen del capital	PRIVADA
Explotación y conformación del capital	LOCAL
Pago de impuestos	PERSONA NATURAL
Función social	CON ÁNIMO DE LUCRO

Tabla 1.8: Clasificación de la empresa en estudio Elaborado por: David Rueda

CAPÍTULO 3: METODOLOGÍAS DE DESARROLLO

3.1 Definición.-

Una metodología de desarrollo de software permite producir organizada y económicamente software de alta calidad, siguiendo una serie de pasos donde se utilizan un conjunto de técnicas, notación y normas de documentación preestablecidas¹.

Presenta una forma de modelar el mundo real con el fin de llevarlo al dominio del computador, a través del modelo se puede obtener una visión global del sistema, para facilitar la especificación de los requerimientos, las restricciones del sistema, y de la solución del problema.

3.2 Tipos

El análisis y diseño, como elementos esenciales del proceso de desarrollo; obligan a tener especial atención y por tal motivo se han ido creando metodologías que sirven de base para tomar las decisiones que afectarán el producto final. Con el advenimiento de la disciplina de la ingeniería del software se inicia el proceso de desarrollo de metodologías, las primeras de ellas fueron las **estructuradas**. Luego, con el avance de las herramientas y tendencias aparecen las denominadas **metodologías ágiles**.

¹<http://www.monografias.com/trabajos15/ingenieria-software/ingenieria-software.shtml>

3.2.1 Metodologías Estructuradas

Los modelos de los métodos clásicos difieren bastante en su conformación y en su naturaleza, pero exaltan casi siempre las virtudes del planeamiento y poseen un espíritu normativo. Comienzan con la licitación y el análisis completo de los requerimientos del usuario. Después de un largo período de intensa interacción con usuarios y clientes, los ingenieros establecen un conjunto definitivo y exhaustivo de rasgos, requerimientos funcionales y no funcionales. Esta información se documenta en forma de especificaciones para la segunda etapa, el diseño, en el que los arquitectos, trabajando junto a otros expertos en temas puntuales (como ser estructuras y bases de datos), generan la arquitectura del sistema. Luego los programadores implementan ese diseño bien documentado y finalmente el sistema completo se prueba y se despacha.¹

El cuadro que se muestra en la siguiente página es un resumen de algunos de los tipos mencionados.

¹ <http://www.monografias.com/trabajos15/ingenieria-software/ingenieria-software.shtml>

Modelo	Versión de origen	Características
Modelo en cascada	Secuencial: Bennington 1956 – Iterativo: Royce 1970 – Estándar DoD 2167-A	Secuencia de requerimiento, diseño del sistema, diseño de programa, codificación, pruebas, operación y mantenimiento
Modelo en cascada con fases superpuestas	Cf. McConnell 1996:143-144	Cascada con eventuales desarrollos en paralelo (Modelo Sashimi)
Modelo iterado con prototipado	Brooks 1975	Iterativo – Desarrollo incremental
Desarrollo rápido (RAD)	J. Martin 1991 – Kerr/Hunter 1994 – McConnell 1996	Modelo lineal secuencial con ciclos de desarrollo breves
Modelo V	Ministerio de Defensa de Alemania 1992	Coordinación de cascada con iteraciones
Modelo en espiral	Barry Boehm 1988	Iterativo – Desarrollo incremental. Cada fase no es lineal, pero el conjunto de fases sí lo es.
Modelo en espiral win- win	Barry Boehm 1998	Iterativo – Desarrollo incremental – Aplica teoría- W a cada etapa
Modelo de desarrollo concurrente	Davis y Sitaram 1994	Modelo cíclico con análisis de estado
Modelo de entrega incremental (<i>Staged delivery</i>)	McConnell 1996: 148	Fases tempranas en cascada – Fases posteriores descompuestas en etapas

Algunas de las Metodologías Estructuradas¹

¹ <http://www.monografias.com/trabajos15/ingenieria-software/ingenieria-software.shtml>

A fines del siglo XX había un abanico de tipos de procesos o modelos disponibles: el más convencional era el modelo en cascada o lineal-secuencial, pero al lado de él había otros como el modelo “V”, el modelo de construcción de prototipos, el de desarrollo rápido o RAD, el modelo incremental, el modelo en espiral básico, el espiral win-win, el modelo de desarrollo concurrente y un conjunto de modelos iterativos o evolutivos (basados en componentes, por ejemplo) que en un primer momento convivían apaciblemente con los restantes, aunque cada uno de ellos se originaba en la crítica o en la percepción de las limitaciones de algún otro. Algunos eran prescriptivos, otros descriptivos. Algunos de los modelos incluían iteraciones, incrementos, recursiones o bucles de retroalimentación; y algunos se preciaban también de rápidos, adaptables y expeditivos.¹

Para este análisis tomaremos los modelos más conocidos:

3.2.1.1 Modelo en cascada

Es el enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, de forma tal que el inicio de cada etapa debe esperar a la finalización de la inmediatamente anterior.

Un ejemplo de una metodología de desarrollo en cascada es:

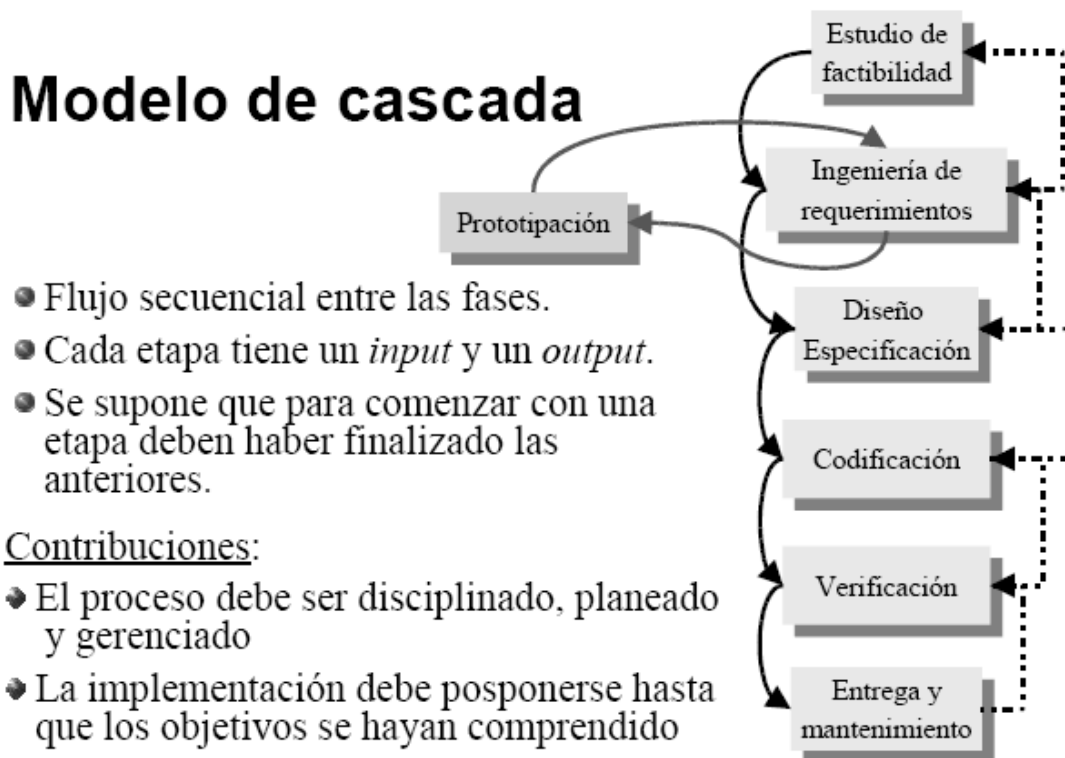
1. Análisis de requisitos
2. Diseño
3. Codificación
4. Pruebas
5. Implantación
6. Mantenimiento²

¹ http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/heterodox.asp

² http://es.wikipedia.org/wiki/Modelo_en_cascada

De esta forma, cualquier error de diseño detectado en la etapa de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costes del desarrollo. La palabra *cascada* sugiere, mediante la metáfora de la fuerza de la gravedad, el esfuerzo necesario para introducir un cambio en las fases más avanzadas de un proyecto.

A continuación tenemos una representación gráfica de dicho modelo:



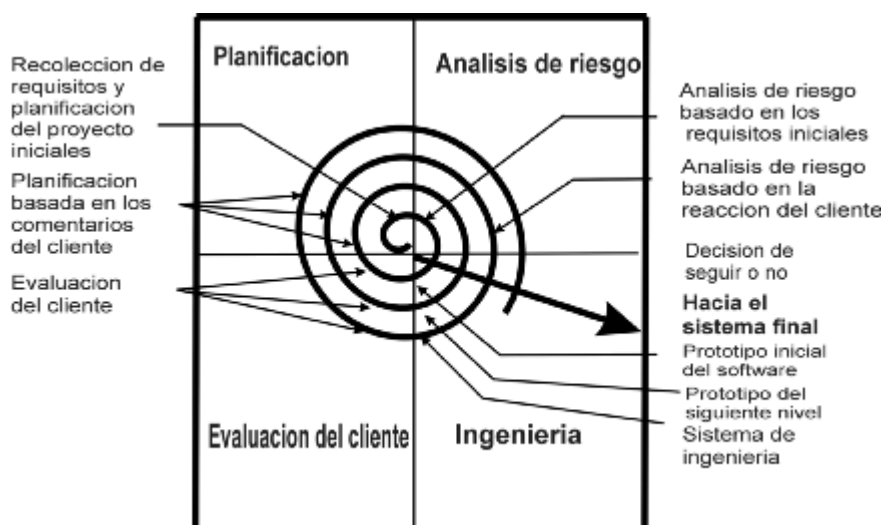
Modelo en cascada¹

¹ http://es.wikipedia.org/wiki/Modelo_en_cascada

3.2.1.2 Modelo en espiral

El modelo espiral para la ingeniería de software ha sido desarrollado para cubrir las mejores características tanto del ciclo de vida clásico, como de la creación de prototipos, añadiendo al mismo tiempo un nuevo elemento: el análisis de riesgo. El modelo representado mediante la espiral de la figura a continuación, define cuatro actividades principales:

1. **Planificación:** determinación de objetivos, alternativas y restricciones.
2. **Análisis de riesgo:** análisis de alternativas e identificación/resolución de riesgos.
3. **Ingeniería:** desarrollo del producto del "siguiente nivel",
4. **Evaluación del cliente:** Valorización de los resultados de la ingeniería.¹



Modelo en Espiral ²

¹⁻² <http://www.itlp.edu.mx/publica/tutoriales/analisis/24.htm>

Durante la primera vuelta alrededor de la espiral se definen los objetivos, las alternativas y las restricciones, y se analizan e identifican los riesgos. Si el análisis de riesgo indica que hay una incertidumbre en los requisitos, se puede usar la creación de prototipos en el cuadrante de ingeniería para dar asistencia tanto al encargado de desarrollo como al cliente.

El cliente evalúa el trabajo de ingeniería (cuadrante de evaluación de cliente) y sugiere modificaciones. Sobre la base de los comentarios del cliente se produce la siguiente fase de planificación y de análisis de riesgo. En cada bucle alrededor de la espiral, la culminación del análisis de riesgo resulta en una decisión de "seguir o no seguir".

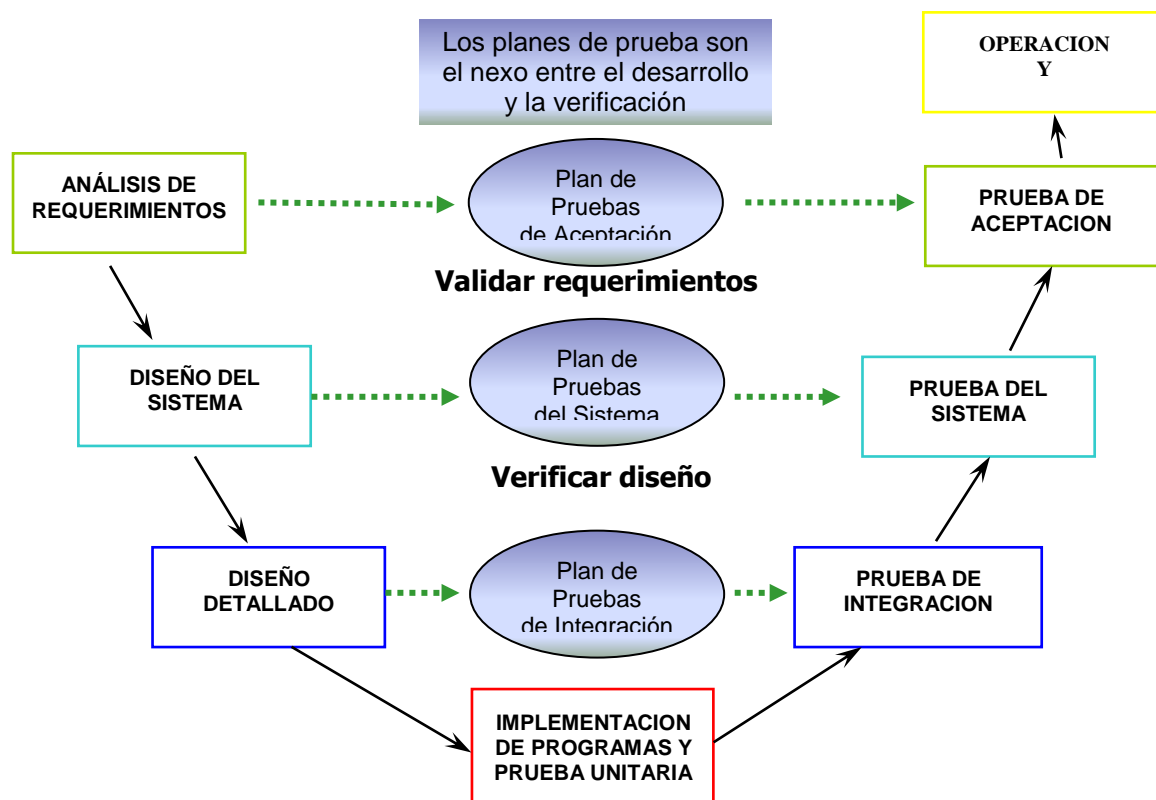
Con cada iteración alrededor de la espiral (comenzando en el centro y siguiendo hacia el exterior), se construyen sucesivas versiones del software, cada vez más completa y, al final, al propio sistema operacional.

El paradigma del modelo en espiral para la ingeniería de software es actualmente el enfoque más realista para el desarrollo de software y de sistemas a gran escala. Utiliza un enfoque evolutivo para la ingeniería de software, permitiendo al desarrollador y al cliente entender y reaccionar a los riesgos en cada nivel evolutivo. Utiliza la creación de prototipos como un mecanismo de reducción de riesgo, pero, lo que es más importante permite a quien lo desarrolla aplicar el enfoque de creación de prototipos en cualquier etapa de la evolución de prototipos.¹

¹ <http://www.itlp.edu.mx/publica/tutoriales/analisis/24.htm>

3.2.1.3 Modelo en V

El Modelo V tiende a ser muy relacionado con el Modelo de Cascada puesto que es una evolución del mismo.



Modelo en V¹

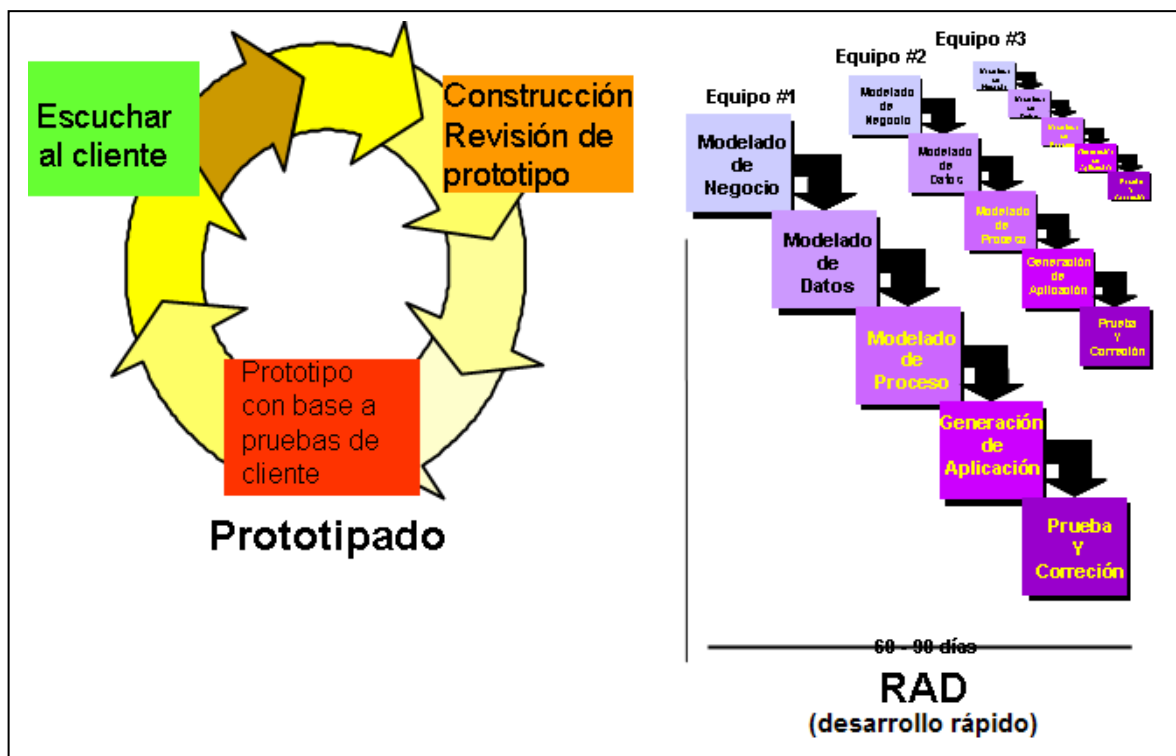
Puede notarse que su primera mitad es similar al Modelo en Cascada, y la otra mitad tiene como finalidad hacer pruebas e integración asociado a cada una de las etapas de la mitad anterior.

Se puede identificar una ventaja principal con respecto al Modelo Cascada más simple, y se refiere a que este modelo involucra chequeos de cada una de las etapas del modelo de cascada.²

¹ <http://carolina.terna.net/ingsw2/Datos/Cascada-ModeloV.doc>

3.2.1.4 Modelo Interactivo

En estas situaciones, el usuario interviene en todas las fases de construcción como se ilustra en la figura:

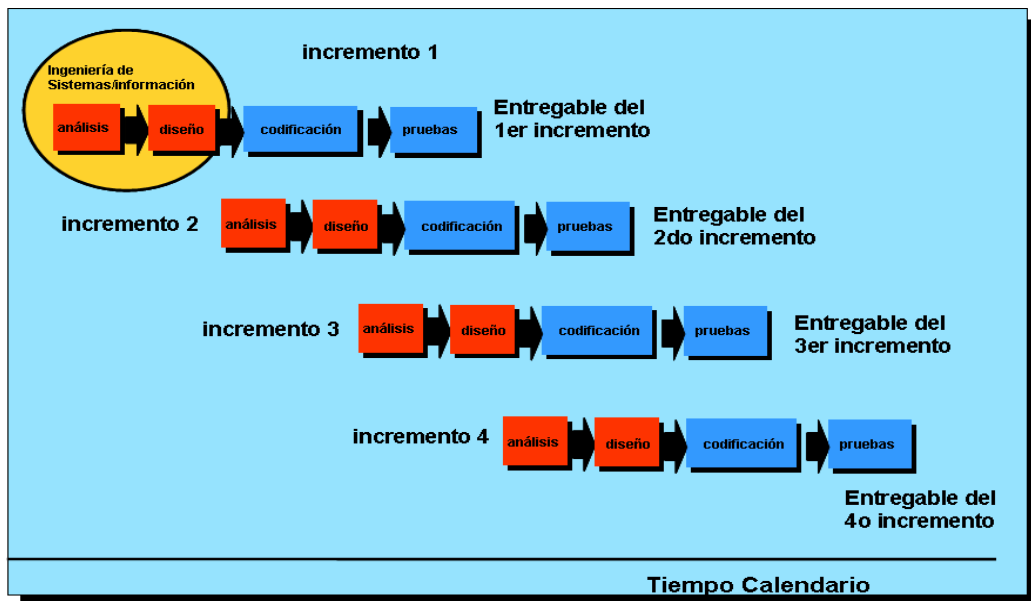


Modelos Interactivos¹

¹ <http://www.itver.edu.mx/comunidad/material/ing-software/unidad1.ppt>

3.2.1.5 Modelo Incremental

El trabajo se realiza en varios módulos o partes. Este tipo de trabajo es usual en empresas de desarrollo de software mayores.



Modelo Incremental¹

¹ <http://www.itver.edu.mx/comunidad/material/ing-software/unidad1.ppt>

3.2.2 Metodologías Ágiles

El desarrollo de software, que inicialmente era básicamente un proceso de "prueba y error" se convirtió, con el paso del tiempo en un proceso disciplinado y de mucha planificación. Sin embargo, posteriormente recibieron críticas por la exagerada burocracia que implican. Parecería que seguir la metodología implicaba más esfuerzo que el desarrollo mismo.¹

En esas circunstancias y hace algunos pocos años, surgieron las denominadas "metodologías ágiles" conocidas anteriormente como *metodologías livianas*, que evitan en lo posible aquella burocracia de las ahora conocidas como "metodologías pesadas", enfocándose en la gente y los resultados.

Es así que grupos de desarrollo han experimentado soluciones que basan su fundamento en la adaptabilidad de los procesos de desarrollo, en lugar de seguir esperando lograr resultados predecibles de un proceso que no evoluciona.²

Las metodologías ágiles como puede entenderse mal, no están en contra de administrar procesos de desarrollo. Por el contrario promueve la formalización de procesos adaptables, tomando en consideración factores como el tiempo y utilidad.

¹ <http://www.angelfire.com/ult/lupa/metodologia/metodologiaagil.html>

² <http://www.acis.org.co/index.php?id=551>

Las características de las metodologías ágiles pueden explicarse a través de los siguientes cuatro principios fundamentales:

- Los **individuos** e **interacciones** son más importantes que los **procesos** y **herramientas** dado que el proceso de desarrollo es creativo, no es posible pensar que las personas funcionen respondiendo a órdenes, a procesos rígidos.
- Que el **software funcione** es más importante que la **documentación exhaustiva**, puesto que si el software no funciona la documentación no vale de nada. A nivel interno puede haber documentación, pero solo la necesaria y a nivel externo lo que el cliente requiera.
- La **colaboración con el cliente** es más importante que la **negociación de contratos**, lo que supone que la satisfacción del cliente con el producto será mayor, mientras exista una conversación y realimentación continua entre éste y la empresa.
- La **respuesta ante el cambio** es más importante que el **seguimiento de un plan**, puesto que si un proyecto de software no es capaz de adaptarse a los cambios fracasará, especialmente en productos de gran envergadura. La estrategia de planificación se basa en: planes detallados para las próximas semanas, planes aproximados para los próximos meses y muy generales para plazos mayores.¹

¹ http://curso-sobre.berlios.de/curso/trab/cjacuna/Open_Source_y_Met_Agiles.pdf

Existen, como es de suponer; varios tipos de Metodologías Ágiles, de las cuales tomaremos las más importantes:¹

- Programación Extrema (XP)
- Scrum
- Modelamiento ágil
- Desarrollo de software adaptativo (ASD)
- Crystal Clear
- Método de desarrollo de sistemas dinámicos (DSDM)
- Desarrollo conducido de características
- Desarrollo de software con tendencias
- Proceso ágil unificado (AUP)

Otras aproximaciones incluyen:

- Documentación ágil
- Agile ICONIX
- Entorno de trabajo para soluciones Microsoft o Microsoft Solutions Framework (MSF)
- Metodología Ágil de datos
- Reconstrucción de bases de datos

¹ http://en.wikipedia.org/wiki/Agile_software_development

Valores, Principios y prácticas

En general, todos los procesos de las metodologías ágiles poseen ciertas características que son:

Valores:

- Comunicación
- Simplicidad
- Feedback, o recolección de opiniones del cliente-usuario
- Coraje
- Humildad¹

Principios

- Asumir simplicidad
- Acoger el cambio
- Asumir que el siguiente esfuerzo es el objetivo secundario
- Cambio incremental
- Maximizar la inversión del inversionista
- Modelar con un propósito
- Múltiples modelos
- Trabajo de calidad
- Feedback de rapidez
- El software es el objetivo primario
- Viajar ligero²

¹ http://en.wikipedia.org/wiki/Agile_software_development

² <http://www.agilemodeling.com/principles.htm>

Prácticas

- Participación activa del inversionista
- Aplicar los correctos artefactos (herramientas)
- Conocimiento colectivo
- Crear varios modelos en paralelo
- Crear contenidos simples
- Representar modelos simples
- Mostrar públicamente los modelos
- Interactuar con otros artefactos
- Modelar en pequeños incrementos
- Modelar con otros
- Probar lo que se hace con código
- Usa sola fuente de información
- Usar las herramientas más simples¹

3.2.2.1 Programación Extrema

La programación extrema o extreme Programming (XP) es una aproximación a la ingeniería de software formulada por Kent Beck, autor del primer libro sobre la materia: Extreme Programming Explained: Embrace Change.²

Es una metodología basada en los principios de simplicidad, comunicación, feedback y valor añadido.

¹ <http://www.agilemodeling.com/principles.htm>

Su funcionamiento trata de que todo el equipo de desarrollo se una alrededor de tareas sencillas y con suficiente feedback para permitir al equipo de desarrollo conocer en que estadio de desarrollo se encuentra.¹

Las directivas en las que se basa el modelo de programación extrema son:

- Equipo de desarrollo compacto: En el modelo XP cada programador forma parte integrante del "equipo de desarrollo".
- Planificación, entregas rápidas y verificación de los usuarios: Se utilizan mecanismos simples de planificación para decidir que es lo siguiente a desarrollar. Centrados en que el resultado a obtener beneficie al cliente, el equipo produce nuevas aplicaciones en versiones constantes, integradas y verificadas por el propio cliente y el equipo de desarrollo.
- Diseño simple y mejora continua: Los programadores trabajan de forma conjunta con un diseño simple y con la obsesión de verificar todo el código que se produce.
- Integración, propiedad colectiva y codificación estándar: El equipo de desarrollo mantiene las aplicaciones en funcionamiento constantemente. Las funcionalidades se codifican y se integran inmediatamente en la aplicación, verificando que cumplen todos los tests de calidad.²

¹ http://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema

² <http://www.deister.es/deisterwww/company/xprogramming.html>

3.2.2.2 Scrum

Es otro de los métodos ágiles para la administración de proyectos. Ha sido documentado para mejorar dramáticamente la productividad en equipos previamente paralizados por metodologías pesadas. La primera documentación fue realizada por Takeuchi and Nonaka en "The New New Product Development Game" (Harvard Business Review, Jan-Feb 1986).¹

Se ha utilizado exitosamente para envolver a la Programación Extrema con otras metodologías de desarrollo. Sin embargo, puede ser aplicado en teoría a cualquier contexto donde un grupo de gente desea trabajar en equipo para alcanzar un objetivo común, como es el caso de montar una pequeña escuela, proyectos de investigación científica, e incluso, la planificación de una boda.

Por otro lado, también puede ser utilizado en la ejecución de grupos de mantenimiento, o como una aproximación a un programa de mantenimiento, lo que se conoce como Scrum de Scrums

Se caracteriza por:

- Un panel de tareas dinámico actualizado constantemente de las tareas que deben ser realizadas y sus prioridades
- Terminación de un amplio conjunto de ítems del panel de tareas en series de interacciones cortas conocidas como sprints
- Una breve reunión diaria o Scrum en la que el progreso es detallado, se describe el trabajo por hacer y los impedimentos son resueltos

¹ <http://en.wikipedia.org/wiki/SCRUM>

- Una breve sesión de planeamiento en la cual los ítems del panel de tareas que serán terminadas en los sprints son definidas
- Un breve resumen de opiniones en retrospectiva, en la que cada miembro comenta sobre el último sprint¹

El Scrum tiene un moderador conocido como ScrumMaster, cuyo principal trabajo es remover los impedimentos que se interponen entre el equipo y el cumplimiento del objetivo del sprint. Dicho moderador no es un líder del equipo, puesto que éste es auto-organizable; pero actúa como un equilibrante de productividad entre todos los miembros del equipo y cualquier influencia desestabilizante.

Permite así mismo, la creación de equipos auto-organizativos mediante la comunicación verbal impulsadora entre todos los miembros del grupo y entre todas las disciplinas envueltas en el proyecto.

Un principio primordial del Scrum es el reconocimiento de que los retos empíricos no pueden ser manejados exitosamente en un proceso de control tradicional.

¹ <http://en.wikipedia.org/wiki/SCRUM>

Scrum Simplificado

Muchas organizaciones se muestran resistentes a metodologías introducidas a nivel de gente común. Sin embargo la adaptabilidad del Scrum permite que sea introducido "saludablemente".

Por ejemplo, siguiendo estos tres pasos:

- Concertar un demo del software con el cliente en el plazo de un mes
- Como equipo, tomar un mes para tener listo el software para este demo, funcionando en ejecución, y no solamente con imágenes descriptivas
- Al momento de realizar el demo, tomar un resumen de opiniones del cliente (feedback) y usarlo como guía para el trabajo de desarrollo del siguiente mes
- Con este proceso, la esencia del patrón del Scrum está implementado, i puede seguirse desarrollando.¹

3.2.2.3 Modelamiento ágil

El modelamiento ágil es una metodología basada en la práctica para el efectivo modelamiento y documentación de sistemas basados en software. Para decirlo simple, es una colección de valores, principios y prácticas para modelamiento de software que puede ser aplicado en un proyecto de desarrollo de software en una forma efectiva y ligera. Es en ocasiones un suplemento de otras metodologías ágiles como la Programación Extrema, el Proceso Ágil Unificado, y el Scrum.²

¹ <http://en.wikipedia.org/wiki/SCRUM>

² http://en.wikipedia.org/wiki/Agile_Modeling

Los principios y valores de las prácticas de ésta metodología ayudan a mitigar las críticas de las Metodologías Ágiles. El principio "Maximizar el la inversión del inversionista" inspira al desarrollador a colaborar con el cliente para proveer un adecuado nivel de documentación.

El principio "Modelar con otros" encamina a designar cuál es la solución que el cliente necesita.

Limitaciones

Existe una significativa dependencia en la comunicación cara a cara y la colaboración del cliente.

El modelamiento ágil es difícil de aplicar cuando se trata de equipos grandes; los miembros del equipo a veces no son bien ubicados, y sus habilidades no son explotadas. Sin embargo, puede ser escalado con técnicas de arquitectura ágil.¹

¹ http://en.wikipedia.org/wiki/Agile_Modeling

3.2.2.4 Método de desarrollo de sistemas dinámicos

Es un método que provee un entorno de trabajo para el Desarrollo de Aplicaciones Rápidas (Rapid Application Development RAD), soportado por la intervención continua del usuario en un desarrollo interactivo e incremental, el cual es responsable de cambiar los requerimientos para desarrollar un sistema que satisfaga las necesidades empresariales a tiempo y dentro de un presupuesto.

Fue desarrollado en el Reino Unido en los años 90 por un consorcio de vendedores y expertos en el campo del Desarrollo de Sistemas de Información, conocido como el consorcio DSDM por sus siglas en inglés (Dynamic Systems Development Method) el cual combinó sus mejores experiencias puestas en práctica.

Como una extensión del desarrollo de aplicaciones rápidas, se enfoca en proyectos de Sistemas de Información que se caracterizan por escalas de tiempo no muy amplias así como los presupuestos.

Consiste en tres fases:

- pre-proyecto
- ciclo de vida del proyecto
- y post proyecto¹

¹ http://en.wikipedia.org/wiki/Dynamic_Systems_Development_Method

La fase de ciclo de vida del proyecto se subdivide en cinco etapas:

- estudio de factibilidad
- estudio de mercado
- interacción del modelo funcional
- interacción de diseño y construcción
- implementación¹

Esta metodología reconoce que los proyectos son limitados por tiempo y recursos, y planifica acordemente para alcanzar las necesidades empresariales. Para el alcance de los objetivos, impulsa el uso del desarrollo de aplicaciones rápidas con el consecuente riesgo de que muchas "esquinas" sean cortadas, es decir; se toman atajos. En ciertas circunstancias puede ser integrado con otras metodologías ágiles.

¹ http://en.wikipedia.org/wiki/Dynamic_Systems_Development_Method

3.2.2.5 Entorno de trabajo para soluciones Microsoft o Microsoft Solutions

Framework (MSF)

Es un conjunto de principios, modelos, disciplinas, conceptos y guías para el diseño de aplicaciones Microsoft; que permiten al desarrollador alcanzar el éxito en el Ciclo de Vida de Desarrollo de Software (software development life cycle, SDLC). Provee una guía adaptable basada en experiencias y las mejores prácticas de dentro y fuera de Microsoft, para incrementar la posibilidad de entrega exitosa de soluciones de tecnologías de información al usuario, mediante el trabajo ágil, disminuyendo el número de personas en el grupo de proyecto, advirtiendo los riesgos mientras se obtienen resultados de alta calidad.

Provee un entorno de trabajo de alto nivel como guía, y principios que pueden ser ubicados en una variedad de plantillas de procesos prescriptivos. Está estructurado en metodologías descriptivas y prescriptivas. El componente descriptivo es conocido al momento como el Metamodelo MSF 4.0¹, el cual es una descripción teórica de las mejores prácticas del ciclo de vida del desarrollo de software, para crear metodologías.

Según Microsoft, las diferentes organizaciones tienen dinámicas divergentes y prioridades contradictorias durante el desarrollo de software. Algunas de estas empresas necesitan un entorno de desarrollo responsable y adaptable, mientras que otras requieren de entornos repetitivos y más controlados.

¹ http://en.wikipedia.org/wiki/Microsoft_Solutions_Framework

Para satisfacer estas necesidades, Microsoft representa el Metamodelo del MSF 4.0 en dos plantillas de metodologías prescriptivas que proveen guía específica en el proceso, llamadas Entorno de Soluciones Microsoft para Desarrollo de software Ágil (Microsoft Solutions Framework for Agile Software Development, MSF4ASD) y Entorno de Soluciones Microsoft para el proceso de mejora del Modelo de Madurez y Capacidad (Microsoft Solutions Framework for Capability Maturity Model Integration Process Improvement (MSF4CMMI)).¹

Consiste de series de cortos ciclos de desarrollo e interacciones. Acoge el desarrollo rápido e interactivo, con continuo aprendizaje y refinamiento, generado por el entendimiento progresivo del negocio y proyecto de los accionistas de la empresa. La identificación de requerimientos, desarrollo del producto, y pruebas entre las interacciones resulta en una finalización incremental para asegurar la supervivencia del valor del proyecto. Cada interacción tiene un enfoque diferente y desemboca en una porción estable del sistema general.

Esta metodología se basa en los principios de la aproximación al desarrollo ágil, formulada por la Alianza Ágil, y provee una guía de proceso que se enfoca en la gente y los cambios. Incluye oportunidades de aprendizaje mediante el uso de interacciones y evaluaciones en cada interacción.

¹ http://en.wikipedia.org/wiki/Microsoft_Solutions_Framework

3.3 Evaluación y elección de la metodología

Como hemos podido apreciar, existe una gran variedad de metodologías a escoger, basadas en diferentes características; las cuales son útiles siempre y cuando se escojan para el proyecto adecuado. Dado el caso, y por experiencia personal de quien desarrollará este proyecto en particular; las Metodologías Clásicas o Estructuradas no son un modelo acertado a seguir debido a que sus procesos de documentación son muy estrictos si se quiere seguir fielmente sus procesos. Dicha documentación, para el caso, y a criterio personal, sólo repercutirá en un tiempo adicional muy largo, por los detalles que se deben tomar en cuenta. Además de la documentación, se podría considerar que son procesos que alejan durante varios períodos de tiempo del cliente para el que se está buscando la solución; tiempo que bien podría ser utilizado en el análisis más profundo de las circunstancias que influyen en el ambiente de trabajo. Por otro lado, las Metodologías Ágiles, además de evitar generación excesiva de documentación en papel; se enfocan también en el aspecto humano, situación que es de gran conveniencia, puesto que en la empresa en cuestión este es el pilar de desarrollo. Además, estando en continuo contacto con el cliente y los usuarios, se puede mejorar las ideas, haciendo estos procesos más óptimos en lo que se refiere a tiempo y recursos puesto que es la experiencia cotidiana lo que va a regir en el trabajo, siendo éste un proceso que carece, o del que existen pocos modelos o plantillas a seguir. Profundizando entonces en estas nuevas tendencias de desarrollo, y otra vez, basados en la experiencia, la Programación Extrema resalta entre las demás opciones gracias a su similitud en procesos y objetivos a la manera en la que se ha trabajado en soluciones anteriores. Es por esto, que se ha decidido utilizar la **Programación Extrema**, considerando también que las bondades de las otras metodologías pudieran ser aprovechadas también en su momento. Esto ofrece la libertad necesaria para la exitosa consecución de los objetivos propuestos.

CAPÍTULO 4: HERRAMIENTAS

En este capítulo se hará un análisis y descripción de las distintas herramientas para el desarrollo de software, tanto en la parte de BASES DE DATOS como en las HERRAMIENTAS DE DESARROLLO, lo que en este caso serán los LENGUAJES DE PROGRAMACIÓN.

4.1 BASE DE DATOS

4.1.1 Sistemas Gestores de Bases de Datos (SGBD) o Motores de Bases de Datos

Los **Sistemas Gestores de Bases de Datos** son un tipo de software muy específico, dedicado a servir de interfaz entre la Base de datos y el usuario, y que son usados por las aplicaciones que la utilizan. Se componen de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

En los textos que tratan este tema, o temas relacionados, se mencionan los términos SGBD y DBMS, siendo ambos equivalentes, y acrónimos, respectivamente, de Sistema Gestor de Bases de Datos y *DataBase Management System*, su expresión inglesa.

Objetivos:

- **Abstracción de la información.** Los usuarios de los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario. Así, se definen varios *niveles de abstracción*.
- **Independencia.** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.

- **Redundancia mínima.** Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante. De entrada, lo ideal es lograr una redundancia nula; no obstante, en algunos casos la complejidad de los cálculos hace necesaria la aparición de redundancias.
- **Consistencia.** En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.
- **Seguridad.** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra asegurada frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.
- **Integridad.** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada.
- **Respaldo y recuperación.** Los SGBD deben proporcionar una forma eficiente de realizar copias de seguridad de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.
- **Tiempo de respuesta.** Lógicamente, es deseable minimizar el tiempo que el SGBD tarda en darnos la información solicitada y en almacenar los cambios realizados.

4.1.1.1 Sistemas Gestores de Bases de Datos utilizados en la actualidad

Como es de suponer, existen al momento varios gestores diferentes que son utilizados en diferentes formas. Sin embargo, si se quiere hacer una distinción de grupos; encontramos un factor muy importante e influyente al momento de la elección del gestor de bases de datos: el FACTOR ECONÓMICO. Es así que en base a este, encontramos que existen motores LIBRES y COMERCIALES:

SGBD libres

- PostgreSQL, Licencia BSD
- MySQL Licencia Dual, depende el uso.
- Firebird basada en la versión 6 de Interbase, Initial Developer's PUBLIC LICENSE Versión 1.0.

SGBD comerciales

- Microsoft Access
- Microsoft SQL Server
- MySQL
- Oracle
- IBM DB2 Universal Database (DB2 UDB)

SISTEMAS LIBRES

4.1.1.1 PostgreSQL

Es un servidor de base de datos relacional libre, liberado bajo la licencia BSD. Es una alternativa a otros sistemas de bases de datos de código abierto (como MySQL, Firebird y MaxDB), así como sistemas propietarios como Oracle o DB2.

Algunas de sus principales características son:

- Claves ajenas también denominadas Llaves ajenas o Llaves Foráneas (*foreign keys*).
- Disparadores (*triggers*).
- Vistas.
- Integridad transaccional.
- Acceso concurrente multiversión (no se bloquean las tablas, ni siquiera las filas, cuando un proceso escribe).
- Capacidad de albergar programas en el servidor en varios lenguajes.
- Herencia de tablas.
- Tipos de datos y operaciones geométricas.

4.1.1.1.2 MySQL

MySQL es uno de los Sistemas Gestores de bases de Datos (SQL) más populares desarrolladas bajo la filosofía de código abierto.

La desarrolla y mantiene la empresa MySQL AB pero puede utilizarse gratuitamente y su código fuente está disponible.

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas Web con contenido dinámico, justamente por su simplicidad; aquellos elementos faltantes fueron llenados por la vía de las aplicaciones que la utilizan.

Poco a poco los elementos faltantes en MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.




4.1.1.1.3 FireBird

Firebird es un sistema de administración de base de datos (o RDBMS) de código abierto, basado en la versión 6 de Interbase, cuyo código fue liberado por Borland en 1999. Su código fue re-escrito de C a C++.

Características

- Es multiplataforma, y actualmente puede ejecutarse en los sistemas operativos: Linux, HP-UX, FreeBSD, Mac OS, Solaris y Microsoft Windows.
- Ejecutable pequeño, con requerimientos de hardware bajos.
- Arquitectura Cliente/Servidor sobre protocolo TCP/IP y otros (embedded).
- Soporte de transacciones y claves foráneas.
- Es medianamente escalable.
- Buena seguridad basada en usuarios/roles.
- Diferentes arquitecturas, entre ellas el Firebird empotrado (*embedded server*) que permite ejecutar aplicaciones monousuario en ordenadores sin instalar el software Firebird.
- Bases de datos de sólo lectura, para aplicaciones que corran desde dispositivos sin capacidad de escritura, como CD-roms.
- Existencia de controladores ODBC, OLEDB y JDBC
- Controlador nativo para PHP
- Requisitos de administración bajos, siendo considerada como una base de datos libre de mantenimiento, al margen de la realización de copias de seguridad y restauraciones periódicas.
- Pleno soporte del estándar SQL-92, tanto de sintaxis como de tipos de datos.
- Completo lenguaje para la escritura de disparadores (*triggers*) y procedimientos almacenados.
- Capacidad de almacenar elementos BLOB (*binary large objects*).

Resumen de motores libres mas comunes

Motor	Desarrollador	Ultima version	Licencia	WebSite
	PostgreSQL Global Development Group	8.2.4 23-Abr-2007	BSD (Berkeley Software Distribution)	www.postgresql.org
	Firebird Project	2.0.1 12-Nov-2006	IDPL (Initial Developer's Public License)	www.firebirdsql.org
	MySQL AB	5.0.41 (Community server) 01-May-2007	GPL (General Public License), Licencia commercial o Acuerdo de suscripcion	www.mysql.com

SISTEMAS COMERCIALES

4.1.1.1.4 Microsoft Access

Microsoft Access es un sistema de gestión de bases de datos creado y modificado por Microsoft (DBMS) para uso personal o de pequeñas organizaciones. Es un componente de la suite Microsoft Office aunque no se incluye en el paquete básico. Para bases de datos de gran calibre (en cuanto a volumen de datos o de usuarios) es recomendable usar otros sistemas como Microsoft SQL Server, MySQL, Postgres u Oracle. Su principal función es ser una potente base de datos, capaz de trabajar en sí misma o bien con conexión hacia otros lenguajes de programación, tales como Visual Basic 6.0 o Visual Basic .NET. Pueden realizarse consultas directas a las tablas contenidas mediante instrucciones SQL. Internamente trae consigo el lenguaje Visual Basic for Application (VBA) el cual es similar en forma a VB6.

Permite el ingreso de datos de tipos: Numéricos, Texto, Fecha, Sí/No, OLE, Moneda, Memo y Boolean. Pueden desarrollarse aplicaciones completas basadas en Microsoft Access, pues trae consigo las herramientas necesarias para el diseño y desarrollo de formularios para el ingreso y trabajo con datos e informes para visualizar e imprimir la información requerida.

Su funcionamiento se basa en un motor llamado Microsoft Jet, y permite el desarrollo de pequeñas aplicaciones autónomas formadas por formularios Windows y código VBA (Visual Basic para Aplicaciones). Una posibilidad adicional es la de crear ficheros con bases de datos que pueden ser consultados por otros programas. Entre las principales funcionalidades de Access se encuentran:

- Crear tablas de datos indexadas.
- Modificar tablas de datos.
- Relaciones entre tablas (creación de bases de datos relacionales).
- Creación de consultas y vistas.
- Consultas referencias cruzadas.
- Consultas de acción (INSERT, DELETE, UPDATE).
- Formularios.
- Informes.
- Llamadas a la API de Windows.
- Interacción con otras aplicaciones que usen VBA (resto de aplicaciones de Microsoft Office, Autocad, etc.).
- Macros.

Además, permite crear frontends - o programa que muestra la interfaz de usuario - de bases de datos más potentes ya que es un sistema capaz de acceder a tablas externas a través de ODBC como si fueran tablas Access.

Es un software de gran difusión entre pequeñas empresas (PYMES) cuyas bases de datos no requieren de excesiva potencia, ya que se integra perfectamente con el resto de aplicaciones de Microsoft y permite crear pequeñas aplicaciones con unos pocos conocimientos de programación. Entre sus mayores inconvenientes figuran que no es multiplataforma, pues sólo está disponible para sistemas operativos de Microsoft, y que no permite transacciones. Su uso es inadecuado para grandes proyectos de software que requieren tiempos de respuesta críticos o muchos accesos simultáneos a la base de datos.

4.1.1.1.5 Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de bases de datos relacionales (SGBD) basada en el lenguaje SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

Entre sus características figuran:

- Soporte de transacciones.
- Gran estabilidad.
- Gran seguridad.
- Escalabilidad.
- Soporta procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo accesan a la información.
- Además permite administrar información de otros servidores de datos

Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños.

Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle o Sybase ASE.

Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQL Server incluye interfaces de acceso para la mayoría de las plataformas de desarrollo, incluyendo .NET.

Microsoft SQL Server, al contrario de su más cercana competencia, no es multiplataforma, ya que sólo está disponible en Sistemas Operativos de Microsoft.

4.1.1.1.6 MySQL

MySQL es un sistema de gestión de base de datos, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre bajo licencia GNU, pero, mediante licencia dual, también proporcionan MySQL bajo la licencia tradicional de software propietario para los casos en los que su uso sea incompatible con la licencia GPL.

Está desarrollado en su mayor parte en ANSI C.

MySQL está poseído y patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. La compañía desarrolla y mantiene la aplicación, vendiendo soporte y servicios, como lo hacen las empresas con software propietario, y contratan trabajadores alrededor del mundo que colaboran vía Internet.

Lenguajes de programación

Existen varias API's que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Delphi (via dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, REALbasic (Mac), FreeBASIC, y Tcl; cada uno de estos utiliza una API específica. También existe un interfaz ODBC, llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL.

4.1.1.1.7 Oracle

Oracle es un sistema de administración de base de datos (o RDBMS por el acrónimo en inglés de Relational Data Base Management System), fabricado por Oracle Corporation.

Se considera a Oracle como uno de los sistemas de bases de datos más completos, destacando su:

- Soporte de transacciones.
- Estabilidad.
- Escalabilidad.
- Es multiplataforma.

Su mayor defecto es su enorme precio, que es de varios miles de dólares (según versiones y licencias). Otro aspecto que ha sido criticado por algunos especialistas es la seguridad de la plataforma, y las políticas de suministro de parches de seguridad, modificadas a comienzos de 2005 y que incrementan el nivel de exposición de los usuarios. En los parches de actualización provistos durante el primer semestre de 2005 fueron corregidas 22 vulnerabilidades públicamente conocidas, algunas de ellas con una antigüedad de más de 2 años.

Aunque su dominio en el mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia del Microsoft SQL Server de Microsoft y de la oferta de otros RDBMS con licencia libre como PostgreSQL, MySQL o Firebird. Las últimas versiones de Oracle han sido certificadas para poder trabajar bajo Linux.

4.1.1.1.8 IBM DB2 Universal Database

Este gestor de propiedad de IBM esta considerado por algunos como el primer producto de bases de datos en utilizar el lenguaje SQL (Structured Query Language). Posee APIs para .NET CLI, Java, Python, Perl, PHP, Ruby, C++, C, REXX, PL/I, COBOL, RPG, FORTRAN, entre otros. También soporta integración con entornos de desarrollo Eclipse y Visual Studio .NET.

Entrega la posibilidad de implementar una solución de replicación de datos en cualquier tipo de ambiente distribuido o heterogéneo, pues permite enviar los datos a cualquier sitio para cubrir todos los requerimientos de una empresa, desde oficinas centrales a sucursales, usuarios móviles, proveedores, clientes y socios de negocios.

Gracias a su alcance global y de bajo costo, Internet puede ser una solución de negocios muy poderosa para realizar operaciones comerciales garantizando un nivel de seguridad y confiabilidad con sus servicios de autorización y autenticación integrados a redes y sistema operativos, soportando el network-computing utilizando Java y JDBC, incluyendo capacidad nativa de almacenar varios tipos de datos: alfanuméricos, video, imagen, audio y los definidos por el usuario.

DB2 y SQL Server tenían grandes compañías detrás con otros negocios que les permitió aguantar la política agresiva de Oracle. Recientemente IBM adquirió Informix con lo que el mercado de las bases de datos comerciales en UNIX (Linux) quedó entre IBM y Oracle.

Sus orígenes se remontan al año 1970, pero no es sino hasta el año 1983 que se empieza a vender con la versión 2.0. Finalmente en 1994 DB2 UDB (DB2 Universal Database) fue construido en base a dos productos incluidos en el DB2 de AIX, DB2 Common Server, que para propósitos generales incluía funciones avanzadas para el mercado de servidores de bases de datos, con soporte de hardware SMP y OLTP; y el DB2 Parallel Edition, que fue desarrollado para soportar aplicaciones de gran escala, como Data Warehousing y Data Mining.

Las plataformas que admite el producto son en principio OS/2, Windows NT y UNIX. Esta última versión del software del servidor, ha sido adaptado para funcionar también con OS/2 WARP, HP-UX, AIX, SCO-UNIX UE7, Windows 98, NT y Solaris.

Permite el manejo de objetos grandes (hasta 2 GB), la definición de datos y funciones por parte del usuario, el chequeo de integridad referencial, SQL recursivo, soporte multimedia: texto, imágenes, video, audio; queries paralelos, commit de dos fases, backup/recuperación on-line y offline.

Además cuenta con un monitor gráfico de performance el cual posibilita observar el tiempo de ejecución de una sentencia SQL y corregir detalles para aumentar el rendimiento.

Mediante los extensores se realiza el manejo de los datos no tradicionales, por ejemplo si se tiene almacenados los curriculums de varias personas, mediante este puedo realizar búsquedas documentos con los datos que me interesen sin tener que ver los CV uno por uno.

Esta capacidad se utiliza en sistemas de búsqueda de personas por huellas digitales, en sistemas información geográfica, etc.

En Internet con DB2 es posible acceder a los datos usando JDBC (tan potente como escribir directamente C contra la base de datos), Java y SQL (tanto el SQL estático, como complementa el SQL dinámico).

Plataformas host:

OS/390(MVS), VM & VSE, OS/400





Plataformas de servidor:

OS/2 Warp Server, Sinix, SCO Openserver, Windows NT, Aix, HP Ux, Solaris.

Plataformas Cliente:

OS/2, DOS, Sinix, SCO OpenServer, Windows 3.1/95/NT, Macintosh System 7, Aix, HP Ux, Solaris.

Resumen de motores comerciales

Motor	Desarrollador	Ultima version	Licencia	WebSite
Microsoft Office Access 	Microsoft	12.0.4518.1014 06-Nov-2006	EULA (End User License Agreement)	www.microsoft.com
Microsoft SQL Server™	Microsoft	2005 07-Nov-2005	EULA (End User License Agreement)	www.microsoft.com
	MySQL AB	5.0.41 (Community server) 01-May-2007	GPL (General Public License), Licencia comercial o Acuerdo de suscripcion	www.mysql.com
	Oracle Corporation	10g	EULA (End User License Agreement)	www.oracle.com
DB2 Universal Database 	IBM	DB2 9 DB2 Express	EULA (End User License Agreement)	www.ibm.com

4.1.2 Análisis y selección del motor de base de datos

En base a la experiencia obtenida en trabajos anteriores, el producto **Microsoft SQL Server** ha sido seleccionado por su seguridad y manejabilidad, a lo que podemos adicionar el peso que tiene el haber trabajado anteriormente en dicho software, sin mayores complicaciones, y con el debido conocimiento adquirido por la práctica.

En el aspecto económico, el Señor Gerente y Propietario de la empresa en estudio ha sido informado de que esta decisión involucra costos por licencias y/o mantenimiento; a lo que ha respondido satisfactoriamente aceptando la decisión y haciéndose responsable por cualquier rubro que pueda implicar el producto. Sin embargo, y ventajosamente, Microsoft ha liberado un producto de uso gratuito: **SQL Server Express with advanced services**, el cual posee ciertas restricciones en comparación con la versión de pago, pero que no afectan en el desempeño y administración del presente proyecto ya que las características del mismo son mínimas para las capacidades de la versión de libre uso.

Otra justificación para la elección de éste gestor de datos está basada en la herramienta de programación que será seleccionada y cuya justificación está en las páginas continuas: **C# de Microsoft Visual Studio .NET**. Al ser creado por el mismo grupo corporativo, dicho motor de bases de datos se presume con facilidades de interconexión y gestores de ayuda orientados naturalmente al lenguaje mencionado, cosa que al menos en la experiencia ha saltado fácilmente a la vista.

4.2 HERRAMIENTAS DE DESARROLLO

LENGUAJES DE PROGRAMACIÓN

Un lenguaje de programación es una técnica estándar de comunicación que permite expresar las instrucciones que han de ser ejecutadas en una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen un lenguaje informático.

Aunque muchas veces se usa lenguaje de programación y lenguaje informático como si fuesen sinónimos, no tiene por qué ser así, ya que los lenguajes informáticos engloban a los lenguajes de programación y a otros más, como, por ejemplo, el HTML.

Algunos lenguajes de programación:

C++

C#

Delphi

Java

JavaScript

PHP

PowerBuilder

Visual Basic

Visual Basic.NET

Visual Foxpro

4.2.1 C++

C++ es un lenguaje de programación, diseñado a mediados de los ochenta, por Bjarne Stroustrup, como extensión del lenguaje de programación C.

Es un lenguaje híbrido, que se puede compilar y resulta más sencillo de aprender para los programadores que ya conocen C. Actualmente existe un estándar, denominado ISO C++, al que se han adherido la mayoría de los fabricantes de compiladores más modernos. Existen también algunos intérpretes como ROOT ([enlace externo](#)).

Las principales características del C++ son: el soporte para programación orientada a objetos y el soporte de plantillas o programación genérica (templates). Por ende, se puede decir que C++ es un lenguaje que abarca tres paradigmas de la programación: La programación estructurada, la programación genérica y la programación orientada a objetos.

Además posee una serie de propiedades difíciles de encontrar en otros lenguajes de alto nivel:

- Posibilidad de redefinir los operadores (sobrecarga de operadores)
- Identificación de tipos en tiempo de ejecución (RTTI)

C++ está considerado por muchos como el lenguaje más potente, debido a que permite trabajar tanto a alto como a bajo nivel, sin embargo es a su vez uno de los que menos automatismos trae (obliga a hacerlo casi todo manualmente al igual que C) lo que "dificulta" mucho su aprendizaje.

En C++, "C++" significa "uno más de C" y se refiere a que C++ es una extensión de C.

4.2.2 C#

Es un lenguaje de programación orientado a objetos desarrollado por Microsoft y estandarizado, como parte de su plataforma .NET.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes. C# fue diseñado para combinar el control a bajo nivel de lenguajes como C y la velocidad de programación de lenguajes como Visual Basic.

C# significa, "do sostenido" (C corresponde a do en la terminología musical anglo-sajona). El símbolo # viene de sobreponer "++" sobre "++" y eliminar las separaciones, indicando así su descendencia de C++.

C#, como parte de la plataforma .NET, está normalizado por ECMA desde diciembre de 2001 (ECMA-334 "Especificación del Lenguaje C#").

Aunque C# forma parte de la plataforma .NET, ésta es una interfaz de programación de aplicaciones; mientras que C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma. Aunque aún no existen, es posible implementar compiladores que no generen programas para dicha plataforma, sino para una plataforma diferente como Win32 o UNIX.

En la actualidad existen los siguientes compiladores para el lenguaje C#:

- Microsoft Visual C# versión 2002, 2003 y 2005.
- Mono, es una implementación GPL de todo el entorno .NET desarrollado por Novell. Como parte de esta implementación se incluye un compilador de C#.
- Delphi 2005, de Borland Software Corporation.
- dotGNU Portable.NET, de la Free Software Foundation.

4.2.3 Delphi

Delphi es un entorno de desarrollo rápido de software diseñado para la programación de propósito general con énfasis en la programación visual. En Delphi se utiliza como lenguaje de programación una versión moderna de Pascal llamada Object Pascal. Es producido comercialmente por la empresa estadounidense Borland. En sus diferentes variantes, permite producir archivos ejecutables para Windows, GNU/Linux y la plataforma .NET.

En la actualidad, Borland parece haber decidido vender Delphi

El principal uso de Delphi es para el desarrollo de aplicaciones visuales y de bases de datos cliente-servidor y multicapas. Debido a que es una herramienta de propósito múltiple, se usa también para proyectos de casi cualquier tipo, incluyendo aplicaciones de consola, CGI y servicios del sistema operativo.

Delphi inicialmente sólo producía ejecutables binarios para Windows (Win16 y Win32).

En la actualidad da más posibilidades:

En la última versión Delphi 2006 incluye en el mismo entorno de desarrollo los lenguajes:

- Delphi para Win32,
- Delphi para .NET, y
- C# para .NET.

Existe una versión de Delphi para sistemas Unix y Linux, denominada Kylix (de la cual existe una versión gratuita, aunque limitada). Sin embargo Kylix fue abandonada por Borland en su versión 3.0.

4.2.4 Java

Java es una plataforma virtual de software desarrollada por Sun Microsystems, de tal manera que los programas creados en ella puedan ejecutarse sin cambios en diferentes tipos de arquitecturas y dispositivos computacionales ("Diferentes plataformas").

La plataforma Java consta de las siguientes partes:

- El lenguaje de programación, mismo.
- La máquina virtual de Java o JRE, que permite la portabilidad en ejecución.
- El API Java, una biblioteca estándar para el lenguaje.

Originalmente llamado OAK por los ingenieros de Sun Microsystems, Java fue diseñado para correr en computadoras incrustadas. Sin embargo, en 1995, dada la atención que estaba produciendo la Web, Sun Microsystems la distribuyó para sistemas operativos tales como Microsoft Windows.

El lenguaje mismo se inspira en la sintaxis de C++, pero su funcionamiento es más similar al de Smalltalk que a éste. Incorpora sincronización y manejo de tareas en el lenguaje mismo (similar a Ada) e incorpora interfaces como un mecanismo alternativo a la herencia múltiple de C++.

4.2.5 JavaScript

JavaScript es un lenguaje interpretado orientado a las páginas Web, con una sintaxis semejante a la del lenguaje Java.

El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que fabricó los primeros navegadores de Internet comerciales.

Apareció por primera vez en el producto de Netscape llamado Netscape Navigator 2.0.

Se utiliza en páginas Web, para realizar tareas y operaciones en el marco de la aplicación cliente.

Los autores inicialmente lo llamaron Mocha y más tarde LiveScript pero fue rebautizado como JavaScript en un anuncio conjunto entre Sun Microsystems y Netscape, el 4 de diciembre de 1995.

En 1997 los autores propusieron JavaScript para que fuera adoptado como estándar de la The European Computer Manufacturers' Association ECMA, que a pesar de su nombre no es europeo sino internacional, con sede en Ginebra. En junio de 1997 fue adoptado como un estándar ECMA, con el nombre de ECMAScript. Poco después también lo fue como un estándar ISO.

JScript es la implementación de ECMAScript de Microsoft, muy similar al JavaScript de Netscape, pero con ciertas diferencias en el modelo de objetos del navegador que hacen a ambas versiones con frecuencia incompatibles.

4.2.6 PHP

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios Web. El nombre es el acrónimo recursivo de "PHP: Hypertext Preprocessor" (inicialmente PHP Tools, o, Personal Home Page Tools), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios Web.

El fácil uso y la similitud con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas.

Debido al diseño de PHP, también es posible crear aplicaciones con una interfaz gráfica para el usuario (también llamada GUI), utilizando la extensión PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo, esta versión de PHP se llama PHP CLI (Command Line Interface).

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página Web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones Web muy robustas.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Windows y Mac OS X, y puede interactuar con los servidores de Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

El modelo PHP puede ser visto como una alternativa al sistema de Microsoft que utiliza ASP.NET/C#/VB.NET, a ColdFusion de la compañía Macromedia, a JSP/Java de Sun Microsystems, y al famoso CGI/Perl. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un compilador comercial llamado Zend Optimizer.

4.2.7 PowerBuilder

PowerBuilder es una herramienta de desarrollo de clase empresarial desarrollada por la empresa Sybase. PowerBuilder es orientada a objetos y permite el desarrollo de diferentes tipos de aplicaciones y componentes para arquitecturas cliente/servidor, distribuidas y Web.

PowerBuilder fue inicialmente desarrollada por la compañía Powersoft; la versión 1.0 se lanzó oficialmente al mercado en julio de 1991. Con ésta versión, los desarrolladores codificaban sus aplicaciones desde el "pintor de ventanas". Con la versión 2.0, lanzada al mercado en 1992, PowerBuilder incluyó herencia y encapsulación - por primera vez una ventana podía heredar el comportamiento de su antecesor. En 1995, cuando PowerBuilder se había afianzado de manera sólida dentro del mercado de herramientas de desarrollo, Sybase adquirió a Powersoft y a finales de 1996 introdujo la versión 5.0 de PowerBuilder. De ahí en adelante, Sybase ha continuado con el desarrollo de la herramienta, lanzando al mercado las versiones 6, 7, 8, 9 y 10, que reflejan la evolución de PowerBuilder de acuerdo a las tendencias del mercado, con mejoras en sus características y con el soporte a nuevas tecnologías como JSP, XML y Servicios Web. En abril de 2006, Sybase liberó la más reciente versión de la herramienta, PowerBuilder 10.5.

PowerBuilder incluye, dentro de su ambiente integrado de desarrollo, herramientas para crear la interfaz de usuario, generar reportes y tener acceso al contenido de una base de datos. PowerBuilder también incluye un lenguaje de programación llamado *Powerscript*, el cual es usado para especificar el comportamiento de la aplicación en respuesta a eventos del sistema o del usuario, tal como cerrar una ventana o presionar un botón.

Las aplicaciones desarrolladas con PowerBuilder se ejecutan exclusivamente en el sistema operativo Microsoft Windows, aunque cierto tipo de componentes que no incluyen una

interfaz de usuario (llamados componentes no visuales o NVOs) y que encapsulan sólo lógica de aplicación, se pueden ejecutar en otros sistemas operativos como Unix, usando la "Máquina Virtual de PowerBuilder" o PBVM incluida dentro del servidor de aplicaciones EAServer de Sybase.

Adicionalmente, PowerBuilder posee un objeto nativo para la gestión de datos llamado DataWindow, el cual puede ser usado para crear, editar y visualizar datos de una base de datos. Este objeto patentado por Sybase da al desarrollador un conjunto amplio de herramientas para especificar y controlar la apariencia y comportamiento de la interfaz de usuario, y también brinda acceso simplificado al contenido de la base de datos. Hasta cierto punto, el DataWindow libera al programador de las diferencias entre sistemas de gestión de base de datos de diferentes proveedores. Recientemente, Sybase introdujo al mercado el DataWindow .NET, un componente que extiende las bondades del DataWindow y acelera el rendimiento de aplicaciones en ambientes de desarrollo .NET.

4.2.8 Visual Basic

Visual Basic es un lenguaje de programación, así como el único IDE visual y compilador que existen para él, desarrollado por Alan Cooper para Microsoft. El lenguaje de programación es un dialecto de BASIC, con importantes añadidos. Su primera versión fue presentada en 1991 con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y en cierta medida también la programación misma.

Es un lenguaje de fácil aprendizaje (pero algunos son más complicados debido a que la persona que lo usa tenga conocimiento de este programa amplio) pensado tanto para programadores principiantes como expertos, guiado por eventos, y centrado en un motor de formularios poderoso que facilita el rápido desarrollo de aplicaciones gráficas.

Su principal innovación, que luego fue adoptada por otros lenguajes, fue el uso de un tipo de dll, llamado inicialmente vbx y posteriormente ocx, que permiten contener toda la funcionalidad de un control y facilitar su rápida incorporación a los formularios.

Su sintaxis, derivada del antiguo BASIC, ha sido ampliada con el tiempo al agregarse las características típicas de los lenguajes estructurados modernos. Se ha agregado una implementación limitada de la Programación Orientada a Objetos (los propios formularios y controles son objetos), aunque si que admite el polimorfismo mediante el uso de los Interfaces no admite la herencia. No requiere de manejo de punteros y posee un manejo muy sencillo de cadenas de caracteres. Posee varias bibliotecas para manejo de bases de datos, pudiendo conectar con cualquier base de datos a través de ODBC (Informix, DBase, Access, MySQL, SQL Server, PostgreSQL, etc.) a través de ADO.

Es utilizado principalmente para aplicaciones de gestión de empresas, debido a la rapidez con la que puede hacerse un programa que utilice una base de datos sencilla, además de la abundancia de programadores en este lenguaje.

El compilador de Microsoft genera ejecutables que requieren una DLL para que sus ejecutables funcionen, en algunos casos llamada MSVBVMxy.DLL (acrónimo de "Microsoft Visual Basic Virtual Machine x.y", siendo x.y la versión) y en otros VBRUNXXX.DLL ("Visual Basic Runtime X.XX"), que provee todas las funciones implementadas en el lenguaje. Además existen un gran número de bibliotecas (DLL) que facilitan el acceso a muchas funciones del sistema operativo y la integración con otras aplicaciones.

Derivados

Su derivado VBScript es el lenguaje predeterminado para Active Server Pages (ASP).

Además, una extensión propia del lenguaje llamada Visual Basic for Applications (VBA) permite codificar módulos (a veces llamados macros) para las aplicaciones de Microsoft Office.

Especialmente a partir de la versión 6 del lenguaje, se permite la interacción y generación de objetos remotos que puedan ser invocados desde páginas de scripts (concretamente las ASP, aunque no es imposible hacer un enlace empleando JSP), alojándolos en servidores de Web.

Visual Basic.NET, parte de .NET, es un lenguaje prácticamente equivalente en funcionalidades a C# (por ejemplo, no admite pseudo-punteros), añadiendo la capacidad de POO que sus anteriores versiones (como Visual Basic 6) no poseían: Herencia, polimorfismo.

4.2.9 Visual Basic.NET

Visual Basic.NET (VB.NET) es una versión de Visual Basic enfocada al desarrollo de aplicaciones .NET. El lenguaje de programación es Visual Basic, que apareció en el año 1991 como una evolución del QuickBasic que fabricaba Microsoft.

Es un lenguaje de programación orientado a objetos, y como novedades más importantes en la versión .NET, podemos citar la posibilidad de definir ámbitos de tipo, clases que pueden derivarse de otras mediante herencia, sobrecarga de métodos, nuevo control estructurado de excepciones o la creación de aplicaciones con múltiples hilos de ejecución, además de contar con la extensa librería de .NET, con la que es posible desarrollar tanto Windows Applications y Web Forms, así como un extenso número de clientes para bases de datos. Gracias a estas mejoras en lo que vendría siendo Visual Basic 7.0 los programadores de este lenguaje pueden desarrollar aplicaciones más robustas que en el pasado con una base sólida orientada a objetos.

Otras de sus características más importantes son:

- Diseño de controles de usuario para aplicaciones Windows y Web.
- Programación de bibliotecas de clase.
- Envío de datos vía documentos XML.
- Generación de reportes basados en Crystal Reports a partir de información obtenida de orígenes de datos (archivos de texto, bases, etc.)

En fin, una amplia gama de características nuevas que permiten diseñar aplicaciones escalables en pequeñas inversiones de tiempo.

4.2.10 Visual Fox Pro

Visual FoxPro es un sistema de administración de bases de datos relacionales. Posee un lenguaje propio, centrado en datos y orientado a objetos muy robusto.

Visual FoxPro ofrece a los desarrolladores un conjunto de herramientas para crear aplicaciones de bases de datos para el escritorio, entornos cliente/servidor, tablet PC o la Web.

Entre sus características tenemos:

- Capacidades poderosas y muy veloces para el manejo de datos nativos y remotos.
- Flexibilidad para crear todo tipo de soluciones de bases de datos.
- Lenguaje de programación Orientado a objetos.
- Utilización de sentencias SQL en forma nativa.
- Poderoso manejo de vistas y cursores y control completo de estructuras relacionales.
- Su propio gestor de base de datos incorporado. Sin embargo, también puede conectarse con servidores de base de datos, tales como Oracle o SQL Server.
- Cuenta con un motor de generación de informes renovado y muy flexible para soluciones más robustas.

La última versión liberada es la 9.0. La próxima versión, 'Sedna', será un poderoso y completo lenguaje que permitirá al producto interactuar aún más con VisualStudio.net, SQLServer2005, SQLEXPRESS2005 y Office12, Windows Vista.

Los precursores de Visual FoxPro fueron diferentes versiones de dBASE, FoxBase, y FoxPro.

4.2.11 Resumen de lenguajes de programación

Lenguaje	Desarrollador	Fecha de aparición	Paradigma (Estilo de programación)
C++	Bjarne Stroustrup (Bell Labs)	1985	Generico , orientado a objetos, procedimental
C#	Microsoft	2001	Estructurado, imperativo, orientado a objetos
Delphi	Borland	1995	Orientado a objetos
Java	Sun Microsystems	1995	Orientado a objetos, estructurado, imperativo
JavaScript	Netscape Communications Corporation, Mozilla Foundation	1995	Multi-paradigma
PHP	The PHP Group	1994	Imperativo, orientado a objetos, funcional
PowerBuilder	Powersoft, Sybase	1991	Orientado a objetos, funcional
Visual Basic	Microsoft	1991	Manejo de eventos
Visual Basic .NET	Microsoft	2001	Estructurado, imperativo, oreintado a objetos
Visual Fox Pro	Microsoft	1984	Orientado a objetos

4.2.12 Análisis y selección de la herramienta de programación

Luego de conocer las distintas herramientas disponibles en el mercado, y su debido análisis con personas que han usado algunas de ellas, se ha optado por el uso de **C# de Microsoft.NET**; debido a su agilidad y facilidad de uso. Además, se desea utilizarlo ya que este lenguaje no es conocido por quién va a realizar esta disertación de grado, pero al generar un producto con el mismo se adquirirá los conocimientos y experiencia suficientes para ser competitivo en el mercado actual. Adicionalmente, el entorno de trabajo de quien realiza este trabajo está involucrado mayoritariamente en dicha plataforma de desarrollo, lo que significa que al término del presente proyecto los conocimientos y experiencias adquiridas representan inmediatamente un agregado de competitividad y valoración de trabajo en la empresa para la cual se prestan actualmente los servicios.

4.3 Resumen de selección de herramientas para el desarrollo del proyecto

Metodologia de desarrollo	Programacion extrema
Motor de base de datos	Microsoft SQL Server 2000
Lenguaje de programacion	Microsoft C# .NET

CAPÍTULO 5: DESARROLLO

5.1 Análisis y descripción de la empresa AZAHARES

Azahares es un negocio de alquiler de productos y venta de servicios relacionados al Catering; es decir:

- Alquiler de cristalería y vajilla.
- Alquiler de mesas, sillas, carpas, etc.
- Alquiler de vestidos de novia.
- Alquiler de trajes de etiqueta.
- Alquiler de ropa formal para fiestas.
- Servicio de recepción de banquetes.
- Servicio de cocina tipo buffet para llevar.

Existen dos locales de atención al público, ubicados al sur de la ciudad de Quito, cada uno con tres empleados y un administrador.

Se trabaja de la siguiente manera:

- Al acercarse un cliente a uno de los locales, hace una consulta sobre la disponibilidad de uno o varios productos y/o servicios.
- La persona que recibe la consulta se encarga de verificar si hay disponibilidad en de stock en inventario, el cual decrece conforme se sigan haciendo reservaciones.
- En caso de existir la posibilidad, se explica al cliente en detalle los términos de un CONTRATO DE ALQUILER que será celebrado por ambas partes (el cliente y un representante de AZAHARES); los costos, y la garantía que debe ser entregada mientras los productos estén en poder del cliente.

- Para hacer definitiva la reservación, el cliente deberá abonar un valor en efectivo o cheque (dependiendo el caso) de por lo menos el 50% (Cincuenta por ciento) del valor total del contrato.
- Una vez celebrado el contrato de alquiler se procede a realizar una FACTURA con valor del alquiler mas el valor del Impuesto al Valor Agregado (I.V.A.) sobre el valor mencionado si es que el cliente lo requiere, de lo contrario el proceso de facturación es realizado al final de todo el ejercicio.
- Ambos documentos: contrato de alquiler y factura, se harán por duplicado, con el fin de que ambas partes conserven un documento de respaldo.
- La garantía puede ser entregada hasta con un día de anticipación, y será detallada al momento de recepción en el contrato celebrado y en la copia del mismo.
- El encargado del contrato realizado procederá a registrar el requerimiento de productos, de tal forma que en una próxima consulta al stock el material reservado conste ya como no disponible, y así evitar un alquiler de productos no disponibles en stock.
- De igual manera, se registrará en una hoja de control diario el valor abonado en la transacción.
- Al final del día, la persona responsable se encargará de consolidar el valor total de dinero ingresado en el día, verificando que el dinero registrado en la hoja de control sea el mismo que el existente en efectivo, el mismo que se hará entrega al administrador del local o en su caso al propietario del establecimiento.
- Cuando ha llegado el día convenido para la entrega de los productos, se cancelará el valor total del contrato de ser necesario, y además se procederá a verificar la entrega de la garantía por parte del cliente antes de despachar los mismos.
- Una sola persona será responsable de despachar los artículos, pudiendo pedir auxilio a otros trabajadores para la tarea; pero siempre llevando el control de lo entregado, esto

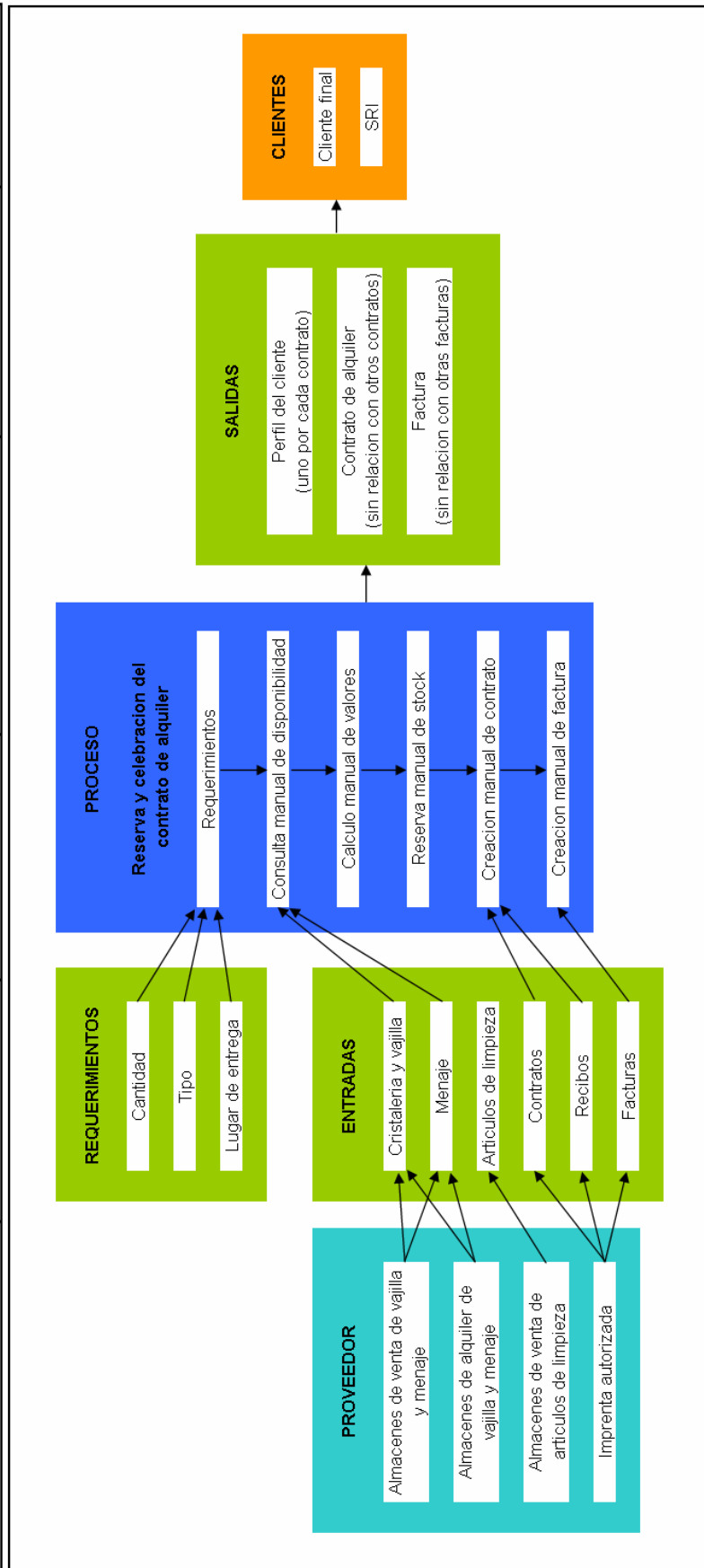
es: mientras se despacha, y al momento de entrega definitiva al cliente cuando se hace constar la entrega ítem por ítem para proceder a una firma de conformidad del cliente.

- Pasado el plazo de alquiler, el cliente se acercará a devolver los artículos conforme a lo estipulado en el contrato, mientras que el personal a cargo los recibirá controlando el estado de las piezas y su limpieza, debiendo reportar en caso de encontrar alguna novedad a la persona responsable del contrato, el cual registrará los rubros respectivos, para el posterior cobro al cliente.
- Cuando el cliente ha devuelto todo lo que utilizó y ha pagado en caso de haber roturas, faltantes o mal aseo; se devolverá la garantía y posteriormente el cliente firmará para constancia de la devolución de dicha garantía.
- Semanalmente se hará el control de los pedidos por reservar y de aquellos que se debían devolver para notificar al cliente la demora.
- Mensualmente el Administrador de cada local hará un resumen de todas las hojas de control diario por un lado, y de los contratos de alquiler por el otro, para verificar que todas las transacciones fueron correctamente registradas, y a su vez; para decrementar en el inventario los artículos que se registraron dañados o faltantes.
- De igual manera, el Administrador deberá hacer el correspondiente ingreso de nueva mercadería, para incremento en los inventarios.

El siguiente es un diagrama SIPOC que resume lo anteriormente dicho:

DIAGRAMA SIPOC

PROVEEDORES	ENTRADAS	REQUERIMIENTOS	PROCESOS	SALIDAS	CLIENTES
* Almacenes de venta de cristalería y vajilla * Almacenes de venta de menaje * Almacenes de alquiler de cristalería y vajilla * Almacenes de alquiler de menaje * Almacenes de venta de artículos de limpieza * Imprenta autorizada	* Cristalería y vajilla * Menaje * Artículos de limpieza * Combustible para transporte * Recibos, contratos y facturas	* Cantidad * Tipo * Lugar de entrega	* Reserva y celebración de contrato de alquiler * Entrega de mercadería * Recepción de mercadería	* Perfil de cliente * Contrato de alquiler * Factura * Recibo de pago por daños	* Cliente final * SRI



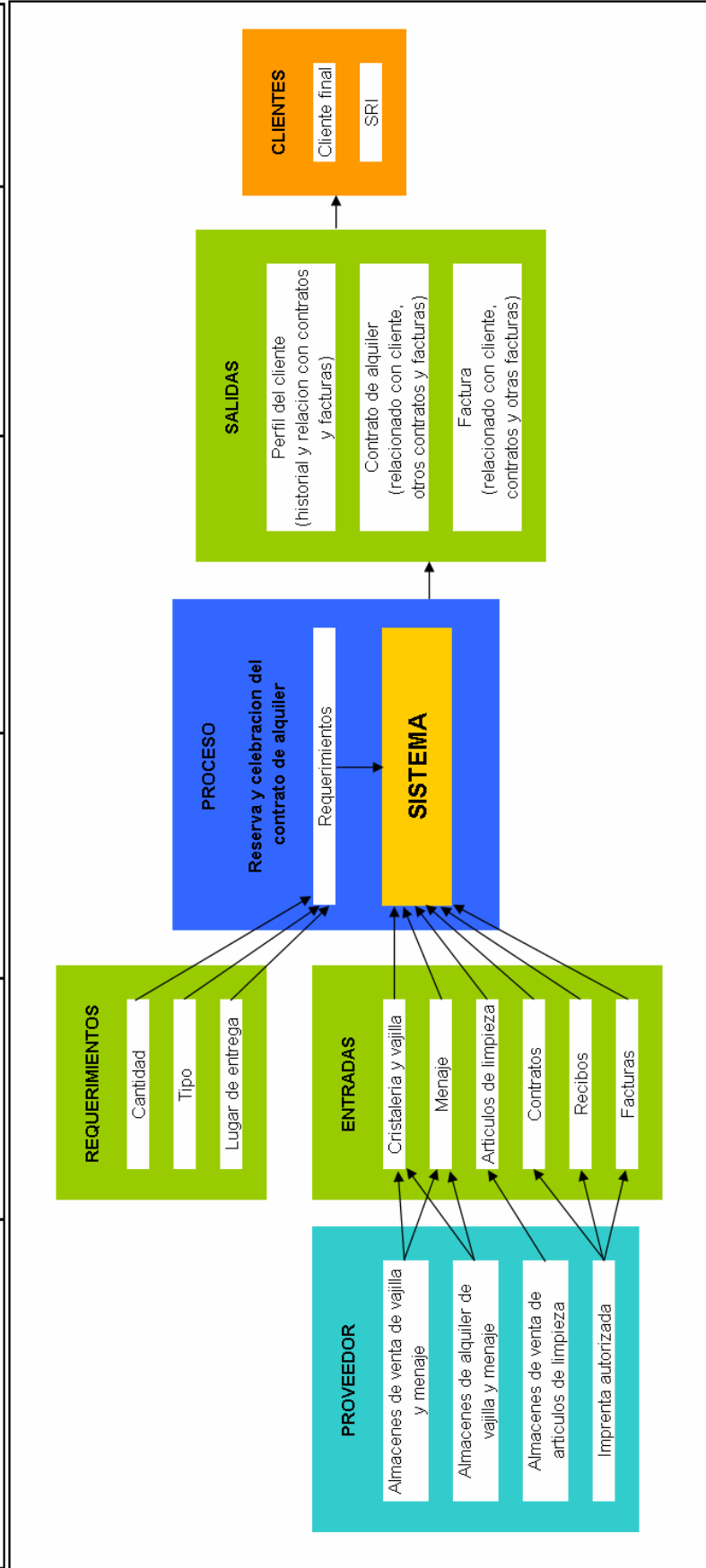
Técnicamente hablando, el sistema es efectivamente requerido puesto que el control que se pretende realizar es altamente complicado en la situación actual; y será muy difícil continuarlo manualmente si se desea cubrir las expectativas de crecimiento fijadas por la empresa en cuestión.

Los problemas que se generan en la actualidad son entre otros varios:

- Tiempos excesivamente altos de atención al cliente.
- Alta susceptibilidad de error en la reserva de artículos, sobretodo en el alquiler de mercadería que ya no está disponible.
- Facturación manual lenta e ineficiente, con probabilidad de error.
- Omisión de cobro de valores al cliente, ya sea por alquiler o reposición y/o lavado de productos.
- Control de dinero de caja manual, es decir; suma de valores factura por factura y las posibles pérdidas económicas que esto puede generar.
- Facilidad para el empleado de realizar operaciones deshonestas.
- Seguimiento manual de clientes.
- Falta de control exacto en inventario.

El siguiente es un diagrama SIPOC en el cual se incluye el sistema como herramienta, y se puede apreciar la mejora en el proceso.

DIAGRAMA SIPOC CON EL SISTEMA COMO HERRAMIENTA					
PROVEEDORES	ENTRADAS	REQUERIMIENTOS	PROCESOS	SALIDAS	CLIENTES
* Almacenes de venta de cristalería y vajilla * Almacenes de venta de menaje	* Cristalería y vajilla * Menaje * Artículos de limpieza	* Cantidad * Tipo * Lugar de entrega	* Reserva y celebración de contrato de alquiler * Entrega de mercadería * Recepción de mercadería	* Perfil de cliente * Contrato de alquiler * Factura * Recibo de pago por daños	* Cliente final * SRI
* Almacenes de alquiler de cristalería y vajilla * Almacenes de alquiler de menaje * Almacenes de venta de artículos de limpieza * Imprenta autorizada	* Combustible para transporte * Recibos, contratos y facturas				



CAPÍTULO 6: CONCLUSIONES

- El manejo de estándares de codificación representa un ahorro considerable de tiempo, es decir; elaborar pequeños templates para mensajes y/o reportes que se generan a menudo, como por ejemplo errores en ejecución. Cada vez que se llega a una posibilidad de éstos, se acude a éste pequeño template y se copia, para simplemente cambiar el número de error. Dicho template contiene al menos el nombre del form y un número de mensaje; de esta manera, cuando aparecen errores sabemos exactamente en donde encontrarlos para corregirlos.
- El uso de la NOTACIÓN HÚNGARA (inicial en minúscula con el tipo de dato de acuerdo a un estándar, seguido de el nombre de variable lo mejor detallada) es también una gran ventaja sobre todo al momento de retomar las variables en líneas de código alejadas de las declaraciones de las mismas.
- El abundante uso de comentarios es uno de los mejores hábitos de programación, no sólo para el creador del código, quien después de mucho tiempo puede retomar muy fácilmente el punto analizado, sino que presenta la gran ventaja de que cualquier colega que deba modificar el código ahorrará una gran cantidad de tiempo en entender su tarea al tener una guía detallada del punto en el que se encuentra.
- El mayor detalle en las tablas de base de datos otorga un gran potencial de crecimiento de tareas del sistema, pero sobretodo; los controles que se generan al registrar en los movimientos: fechas, horas y más importante el operador responsable de las transacciones, de esta manera se evitan vacíos en la definición de responsabilidades.
- El uso de la herramienta mejora dramáticamente los tiempos de atención al cliente
- Los cálculos de valores ya no son susceptibles de errores o manipulaciones
- La información económica está por ende también libre de errores y suspicacias por parte de los operarios
- Se puede concluir finalmente, que es una decisión muy acertada la implementación del sistema en cuestión, en base a los resultados obtenidos.

CAPÍTULO 7: RECOMENDACIONES

- El mejor conocimiento de la herramienta obtiene los mejores tiempos y resultados: cuando la herramienta es nueva, o poco familiar, lo mejor que se puede hacer es consultar con la documentación de la misma, o mejor aún con colegas que conozcan del tema. En experiencia propia, en más de una ocasión alguien conocía una mejor manera de obtener el resultado deseado, e incluso mejores resultados.
- Realizar un estudio de lo requerido en el campo de trabajo, es decir, acudir al entorno donde el sistema será ejecutado, analizar lo que sucede mientras los operarios realizan su trabajo cotidiano, mientras más sea el tiempo mejor idea se tiene de los problemas a solucionar.
- Hacer las preguntas necesarias: uno de los mayores errores en este tipo de trabajos es ASUMIR ciertas circunstancias, pues muchas de esas asunciones son incluso totalmente erradas. Solamente el operador/cliente sabe qué es lo que sucede exactamente en ciertas circunstancias y cómo se debe actuar ante ellas.
- Darse el tiempo necesario y prestar mucha atención al momento de estructurar la base de datos. El objetivo ideal es que la misma debería estar lista antes de iniciar el proceso de codificación.
- Ya en el sistema: se recomienda al cliente que tome ventaja de la información almacenada en las tablas de datos para:
 - Realizar análisis financieros detallados, para encontrar índices de utilidad/pérdida.
 - Generar cuadros estadísticos económicos
 - Generar cuadros estadísticos de temporadas altas y bajas de movimiento comercial, pérdidas, ingresos emergentes de stock, entre otros.
- Agregar al sistema herramientas de EGRESOS DE CAJA
- Agregar al sistema herramientas de NOTAS DE CRÉDITO
- Agregar al sistema herramientas de CONTROL DE TIEMPOS, para detectar cuales son los procesos a mejorar en caso de ser necesario.

CAPÍTULO 7: MEDIOS DE CONSULTA / BIBLIOGRAFÍA

<http://www.pymes-ecuador.com/nuevo-sitio/inicio.htm>
LEY DE COMPAÑIAS, Codificación 000, Registro Oficial 312 de 5 de Noviembre de 1999
http://www.sri.gov.ec/pages/guia_contribuyente/guia_facturacion/guia_facturacion.html
Ley de Propiedad Intelectual, Ley No. 83, R. O. No.320 de 19 de mayo de 1998
<http://es.wikipedia.org/wiki/Negocio>
medioambiente.geoscopio.com/medioambiente/temas/profit/terminos.php
<http://es.wikipedia.org/wiki/Empresa>
<http://www.monografias.com/trabajos21/clasificacion-empresas/clasificacion-empresas.shtml#>
<http://www.monografias.com/trabajos15/ingenieria-software/ingenieria-software.shtml>
http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/heterodox.asp
http://es.wikipedia.org/wiki/Modelo_en_cascada
<http://www.itlp.edu.mx/publica/tutoriales/analisis/24.htm>
<http://carolina.terna.net/ingsw2/Datos/Cascada-ModeloV.doc>
<http://www.itver.edu.mx/comunidad/material/ing-software/unidad1.ppt>
<http://www.angelfire.com/ult/lupa/metodologia/metodologiaagil.html>
<http://www.acis.org.co/index.php?id=551>
http://curso-sobre.berlios.de/curso/trab/cjacuna/Open_Source_y_Met_Agiles.pdf
http://en.wikipedia.org/wiki/Agile_software_development
<http://www.agilemodeling.com/principles.htm>
http://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema
<http://www.deister.es/deisterwww/company/xprograming.html>
<http://en.wikipedia.org/wiki/SCRUM>
http://en.wikipedia.org/wiki/Agile_Modeling
http://en.wikipedia.org/wiki/Dynamic_Systems_Development_Method
http://en.wikipedia.org/wiki/Microsoft_Solutions_Framework

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
ESCUELA DE SISTEMAS**

PLAN DE DISERTACIÓN DE GRADO

TEMA: Desarrollo e implementación de un sistema de control de alquiler de menaje, inventarios y facturación de la empresa “AZAHARES”.

ALUMNO: David Rueda O.

RESUMEN

La automatización es definitivamente una necesidad de las empresas que se encuentran en desarrollo, no sólo porque representa una ventaja competitiva, sino también por sus grandes beneficios en reducción de costos, además de tiempos de atención y respuesta con los clientes. Esta es una realidad que no se puede ocultar, y mucho menos con el apogeo de sistemas cada vez más eficientes, que cumplen papeles muy importantes en el comercio en general, tan importantes que simplemente no pueden ser reemplazados por el trabajo humano simplemente.

Es debido a esta situación, que la empresa “AZAHARES”, dedicada al catering desde hace más de 25 años; ha decidido a través de su Gerente General Xavier Rueda; enfrentar los nuevos desafíos comerciales mediante el desarrollo e implantación de un **sistema de control de alquiler de menaje, inventarios y facturación**; el mismo que fue propuesto como una solución que beneficia a ambas partes, tanto como al propietario como para el estudiante que realizará el desarrollo, y que utilizará el mismo como DISERTACIÓN DE GRADO para lograr obtener el título de Ingeniero en Sistemas.

El presente trabajo consta de un estudio del Marco Legal y Teórico concerniente; luego del cual se describen las posibles herramientas a utilizar con una evaluación de las mismas para una adecuada selección.

Finalmente la implementación práctica del Sistema en sí en basado en los puntos descritos anteriormente.

JUSTIFICACIÓN

Como ya se mencionó, una de las razones principales para esta inclinación, es la del lazo familiar, que, sin dejar de lado el aspecto profesional, impulsa a realizar un trabajo que beneficie a aquellos que son parte de su vínculo familiar más cercano. Sería por tanto este trabajo muy provechoso para el estudiante en dos sentidos: culminar satisfactoriamente la carrera de ingeniería, y a su vez ayudar y ayudarse en el desarrollo comercial de la empresa familiar.

Técnicamente hablando, el sistema es efectivamente requerido puesto que el control que se pretende realizar es altamente complicado en la situación actual; y será muy difícil continuarlo manualmente si se desea cubrir las expectativas de crecimiento fijadas por la empresa en cuestión.

Los problemas que se generan en la actualidad son entre otros varios:

- Tiempos excesivamente altos de atención al cliente.
- Alta susceptibilidad de error en la reserva de artículos.
- Facturación manual lenta e ineficiente, con probabilidad de error.
- Control de dinero de caja manual, es decir; suma de valores factura por factura.
- Facilidad para el empleado de realizar operaciones deshonestas.
- Seguimiento manual de clientes.

OBJETIVO GENERAL

Desarrollo e implementación de un **sistema de control de alquiler de menaje, inventarios y facturación**, para la optimización de recursos y tiempos; control preciso monetario y de inventarios, y minimización de errores en cada uno de los procesos.

OBJETIVOS ESPECÍFICOS

- Adquirir conocimientos nuevos y experiencia suficiente en el campo de la Programación Extrema y sus herramientas; además de profundizar en el manejo del lenguaje de programación a utilizar, así como el trabajo en Bases de Datos.
- Desarrollar un prototipo que cubra las necesidades de mercado de la empresa en la que se va a trabajar.
- Omitir la intervención humana en cálculos monetarios para evitar cualquier tipo de error o manipulación malintencionada.
- Control estricto y sin errores del stock de productos, para evitar la sobre-oferta de los mismos, y sus respectivos problemas.
- Automatizar procesos tales como facturación, control de caja, control de inventario, entre otros; para tener resultados altamente confiables en el menor tiempo posible.
- Evitar las pérdidas de producto con un riguroso seguimiento de los problemas encontrados con los clientes y sus pedidos.

ALCANCE

El presente trabajo concluirá con la aceptación del sistema en la empresa “Azahares”, con sus respectivas pruebas en un prototipo del producto.

METODOLOGIA

- Programación Extrema (XP).
- Arquitectura Cliente-Servidor - 2 Capas.
- Uso de Internet para consultas e investigación.
- Elección del lenguaje de programación y del motor de bases de datos a utilizar en base al análisis de requerimientos.
- Asesoría de profesionales expertos en las herramientas seleccionadas en el punto anterior.

CRONOGRAMA

Se estima que el prototipo a realizar tomará aproximadamente cinco meses en ser finalizado e implementado, a partir de la fecha de inicio. Para tales efectos, se ha realizado un Diagrama de Gantt con la previsión del tiempo requerido para cada tarea; dicho diagrama se encuentra como Anexo al presente documento.

TEMARIO

GRUPO I: Marco Legal

1. Legislación

- 1.1. Definición legal
 - 1.1.1. Definición de la Ley de Compañías del Ecuador
 - 1.1.2. Otras definiciones.
- 1.2. Tipos de PYMES
- 1.3. Facturación
 - 1.3.1. Aspectos importantes de los comprobantes de venta
 - 1.3.2. Tipos de comprobantes de venta y de retención autorizados
 - 1.3.3. Emisión de los comprobantes de venta
 - 1.3.3.1. Facturas
 - 1.3.3.2. Notas o boletas de venta
 - 1.3.3.3. Tickets o vales emitidos por máquinas registradoras
 - 1.3.4. Información pre-impresa e información no impresa
 - 1.3.4.1. Información pre-impresa (Todos los casos)
 - 1.3.4.2. Información no pre-impresa
 - 1.3.4.2.1. Información no impresa para el caso de facturas
 - 1.3.4.2.2. Información no impresa para el caso de notas o boletas de venta
 - 1.3.4.2.3. Información sobre la transacción en liquidaciones de compras de bienes o prestación de servicios
 - 1.3.4.2.4. Información en tiquetes o vales emitidos por máquinas registradoras
- 1.4. Infracciones al reglamento de facturación
- 1.5. Facultad de la Superintendencia de Compañías
- 1.6. Marcas, patentes y Ley de Propiedad Intelectual

GRUPO II: Marco Teórico

2. Empresas

- 2.1. Definición
- 2.2. Características
 - 2.2.1. Elementos
 - 2.2.1.1. Factores activos
 - 2.2.1.2. Factores pasivos
 - 2.2.1.3. Organización
- 2.3. Buen gobierno empresarial
- 2.4. Clasificación
 - 2.4.1. Por sector económico
 - 2.4.2. Por tamaño
 - 2.4.3. Por el origen del capital
 - 2.4.4. Por la explotación y conformación de su capital
 - 2.4.5. Por el pago de impuestos
 - 2.4.6. Por la función social
- 2.5. Clasificación de la empresa Azahares según los puntos documentados

3. Metodologías de desarrollo

3.1. Definición

3.2. Tipos

3.2.1. Metodologías estructuradas

3.2.1.1. Modelo en cascada

3.2.1.2. Modelo en espiral

3.2.1.3. Modelo en V

3.2.1.4. Modelo interactivo

3.2.1.5. Modelo incremental

3.2.2. Metodologías Ágiles

3.2.2.1. Programación extrema

3.2.2.2. Scrum

3.2.2.3. Modelamiento ágil

3.2.2.4. Método de desarrollo de sistemas dinámicos

3.2.2.5. Entorno de trabajo para soluciones Microsoft

3.3. Evaluación y elección de la metodología

4. Herramientas

4.1. Base de Datos

4.1.1. Definición de Bases de Datos

4.1.2. Tipos de Bases de datos

4.1.2.1. Bases de datos jerárquicas

4.1.2.2. Bases de datos de red

4.1.2.3. Bases de datos relacionales

4.1.2.4. Bases de datos orientadas a objetos

4.1.2.5. Bases de datos documentales

4.1.2.6. Bases de datos deductivas

4.1.2.7. Gestión de Bases de datos distribuida

4.1.3. Sistema Gestores de Bases de Datos (SGBD) o Motores de Bases de Datos

4.1.3.1. Sistemas Gestores de Bases de Datos utilizados en la actualidad

Sistemas Libres

4.1.3.1.1. PostgreSQL

4.1.3.1.2. MySQL

4.1.3.1.3. FireBird

4.1.3.1.4. Adaptive Server Enterprise (SyBase ASE)

Sistemas Comerciales

4.1.3.1.5. dBase

4.1.3.1.6. Fox Pro

4.1.3.1.7. Microsoft Access

4.1.3.1.8. Microsoft SQL Server

4.1.3.1.9. MySQL

4.1.3.1.10. Oracle

4.1.3.1.11. Adaptive Server Anywhere (SyBase ASA)

4.1.4. Análisis y Selección del motor de bases de datos

4.2. Herramientas de Desarrollo Lenguajes de programación

4.2.1. Definición

Productos

- 4.2.1.1. ABAP
- 4.2.1.2. ActionScript
- 4.2.1.3. Afnix
- 4.2.1.4. ASP
- 4.2.1.5. Basic
- 4.2.1.6. C++
- 4.2.1.7. C#
- 4.2.1.8. COBOL
- 4.2.1.9. Delphi
- 4.2.1.10. FORTRAN
- 4.2.1.11. Java
- 4.2.1.12. JavaScript
- 4.2.1.13. MAGIC
- 4.2.1.14. PASCAL
- 4.2.1.15. PHP
- 4.2.1.16. PowerBuilder
- 4.2.1.17. Prolog
- 4.2.1.18. Self
- 4.2.1.19. Simula
- 4.2.1.20. SmallTalk
- 4.2.1.21. TI-Basic
- 4.2.1.22. VBA
- 4.2.1.23. Visual Basic
- 4.2.1.24. Visual Basic.NET
- 4.2.1.25. Visual Fox Pro

4.3. Análisis y Selección de la herramienta

5. Desarrollo

- 5.1. Análisis y descripción de la empresa
- 5.2. Análisis de Requerimientos
- 5.3. Modelo Relacional
- 5.4. Modelo Físico
- 5.5. Diagrama de Casos de Uso
- 5.6. Prototipos
 - 5.6.1. Generación del prototipo
 - 5.6.2. Evaluación
 - 5.6.3. Cambios
- 5.7. Implementación
- 5.8. Pruebas

6. Conclusiones

7. Recomendaciones