



**INSTITUTO SUPERIOR MINERO METALURGICO**

**“Dr. Antonio Núñez Jiménez”**

**Facultad Metalurgia – Electromecánica**

**Carrera de Ingeniería Informática**

# **TRABAJO DE DIPLOMA**

**Para optar por el Título de Ingeniero Informático**

**SISTEMA DE GESTION INTEGRAL DE LA EMPRESA EMPLEADORA DEL  
NIQUEL - MODULO GESTION DE CONTRATOS DE COMPRAS.**

Autor: Jorge Mario Basulto Aguilera.

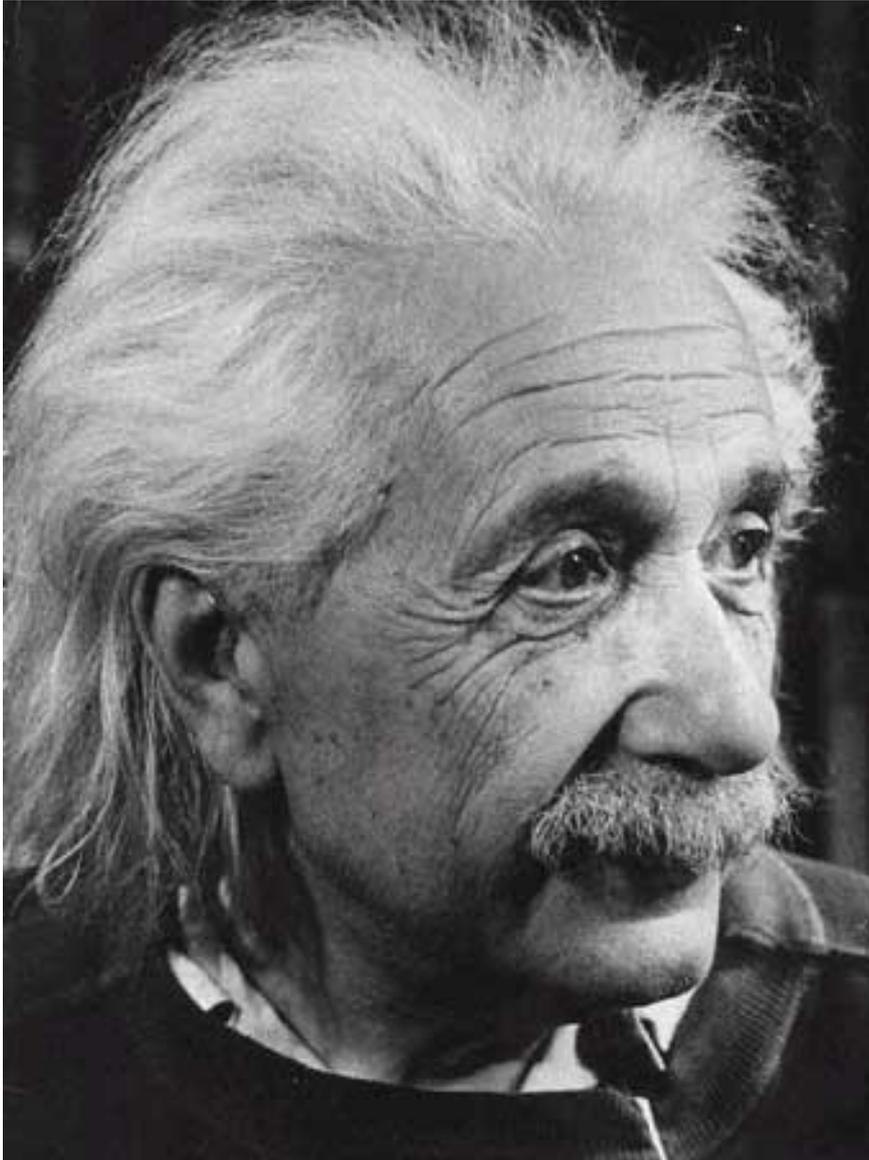
Tutores: Ing: Yodexy Mosqueda Naranjo.

Ing: Edgar Núñez Torres.

Moa, Cuba

Junio 2010

## *Pensamiento.*



*“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber.”*

*Albert Einstein*

## *Agradecimientos.*

*A mi familia, y muy especialmente a mi madre por tanto amor y apoyo en todo momento.*

*A mis amigos; en especial a Roli, Edgar, Leyanis, Bozena, Osmany y Junier, por darme apoyo, comprenderme y por la preocupación que han tenido por mí.*

*A todos aquellos que de una forma u otra han contribuido con mi formación profesional y la realización de este trabajo.*

*A mis tutores, Ing. Yodexy Mosqueda Naranjo e Ing. Edgar Núñez Torres por su inestimable ayuda y guía en todo momento*

*A todos, gracias...*

*Jorge Mario Basulto Aguilera.*

## *Dedicatoria.*

*A mis padres, a mi familia y a mis amigos...*

*Jorge Mario Basulto Aguilera*

## DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Empresa Empleadora del Níquel para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los 24 días del mes Junio del año 2010.

Jorge Mario Basulto Aguilera

Nombre completo del autor

Ing. Yodexy Mosqueda Naranjo

Nombre completo del primer tutor

Ing. Edgar Núñez Torres

Nombre completo del segundo tutor

## OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El trabajo de diploma, titulado ***Sistema de Gestión Integral de la Empresa Empleadora del Níquel - Módulo Gestión de Contratos de Compras***, fue realizado en nuestra entidad Empresa Empleadora del Níquel "EMPLENI". Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

- Totalmente
- Parcialmente en un \_\_\_\_%.

Los resultados de este informe técnico le reportan a esta entidad los beneficios siguientes (cuantificar):

- ❖ Protege la utilización de las partidas aprobadas en el presupuesto de la empresa para el año, buscando su adecuada administración ante riesgos potenciales y reales que los pueden afectar.
- ❖ Garantiza la eficacia, eficiencia y economía en todas las operaciones del sistema de contratación, promoviendo y facilitando la correcta ejecución de los montos aprobados para la ejecución de las compras.
- ❖ Asegura que todas las acciones comerciales se desarrollen en el marco de las normas constitucionales, legales y reglamentarias.

Y para que así conste, se firma la presente a los 24 días del mes de Junio del año 2010

Yaquelín Barzaga García

Nombre del representante de la entidad

Especialista en Gestión Comercial

Cargo

\_\_\_\_\_  
Firma

\_\_\_\_\_  
Cuño

## OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: “Sistema de Gestión Integral de la Empresa Empleadora del Níquel - Módulo Gestión de Contratos de Compras”.

Autor: Jorge Mario Basulto Aguilera.

El tutor del presente trabajo de diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia
- Originalidad
- Creatividad
- Laboriosidad
- Responsabilidad>

<Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de <nota 2-Desaprobado, 3-Aprobado, 4-Bien, 5-Excelente>. <Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>

---

Ing. Yodexy Mosqueda Naranjo.

---

Ing. Edgar Núñez Torres.

Instructor

Fecha: \_\_\_\_\_

## **RESUMEN**

El ascendente desarrollo que ha acontecido en los últimos años, ha situado las tecnologías de la información en un lugar relevante en el ámbito del intercambio y gestión de la información e informatización de los diferentes sistemas.

La Unidad Empresarial de Base de Compras perteneciente a la Empresa Empleadora del Níquel, se desenvuelve en un entorno de niveles considerables de información provenientes de los contratos que se tienen con las empresas del Grupo Empresarial Cubaníquel y otras entidades a lo largo del país, esta información es manejada de forma manual. La gestión de la información se ha convertido en un tema de amplia significación, lo que propicia la necesidad de crear un sistema automatizado que permita llevar el control de los contratos monitoreados por la unidad empresarial de compras.

Con el desarrollo de esta investigación, se propone la informatización de la gestión de la información correspondiente a los contratos de la entidad, a partir de la creación de una aplicación Web que gestione la información de manera ágil y segura.

Para la elaboración de la misma se realizó un estudio bibliográfico sobre las aplicaciones Web y las herramientas para la construcción de las mismas. En este documento se recoge un resumen del estudio realizado, y se presenta la metodología de Ingeniería de Software que se siguió para el desarrollo del software que se propone como solución de la problemática encontrada.

## **ABSTRACT**

The growing development that has been happening in last few years has placed the information technologies at a significant level in the interchange and information management fields and also in the computing of different systems.

The Business Unit of Purchasing from the Empresa Empleadora del Níquel (Nickel Employing Enterprise) operates in a working environment of considerable levels of information coming from contracts with other enterprises from the Cubaníquel Entrepreneurial Group and other nationwide enterprises. Such info is manually handled. Information management has become an essential subject, which supports the necessity to create an automated system that leads to keeping the whole control of contracts performed by the Purchasing business unit.

This research proposes to undertake an information process of the contractual documents by creating a webpage with easy and safe links to different informative areas.

A bibliographical revision about web applications and tools for the creation of a webpage was made for the purpose of this research. This paperwork compiles a summary of the conducted study, and the software engineering methodology that was used for the development of the software proposed as a solution to the stated problem is also presented.

# ÍNDICE

<i>PENSAMIENTO</i> .....	I
<i>AGRADECIMIENTOS</i> .....	II
<i>DEDICATORIA</i> .....	III
<b>DECLARACIÓN DE AUTORÍA</b> .....	IV
<b>OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA</b> .....	V
<b>OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA</b> .....	VI
<b>RESUMEN</b> .....	VII
<b>ABSTRACT</b> .....	VIII
<b>ÍNDICE</b> .....	IX
<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO I - FUNDAMENTACIÓN TEÓRICA</b> .....	6
1.1 INTRODUCCIÓN .....	6
1.2 ESTADO DEL ARTE .....	6
1.2.1 <i>Antecedentes de los contratos</i> .....	6
1.2.2 <i>La gestión de los contratos en las empresas</i> .....	7
1.2.3 <i>Conceptualización y características de los contratos</i> .....	10
1.3 SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN .....	11
1.4 LAS APLICACIONES WEB .....	14
1.4.1 <i>Tipos de aplicaciones web existentes</i> .....	15
1.4.2 <i>Ventajas que proporciona una aplicación web</i> .....	15
1.4.3 <i>Cualidades de las aplicaciones Web</i> .....	16
1.5 HERRAMIENTAS .....	16
1.5.1 <i>Herramientas CASE</i> .....	16
1.5.2 <i>Lenguajes de programación</i> .....	18
1.5.3 <i>Sistemas gestores de bases de datos</i> .....	19
1.5.4 <i>Servidor para aplicaciones web</i> .....	20
1.6 TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	21
1.6.1 <i>Metodología propuesta para el desarrollo de la aplicación</i> .....	22
1.6.2 <i>Programación Extrema (Extreme Programming, XP)</i> .....	23
1.6.3 <i>Valores que promueve XP</i> .....	23
1.6.3.1 <i>Prácticas en las que se fundamenta XP</i> .....	25
1.6.3.2 <i>Fases de la metodología XP</i> .....	30
1.7 PATRONES ARQUITECTÓNICOS .....	32
1.8 TECNOLOGÍAS A UTILIZAR .....	37
1.9 ARQUITECTURA A UTILIZAR.....	41
1.10 HERRAMIENTAS A EMPLEAR EN LA PROPUESTA DE SOLUCIÓN .....	41
1.11 CONCLUSIONES .....	43
<b>CAPÍTULO II - PLANEACIÓN Y DISEÑO</b> .....	44
2.1 INTRODUCCIÓN .....	44
2.2 FUNCIONALIDADES GENERALES .....	44
2.1.1 <i>Personal relacionado con el sistema</i> .....	45
2.1.2 <i>Lista de reserva</i> .....	45

2.2.3 <i>Historias de usuario</i> .....	46
2.3 PLANIFICACIÓN DE ENTREGAS.....	48
2.3.1 <i>Estimación de esfuerzo por historias de usuario</i> .....	49
2.3.2 <i>Planificación de iteraciones</i> .....	49
2.3.3 <i>Plan de duración de las iteraciones</i> .....	51
2.4 CONCLUSIONES .....	51
<b>CAPÍTULO III - DESARROLLO Y PRUEBAS.....</b>	<b>52</b>
3.1 INTRODUCCIÓN .....	52
3.2 MODELO DE DATOS.....	52
3.3 CLASES, RESPONSABILIDADES Y COLABORADORES .....	54
3.4 DESARROLLO DE LAS ITERACIONES .....	54
3.4.1 <i>Tareas por historias de usuario</i> .....	54
3.5 PRUEBAS.....	56
3.5.1 <i>Desarrollo dirigido por pruebas</i> .....	57
3.5.2 <i>Pruebas de aceptación</i> .....	57
3.6 CONCLUSIONES .....	58
<b>CAPÍTULO IV - ESTUDIO DE FACTIBILIDAD .....</b>	<b>59</b>
4.1 INTRODUCCIÓN .....	59
4.2 MÉTODOS PARA LA ESTIMACIÓN DEL ESFUERZO .....	59
4.3 ESTUDIO DE FACTIBILIDAD.....	60
4.4 ANÁLISIS DE SOSTENIBILIDAD .....	66
4.5 BENEFICIOS TANGIBLES E INTANGIBLES.....	67
<b>CONCLUSIONES GENERALES .....</b>	<b>69</b>
<b>RECOMENDACIONES.....</b>	<b>70</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>71</b>
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>74</b>
<b>ANEXO 1: INTERFAZ DE USUARIO.....</b>	<b>78</b>
<b>ANEXO 2: HISTORIAS DE USUARIO.....</b>	<b>90</b>
<b>ANEXO 3: TARJETAS CRC .....</b>	<b>94</b>
<b>ANEXO 4: MODELO DE DATOS .....</b>	<b>105</b>
<b>ANEXO 5: TARJETAS DE TAREAS .....</b>	<b>106</b>
<b>ANEXO 6: PRUEBAS DE ACEPTACIÓN .....</b>	<b>117</b>
<b>ANEXO 7: AVAL .....</b>	<b>121</b>



---

## INTRODUCCIÓN

El mundo de hoy se caracteriza por el desarrollo acelerado de las tecnologías. Ninguna empresa o negocio podrá tener éxito si no es capaz de adaptarse a los diferentes cambios aprovechando los beneficios de esta nueva era, en la que la economía global depende cada vez más de sistemas automatizados, que en épocas pasadas.

La automatización solo es viable, si al evaluar los beneficios económicos y sociales de las mejoras que se puedan obtener al automatizar, son mayores a los costos de operación y mantenimiento del sistema, sin embargo, no hay dudas de que la automatización de un proceso frente a la ejecución manual del mismo, ofrece ventajas de orden económico, social, y tecnológico; esto se debe a que se asegura una mejora en la calidad del trabajo por parte del que lo realiza y en el desarrollo del proceso, la que dependerá de la eficiencia del sistema implementado, que se traduce en la reducción de los costos, debido a la racionalidad del trabajo, logra reducir los tiempos de procesamiento de información y un conocimiento más detallado, mediante la recopilación de información y datos estadísticos del proceso.

Cuba no está ajena a los cambios que tienen lugar en el ámbito internacional por lo que las entidades, han adoptado nuevas tendencias que mejoran la gestión de las organizaciones y se han visto inmersas en la automatización de las actividades que lo requieran, como vía indispensable para incrementar su eficiencia y eficacia.

La Industria Cubana del Níquel, uno de los sectores más importantes en el desarrollo de la economía, ha venido realizando un proceso de perfeccionamiento general con el fin de asegurar su estabilidad en el mercado,



por lo que sus entidades han hecho uso de la automatización con el propósito de lograr la excelencia en su gestión.

La Empresa Empleadora del Níquel, encargada de suministrar fuerza de trabajo calificada a todas las entidades del Grupo Empresarial Cubaníquel como objeto social, ejecuta servicios de organización del trabajo y otros servicios gráficos a estas entidades y a terceros.

La cartera de clientes de la entidad es muy amplia y la inclusión de nuevos usuarios requiere de un efectivo proceso de contratación. Cada servicio que se ejecute debe estar respaldado por un contrato.

La gestión de la información de los contratos en la Unidad Empresarial de Base de Compras se realiza de forma manual, lo que hace el proceso más lento, con el consiguiente aumento de gastos de papel y otros materiales de oficina e incrementa la posibilidad de que se produzcan errores en la manipulación de la información de los mismos, lo que puede traer como consecuencias afectaciones en la gestión comercial de la entidad debido a que pueden ejecutarse compras, cuyos contratos no estén vigentes o por montos superiores a los aprobados en los mismos, lo que constituye el **problema social** de esta investigación.

En respuesta a las irregularidades mencionadas, se define como **problema científico**, la necesidad de garantizar el control de la información en el proceso de gestión de los contratos en la Unidad Empresarial de Base de Compras de la Empresa Empleadora del Níquel de forma automatizada.

El **objeto de estudio** de la investigación, lo constituye el proceso de gestión de contratos en la Unidad Empresarial de Base de Compras de la entidad mencionada y el **campo de acción** el proceso de automatización de la gestión



de los contratos en la Unidad Empresarial de Base de Compras de la referida entidad.

El **objetivo general** de la investigación consiste en desarrollar un software que permita mantener el control de forma automatizada de la información en el proceso de gestión de los contratos en la Unidad Empresarial de Base de Compras de la Empresa Empleadora del Níquel.

Como **objetivos específicos**:

1. Revisar la literatura relacionada con el tema.
2. Realizar observaciones en el campo del objeto de estudio y entrevistas efectuadas a especialistas.
3. Estudiar la metodología para la elaboración del software.
4. Realizar el análisis y diseño del software.
5. Desarrollar el software.
6. Realizar el estudio de factibilidad.
7. Implantar el software.

Para dar respuesta al problema científico se plantea como **idea a defender** que el diseño de un sistema automatizado para el control de la información en el proceso de gestión de los contratos, agilizará el desarrollo de esta actividad en la Unidad Empresarial de Base de Compras, y por consiguiente la optimización de la gestión comercial de la organización.

Los **métodos de investigación** empleados son los teóricos y empíricos. Los métodos empíricos utilizados fueron la observación, entrevistas y análisis de documentos para la recopilación de la información. La observación se utilizó para conocer la funcionalidad de la “UEB” de Compras de la entidad y el comportamiento del problema. La entrevista a partir de la conversación



planificada, con el fin de obtener información individual y colectiva para determinar los principales requerimientos del sistema. Mediante el análisis de documentos se pudo conocer el funcionamiento actual de los procesos del departamento.

Los métodos teóricos utilizados fueron, el método histórico y lógico para la búsqueda de los antecedentes del software. El análisis y síntesis para la recopilación y el procesamiento de la información obtenida en los métodos empíricos y arribar a las conclusiones de la investigación; y el hipotético deductivo para la elaboración de la idea a defender y su verificación. Mediante la modelación se realizó el estudio de la gestión de la información de la Unidad Empresarial de Base de Compras.

Las etapas de la investigación son:

- ❖ El estudio de las distintas metodologías de desarrollo de software, las herramientas, tecnologías y patrones arquitectónicos para hacer una selección de las mejores, y utilizarlos en el diseño del software de acuerdo a sus características.
- ❖ La contextualización del problema, enmarcando los objetivos específicos requeridos por el cliente.
- ❖ Análisis y diseño del software con todas las especificaciones requeridas.
- ❖ Realización del estudio de factibilidad.

El presente trabajo consta de introducción, 4 capítulos, conclusiones, recomendaciones, bibliografía, glosario de términos y anexos.

En el capítulo 1, *Fundamentación teórica*, se analizan aspectos relacionados con los antecedentes de la contratación, conceptualización y características, se realiza un estudio acerca de los diferentes sistemas existentes vinculados al



campo de acción, además se presenta la metodología y las herramientas a utilizar en el desarrollo del sistema propuesto.

En el capítulo 2, *Planeación y diseño*, se hace uso de la metodología expuesta en el capítulo inicial para el desarrollo del proyecto, abordando en detalles cada una de sus fases.

El capítulo 3, *Desarrollo y Pruebas*, se presentan los principales métodos y definiciones dentro de la implementación de los flujos de trabajo. Se describen además las pruebas realizadas y sus resultados.

En el capítulo 4, *Estudio de factibilidad y sostenibilidad*, se realiza un estudio de los esfuerzos requeridos para la realización del sistema, y se valora la sostenibilidad del producto.

Para concluir se muestran las conclusiones a las que se arribaron, las recomendaciones propuestas, bibliografía empleada, glosario de términos y anexos con la información necesaria sobre el trabajo.



---

## CAPÍTULO I - FUNDAMENTACIÓN TEÓRICA

### 1.1 Introducción

En la actualidad, la informática ha evolucionado considerablemente y junto a ella la producción de diversos software. En este capítulo se realiza un esbozo acerca de los antecedentes de los contratos, así como sus conceptos fundamentales y características, además se hace una descripción sobre las aplicaciones web. Se muestran algunos de los sistemas automatizados existentes unido al campo de acción y se exponen tanto la metodología a seguir en el desarrollo de la aplicación, como las herramientas a emplear en la misma.

### 1.2 Estado del arte

#### 1.2.1 Antecedentes de los contratos

Los primeros vestigios de la investigación se remontan a las fuentes romanas, en las que se concibe ya la posibilidad de que, al concertar un contrato, las partes dispusieran la atribución de un derecho a favor de quien no había intervenido en él. En el *derecho justiniano*, el contrato es el acuerdo de voluntades capaz de instituir a una persona en deudora de otra, incluyendo como tal, a toda clase de negocio que tuviera por fin la creación, modificación o extinción de cualquier relación jurídica. (Guzmán, Argüello, 1962)

El *nexum* fue el primer contrato romano que se caracteriza por las rígidas solemnidades que debían seguirse para su perfeccionamiento, como la pesada del cobre y la balanza, la presencia del *librepiens* y de los cinco testigos.



Una derivación del nexum, es el *sponsio*, contrato que consistía en el empleo de palabras sacramentales, como *¿spondes?*, a lo que el obligado contestaba *spondeo*, sin necesidad del *per aes et libram*. No obstante, como este contrato podía llevarse a cabo entre ciudadanos, aparece la *stipulatio* para que también se pudieran contratar los no ciudadanos, donde las partes podían interrogarse usando cualquier expresión, a lo que el obligado contestaba siempre *promitto*, de esta manera nacieron los contratos verbales.

A partir de la práctica de que un ciudadano romano llevara el libro de registro doméstico, el *codex accepti et expensi*, donde anotaba los créditos contra el deudor, surge la *nómina transcriptitia*, que era usada cuando el obligado era otro ciudadano, y con la *chirographa* o *syngrapha* para el deudor extranjero. De estas formas de efectuar una convención cuyo perfeccionamiento radicaba en las anotaciones, se derivan los contratos literales.

Posteriormente, se agregaron el *mutuo*, el *comodato*, el *depósito* y la *prenda*, que surgen cuando deja de ser el nexum el medio más idóneo para desarrollarlos, y basta la simple usanza de un objeto, para establecer los contratos reales.

Finalmente, cuando la evolución del *derecho romano* hizo del acuerdo de voluntades el elemento característico del contrato, se acepta que puedan ser perfeccionados por el mero consentimiento de las partes, apareciendo así, los contratos consensuales. (Demarchi, 2008)

### **1.2.2 La gestión de los contratos en las empresas**

En todas las empresas, la *gestión de contratos* con clientes, proveedores y colaboradores consume buena parte del tiempo de su asesoría jurídica y directivos. Estos deben ser renovados, modificados o cancelados en



condiciones muy diversas. Para determinadas empresas de crédito al consumo, de venta por internet o de productos de gran difusión, la gestión de contratos supone parte esencial de su actividad. Todo ello implica una gestión de procesos de información compleja sujeta a incumplimientos, demandas y litigios en los que la empresa pone en juego su dinero y su prestigio en el mercado.

Los contratos regulan la actividad comercial entre los socios comerciales. El incumplimiento de los mismos puede causar costes incalculables y poner en peligro la existencia de una empresa u organización. La ausencia de un control sistemático, herramientas insuficientes y alto gasto de administración en el ámbito de la gestión de contratos asociada al papel, junto al riesgo permanente de pérdida de información, pueden ser evitados mediante una *gestión electrónica de contratos integrada*.

### **Los contratos en Cuba**

En 1978, se dicta en Cuba el Decreto-Ley 15, para establecer las “Normas Básicas para los Contratos Económicos” que regirían a partir de ese momento las relaciones monetario-mercantiles en el país y que contó con una legislación complementaria que definía y regulaba un conjunto de figuras contractuales, que actualmente se siguen utilizando dentro del proceso económico. No obstante, a raíz de los diferentes cambios que a partir de la década de los años 90 operan en Cuba, se han incorporado nuevas modalidades de contratos que son frecuentemente empleadas dentro del proceso de gestión empresarial y por tanto se requiere atemperar la vieja legislación vigente en materia de contratación económica a las nuevas condiciones técnico económicas del país.

Como un hecho singular de la nueva normativa, y tomando en cuenta regulaciones posteriores a la legislación contractual de finales de los años 70 y principios de los 80 del siglo pasado, cabe señalar que el Resuelvo segundo de



la Resolución 2253/2005 incluye en su inciso f) a las empresas mixtas como sujeto de estas indicaciones. (Valdés Domínguez, 2005)

### **Gestión de los contratos de compras en la Empresa Empleadora del Níquel (EMPLENI)**

Los directores de las áreas de regulación y control y directores de unidades empresariales de base (UEB) se encargan de identificar los recursos necesarios para la realización de los servicios, posteriormente deben llenar las solicitudes de materiales e insumos de clientes internos y entregarlas al especialista principal de compras, para el caso de las solicitudes que surgen en las áreas por situaciones coyunturales, son entregadas al mismo con su debida fundamentación. Las solicitudes son presentadas ante el comité económico financiero (CEF) para su aprobación. Una vez que las mismas son aprobadas comienza el proceso de contratación.

Primeramente, los especialistas del grupo de compras tienen la tarea de actualizar el registro de comportamiento de los recursos solicitados por los clientes internos, teniendo en cuenta el máximo y mínimo de inventario establecido para el cumplimiento de los planes de producción y servicios, con el objetivo de llevar la trazabilidad de los recursos solicitados hasta la ejecución de las compras. Los mismos deben comunicar al solicitante el seguimiento de la ejecución de la compra, y la explicación de las acciones realizadas con los proveedores aprobados o los posibles a contratar.

Por último los especialistas en gestión comercial que atienden los contratos son responsables de entregar el contrato original al asesor jurídico para su archivo y custodia. Del mismo modo deben actualizar el registro central de contratos de compras, con el objetivo de controlar todo lo relacionado con los estos: vigencia, importe, caducidad etc. Además de la actualización del registro de



solicitud de aprobación de contratos, con el propósito de resumir los principales datos del mismo.

### 1.2.3 Conceptualización y características de los contratos

#### Definición

En términos generales, el *contrato* es definido como un acuerdo privado, oral o escrito, entre partes que se obligan sobre materia o cosa determinada, y a cuyo cumplimiento pueden ser exigidas. Es un acuerdo de voluntades que genera derechos y obligaciones para las partes.<sup>1</sup>

Se señala que habrá contrato, cuando varias partes se ponen de acuerdo sobre una manifestación de voluntad destinada a regularizar sus derechos. Teóricamente, ha sido definido como un negocio jurídico bilateral o multilateral, porque intervienen dos o más personas (a diferencia de los actos jurídicos unilaterales que interviene una sola persona), y que tiene por finalidad crear derechos y obligaciones (a diferencia de otros actos jurídicos que están destinados a modificar o extinguir derechos y obligaciones, como las convenciones). Igualmente, se denomina contrato el documento que recoge las condiciones de dicho acto jurídico, en éste, hay dos polos o extremos de la relación jurídica obligacional, cada uno puede estar constituido por más de una persona respaldando la calidad de parte.

El contrato, en general, tiene una connotación patrimonial, y forma parte de la categoría más amplia de los negocios jurídicos. La función del mismo es originar efectos jurídicos. En cada país puede existir un concepto de contrato diferente, y esa divergencia tiene que ver con la realidad socio-cultural y jurídica de cada país, existen ordenamientos en que el contrato no se limita al campo de los derechos patrimoniales únicamente, sino que abarca también

---

<sup>1</sup> Según el artículo 1254 del Código Civil de España



derechos de familia como, por ejemplo, los países en los que el matrimonio es considerado un contrato. (citado en: <http://es.wikipedia.org/wiki/Contrato>)

### **Contrato legal de acuerdo a códigos civiles**

La mayoría de los códigos civiles contienen una definición de "contrato". Muchos de ellos, siguen los lineamientos del código civil francés, que en su artículo No. 1101 se expresa..."El contrato es la convención por la cual una o más personas se obligan, con otra u otras, a dar, hacer, o no hacer alguna cosa".

El código civil alemán prescribe que tanto "para la formación de un negocio obligacional por actos jurídicos, como para toda modificación del contenido de un negocio obligacional se exige un contrato celebrado entre las partes, salvo que la ley disponga de otro modo".

Mientras el código civil suizo señala que "hay contrato, si las partes manifiestan de una manera concordante su voluntad recíproca; esta manifestación puede ser expresa o tácita".

Por otra parte el código civil soviético solo expresa que "los actos jurídicos, son los que tienden a establecer, modificar o extinguir relaciones de derecho civil, que pueden ser unilaterales o bilaterales. (citado en: <http://es.wikipedia.org/wiki/Contrato>)

## **1.3 Sistemas automatizados existentes vinculados al campo de acción**

### **Sistema control de contratos**

El módulo Sistema Control de Contratos (SCC), comprende todas las actividades necesarias para el manejo y control de contratos de ordenes de



servicio, cartas contratos, contratos a todo costo, de mano de obra, equipos, materiales, transportes, varios, ofertas mercantiles y maquila.

Permite dar seguimiento a todos y cada uno de los contratistas, y logra elaborar su contrato inicial así como el pago de las actas parciales de obra. Incluye un sistema de aplicación de novedades, referidas éstas a todos los conceptos de costos que afectan a un contrato, como los anticipos, amortizaciones, garantía de obra, retención en la fuente, reteica, devolución de la garantía, parafiscales, descuento por obra mala, etc. Cuando se modifican las condiciones iniciales del contrato el módulo incluye el manejo de OtroSi<sup>2</sup>.

Los valores a contratar no pueden exceder los del presupuesto ni en cantidad ni en precio. La interfaz contable le permite al usuario subir información tipo batch en lo que se refiere a transacciones por pago de anticipos, actas y/o planillas e impuesto de timbre. (citado en <http://www.gtcia.com.co/scc.htm>)

Hasta el momento se tiene conocimiento de la implementación de software para efectuar contratos compras, como sigue:

### **Software para el control de contratos de cereal - descripción y objetivos del Aplicativo<sup>3</sup>**

- Realizar la entrega de los camiones, el resultado de los análisis realizados al cereal y controla los descuentos y bonificaciones realizadas por la calidad del cereal en base a la codificación “*condiciones cámara*”.

---

<sup>2</sup> Ocasionalmente los contratos de mano de obra sufren alteraciones respecto al contrato inicial, dándose el caso en que es necesario modificar las cantidades originales contratadas así como los precios inicialmente pactados. A través de la opción de OtroSi el usuario puede hacer todas estas modificaciones.

<sup>3</sup> Nota: agroads.com no vende este producto o servicio y no participa en ninguna negociación, venta o perfeccionamiento de operaciones. El usuario asume toda la responsabilidad por la publicación.



- Registrar las *condiciones especiales de venta* mediante fórmulas, chequea las bonificaciones y los descuentos realizados.
- Confrontar los distintos análisis realizados al cereal (origen, laboratorio, cámara y destino).
- Controlar los cobros realizados por el contrato.
- Registrar las liquidaciones de cereal y generar el informe de I.V.A.
- Controlar las cuentas corrientes de los contratos.

(citado en: <http://www.agroads.com.ar/guias/software-agropecuario.html>)

### **Software administración de contratos Ebackup S.A**

Ebackup S.A, se ha desarrollado en base a conocimiento y experiencia, sobre todo con requerimientos de sus clientes, una plataforma documental para la administración de contratos, y toma tanto información histórica que se requiera almacenar y consultar, como contratos de consulta recientes.

En la actualidad Ebackup S.A cuenta con una plataforma vía internet en línea para administrar los distintos tipos de contratos, comodatos, arrendamiento, compraventas e información anexa a ellas, con una herramienta que además de aviso automático de vencimiento de los contratos, tiene la finalidad de constituir no sólo una biblioteca de almacenamiento de contratos, sino que ellos tengan vigencia y existan dentro del sistema. (citado en: <http://www.hotfrog.cl/Companies/Administracion-de-Contratos-Ebackup>)

### **Sistema de Información para la Vigilancia de la Contratación Estatal**

El Sistema de Información para la Vigilancia de la Contratación Estatal (SICE) es un sistema de información que integra todos los datos relevantes de proceso de contratación estatal, permite su autorregulación, control institucional y publicidad de las operaciones. (citado en: <http://www.sice-cgr.gov.co/>)



Los softwares antes mencionados tienen una amplia gama de aplicabilidad, sin embargo, estos no se adaptan a las necesidades que se requieren en la empresa para resolver de forma objetiva y factible el problema existente. Además, son inalcanzables debido a los elevadísimos precios que poseen.

#### **1.4 Las aplicaciones web**

Cuando la informática se introdujo por primera vez en la empresa lo hizo bajo un modelo cliente/servidor. Los usuarios disponían de una terminal con una pequeña pantalla conectada al servidor donde coexistían los datos y el software.

Más tarde con la aparición de la PC, el modelo cambió y la lógica y los datos pasaron a residir completamente en ésta, con los consecuentes problemas de administración, soporte y coste de las licencias. Esto no varió durante mucho tiempo, actualmente parece retornar nuevamente al principio.

La era de los ordenadores ha terminado para ceder paso a la era de la Internet y es justamente ahí donde se alojarán los programas y los datos. Con las aplicaciones web se revitaliza el papel del servidor web. Los datos se almacenan en las bases de datos accesibles desde un navegador o una terminal móvil gracias a la lógica que se ejecuta en el servidor y al diseño de la interfaz que es transferido a dichas terminales. Se conocen buenos ejemplos de este tipo de programa, como el correo electrónico basado en web o las comunidades en línea.

Una aplicación web es un conjunto de páginas web enlazadas que visualizan diferentes partes de la información que se quiere mostrar a través de ella. Constituye una de las mejores herramientas para divulgar, gestionar y



---

compartir información, lo que trae consigo un aumento de la eficiencia en cuanto a la manipulación de gran cantidad de información. (Monmany, 2009)

#### **1.4.1 Tipos de aplicaciones web existentes**

Una de las primeras revelaciones que pueden originarse, al planear migrar los datos a Internet, es el riesgo que conlleva en cuanto a seguridad y disponibilidad. Esta disyuntiva se utiliza para introducir la distinción entre dos tipos de aplicaciones web según su entorno de funcionamiento, que puede ser una intranet o extranet. En el primer caso, la aplicación se aloja y sólo está disponible dentro de la red interna de la empresa, mientras que en el segundo, está en cualquier punto con acceso a Internet. Si no se dispone de una buena conexión y/o se prefiere tener las aplicaciones dentro de la organización, se podrá hacer sin renunciar a las ventajas de las tecnologías de Internet. Si se necesita llegar a usuarios de redes externas, se opta por una extranet y se coloca la aplicación en un servidor con óptima conexión a Internet, que puede ser contratado a una compañía especializada. (Monmany, 2009)

#### **1.4.2 Ventajas que proporciona una aplicación web**

Debido a que el mantenimiento se centra en el servidor, el costo se reduce. En general, es el proveedor del servicio quien se preocupa de tener la aplicación disponible y actualizada a cambio de una cuota razonable. Existen numerosas ventajas, pero posiblemente, la propiedad más destacada sea la conectividad que proporciona la Internet permitiendo el acceso a la aplicación desde cualquier punto. Esto permite ahorrarse el intervenir en costosas infraestructuras de comunicaciones que en muchos casos podrían ser no viables. (Monmany, 2009)



### **1.4.3 Cualidades de las aplicaciones Web**

Para que la web sea efectiva, debe poseer algunas cualidades indispensables, como la seguridad, escalabilidad, portabilidad y un diseño eficiente de la interfaz y el programa. La escalabilidad y la portabilidad son las que con más frecuencia caen en el olvido.

Escalabilidad: la aplicación tendrá la capacidad de crecer junto con las necesidades de sus usuarios.

Portabilidad: se enlaza al concepto de escalabilidad, y se refiere a la compatibilidad de la aplicación con múltiples sistemas operativos y plataformas existentes en el mercado. En general, cuando las demandas de los usuarios crecen, la aplicación debe cambiar de plataforma para pasar de un servidor pequeño a uno mediano o grande. Es entonces, cuando la portabilidad es crítica para poder “escalar” sin problemas, independientemente del entorno que se requiera.

Sin dudas, las aplicaciones web tienen aún un largo camino por recorrer, pero son ya una opción muy interesante, especialmente, en tareas donde las bases de datos juegan un papel predominante y los usuarios se encuentran dispersos. (Monmany, 2009)

## **1.5 Herramientas**

### **1.5.1 Herramientas CASE**

Las herramientas CASE (Ingeniería de Software Asistida por Computadoras), son aplicaciones informáticas que tienen como objetivo fundamental solucionar



y afrontar los problemas de mala calidad de software y documentación inadecuada. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo de un determinado proyecto o software, ya que brindan la posibilidad de realizar cálculos de costos, generan código fuente automáticamente de un diseño previamente dado, poseen compilación automática, ayudan con la documentación y juegan un papel importante en la detección de errores.

Por tales ventajas algunas personas han dado su criterio personal sobre el concepto CASE, que se define como:

“Herramientas individuales para ayudar al desarrollador de software o administrador de proyecto durante una o más fases del desarrollo de software o mantenimiento del mismo” (citado en: [http://www.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro\\_case\\_SA.pdf](http://www.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf)).

Además, estas herramientas son ajustables a distintas metodologías de desarrollo de software, mejoran la productividad del desarrollo y mantenimiento del software, aumentan la calidad de los mismos y ayudan a la reutilización del software, portabilidad y estandarización de la documentación.

**Rational Rose** - Es una herramienta software para el modelado visual mediante el lenguaje unificado de modelado (unified modeling lenguaje UML) de sistemas software, permite especificar, analizar, diseñar el sistema antes de codificar. Mantiene la consistencia de los modelos del sistema software, además de la generación de documentos automáticamente. Otra funcionalidad de la herramienta es que permite realizar la ingeniería inversa de un producto.



### 1.5.2 Lenguajes de programación

**Personal Home Page** - (PHP) es el acrónimo de procesador hipertexto (Hypertext Preprocessor). Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Es también un lenguaje interpretado y embebido en el HTML. Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores.

PHP es un lenguaje de programación de estilo clásico, esto significa que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como podría ser HTML, XML o WML.

A diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor, como por ejemplo podría ser, una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML.

PHP es la gran tendencia en el mundo de Internet. Últimamente se puede observar un ascenso imparable, puesto que cada día son muchas las páginas web que lo utilizan para su funcionamiento, según las estadísticas, PHP se utiliza en más de 10 millones de páginas, y cada mes realiza un aumento del 15%.

Como síntesis, PHP corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte sobre unas 20 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 sólidas, y contiene unas 40 extensiones



estables sin contar las que se están experimentando, igualmente tiene soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros, además de que:

- ❖ Es software libre, lo que implica menos costes y servidores más baratos que otras alternativas.
- ❖ Es muy rápido. Su integración con la base de datos MySQL y el servidor Apache, le permite constituirse como una de las alternativas más atractivas del mercado.
- ❖ Su sintaxis está inspirada en C, ligeramente modificada para adaptarlo al entorno en el que trabaja, de modo que si está familiarizado con esta sintaxis, resultará un poco mejor aprender PHP.
- ❖ Su librería estándar es realmente amplia, lo que permite reducir los llamados “costes ocultos”, uno de los principales defectos de ASP.
- ❖ PHP tiene una de las comunidades más grandes en internet, esto permite encontrar fácilmente ayuda, documentación, artículos, noticias y otros recursos.
- ❖ Permite las técnicas de Programación Orientada a Objetos (POO).
- ❖ Posibilita crear los formularios para la web.
- ❖ No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

### 1.5.3 Sistemas gestores de bases de datos

**MySQL** - es un sistema de gestión de base de datos relacional, multi-hilo y multiusuario, con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. MySQL AB pertenece a Sun Microsystems desde enero de 2008. Por un lado lo ofrece bajo la GNU GPL, pero, entidades que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia que les permita ese uso. Está desarrollado en su mayor parte en ANSIC.



Al contrario de proyectos como el Apache, donde el software es desarrollado por una comunidad pública, y el copyright del código está en poder del autor individual, MySQL está poseído y patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. (citado en: <http://es.wikipedia.org/wiki/MySQL>)

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson, y Michael Widenius.

MySQL funciona sobre múltiples plataformas, incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows. También existe MySQL para OpenVMS.

#### 1.5.4 Servidor para aplicaciones web

**XAMPP** - es un paquete formado por un servidor web Apache, una base de datos MySQL y los intérpretes para los lenguajes PHP y Perl. El nombre proviene de **X** (para cualquier sistema operativo), **A** (Apache), **M** (MySQL), **P** (PHP), **P** (Perl). El programa está liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris, y MacOS X. (Blanco, 2008)



XAMPP es regularmente actualizado para incorporar las últimas versiones de Apache/MySQL/PHP y Perl. Incluye otros módulos como OpenSSL, y phpMyAdmin.

Independientemente de que existen otras herramientas como son, los lenguajes de programación web, gestores de bases de datos y servidores web, que, aunque no se mencionan y describen, no pierden su vigencia, funcionalidad e importancia, en esta investigación sólo se hace referencia:

- ✓ Al lenguaje de programación, Personal Home Page (PHP).
- ✓ El gestor de bases de datos, MySQL.
- ✓ Y el servidor web, XAMPP.

Ya que la EMPLeni, cuenta con un sistema de gestión integral, implementado con las herramientas antes mencionadas, y en esta investigación, el software a desarrollar, se integrará al mismo, por tanto, la política de la empresa estipula que se empleen las herramientas citadas.

## **1.6 Tendencias y tecnologías actuales**

Ante el incesante avance de las tecnologías, la sociedad, ávida de nuevas herramientas y funcionalidades, exige a los desarrolladores de software nuevos retos y nuevas concepciones para satisfacer sus exigencias, cada vez más ambiciosas. Para satisfacer estas exigencias, los desarrolladores deben buscar nuevas ideas surgiendo así nuevas metodologías y formas de desarrollo que permitan confeccionar productos cada vez más complejos.



### 1.6.1 Metodología propuesta para el desarrollo de la aplicación

El desarrollo de software no es una tarea fácil. Prueba de ello es que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte tenemos aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, las herramientas y anotaciones que se usarán. Estas propuestas han demostrado ser efectivas y necesarias en un gran número de proyectos, pero también han presentado problemas en muchos otros.

Una posible mejora es incluir en los procesos de desarrollo más actividades, artefactos y restricciones, basando en los puntos débiles detectados. Sin embargo, el resultado final sería un proceso de desarrollo más complejo que puede incluso limitar la propia habilidad del equipo para llevar a cabo el proyecto. Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software. Esta es la filosofía de las metodologías ágiles, las cuáles dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas.

Este enfoque ha mostrando su efectividad en proyectos con requisitos muy cambiantes y cuando se exige reducir drásticamente los tiempos de desarrollo, no obstante, mantiene una alta calidad. Las metodologías ágiles están revolucionando la manera de producir software, y a la vez generando un amplio debate entre sus seguidores y quienes por escepticismo o convencimiento no las ven como alternativa para las metodologías tradicionales.



### 1.6.2 Programación Extrema (Extreme Programming, XP)

#### Definición

XP<sup>4</sup>, es una metodología ágil, centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, se preocupa por el aprendizaje de los desarrolladores, y propicia un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Beck, 1999)

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck, el padre de XP, que describe la filosofía de XP en el "Manifiesto Ágil"<sup>5</sup>, sin cubrir los detalles técnicos y de implantación de las prácticas. Posteriormente, otras publicaciones de experiencias se han encargado de dicha tarea.

### 1.6.3 Valores que promueve XP

#### Comunicación

XP se nutre del ancho de banda más grande que se puede obtener cuando existe algún tipo de comunicación como la comunicación directa entre personas. Es muy importante entender cuáles son las ventajas de este medio. Cuando dos o más personas se comunican directamente pueden no sólo consumir las palabras formuladas por la otra persona, sino que también aprecian los gestos, miradas etcétera, que hace su compañero. Sin embargo,

---

<sup>4</sup> [www.extremeprogramming.org](http://www.extremeprogramming.org), [www.xprogramming.com](http://www.xprogramming.com), [c2.com/cgi/wiki?ExtemeProgramming](http://c2.com/cgi/wiki?ExtemeProgramming)

<sup>5</sup> [agilemanifesto.org](http://agilemanifesto.org)



---

en una conversación mediante el correo electrónico, hay muchos factores que hacen de esta una comunicación, por así decirlo, mucho menos efectiva.

### **Coraje**

El coraje es un valor muy importante dentro de la programación extrema. Un miembro de un equipo de desarrollo extremo debe tener el coraje de exponer sus dudas, miedos, experiencias sin “embellecer” éstas de ninguna de las maneras. Esto es muy importante ya que un equipo de desarrollo extremo se basa en la confianza para con sus miembros. Faltar a esta confianza es grave.

### **Simplicidad**

Dado que no se puede predecir como va a ser en el futuro, el software que se desarrolla; por un equipo de programación extrema intenta mantener el mismo lo más sencillo posible. Esto quiere decir que no se va a invertir ningún esfuerzo en hacer un desarrollo que en un futuro pueda llegar a tener valor. En XP, frases como, “...en un futuro vamos a necesitar...” o, “...haz un sistema genérico de...”, no tienen ningún sentido ya que no aportan ningún valor en el momento.

### **Retroalimentación**

La agilidad se define, entre otras cosas, por la capacidad de respuesta ante los cambios que se van haciendo necesarios a lo largo del camino. Por este motivo uno de los valores que nos hace más ágiles es el continuo seguimiento o retroalimentación que recibimos a la hora de desarrollar en un entorno ágil de desarrollo. La retroalimentación se toma del cliente, de los miembros del equipo y de todo el entorno en el que se mueve un equipo de desarrollo ágil.



### 1.6.3.1 Prácticas en las que se fundamenta XP

#### Planificación incremental

La programación extrema asume que la planificación nunca será perfecta, y que variará en función de cómo varíen las necesidades del negocio. Por tanto, el valor real reside en obtener rápidamente un plan inicial, y contar con mecanismos de retroalimentación que permitan conocer con precisión dónde se está. Como es lógico, la planificación es iterativa: un representante del negocio decide al inicio de cada iteración qué características concretas se van a implementar.

El objetivo de XP es generar versiones de la aplicación tan pequeñas como sea posible, pero que proporcionen un valor adicional claro, desde el punto de vista del negocio. A estas versiones se les denomina *releases*.

Un release cuenta con un cierto número de historias. La historia es la unidad de funcionalidad en un proyecto XP, y corresponde a la mínima funcionalidad posible que tiene valor desde el punto de vista del negocio. Durante cada iteración se cierran varias historias, lo que hace que toda iteración añada un valor tangible para el cliente.

Gran parte de la eficacia de este modelo de planificación se deriva de una división clara de responsabilidades, que tiene en cuenta las necesidades del negocio en todo momento. Dentro de esta división, el representante del cliente tiene las siguientes responsabilidades:

- ❖ Decidir qué se implementa en cada release o iteración.
- ❖ Fijar las fechas de fin de un release, recortando unas características o añadiendo otras.



- ❖ Priorizar el orden de implementación, en función del valor de negocio.

Las responsabilidades del equipo de desarrollo son las siguientes:

- ❖ Estimar cuanto tiempo llevará una historia de usuario, esto es fundamental para el cliente, y puede llevarle a reconsiderar que historias se deben incluir en una iteración.
- ❖ Proporcionar información sobre el coste de utilizar distintas opciones tecnológicas.
- ❖ Organizar el equipo.
- ❖ Estimar el riesgo de cada historia.
- ❖ Decidir el orden de desarrollo de historias dentro de la iteración.

### **Testing**

La ejecución automatizada de tests es un elemento clave de la metodología XP. Existen tanto tests internos (o tests de unidad), para garantizar que el mismo es correcto, como tests de aceptación, para garantizar que el código hace lo que debe hacer. El cliente es el responsable de definir los tests de aceptación, no necesariamente de implementarlos. Él es la persona mejor calificada para decidir cuál es la funcionalidad más valiosa.

Un efecto lateral importante de los tests, es que dan una gran seguridad a los desarrolladores; es posible llegar a hacer cambios más o menos importantes sin miedo a problemas inesperados, dado que proporciona una red de seguridad. La existencia de tests hace al código muy maleable.

### **Programación en parejas**

XP incluye, como una de sus prácticas estándar, la programación en parejas. Nadie programa en solitario, siempre hay dos personas frente al ordenador.



Esta es una de las características que más se cuestiona al comienzo de la adopción de la metodología dentro de un equipo, pero en la práctica se acepta rápidamente y de forma entusiasta.

El hecho de que todas las decisiones las tomen al menos dos personas, proporciona un mecanismo de seguridad enormemente valioso. Es muy difícil que dos personas se salten tareas por descuido o negligencia. El código siempre está siendo revisado por otra persona. La revisión de código es el método más eficaz de conseguir código de calidad, algo corroborado por numerosos estudios, muchos de los cuáles son anteriores a la programación extrema.

En contra de lo que pueda parecer, los dos desarrolladores no hacen lo mismo, mientras el que tiene el teclado adopta un rol más táctico, el otro adopta un rol más estratégico, se preguntan constantemente si lo que se está haciendo tiene sentido desde un punto de vista global.

### **Refactorización**

Uno de los objetivos de XP, es mantener la curva de costes tan plana como sea posible, por lo que existen una serie de mecanismos destinados a mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez. A este proceso básico para mantener el código en buena forma se le llama refactorización. La refactorización no sólo sirve para mantener el código legible y sencillo, sino que también se utiliza cuando resulta conveniente modificar el código existente para hacer más fácil de implementar la nueva funcionalidad.

### **Diseño simple**

Otra práctica fundamental de la programación extrema es utilizar diseños tan simples como sea posible. El principio es "utilizar el diseño más sencillo que



consiga que todo funcione". La metodología XP nos pide, que no se viva bajo la ilusión de que un diseño puede resolver todas o gran parte de las situaciones futuras.

XP define un "diseño tan simple como sea posible" aquél que pasa todos los tests, no contiene código duplicado, deja clara la intención de los programadores (enfatisa el qué, no el cómo) en cada línea de código y contiene el menor número posible de clases y métodos.

### **Propiedad colectiva del código**

XP aboga por la propiedad colectiva del código. En otras palabras, todo el mundo tiene autoridad para hacer cambios a cualquier código, y es responsable de ellos. Esto permite no tener que estar esperando a otros cuando todo lo que hace falta es algún pequeño cambio.

### **Integración continua**

En muchos casos la integración de código produce efectos laterales imprevistos, y en ocasiones esta puede llegar a ser realmente traumática, cuando dejan de funcionar cosas por motivos desconocidos. La programación extrema hace que la integración sea permanente, con lo que todos los problemas se manifiestan de forma inmediata.

### **Clientes en el equipo**

Algunos de los problemas más graves en el desarrollo son los que se originan cuando el equipo toma decisiones de negocio críticas. Esto no debería ocurrir, pero en el momento cumbre, con frecuencia no se obtiene retroalimentación del cliente con la fluidez necesaria.

La metodología XP intenta resolver este tipo de problemas integrando un representante del negocio dentro del equipo de desarrollo. Esta persona



---

siempre está disponible para resolver dudas y para decidir qué y qué no se hace en cada momento, en función de los intereses del negocio.

### **Entregas pequeñas**

Siguiendo la política de la metodología XP, de dar el máximo valor posible en cada momento, se intenta liberar nuevas versiones de las aplicaciones con frecuencia. Estas deben ser tan pequeñas como sea posible, aunque deben añadir suficiente valor, para que resulten meritorias para el cliente.

### **Semana de 40 horas**

La programación extrema lleva un modo de trabajo en el que el equipo siempre está al 100%. Una semana de 40 horas en las que se dedica la mayor parte del tiempo a tareas que suponen un avance puede dar mucho de sí, y hace innecesario recurrir a sobreesfuerzos, excepto en casos extremos. Además, el esfuerzo continuado pronto lleva a un rendimiento menor y a un deterioro de la moral de todo el equipo.

### **Estándares de codificación**

Para conseguir que el código se encuentre en buen estado y que algún miembro del equipo pueda modificar cualquier parte del código es imprescindible que el estilo de codificación sea consistente. Un estándar de codificación es necesario para soportar otras prácticas de XP.

### **Uso de metáforas**

Para conseguir que la comunicación sea fluida es imprescindible, entre otras cosas, utilizar el vocabulario del negocio. Además es fundamental huir de definiciones abstractas. Dicho de otro modo, la metodología XP no pretende seguir la letra de la ley, sino de su espíritu. Dentro de este enfoque es fundamental buscar continuamente metáforas que comuniquen intenciones y resulten descriptivas, enfatizando *el qué* antes *del cómo*.



### 1.6.3.2 Fases de la metodología XP

#### Fase I: Planificación

- ❖ Se escriben historias de usuario, cuya idea principal es describir un caso de uso en dos o tres líneas con terminología del cliente (de hecho, se supone que deben ser escritos por el mismo), de tal manera que se creen *test* de aceptación para historias de usuarios (*user storie*) y permita hacer una estimación de tiempo de desarrollo del mismo.
- ❖ Se crea un plan de lanzamiento (*release planning*), que debe servir para crear un calendario que todos puedan cumplir y en cuyo desarrollo hayan participado todas las personas involucradas en el proyecto. Se usa como base las historias de usuario, participando el cliente en la elección de las que se desarrollarán, y según las estimaciones de tiempo de los mismos se crearán las iteraciones del proyecto.
- ❖ El desarrollo se divide en iteraciones, cada una de las cuales comienzan con un plan de iteración, para el que se eligen las historias de usuario a desarrollar y las tareas de desarrollo.
- ❖ Se cambia el proceso cuanto sea necesario, para adaptarlo al proyecto.

#### Fase II: Diseño

- ❖ Se eligen los diseños funcionales más simples.
- ❖ Se elige una metáfora del sistema para que el nombrado de clases, siga una misma línea, facilitando la reutilización y la comprensión del código.
- ❖ Se escriben tarjetas de clase-responsabilidades-colaboración (CRC) para cada objeto, que permitan abstraerse al pensamiento estructurado y que el equipo de desarrollo completo participe en el diseño.

#### Fase III: Codificación

- ❖ El cliente está siempre disponible, de ser posible, cara a cara. La idea es que forme parte del equipo de desarrollo, y esté presente en todas las fases



de XP. La idea es usar el tiempo del cliente para estas tareas en lugar de crear una detallada especificación de requisitos, y evitar la entrega de un producto insuficiente, que le hará perder tiempo.

- ❖ El código se ajustará a unos estándares de codificación, asegurando la consistencia y facilitando la comprensión y refactorización del código.
- ❖ Las pruebas unitarias se codifican antes que el código en sí, haciendo que la codificación de este último sea más rápida, y que cuando se afronte la misma se tenga más claro, qué objetivos tiene que cumplir lo que se va a codificar.
- ❖ La programación del código se realiza en parejas, para aumentar la calidad del mismo. En cada momento, sólo habrá una pareja de programadores que integre código.
- ❖ Se integra código y se lanza dicha integración de manera frecuente, evitando divergencias en el desarrollo y permitiendo que todo el mundo trabaje con la última versión del desarrollo. De esta manera, se evitará pasar grandes períodos de tiempo integrando el código al final del desarrollo, ya que las incompatibilidades serán detectadas enseguida.
- ❖ Se usa la propiedad colectiva del código, lo que se traduce en que cualquier programador puede cambiar cualquier parte del código. El objetivo es fomentar la contribución de ideas por parte de todo el equipo de desarrollo.
- ❖ Se deja la optimización para el final.
- ❖ No se hacen horas extra de trabajo.

#### **Fase IV: Pruebas**

- ❖ Todo el código debe tener pruebas unitarias, y debe pasarlas antes de ser lanzado.
- ❖ Cuando se encuentra un error de codificación o bug, se desarrollan pruebas para evitar volver a caer en el mismo.
- ❖ Se realizan pruebas de aceptación frecuentemente, publicando los resultados de las mismas. Estas pruebas son generadas a partir de las user



stories elegidas para la iteración, y son "pruebas de caja negra", en las que el cliente verifica el correcto funcionamiento de lo que se está probando. Cuando se pasa la prueba de aceptación, se considera que el correspondiente user storie se ha completado.

## 1.7 Patrones arquitectónicos

**Patrón de arquitectura MCV (Modelo vista controlador)** - Para el diseño de aplicaciones con sofisticadas interfaces se emplea el patrón de diseño MCV. La lógica de una interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. Si se realiza un diseño ofuscado, es decir, una forma de mezclar los componentes de interfaz y de negocio, entonces, la consecuencia será que, cuando se necesite cambiar la interfaz, tendrá que modificarse trabajosamente los componentes de negocio, por lo que propiciará mayor trabajo y más riesgo de error.

Se trata de realizar un diseño que desacople la vista del modelo, con el fin de perfeccionar la reusabilidad. De este modo las modificaciones son las que impactan en menor medida en la lógica de negocio o de datos. (Jiménez, 2009)

Elementos del patrón:

- ✓ Modelo: datos y reglas de negocio.
- ✓ Vista: muestra la información del modelo al usuario.
- ✓ Controlador: gestiona las entradas del usuario.

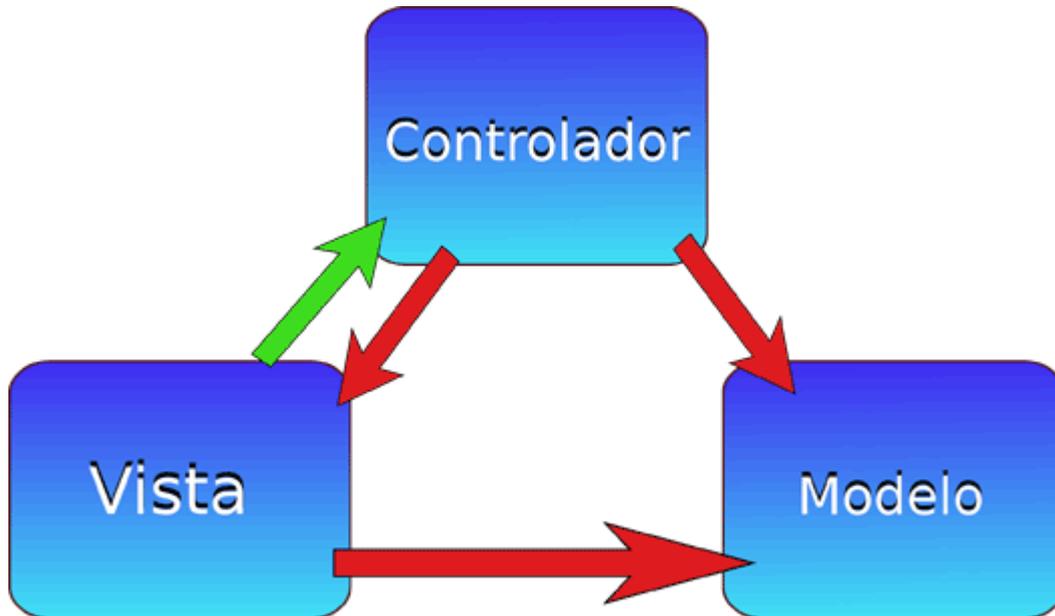


Figura 1.1 Modelo vista controlador

Un modelo puede tener diversas vistas, cada una con su correspondiente controlador. Un ejemplo clásico es el de la información en una base de datos, que puede presentarse de diversas formas: diagrama de pastel, de barras, tabular, etc. Analizan cada componente:

El modelo es responsable de:

- ❖ Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- ❖ Definir las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: “si la mercancía solicitada no está en el almacén, consultar el tiempo de entrega estándar del proveedor”.
- ❖ Llevar un registro de las vistas y controladores del sistema.
- ❖ Si se está en presencia un modelo activo, el mismo notificará a las vistas los cambios que en los datos pueda producir un agente externo (ejemplo: un fichero bat que actualiza los datos, un temporizador que desencadena una inserción, etc.).



---

El controlador es responsable de:

- ❖ Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- ❖ Contiene reglas de gestión de eventos, del tipo “si el evento z, entonces acción w”. estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada a actualizar. Una petición al modelo puede ser “obtener\_tiempo\_de\_entrega dada una nueva orden”.

Las vistas son responsables de:

- ❖ Recibir datos del modelo y mostrarlo al usuario.
- ❖ Tienen un registro de su controlador asociado, (normalmente porque además lo demanda).
- ❖ Pueden suministrar el servicio de actualizar, para que sea solicitado por el controlador o por el modelo, (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

**Arquitectura en capas** - Los sistemas o arquitecturas en capas constituyen uno de los estilos que aparecen con mayor frecuencia mencionados como categorías mayores del catálogo o por el contrario, como una de las posibles imágenes de algún estilo envolvente. (Reynoso, Kiccillof, 2004) definen el estilo en capas como una organización jerárquica tal, que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones de la inmediatamente inferior.

La arquitectura por capas es un estilo de arquitectura en la que el objetivo primordial es la separación de la lógica de negocio de la lógica de diseño, un ejemplo básico es separar la capa de datos, de la capa de presentación al

usuario. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de los niveles, simplemente es necesario conocer las API que existen entre niveles.

El diseño de sistemas informáticos suele usar las arquitecturas multinivel o *programación por capas*. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables, (que pueden ampliarse con facilidad en caso de que las necesidades aumenten). El diseño más en boga actualmente es el diseño en tres capas.

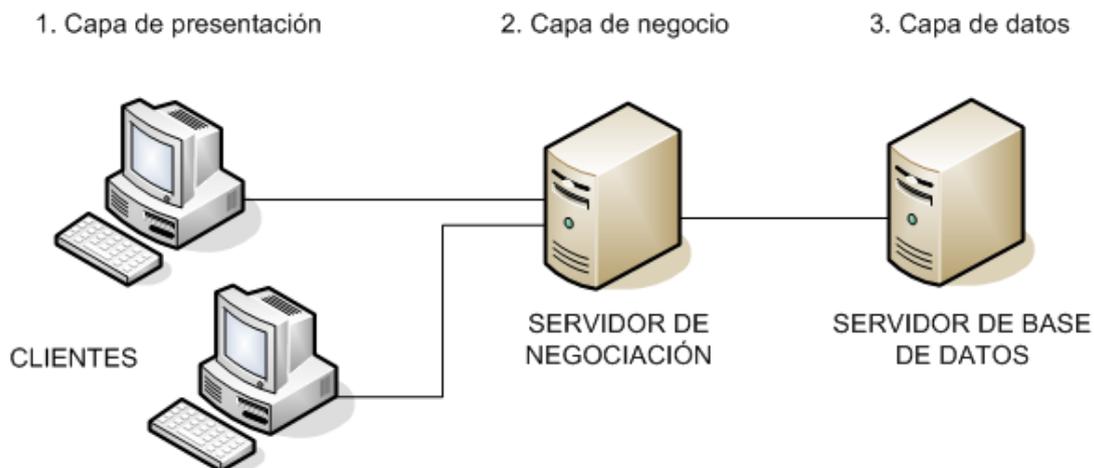


Figura 1.2 arquitectura en tres capas

### Capas o niveles

- ❖ Capa de presentación o interface: es la capa de que le permite al usuario interactuar con el sistema, captura y le comunica la información al mismo, dando un mínimo de proceso, (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la del negocio.



- ❖ Capa de lógica o de negocio: es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio e incluso lógica del negocio, pues es aquí donde se establecen las reglas que deben cumplirse. Esta capa se comunica con la de presentación para recibir las solicitudes y presentar los resultados, y con la capa de datos para solicitar al gestor de bases de datos para almacenar o recuperar datos de él.
- ❖ Capa de datos: es donde se ubican los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de los mismos, reciben solicitudes de almacenamiento o de recuperación de información desde la lógica del negocio.

Todas estas capas pueden residir en un único ordenador, esto no sería lo normal, lo más usual es que haya una multitud de ordenadores donde reside la capa de interface (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en un mismo ordenador, y si el crecimiento de las necesidades lo aconseja, pueden dividirse en dos o más ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, pueden separarse en varios ordenadores los cuáles recibirán las peticiones del ordenador en que resida la capa de negocio. Si por el contrario, la complejidad fuese en la capa de negocio lo que obligase a la separación, esta lógica del negocio podría residir en uno o más ordenadores que realizarían las solicitudes a una única base de datos.

En una arquitectura de tres niveles, los términos “capas” y “niveles” no significan lo mismo ni son similares. El término capa hace referencia a la forma como una solución es segmentada desde el punto de vista lógico: interface/lógica del negocio/datos. En cambio, el término nivel, corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física.



Ejemplo:

Una solución de tres capas (interface, lógica, datos), que residen en un solo ordenador (interface+lógica+datos), se dice que la arquitectura de la solución es de tres capas y un nivel.

Una solución de tres capas que residen en dos ordenadores (interface+lógica, lógica+datos), se dice que la arquitectura de la solución es de tres capas y dos niveles.

Una solución de tres capas que residen en tres ordenadores, la arquitectura que la define es: una solución de tres capas y tres niveles.

Ventajas:

- ❖ El estilo soporta un diseño basado en niveles de abstracción crecientes, lo cual, a su vez permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.
- ❖ El estilo admite muy naturalmente optimizaciones y refinamientos.
- ❖ Proporciona una amplia reutilización. Al igual que los tipos de datos abstractos, se pueden utilizar diferentes implementaciones o versiones de una misma capa en a medida que soporten las mismas interfaces de cara a las capas adyacentes. Esto conduce a la posibilidad de definir interfaces de capa estándar, a partir de las cuáles pueden construirse extensiones o prestaciones específicas.

## 1.8 Tecnologías a utilizar

**PHP v5.0 o superior** - Las iniciales PHP significan "PHP Hypertext Pre-processor y se trata de un lenguaje de programación que es usado para la creación de aplicaciones para servidores, o creación de contenido dinámico



para sitios web. Es un lenguaje de programación usado generalmente en la creación de contenido para sitios o aplicaciones web. La versión 5 de PHP presenta un magnífico trabajo con el paradigma orientado a objeto que permite la reutilización de código entre otras facilidades.

Ventajas de trabajar con PHP comparado con otros lenguajes similares:

- ❖ Es un lenguaje multi plataforma.
- ❖ Rapidez de ejecución.
- ❖ Mantiene un bajo consumo de recursos de máquina.
- ❖ Gran seguridad, muy poca probabilidad de corromper los datos.
- ❖ Capacidad de conexión con la mayoría de los manejadores de bases de datos que se emplean en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- ❖ Posee una amplia documentación en internet, incluyendo una gran variedad de ejemplos y de ayudas.
- ❖ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- ❖ Permite las técnicas de programación orientada a objetos.
- ❖ Permite crear formularios para la web.
- ❖ No requiere definición de tipos de variables ni manejo detallado de bajo nivel.

**AJAX** - AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas se ejecutan en el cliente, es decir, en el navegador de los usuarios y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.



---

AJAX es una combinación de tres tecnologías ya existentes:

- ❖ XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- ❖ Document Object Model (DOM) accedido con un lenguaje script por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y JScript, para mostrar e interactuar dinámicamente con la información presentada.
- ❖ El objeto XMLHttpRequest para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar del XMLHttpRequest para realizar dichos intercambios.
- ❖ XML es el formato usado comúnmente para la transferencia de vuelta al servidor, aunque cualquier formato puede funcionar, incluyendo HTML pre formateado, texto plano, JSON (JavaScript Object Notation) y hasta EBML (meta lenguaje binario extendible).

Ventajas:

- ❖ Recuperación asíncrona de datos, el usuario no tiene que esperar después de una petición.
- ❖ Acercamiento del ambiente de escritorio a la web.
- ❖ No requiere plugins (es una aplicación informática que interactúa con otra aplicación para aportarle una función o utilidad específica).
- ❖ Se reduce el tamaño de la información intercambiada.

**MySQL** - MySQL es muy rápido, fiable y fácil de usar, surge para manipular bases de datos muy grandes. Es un sistema multiplataforma de base de datos relacionales, lo que da velocidad y flexibilidad, cuenta con un sistema de privilegios contraseñas, muy seguro que permite la autenticación básica para el



acceso al servidor. MySQL es un sistema de administración de base de datos. Opera en una arquitectura cliente/servidor. Es el sistema gestor de bases de datos “open source” más popular, o sea que puede ser bajado de Internet y usarlo sin tener que pagar, además que cualquiera puede estudiar su código y adecuarlo a las necesidades que requiera.

Luego de analizadas las características y facilidades del SGBD presentado, y la de la herramienta a desarrollar se decide usar el MySQL como SGBD, por las siguientes razones:

- ❖ No se necesita de un manejo complejo de la información.
- ❖ El PHP maneja más fácil al MySQL que al SQL Server, debido a la gran cantidad de funciones que tiene explícitas.
- ❖ El MySQL es multiplataforma.
- ❖ El MySQL no tiene precio en el mercado, se adquiere libremente.

**Apache** - Es el servidor web más utilizado en el mundo con un 57 % de cuota de mercado, frente al 20 % de Microsoft IIS y el 7 % de Netscape. Dentro de sus puntos fuertes se encuentran:

- ❖ Tiene interfaz con todos los sistemas de autenticación.
- ❖ Facilita la integración como "plug-ins" de los lenguajes de programación de páginas web, y dinámicas más comunes.
- ❖ Tiene integración en estándar del protocolo de seguridad SSL. (más utilizado).
- ❖ Provee interfaz a todas las bases de datos.



## 1.9 Arquitectura a utilizar

Una arquitectura es el conjunto de decisiones significativas sobre la organización del sistema software, la selección de los elementos estructurales y sus interfaces, con los que se compone el sistema, junto a su comportamiento, tal como se especifica en las colaboraciones entre esos elementos, la composición de los elementos estructurales y de comportamiento en subsistemas progresivamente más amplios, y el estilo de arquitectura que guía esta organización, estos elementos y sus interfaces, sus colaboraciones, y su composición, ejemplo: los patrones de diseño relacionados con el diseño de los objetos y frameworks de pequeña y mediana escala, que son aplicables al diseño de una solución para conectar los elementos de gran escala que se definen mediante los patrones de arquitectura, y durante el trabajo de diseño detallado para cualquier aspecto del diseño local. También se conocen como patrones de micro-arquitectura. El patrón fachada, que se puede utilizar para proporcionar la interfaz de una capa a la siguiente. La arquitectura en capas o arquitectura n capas es la que se ha seleccionado para darle solución a la problemática planteada anteriormente, debido a las facilidades presentadas anteriormente.

## 1.10 Herramientas a emplear en la propuesta de solución

**PhpDesigner 2008 v6.0.0** - Es un completo entorno de desarrollo y programación especialmente diseñado para los gurús de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL.

Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo momento las tareas, además de acceso directo a librerías de código o scripts



de uso habitual, utilidades diversas y toda suerte de herramientas, todo ello en una interfaz de diseño sencillo y elegante que se puede personalizar con nada menos que dieciocho temas distintos.

Cuenta con cliente de FTP y navegador de ficheros integrado, utilidades de corrección y autocompletado, búsqueda integrada en google y soporte para proyectos, además de usar un práctico esquema de color para la sintaxis del código fuente que facilita enormemente la programación. PhpDesigner soporta: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL. (citado en: <http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>)

**Embarcadero ER/Studio 6.0.1** - Es una herramienta de modelado de datos, se usa para el diseño y la construcción lógica y física de bases de datos. Su ambiente es de gran alcance y multinivel. ER/Studio se diseña para hacer más fácil de entender el estado actual de los datos de las empresas. Simple y fácil al usuario, ayuda a las organizaciones para tomar decisiones en cómo resolver embotellamientos de los datos, elimina redundancia y alcanza en última instancia usos de más alta calidad que entreguen datos más eficientes y exactos a la empresa. (citado en: [http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa\\_software&key=1153755975](http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa_software&key=1153755975).)

Ventajas:

- ❖ Si se está comenzando un nuevo diseño o está manteniendo una base de datos existente, ER/Studio se combina con las características para ayudarle a conseguir el trabajo hecho con eficacia.
- ❖ La creación de diagramas es clara y rápida.
- ❖ Tiene la posibilidad de realizar diagramas con desempeño rápido.



**XAMPP** - Es un servidor independiente de plataforma, software libre, incluye el servidor web Apache, los servidores de datos MySQL y SQLite, sus respectivos gestores phpMyAdmin y phpSQLiteAdmin, el intérprete del lenguaje homónimo PHP con los extras incluidos en Perl, el intérprete del lenguaje Perl, servidores de FTP como ProFTPD ó FileZilla FTP Server, las estadísticas Webalizer y OpenSSL, eAccelerator, Freetype2, libjpeg, libpng, zlib, Ming, etc. entre otros. El programa esta liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris y MacOS X.

- ❖ Incluye Chequeo de seguridad.
- ❖ Contiene un Panel de control.

Incluye herramientas extras (Webalizer, Mercury Mail, Conmutador PHP).  
(citado en: <http://e.wikipedia.org/wiki/XAMPP>.)

### 1.11 Conclusiones

En el presente capítulo se realizó una investigación sobre el estado del arte, lo que permite una aproximación del lector a los antecedentes, conceptualización y características de los contratos, se efectuó un estudio acerca de los sistemas automatizados vinculados al campo de acción, en el cuál se pudo observar que ninguna de las soluciones existentes se ajusta a las necesidades requeridas por la entidad. Se propuso como metodología de desarrollo de software a emplear XP por sus facilidades, documentación y flexibilidad.

Una vez definidos estos aspectos, es posible pasar al siguiente capítulo donde se presenta la solución propuesta.



---

## CAPÍTULO II - PLANEACIÓN Y DISEÑO

### 2.1 Introducción

En este capítulo, se introduce la fase de planeación y diseño, donde se detallan las necesidades del cliente, se describen las funcionalidades que serán objeto de automatización mediante el empleo de las historias de usuarios (HU), se realiza una estimación del esfuerzo necesario para las mismas y se establece un plan de iteraciones necesarias sobre el sistema, para su terminación.

### 2.2 Funcionalidades generales

La gestión de la información de los contratos en la UEB de Compras se realiza de forma manual, lo que hace el proceso más lento, con el consiguiente aumento de gastos de papel y otros materiales de oficina e incrementa la posibilidad de que se produzcan errores en el momento de manipular la información de los mismos, lo que trae como consecuencia afectaciones en la gestión comercial debido a que pueden ejecutarse servicios cuyos contratos no estén vigentes o por montos superiores a los existentes en los mismos.

De la descripción antes expuesta, se identifican como requisitos generales del sistema, *el análisis del estado de la contratación y el control de su ejecución.*

#### **Análisis del estado de la contratación y control de su ejecución**

Consiste en darle seguimiento a los contratos de compras que se encuentran en ejecución a través de la generación de reportes, los mismos incluyen, conocer el monto que se ha ejecutado de cada uno y el restante, la vigencia de los mismos, si presentan suplementos, la caducidad etc.



### 2.1.1 Personal relacionado con el sistema

Personas relacionadas con el sistema	Justificación
Especialista	Esta es la persona que tiene conocimiento en la materia de contratos de compras, y está encargada de la gestión de información de los mismos.
Administrador	Es la persona encargada de asesorar y dar seguimiento del estado del proceso de desarrollo.
Desarrollador	Es la persona responsable de llevar a cabo la implementación del sistema.

Tabla 2.1 Personas relacionadas con el sistema

### 2.2.2 Lista de reserva

Después de conocer el personal relacionado e identificar los requisitos generales, se procede a realizar el análisis de las funcionalidades que debe cumplir la aplicación para dar respuesta a los mismos. Para ello se enumerarán mediante una lista de reserva, las funcionalidades que el sistema debe ser capaz de cumplir.

De acuerdo a lo antes expuesto, el sistema debe ser capaz de:

- ❖ Insertar datos de los contratos.
- ❖ Agregar suplemento a los contratos.
- ❖ Mostrar estado de ejecución de los contratos.
- ❖ Actualizar información de los contratos
- ❖ Registrar contrato.



- ❖ Actualizar suplemento de los contratos.
- ❖ Insertar responsables de contratos.
- ❖ Modificar responsables.
- ❖ Mostrar responsables de contratos.
- ❖ Eliminar responsables.
- ❖ Insertar información de proveedores.
- ❖ Modificar información de proveedores.
- ❖ Mostrar información de proveedores.
- ❖ Insertar servicios de proveedores.
- ❖ Modificar servicios de proveedores.
- ❖ Mostrar servicios de proveedores.
- ❖ Insertar datos de las facturas de los contratos.
- ❖ Actualizar información de facturas.
- ❖ Listar información de las facturas.
- ❖ Subir documentos de los contratos.
- ❖ Subir documentos de los proveedores.

Para mayor información remitirse a [[Plantilla Lista de Reserva del Producto \(LRP\)](#)] en el expediente del proyecto.

### 2.2.3 Historias de usuario

Las HU, son la técnica utilizada en XP para detallar los requisitos del software. Son el resultado directo del intercambio entre los usuarios y desarrolladores a través de reuniones donde las conocidas *tormenta de ideas (brain storm)* arrojan no solo los requerimientos, sino también las posibles soluciones; representan una forma rápida de administrar las necesidades de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para gestionarlos, debido a que un requerimiento de software es descrito de forma concreta y sencilla utilizando el lenguaje común del usuario.



Las HU permiten responder ágilmente a los requerimientos cambiantes y aunque se redactan desde las perspectivas de los clientes, también los desarrolladores pueden brindar ayuda en la identificación de las mismas. Para definir las se emplea la siguiente plantilla.

### Modelo de planilla de historia de usuario

Historia de usuario	
<b>Número:</b> No. Historia de usuario	<b>Usuario:</b> Usuario entrevistado para obtener la función requerida a automatizar.
<b>Nombre:</b> nombre de la historia de usuario que sirve para identificarla mejor entre los desarrolladores y el cliente.	
<b>Prioridad en el negocio:</b> Importancia: Alta / Media / Baja	<b>Riesgo en desarrollo:</b> Dificultad: Alta / Media / Baja
<b>Puntos estimados:</b> Estimación: de 1 a 3 puntos	<b>Iteración asignada:</b> Iteración a la que corresponde
<b>Programador responsable:</b> Nombre de encargado de programación.	
<b>Descripción:</b> Se especifican las operaciones por parte del usuario y las respuestas del sistema.	
<b>Observaciones:</b> Algunas observaciones de interés, como glosario, información sobre usuario etc.	

Tabla 2.2 Planilla de historia de usuario

### HU No.1: Gestión de contratos

Historia de usuario	
<b>Número:</b> 1	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Gestión de la información de los contratos.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> El usuario debe insertar los datos de los contratos, en caso de que requieran de suplemento, también debe ser insertada esta información, además debe ser almacenada la información de los contratos en el registro central de contratos. Una vez añadidos los contratos, se podrá mostrar y	



---

actualizar la información de los mismos.
<b>Observaciones:</b> Confirmado con el cliente.

Tabla 2.3 HU No.1: Gestión de contratos

Para ver las historias de usuario ver [[ANEXO 2: HISTORIAS DE USUARIO](#)].

### 2.3 Planificación de entregas

En esta fase se establece la prioridad de cada HU, y a continuación, se realiza una estimación del esfuerzo necesario de cada una de ellas por parte de los programadores. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debe obtenerse en no más de dos a tres meses.

Las estimaciones asociadas a la implementación de las historias se establecen empleando como medida el punto de estimación. Un punto de estimación equivale a una semana ideal de programación, donde los miembros de los equipos de desarrollo, trabajan el tiempo planeado sin ningún tipo de interrupción, este punto de estimación que se utiliza para representar la semana ideal, es de 5 días. Las historias generalmente tienen un valor de 1 a 3 puntos. Además, se mantiene un registro de la velocidad de desarrollo, establecida por puntos de iteración, basado fundamentalmente en la suma de los puntos de estimación correspondientes a las HU, que fueron terminadas en la última iteración.



### 2.3.1 Estimación de esfuerzo por historias de usuario

Para el buen desarrollo del sistema propuesto, se realizó una estimación para cada una de las HU identificadas, y se obtienen los resultados que se muestran a continuación:

#### Estimación de esfuerzo por historia de usuario

Historias de usuario	Puntos de estimación
Gestión de contratos	3 semanas
Gestión de responsables de contratos	2 semanas
Gestión de proveedores	2 semanas
Gestión de servicios de proveedores	2 semanas
Gestión de facturación	3 semanas
Subir el documentos de los contratos	1 semana
Subir documentos legales de proveedores	1 semana

Tabla 2.10 Estimación de esfuerzo por historia de usuario

### 2.3.2 Planificación de iteraciones

A partir de las HU antes expuestas y la estimación del esfuerzo propuesto para la realización de las mismas, se procede a realizar la planificación de la etapa de implementación del sistema, apoyándose en el tiempo e intentando concentrar las funcionalidades relacionadas en una misma iteración. En este plan se establece cuántas iteraciones serán necesarias realizar sobre el sistema para su terminación. El plan de iteraciones puede contener indicaciones sobre cuáles HU se incluirán en un release, lo cual debe ser consistente con el contenido de una o dos iteraciones.

En relación con lo antes tratado se decide realizar el sistema en 3 iteraciones, las cuales se explican de forma detalla a continuación:



### **Primera iteración:**

Esta iteración tiene como objetivo darle cumplimiento a las HU que se consideraron de mayor importancia para el desarrollo de la aplicación. Al concluir dicha iteración se contará con todas las funcionalidades descritas en las HU 1 y 2, las cuales hacen alusión a la inserción y visualización de la información de los contratos, la agregación de suplementos a estos, así como la asignación de responsables a los mismos. Además se tendrá la primera versión de prueba, que contará con dos modelos de desarrollo que incorporan todas las funcionalidades antes vistas, estos modelos se presentarán al cliente con el objetivo de obtener una retroalimentación del mismo para posteriores iteraciones del producto.

### **Segunda iteración:**

Esta iteración tiene como finalidad desarrollar las HU 3 y 4. Las mismas son las que brindan las funcionalidades de obtener y mostrar la información de los proveedores a contratar, y la gestión y visualización de información acerca de los servicios que prestan para su posible contratación. La versión que se obtenga de esta iteración en unión con la entregada en la iteración anterior se le facilitará al cliente para comprobar si cumple con las necesidades antes acordadas con él.

### **Tercera iteración:**

Esta última iteración del módulo tiene como propósito llevar a cabo el desarrollo de la HU 5, 6 y 7. La cuáles proporcionan las funcionalidades de: almacenar y visualizar los datos de las facturas de los contratos, subir el documento correspondiente a cada contrato y subir los documentos legales de cada proveedor. Estas HU serán integradas con el resultado de las iteraciones anteriores, y como fruto de esta integración se obtendrá la versión 1.0 del producto final. A partir de este momento el software será puesto a un proceso de prueba para evaluar el desempeño del mismo.



### 2.3.3 Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto guiado por la metodología de desarrollo de software XP, se crea el plan de duración de cada una de las iteraciones que se llevarán a cabo durante el desarrollo del mismo. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada una de las mismas.

#### Plan de duración de las iteraciones

Iteración	Historias de usuario	Duración total
Iteración 1	Gestión de contratos	5 semanas La entrega se realizará en la primera semana de marzo
	Gestión de responsables de los contratos	
Iteración 2	Gestión de proveedores	4 semanas La entrega se realizará en la primera semana de abril
	Gestión de servicios de Proveedores	
Iteración 3	Gestión de facturación	6 semanas La entrega se realizará en la primera semana de mayo
	Subir documentos de los contratos	
	Subir documentos legales de proveedores	

Tabla 2.11 Plan de duración de las iteraciones

## 2.4 Conclusiones

En este capítulo se abordó la fase de planeación y diseño donde se delinearon las HU con la participación del cliente, se llevó a efecto la planificación de iteraciones de cada HU a partir de la estimación del esfuerzo necesario de las mismas, culminando así esta fase y se determina que el equipo de trabajo está listo para pasar a la siguiente etapa de desarrollo.



---

## CAPÍTULO III - DESARROLLO Y PRUEBAS

### 3.1 Introducción

En este capítulo se inicia la fase de desarrollo y pruebas conforme a la metodología XP. Se presenta el modelo de datos empleado para la aplicación concluyente, se presentan además las tarjetas, clases, responsabilidades y colaboradores, que permitirán trabajar con una metodología basada en objetos, se realiza el desarrollo de las iteraciones a partir del desglose de las HU en tareas. Se muestran las interfaces gráficas de usuario diseñadas para la aplicación final. Se describen igualmente las pruebas realizadas y se indican las respuestas de la aplicación en el empleo de las diferentes funcionalidades, así como los posibles mensajes de error, información o aceptación que emite la misma cuando se utiliza una de estas funcionalidades.

### 3.2 Modelo de datos

En esta parte se muestra el modelo de datos empleado para la aplicación. En la gestión de contratos es empleada la tabla cp\_contratos, donde se almacenan los datos de los contratos y la tabla cp\_reg\_central en la cual se registran los contratos. Cuando se le hace un suplemento a un contrato estos datos son almacenados en la tabla cp\_suplemento. En la tabla cp\_responsables se guarda la información de los responsables a los que se les asignan los contratos para su control y seguimiento. Las tablas cp\_tipo\_contrato y cp\_vigencia, son empleadas para almacenar el tipo de contrato que se registra y el tiempo de vida de cada contrato respectivamente.

Para la gestión de proveedores son utilizadas las tablas: cp\_proveedores, donde se almacena la información de los proveedores, y la tabla cp\_servicios,

la cual guarda los datos de los servicios de proveedores. Para la gestión de facturación se utiliza la tabla cp\_facturas, donde se guarda el monto ejecutado de cada contrato. Por último para subir los documentos de los contratos se emplea la tabla cp\_documentos.

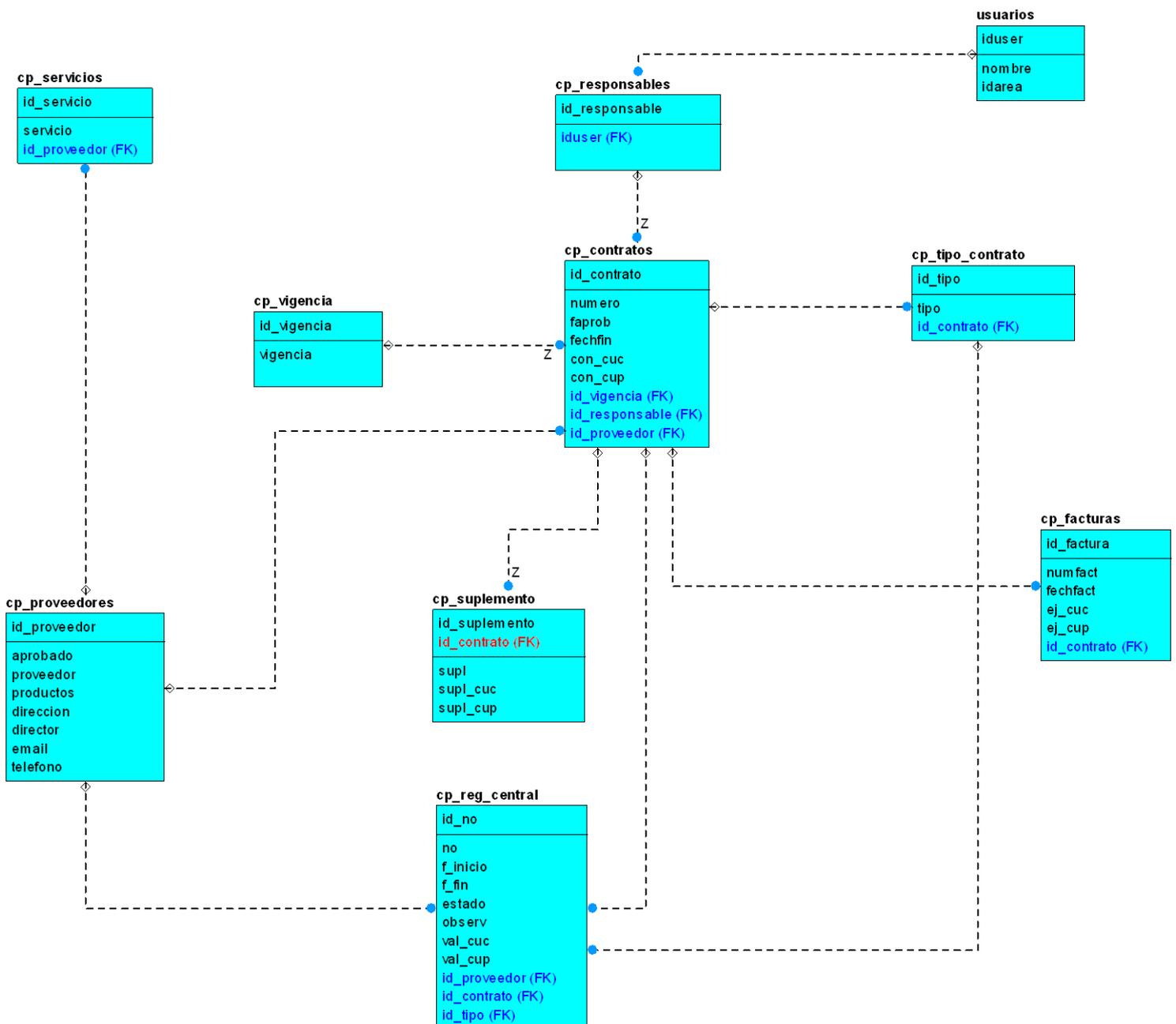


Fig. 3.1 Modelo de datos



### 3.3 Clases, responsabilidades y colaboradores

En este epígrafe tiene lugar la realización de las tarjetas de clases, responsabilidades y colaboración, conocidas tradicionalmente como *tarjetas CRC*, las cuáles se realizan con el objetivo de facilitar la comunicación y documentar los resultados. Además, las mismas permiten la total participación y contribución del equipo de desarrollo en la tarea de diseño. Una tarjeta CRC representa un objeto, por tanto es una clase, cuyo nombre se ubica en forma de título en la parte superior de la tarjeta, los atributos y las responsabilidades más significativas se colocan a la izquierda y las clases implicadas con cada responsabilidad a la derecha, en la misma línea de su requerimiento correspondiente. Para mejor comprensión de las mismas, se determina agruparlas por HU.

Para ver las tarjetas CRC ver [\[ANEXO 3: TARJETAS CRC\]](#).

### 3.4 Desarrollo de las iteraciones

Durante la fase planificación y diseño fueron detalladas las HU correspondientes a cada una de las iteraciones a desarrollar, teniendo en cuenta las prioridades y restricciones de tiempo, previstas por el cliente.

#### 3.4.1 Tareas por historias de usuario

Dentro del contenido de este plan, las HU se descomponen en tareas de programación o ingeniería, y a su vez, estas son asignadas al equipo de desarrollo para su implementación. Las tareas no tienen que ser entendidas necesariamente por el cliente, pues las mismas, sólo son utilizadas por los miembros del equipo de desarrollo, por lo que pueden ser escritas en lenguaje técnico. Las mismas se representan mediante las tarjetas de tareas.

**Distribución de tareas por historia de usuario**

Historia de usuario	Tareas
<b>Gestión de contratos</b>	<ul style="list-style-type: none"> <li>• Insertar los datos de los contratos.</li> <li>• Modificar los datos de los contratos.</li> <li>• Agregar suplemento al contrato.</li> <li>• Mostrar estado de los contratos.</li> </ul>
<b>Gestión de responsables de contratos</b>	<ul style="list-style-type: none"> <li>• Insertar responsable de contrato.</li> <li>• Modificar responsable de contrato.</li> <li>• Eliminar responsable de contratos.</li> <li>• Mostrar responsables de contratos.</li> </ul>
<b>Gestión de proveedores</b>	<ul style="list-style-type: none"> <li>• Insertar información de proveedor.</li> <li>• Modificar información de proveedor.</li> <li>• Eliminar proveedor.</li> <li>• Mostrar información de proveedores.</li> </ul>
<b>Gestión de servicios de proveedores</b>	<ul style="list-style-type: none"> <li>• Insertar servicio de proveedor.</li> <li>• Modificar servicio de proveedor.</li> <li>• Eliminar servicio de proveedor.</li> <li>• Mostrar servicios de proveedores</li> </ul>
<b>Gestión de facturación</b>	<ul style="list-style-type: none"> <li>• Insertar datos de factura.</li> <li>• Modificar datos de factura.</li> <li>• Mostrar facturas</li> </ul>
<b>Subir el documento digital del contrato</b>	❖ Subir documentos de los contratos.
<b>Subir documentos legales de proveedores</b>	❖ Subir documentos legales.

Tabla 3.1 Distribución de tareas por historia de usuario

**Historias de usuario abordadas en la primera iteración**

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestión de contratos	3	2.3
Gestión de responsables de contratos	2	1

Tabla 3.2 Historias abordadas en la primera iteración

**Historias de usuario abordadas en la segunda iteración**

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestión de proveedores	2	1
Gestión de servicios de	2	1



---

proveedores		
-------------	--	--

Tabla 3.10 Historias de usuario abordadas en la segunda iteración

### Historias de usuario abordadas en la tercera iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestión de facturación	2	1.4
Subir documentos de los contratos	2	1
Subir documentos legales de los contratos	2	1

Tabla 3.18 Historias de usuario abordadas en la tercera iteración

Para ver las tarjetas de ingeniería ver [[ANEXO 5: TARJETAS DE TAREAS](#)].

### 3.5 Pruebas

En la Programación Extrema es esencial el desarrollo de las pruebas, permitiendo probar continuamente el código. Cada vez que se desea implementar las funcionalidades que tendrá el software, XP propone una redacción sencilla de prueba, para ser pasada por el código posteriormente. El proceso constante de las pruebas permite la obtención de un producto con mayor calidad, y se ofrece a los programadores una mayor certeza en el trabajo que desempeñan. En la metodología XP hay dos tipos de pruebas; las unitarias o desarrollo dirigido por pruebas (TDD test driven development), desarrolladas por los programadores verificando su código de forma automática, y las pruebas de aceptación, las cuáles son evaluadas luego de culminar una iteración, se verifica así, que se cumplió la funcionalidad requerida por el cliente. Con estas normas se obtiene un código simple y funcional de manera bastante rápida y eficiente. Por esto es importante pasar las pruebas al 100%.



### 3.5.1 Desarrollo dirigido por pruebas

El desarrollo dirigido por pruebas, se enfoca en la implementación orientada a pruebas. El código debe ser probado paso a paso para lograr un resultado, aunque no con lógica para el negocio, pero si funcional. Algunas personas confunden este término con las llamadas “pruebas de caja blanca” las cuáles se les practican a los métodos u operaciones para medir la funcionalidad del mismo, desde el punto de vista de validez del cliente. Sin embargo, el TDD se aplica antes de comenzar a implementar cada paso de la tarea en desarrollo, asumiendo que la prueba es insatisfactoria desde un inicio. Sólo una vez que se haya cumplido de la forma más sencilla posible la lógica del código a probar se asume como cumplida. Luego se realiza un proceso conocido como “refactorización” de código perteneciente a una de las doce prácticas planteadas por la metodología XP, el cual consiste en mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez. Es esencia el TDD y las pruebas de caja blanca se enfocan en la lógica del negocio.

### 3.5.2 Pruebas de aceptación

Las pruebas de aceptación en XP, se pueden asociar con las pruebas de caja negra que se aplican en otras metodologías de desarrollo, sólo que se crean a partir de las historias de usuario y no por un listado de requerimientos. Durante las iteraciones, las HU se traducen a pruebas de aceptación. En ellas se especifican desde la perspectiva del cliente, los escenarios para probar que la HU ha sido implementada correctamente. La misma puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo que persiguen estas pruebas, es garantizar que las funcionalidades solicitadas por el cliente han sido realizadas. Una HU no se considera completa hasta que no ha transitado por sus pruebas de aceptación. Luego de ver los



paradigmas anteriores empleados para la realización de las pruebas y reunirse con el cliente para su análisis, el mismo decidió que se lleve a cabo el proceso mediante las pruebas de aceptación.

La planilla utilizada para plasmar el contenido de las pruebas de aceptación se muestra a continuación.

### Planilla de prueba de aceptación

<b>Prueba de aceptación</b>
<b>HU:</b> Nombre de la historia de usuario que va a comprobar su funcionamiento.
<b>Nombre:</b> Nombre del caso de prueba.
<b>Descripción:</b> Descripción del propósito de la prueba.
<b>Condiciones de ejecución:</b> Precondiciones para que la prueba se realice.
<b>Entrada/Pasos ejecución:</b> Pasos para probar la funcionalidad.
<b>Resultado:</b> Resultado que se desea de la prueba.
<b>Evaluación de la prueba:</b> Aceptada o denegada.

Tabla 3.24 Planilla de prueba de aceptación.

Para ver las pruebas de aceptación ver [\[ANEXO 6: PRUEBAS DE ACEPTACION\]](#).

### 3.6 Conclusiones

En este capítulo se llevó a cabo la fase de desarrollo y diseño donde se presenta el modelo de datos de la aplicación a obtener, logrando una visión detallada de sus atributos y las relaciones entre sus clases, además, las principales clases mediante el empleo de las tarjetas CRC. Se realiza el desarrollo de las iteraciones a partir de la distribución de tareas por HU, y se les practica las pruebas de aceptación a las funcionalidades de mayor importancia.



---

## CAPÍTULO IV - ESTUDIO DE FACTIBILIDAD

### 4.1 Introducción

La probabilidad de restricciones, insuficiencia de recursos y presiones en las fechas de entregas implícitas en el desarrollo en un proyecto, es real, por lo que es ineludible y sensato evaluar la viabilidad del mismo. De identificarse en la fase de definición, posibles riesgos en la concepción del proyecto, es válido minimizar de forma razonable recursos humanos, materiales y financieros. En general los productos informáticos no están exentos de estas dificultades, por lo que es de vital importancia estimar la relación costo-beneficio, así como el esfuerzo, capital humano y el tiempo de desarrollo que se demanda en la ejecución de los mismos.

### 4.2 Métodos para la estimación del esfuerzo

**Análisis de puntos de estimación y COCOMO II** - El SEI (del inglés, Software Engineering Institute), propone desde hace algunos años un método para la estimación del esfuerzo llamado COCOMO II (Construction Cost Model). El mismo está basado en ecuaciones matemáticas que permiten calcular el esfuerzo a partir de ciertas métricas de tamaño estimado, como el análisis de puntos de función y las líneas de código fuente (en inglés SLOC, Source Lines of Code).

**Análisis de puntos de casos de uso** - Existe una posibilidad de predecir el tamaño de un sistema a partir de las características de sus requisitos, expresados en los casos de uso. La estimación mediante el análisis de puntos de casos de uso, es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores. Se trata de



de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de “pesos” a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

### 4.3 Estudio de factibilidad

Para la estimación del proyecto en la investigación se emplea el método del análisis de puntos de casos de uso, ya que el sistema cuenta con siete HU, siendo esta una vía sencilla que cuenta con un software denominado ESTIMAC (cálculo de la estimación de proyectos informáticos basado en puntos de casos de uso) para proporcionar su uso.

A continuación, se detallan los pasos a seguir para la aplicación de este método.

El primer paso para la estimación consiste en el cálculo de *puntos de casos de uso sin ajustar*, este valor se obtiene a partir de la siguiente ecuación:

$$\mathbf{UUCP = UAW + UUCW}$$

Donde:

UUCP (unadjusted use cases points): puntos de casos de uso sin ajustar.

UAW (unadjusted actors weight): factor de peso de los actores sin ajustar.

UUCW (unadjusted use cases weight): factor de peso de los casos de uso sin ajustar.



### Factor de peso de los actores sin ajustar (UAW)

Este valor se calcula mediante un análisis de la cantidad de actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los actores se establece teniendo en cuenta en primer lugar si se trata de una persona otro sistema, y en segundo lugar la forma en que el actor interactúa con el sistema.

### Factor de peso de los actores sin ajustar

Tipo	Peso	Valor	Peso*Valor
Simple	1	0	0
Medio	2	0	0
Complejo	3	3	9
			<b>UAW: 9</b>

Tabla 4.1 Factor de peso de los actores sin ajustar.

### Factor de peso de los casos de uso sin ajustar (UUCW)

Este valor se calcula mediante un análisis de la cantidad de casos de uso (CU) presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los CU se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómicas, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia.

### Factor de peso de los CU sin ajustar

Tipo	Peso	Valor	Peso*Valor
Simple	5	5	25
Medio	10	2	20
Complejo	15	0	0
			<b>UUCW: 45</b>

Tabla 4.2 Factor de peso de los CU sin ajustar.



Finalmente los puntos de CU sin ajustar resultaron:

$$\mathbf{UUCP = UAW + UUCW}$$

$$\mathbf{UUCP = 9 + 45}$$

$$\mathbf{UUCP = 54}$$

Una vez que se tienen los puntos de CU sin ajustar, se debe ajustar este valor calculando los puntos de CU ajustados mediante la ecuación:

$$\mathbf{UCP = UUCP \times TCF \times EF}$$

Donde:

UCP (use cases points): puntos de casos de uso ajustados.

UUCP (unadjusted use cases points): puntos de casos de uso sin ajustar.

TCF (technical complexity factor): factor de complejidad técnica.

EF (environment factor): factor de ambiente.

### **Factor de complejidad técnica**

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica. Cada uno de estos factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

El factor de complejidad técnica se calcula mediante la ecuación:

$$\mathbf{TCF = 0.6 + 0.01 \times \Sigma (\text{Peso} \times \text{Valor asignado})}$$

$$\mathbf{TCF = 0.6 + 0.01 \times 18}$$

$$\mathbf{TCF = 0.78}$$

**Factor de complejidad técnica**

Factor	Peso	Valor	Peso*Valor
T1	2	0	0
T2	1	4	4
T3	1	3	3
T4	1	1	1
T5	1	3	3
T6	0.5	1	0.5
T7	0.5	1	0.5
T8	2	1	2
T9	1	1	1
T10	1	1	1
T11	1	1	1
T12	1	1	1
T13	1	0	0
<b>Total</b>			<b>18</b>
			<b>TCF: 0.78</b>

Tabla 4.3 Factor de complejidad técnica.

**Factor de ambiente**

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del factor de ambiente. El cálculo del mismo es similar al del factor de complejidad técnica, es decir, se trata de un conjunto de factores que cuantifican con valores de 0 a 5.

El factor ambiente se calcula:

$$EF = 1.4 - 0.03 \times \Sigma (\text{Peso} \times \text{Valor asignado})$$

$$EF = 1.4 - 0.03 * 24.5$$

$$EF = 0.665$$

**Factor de ambiente**

Factor	Peso	Valor	Peso*Valor
E1	1.5	3	4.5
E2	0.5	3	1.5
E3	1	3	3
E4	0.5	3	1.5
E5	1	5	5
E6	2	5	10
E7	-1	0	0
E8	-1	1	-1
<b>Total</b>			<b>24.5</b>
			<b>EF: 0.665</b>

Tabla 4.4 Factor de ambiente.

Finalmente los puntos de casos de uso ajustados resultan:

$$\text{UCP} = 54 * 0.78 * 0.665$$

$$\text{UCP} = 28.0098$$

Luego el esfuerzo en horas-hombre viene dado por:

$$\text{E} = \text{UCP} \times \text{CF}$$

$$\text{E} = 28.0098 * 20$$

$$\text{E} = 560.196$$

Donde:

E (effort): esfuerzo estimado en horas-hombre.

UCP (use cases points): puntos de casos de uso ajustados.

CF (conversion factor): factor de conversión.

Este método proporciona una estimación del esfuerzo en horas-hombre contemplado sólo el desarrollo de la funcionalidad especificada en los CU. Para una estimación más exacta de la duración completa del proyecto, hay que agregar a la estimación del esfuerzo obtenida por los puntos de CU, las



estimaciones de esfuerzo de las demás actividades relacionadas con el desarrollo de software.

#### Estimación del esfuerzo total del proyecto

Actividad	% Esfuerzo	Valor esfuerzo
Análisis	10	140,049
Diseño	20	280,098
Implementación	40	560,196
Prueba	15	210,0735
Sobrecarga	15	210,0735
<b>Total</b>	<b>100%</b>	<b>1400,49</b>

Tabla 4.4 Estimación del esfuerzo total del proyecto.

El proyecto requiere de 560.196 horas-hombres para su desarrollo, por tanto si trabaja 1 persona, 40 horas como promedio en una semana, el proyecto se puede terminar en aproximadamente 225,298125 días (8 meses)

Costo por Mes (CM) y Costo General (CG):

CM = Salario Total

CM = Salario \* cantidad de obreros

CM = 100.00 \* 1

CM = 100.00 Pesos por mes.

CG = CM \* TDEV

CG = 100.00 \* 8

CG = 800.00 Pesos

El desarrollo del sistema tendrá un costo estimado de \$ 800.00.



#### 4.4 Análisis de sostenibilidad

Cuando un sistema informático se instaura, éste repercute de forma positiva y/o negativa en los usuarios finales, por lo que es necesario efectuar el análisis de su impacto en las dimensiones administrativas, socio-humanistas, ambientales y tecnológicas, para contribuir al desarrollo sostenible del producto informático. Se ha afirmado con frecuencia que las nuevas tecnologías informáticas tendrán un impacto económico tan fuerte como el de la revolución industrial, de hecho, gracias a los progresos de las tecnologías informáticas y de las comunicaciones, el modo en que los directivos gestionan la información en sus empresas ha cambiado completamente, pasando de los informes impresos al uso de programas informáticos que les ofrezcan la información que precisan.

El desarrollo de este subsistema no supone grandes gastos de recursos, ni de tiempo; la base de datos que contiene puede ser alojada en los sistemas de gestión existentes en la empresa, ya que los mismos tienen buenas prestaciones y acceso rápido. La tecnología empleada para el desarrollo del subsistema es totalmente libre, no requiere incurrir en gastos adicionales, en el pago de licencias de uso. Es portable por lo que un cambio de plataforma para la implantación del mismo es viable y factible, y no hay que infringir en muchos cambios.

Incluso cuando existe un gasto de electricidad asociado al consumo de los ordenadores y el servidor del sistema debe mantenerse encendido 24 horas, el mismo no aumentará el consumo actual al estar considerado en los niveles de gastos.



#### 4.5 Beneficios tangibles e intangibles

El módulo para la gestión de contratos de compras de la EMPLEI, no es un producto con fines comerciales, su principal objetivo es resolver los problemas que existentes durante el desarrollo de esta tarea en la UEB Compras. El beneficio fundamental del sistema es contar con una aplicación web flexible, dinámica y de interfaz agradable que le permita registrar, actualizar y conocer de una forma más precisa y en el menor tiempo posible datos de interés a los participantes en esta actividad.

Con el aprovechamiento de este módulo que se desarrolla, se consiguen beneficios directos al medio ambiente, ya que disminuye considerablemente el volumen de papel y tinta antes usado por parte de los técnicos, eliminando así el uso excesivo de estos recursos.

Los beneficios inmediatos por lo general son los siguientes:

- ❖ Disminución del tiempo y esfuerzo que se invierte en esta tarea realizada hasta el momento de forma manual.
- ❖ Disminución de la acumulación de materiales impresos relacionados con los procesos de contratación, facturación, etc.
- ❖ Ahorro de tiempo en la búsqueda de información de un contrato.
- ❖ Los usuarios pueden revisar el estado de la contratación en todo momento.
- ❖ Seguridad al guardar la información.
- ❖ El cliente ya cuenta con un sistema capaz de guardar grandes volúmenes de información de forma segura, eliminando los inconvenientes de que se pierdan o deterioren y que se acumulen de forma excesiva.
- ❖ La empresa dispone de toda la información relacionada con los contratos de compras de forma automatizada, lo que implica mayor seguridad y confiabilidad.



Unido al subsistema se entregará un manual de usuario donde los mismos podrán consultar en detalle el funcionamiento del sistema. Los tiempos de respuestas del sistema son mínimos, como máximo de dos a tres segundos y los principales retrasos están dados por el tráfico de la red, por lo que los usuarios podrán ejecutar sus operaciones de forma rápida. La implementación de la aplicación no tiene incidencias desfavorables sobre el medio ambiente.

Existe una estrecha comunicación entre los desarrolladores de la aplicación y los usuarios finales, logrando que el módulo que se presenta, satisfaga las necesidades y expectativas de los clientes finales.

Una vez efectuado el análisis del producto informático se puede arribar a las siguientes conclusiones:

- ❖ El producto desplegado ofrece respuesta a la necesidad que lo origina.
- ❖ Los recursos para su desarrollo son mínimos ya que se realizó con los disponibles y las tecnologías sobre las que funciona están basadas en software libre.
- ❖ Se empleó la filosofía de trabajo con que se realizan los sistemas en la empresa, lo que facilitará su mantenimiento y futuras actualizaciones.
- ❖ La solución que se propone es flexible a cambios.



---

## CONCLUSIONES GENERALES

El sistema se desarrolla siguiendo la metodología XP, y se utilizaron representaciones para la modelación de todas las fases del proyecto. El sistema resultante está provisto de un ambiente cómodo, fácil de entender, que cumple los estándares de diseño y utiliza técnicas modernas de programación orientada a objetos, para esto se siguieron los siguientes pasos:

- ❖ Se elaboró el marco teórico metodológico que fundamenta la investigación, permitiendo el análisis del proceso de gestión de la información de los contratos en la UEB de Compras para lograr una mejor comprensión del negocio.
- ❖ Se efectuó un estudio de las diferentes tecnologías y herramientas para la confección del sistema, haciendo una elección, de acuerdo a las especificaciones del cliente.
- ❖ Se efectuó un levantamiento de los requerimientos, los que propiciaron un mejor entendimiento de la problemática en cuestión, facilitando el análisis, diseño e implementación del sistema automatizado, el cual solucionó las dificultades antes expuestas.
- ❖ Se obtuvo un sistema informático que permite el control de la ejecución de los contratos en la “UEB” Compras.

Por todo lo anterior se concluye que los objetivos propuestos en el presente proyecto han sido cumplidos satisfactoriamente.



---

## RECOMENDACIONES

De manera general los objetivos trazados al inicio de esta investigación han sido logrados, al mismo tiempo, en el transcurso del proceso de desarrollo, ha quedado evidenciado, que la propuesta es sólo la primera fase de un proyecto que puede ser mucho más ambicioso. Por tanto se hacen las siguientes recomendaciones.

- ❖ Continuar trabajando en el sistema de contratación de compras con el objetivo de integrarlo al sistema general de contratación de los servicios de la entidad.



---

## REFERENCIAS BIBLIOGRÁFICAS

Agroads: *Software para el control de contratos de cereal*. [en línea].  
[Consultado: 2010-02-10]. Disponible en:  
<http://www.agroads.com.ar/guias/software-agropecuario.html>

BLANCO CRIADO, A. *XAMPP*, 2008. [en línea]. [Consultado: 2010-02-25].  
Disponible en: <http://www.adictosaltrabajo.com/tutoriales>

BECK, K. “Extreme Programming Explained. Embrace Change”, Pearson Education, 1999. Traducido al español como: “Una explicación de la programación extrema. Aceptar el cambio”, Addison Wesley, 2000.

Contrato [en línea]. [Consultado: 2010-01-07]. Disponible en:  
<http://es.wikipedia.org/wiki/Contrato>

Demarchi G. *Contratos*. [en línea]. [Consultado: 2010-01-07]. Disponible en:  
<http://www.monografias.com/trabajos6/cont/cont.shtml>

*Embarcadero ER/Studio*. [en línea]. [Consultado: 2010-02-25]. Disponible en:  
[http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa\\_software&key=1153755975](http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa_software&key=1153755975).

*ER/Studio*. [en línea]. [Consultado 2010-02-25]. Disponible en:  
<http://www.monografias.com/trabajos14/modelodebase/modelodebase.shtml>

FERRER, J. *Metodologías Ágiles*. [en línea]. [Consultado: 2010-02-25].  
Disponible en: <http://libresoft.es/downloads/ferrer-20030312.pdf>



---

Firebird Project. *Firebird*, 2008. [en línea]. [Consultado: 2010-01-22]. Disponible en: <http://es.wikipedia.org/wiki/Firebird>

GT y Cía. Ltda. *SCC: Sistema Control de contratos*. [en línea]. [Consultado: 2010-01-07]. Disponible en: <http://www.gtcia.com.co/scc.htm>

HotFrog. *Software administración de contratos*. [en línea]. [Consultado: 2010-02-10]. Disponible en: <http://www.hotfrog.cl/Companies/Administracion-de-Contratos-Ebackup>

*Introducción a herramientas CASE y System Architect*. Universidad Politécnica de Valencia. [en línea]. [Consultado: 2010-01-15], 2004 . Disponible en: [http://www.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro\\_case\\_SA.pdf](http://www.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf)

JEFFRIES, R. *What is Extreme Programming?* [en línea]. [Consultado: 2010-02-25]. Disponible en: <http://www.xprogramming.com/xpmag/whatisxp.htm>

JIMÉNEZ IGLESIAS, E.R. “*Sistema automatizado para la gestión del producto final de los servicios de la UEB Organización empresarial de la Empresa Empleadora del Níquel. (SGPFS)*”. Yeidel cuenca Aguilar (tutor). Tesis de Grado. Instituto Superior Minero Metalúrgico, 2009. 120 h.

MONMANY, J. *Aplicaciones web*. [en línea]. [Consultado: 2010-01-15]. Disponible en: <http://www.webvillage.info>

MySQL. [en línea]. [Consultado: 2010-02-25]. Disponible en: <http://es.wikipedia.org/wiki/MySQL>



---

PEÑA GUZMÁN, L. A; ARGÜELLO, L.R. *Derecho Romano*. Buenos Aires: Editorial: TEA, 1966. 261 p.

*PHP Designer 2008 v6.0.2.0 Professional*. [en línea]. [Consultado 2010-02-25].  
Disponble en: <http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>

REYNOSO, C; KICCILLOF, N. *Estilos y Patrones en la Estrategia de Arquitectura de Microsoft*. [en línea]. [Consultado: 2010-02-25]. Disponible en: [www.willydev.net/descargas/prev/Estiloypatron.pdf](http://www.willydev.net/descargas/prev/Estiloypatron.pdf)

SICE. *Sistema de Información para la Vigilancia de la Contratación Estatal*. [en línea]. [Consultado: 2010-02-10]. Disponible en: <http://www.sice-cgr.gov.co/>

VALDES DOMINGUEZ, M. El proceso de contratación económica a la luz de los cambios operados en la economía cubana. *La revista del empresario cubano*. [en línea]. [Consultado: 2010-02-10], abril de 2005. Disponible en: [http://www.betsime.disaic.cu/secciones/jur\\_mj\\_05.htm](http://www.betsime.disaic.cu/secciones/jur_mj_05.htm)

XAMPP. [en línea]. [Consultado: 2010-02-25]. Disponible en: <http://e.wikipedia.org/wiki/XAMPP>.

\_\_\_\_\_. [en Línea]. [Consultado: 2010-02-25]. Disponible en: <http://www.genbeta.com/2005/09/27-xampp-servidor-web-correo-base-de-datos-y-mas-extras-en-pocos-minutos>



---

## GLOSARIO DE TÉRMINOS

**Nexum:** El nexum era la figura contractual en el derecho romano. Si bien las referencias históricas son escasas y confusas, con lo que la opinión de los historiadores no es unánime, parece que por ese compromiso, el llamado nexum, el deudor o *nexi* cedía legal (e incluso con fórmulas rituales) poder de dominio sobre su persona al acreedor, fuera de efectos inmediatos o aplazados hasta el momento de quedar incumplida la deuda. Los *nexi* eran ciudadanos empobrecidos, en general plebeyos en tanto que carecían de otras redes de contención, que así se habían dado ellos mismos en prenda de acreedor. El compromiso implicaba dominio, incluyendo la posibilidad de que el acreedor lo vendiera como esclavo, le diera cárcel o incluso muerte.

**Librepiens:** En el derecho romano se llamaba así, el funcionario que tenía a su cargo el empleo de la balanza que servía para pesar el metal que en determinados actos jurídicos, como la *mancipatio* y el *nexum*, entregaba una de la partes a la otra en concepto de pago. Tuvo gran importancia en la época primitiva con relación a los actos *per aes et libram*.

**Sponsio:** Un *sponsio*, en *derecho romano* se definía originalmente como votos o promesas a la divinidad y como *esponsales* o promesa de futuro matrimonio. Más tarde se sostiene la tesis de que el *sponsio* había sido un procedimiento de caución, es decir, una garantía. De acuerdo a esto el *sponsor* no era el deudor propiamente como tal, sino el garante, o sea quien respondía ante el incumplimiento del deudor. A diferencia de la estipulación, el *sponsio* era accesible solo para los ciudadanos romanos (*actio iuris civilis*).

**Per aes et libram:** Quiere decir «por el cobre (o bronce) y la balanza». Suele emplearse como calificativo de ciertos negocios jurídicos formales, que proceden de una época en la que los trozos y barras de cobre no amonedado



constituían el medio usual de pago. La principal aplicación del mismo es la *mancipatio* o *mancipium*, acto privado y solemne, que consiste en una declaración de un adquirente, *mancipio accipiens*, que se apodera formalmente de la cosa, en presencia del propietario de la misma, *mancipio dans*, y en el acto de pesar un metal en una balanza de platillos; se realizaba este negocio en presencia de cinco testigos, más otro encargado de pesar el metal, llamado *libripens*. La *mancipatio* servía para adquirir el poder sobre las personas y cosas integrantes de la familia (bienes que se suelen heredar) y especialmente la propiedad de la *res Mancipi*. Es un acto exclusivo de los ciudadanos, latinos o peregrinos con *commercium*; también uno de los llamados *actus legitimi*, es decir, que no tolera que sus efectos sean sometidos a un acontecimiento futuro incierto (condición) o cierto (término).

**Stipulatio:** Consiste en una pregunta del acreedor (*stipulante*) y una respuesta congruente del deudor (*promitente*). Era un contrato unilateral y de derecho estricto, el acreedor realizaba una pregunta oral al deudor y este debía dar una respuesta afirmativa igualmente oral. En la pregunta que hacía el acreedor debía incluir claramente el objeto de la obligación y en la respuesta el deudor debía comprender expresamente o tácitamente dicho objeto. Por ejemplo: ¿Prometes darme 500 sestercios? Y el deudor contestaba prometo darte 500 sestercios y así quedaba obligado.

**Chirographa:** es un escrito emanado de la propia mano del deudor, donde se reconocía una deuda, sellada por él y entregada por el acreedor. Es sólo un escrito. Se parece a nuestro actual 'pagaré'.

**Syngrapha:** Se hacen dos ejemplares y se intercambian. Se entiende que de estos documentos no nace propiamente una obligación civil pero asimila obligaciones para las relaciones con extranjeros.



**Mutuo:** El mutuo es un contrato de préstamo de dinero que suele ser remunerado mediante el pago de intereses en función del tiempo. Si el mutuo se encuentra garantizado mediante un derecho real de hipoteca se denomina mutuo hipotecario.

**Comodato:** El comodato es un contrato por el cual una parte entrega a la otra gratuitamente una especie, mueble o bien raíz, para que haga uso de ella, con cargo de restituir la misma especie después de terminado el uso.

**Nómina transcriptitia:** Anotaciones vinculadas con el *codex accepti et expensi* (libro del debe y haber). Libro que llevaban los *paters*. Cuando las anotaciones realizadas correspondían a efectivas transferencias de dinero, hacían nacer obligaciones. Era el contrato más fácil para cambiar una obligación por otra.

**I.V.A:** En términos generales el IVA es un impuesto nacional a los consumos que tiene que ser abonado por las personas en cada una de las etapas del proceso económico, en proporción al valor agregado del producto, así como sobre la realización de determinadas obras y locaciones, y la prestación de servicios. En Venezuela comenzó a regir a partir de enero de 1975 por la Ley 20.631. El IVA reemplazó al impuesto nacional a las ventas, también debía reemplazar los impuestos locales de actividades lucrativas.

**HTML:** Hyper Text Markup Language, o simplemente HTML, es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web. Si se hace la traducción de su nombre del inglés al castellano, sería “Lenguaje de Marca de Hipertextos”, ya que es justamente un lenguaje que se basa en las marcas para crear los hipertextos.

**WML:** El *wireless markup language* es un lenguaje cuyo origen es el XML (*extensible markup language*). Este lenguaje se utiliza para construir las



---

páginas que aparecen en las pantallas de los teléfonos móviles y los asistentes personales digitales (PDA) dotados de tecnología WAP. Es una versión reducida del lenguaje HTML que facilita la conexión a Internet de dichos dispositivos y que además permite la visualización de páginas web en dispositivos inalámbricos que incluyan la tecnología WAP. La visualización de la página dependerá del dispositivo que se use y de la forma en que este interprete el código, ya que varían entre sí. WML es un metalenguaje, lo que implica que además de usar etiquetas predefinidas se pueden crear componentes propios y tiene ciertas similitudes con otro lenguaje de etiquetas bastante conocido, el HTML, utilizado para la creación de páginas web convencionales.

**Testing:** Las pruebas de software, en inglés *testing* son los procesos que permiten verificar y revelar la calidad de un producto software. Son utilizadas para identificar posibles fallos de implementación, calidad, o usabilidad de un programa de ordenador o videojuego. Básicamente es una fase en el desarrollo de software consistente en probar las aplicaciones construidas.



## ANEXO 1: INTERFAZ DE USUARIO

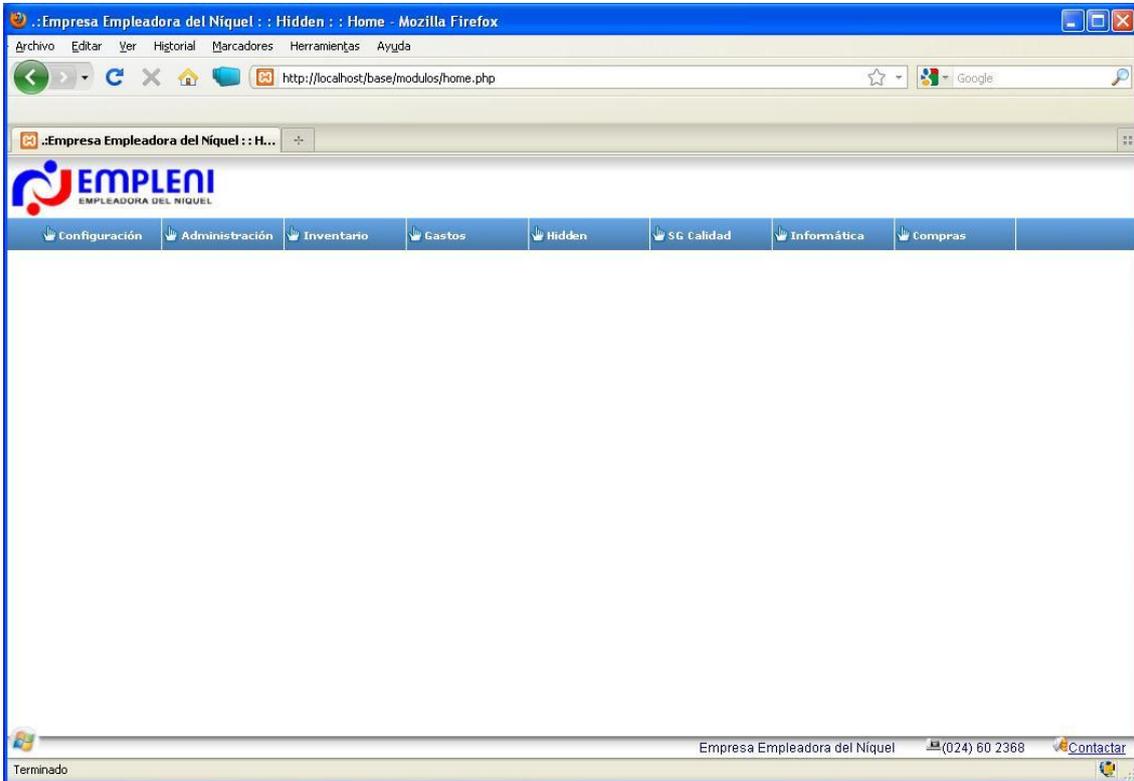


Figura 1.1 Portada del Sistema.

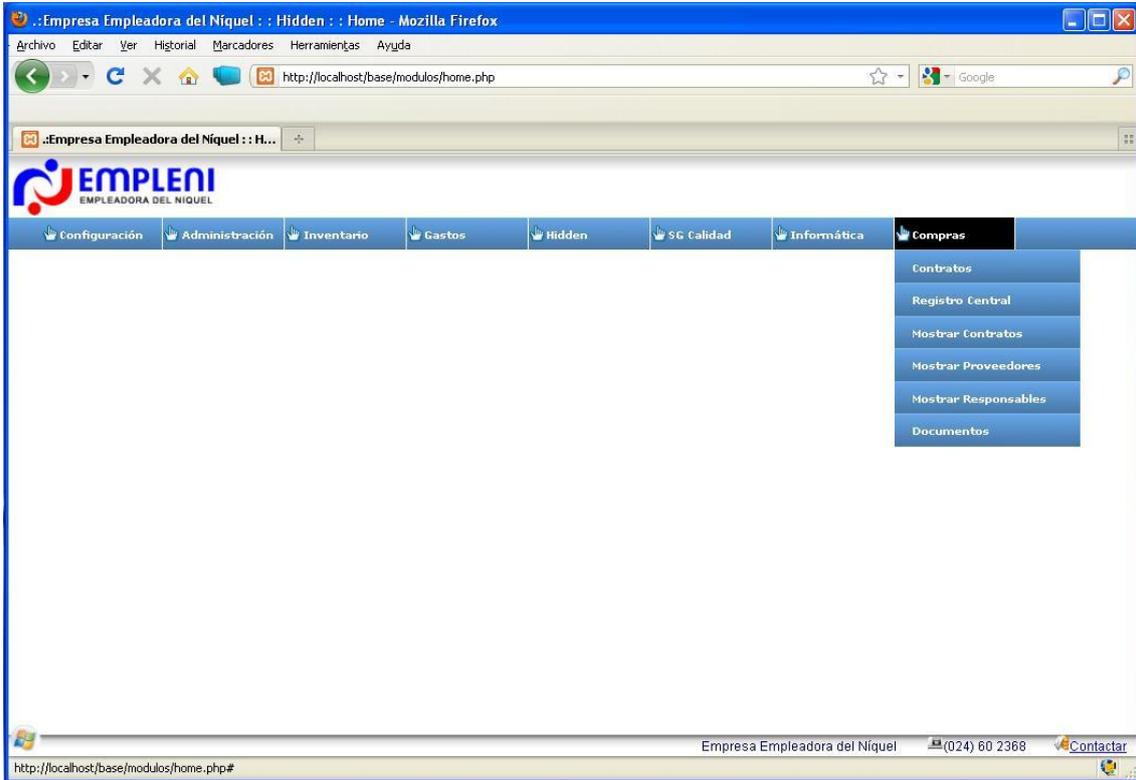


Figura 1.2 Menú: Compras.

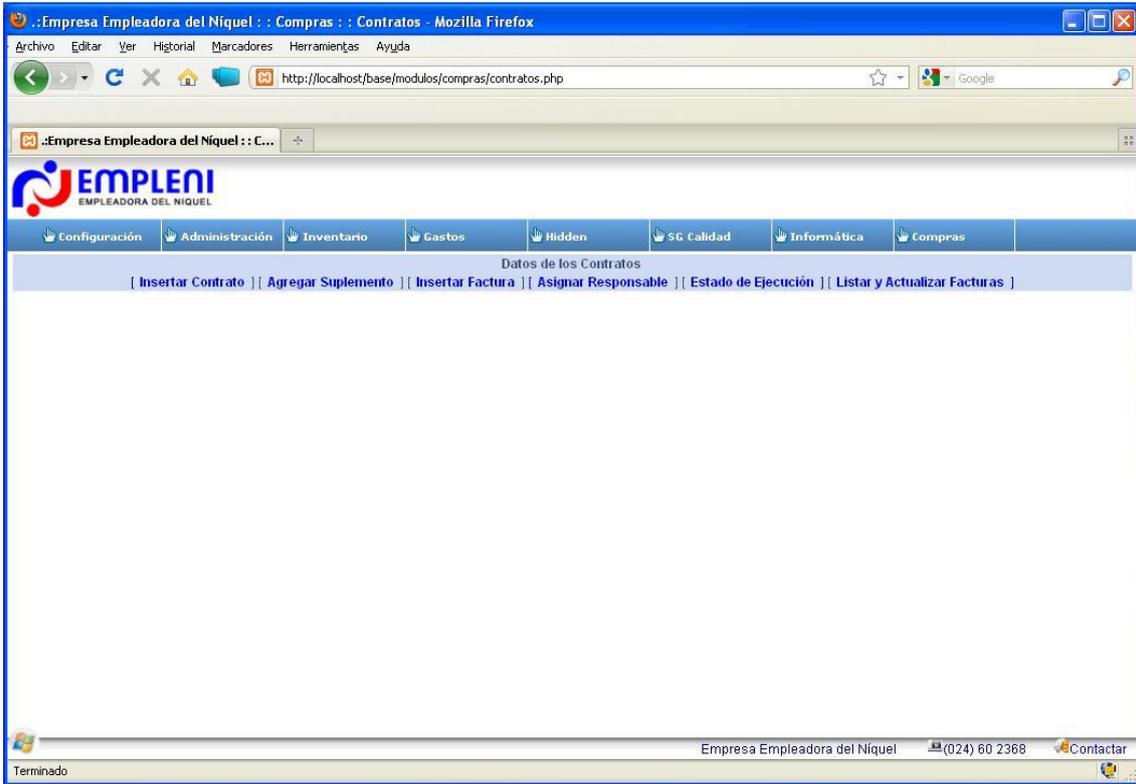


Figura 1.3 Interfaz: Datos de los Contratos.

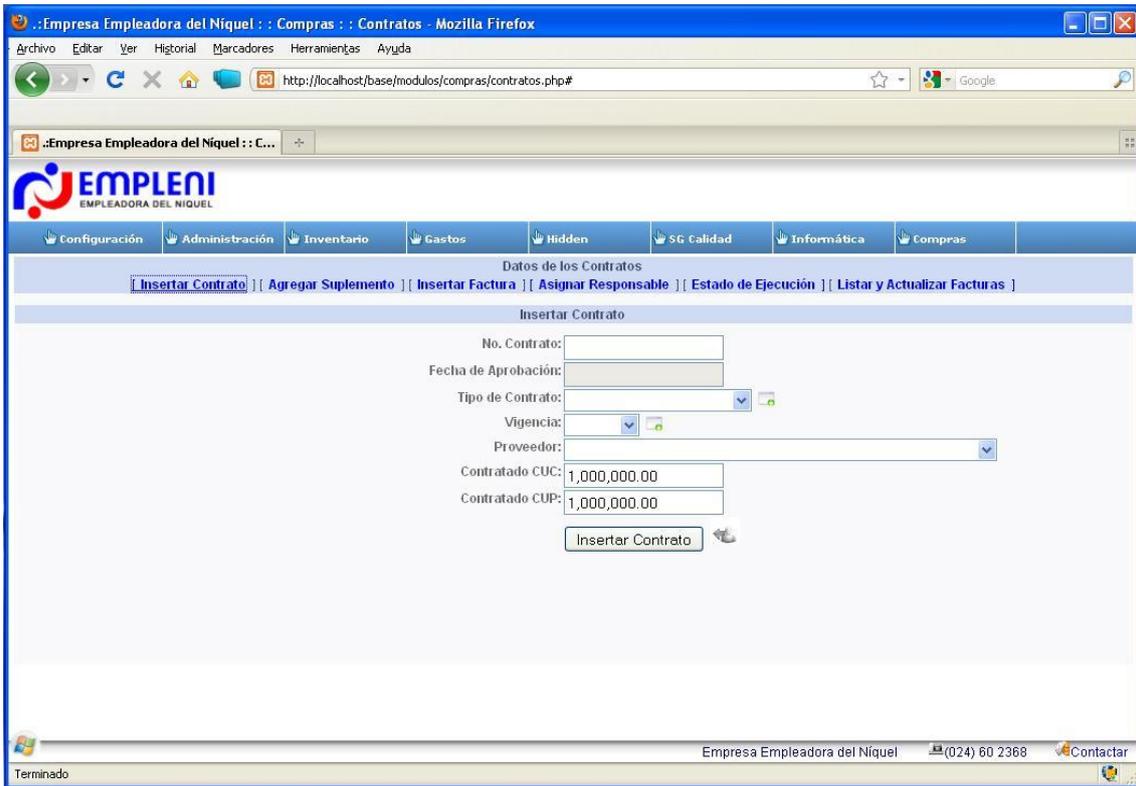


Figura 1.3 Pestaña: Insertar Contrato.

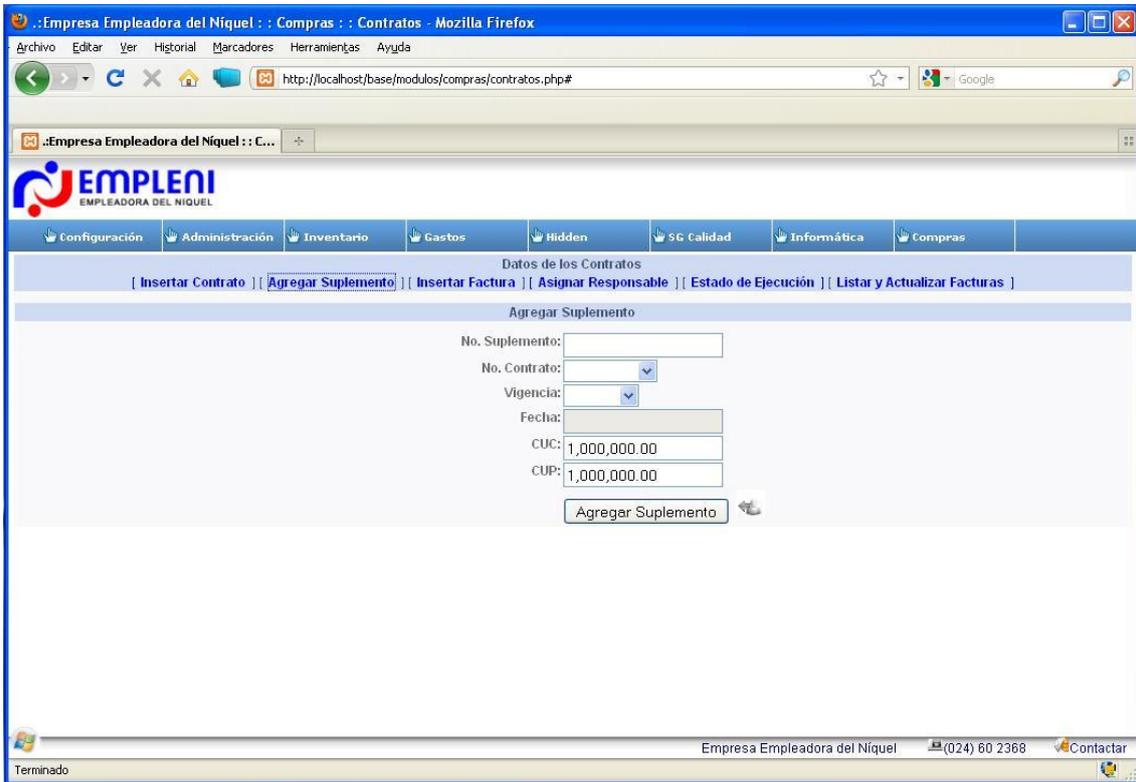


Figura 1.4 Interfaz: Agregar Suplemento.

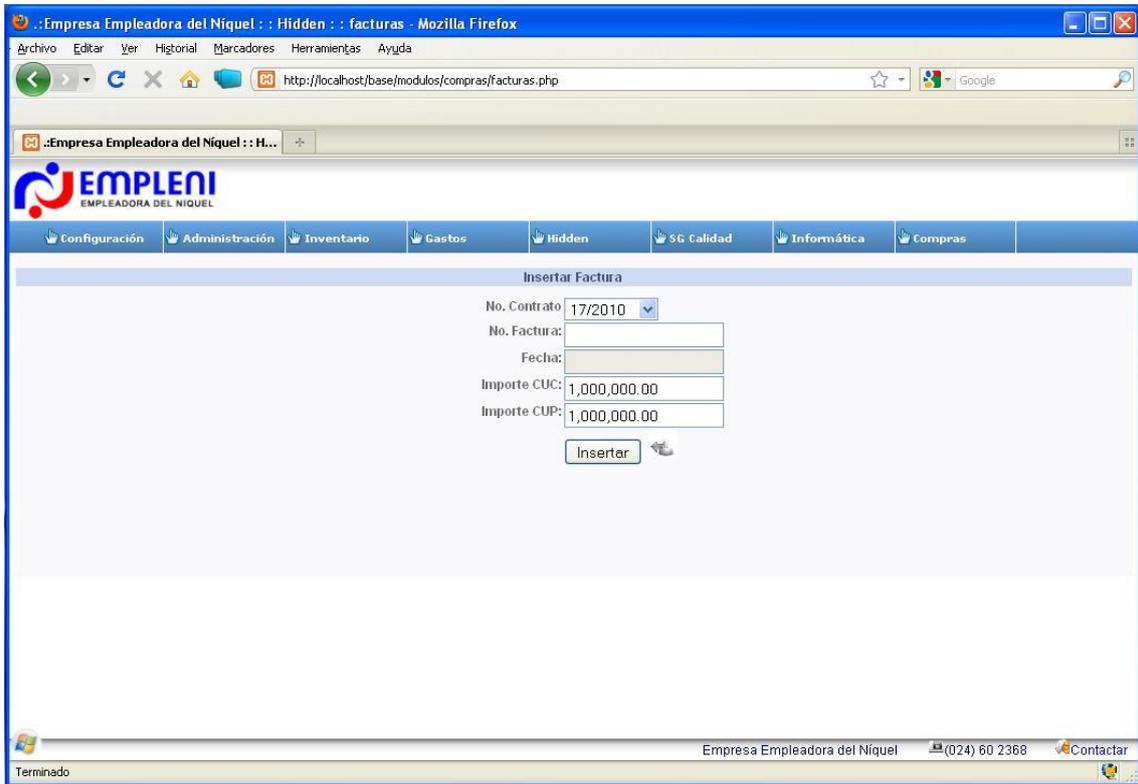


Figura 1.5 Interfaz: Insertar Factura.



## ANEXO 1: INTERFAZ DE USUARIO



Contrato	Aprobado	Tipo	Vigencia	Proveedor	Valor CUC	Suplemento	Ejecutado	Saldo	Valor CUP	Suplemento	Ejecutado	Saldo
17/2010	2010-02-24	Compra-Venta	1 año	Copextel hlg	10000.00	9700.26	599.48	9400.52	5000.00	4930.31	139.38	4860.62
072/2010	2010-02-24	Compra-Venta	1 año	CIMEX S.A, hlg	10000.00	0	3727	6273	0.00	0	0	0
071/2010	2010-02-22	Compra-Venta	1 año	CIMEX S.A, hlg	172966.34	0	8340.4	164625.94	0.00	0	0	0
23/2010	2010-02-19	Compra-Venta	1 año	Copextel Moa	112589.28	0	4855.8	107733.48	39406.25	0	1897.42	37508.83
01/2009	2009-02-27	Compra-Venta	1 año	ECRIN	652.69	373.99	0	652.69	38.00	0.00	0	38

Figura 1.6 Interfaz: Estado de Ejecución.



2010-01-01 2010-06-09 Filtro

[Todos]

No. Contrato	No. Factura	Fecha	Importe CUC	Importe CUP
071/2010	263902	08/06/2010	1800.00	0.00
17/2010	FS-01	07/06/2010	299.74	69.69
071/2010	263474	07/06/2010	52.80	0.00
071/2010	262663	01/06/2010	704.00	0.00
071/2010	262417	31/05/2010	369.60	0.00
071/2010	262053	27/05/2010	1422.00	0.00
072/2010	260031	18/05/2010	106.00	0.00
071/2010	257663	06/05/2010	150.00	0.00
071/2010	257659	06/05/2010	25.30	0.00
071/2010	257161	03/05/2010	59.10	0.00
071/2010	256936	30/04/2010	61.80	0.00
071/2010	256020	27/04/2010	136.20	0.00
071/2010	256019	27/04/2010	600.00	0.00
23/2010	BFB-451000377	23/04/2010	71.55	25.40
23/2010	BFB-451000378	23/04/2010	194.16	258.38
23/2010	BFB-451000361	19/04/2010	2549.79	889.33
072/2010	254521	19/04/2010	14.35	0.00
072/2010	254178	16/04/2010	528.35	0.00
071/2010	253378	13/04/2010	84.00	0.00

Terminado Empresa Empleadora del Niquel (024) 60 2368 Contactar

Figura 1.7 Interfaz: Listar y Actualizar Facturas.

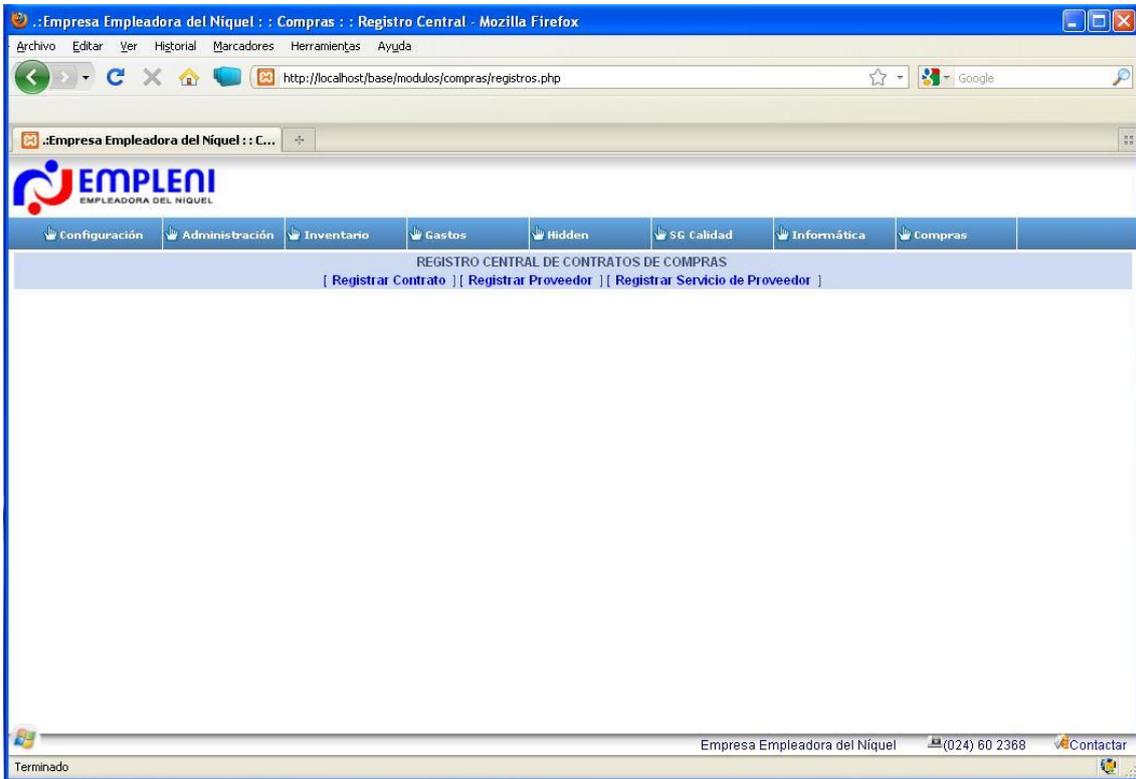


Figura 1.8 Interfaz: Registro Central.

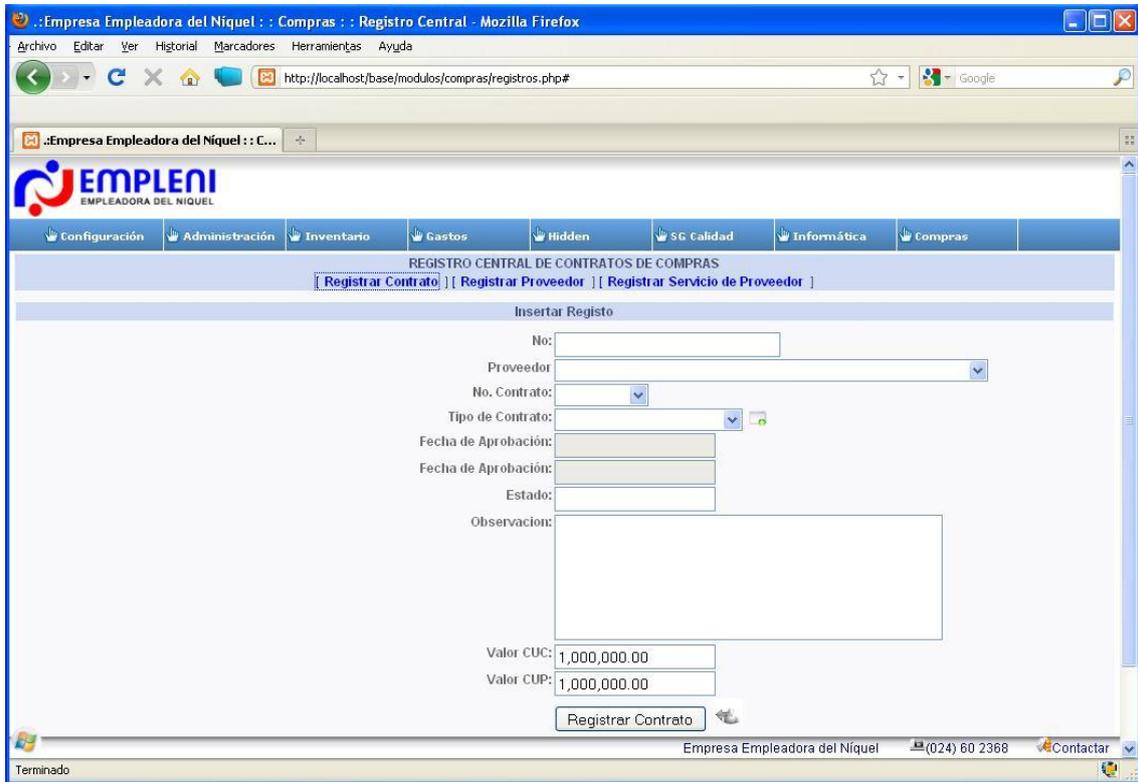


Figura 1.9 Interfaz: Insertar Registro.

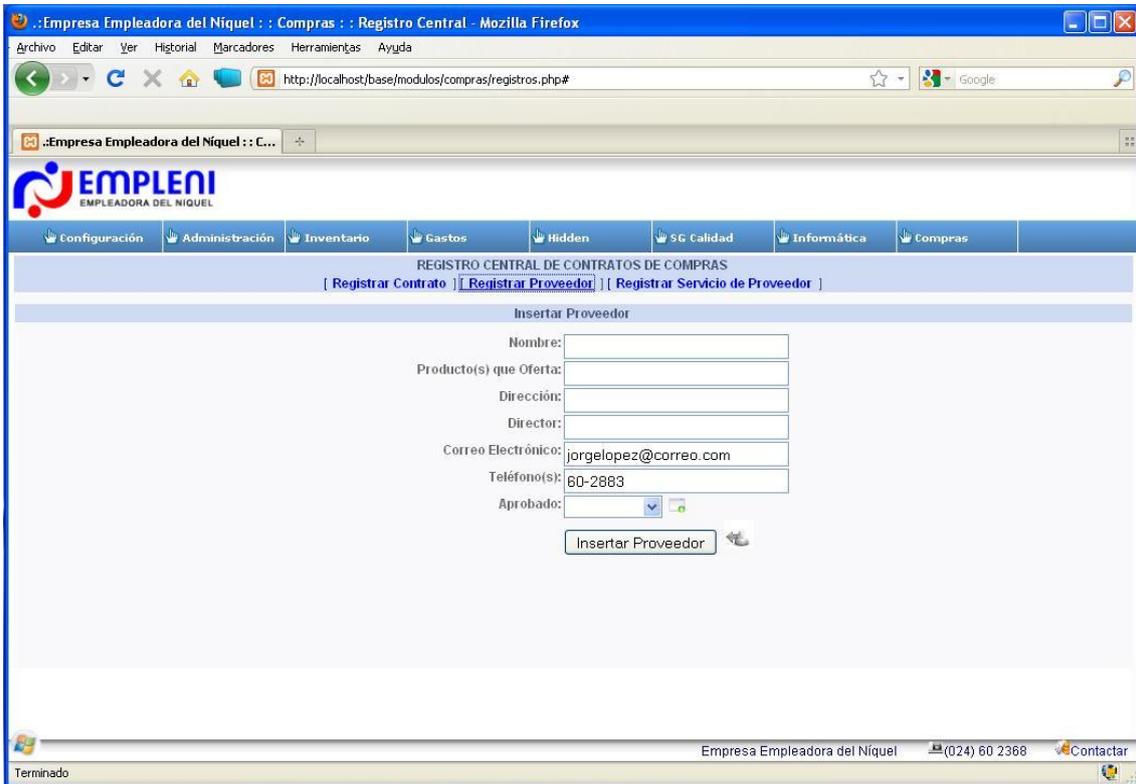


Figura 1.10 Interfaz: Insertar Proveedor.

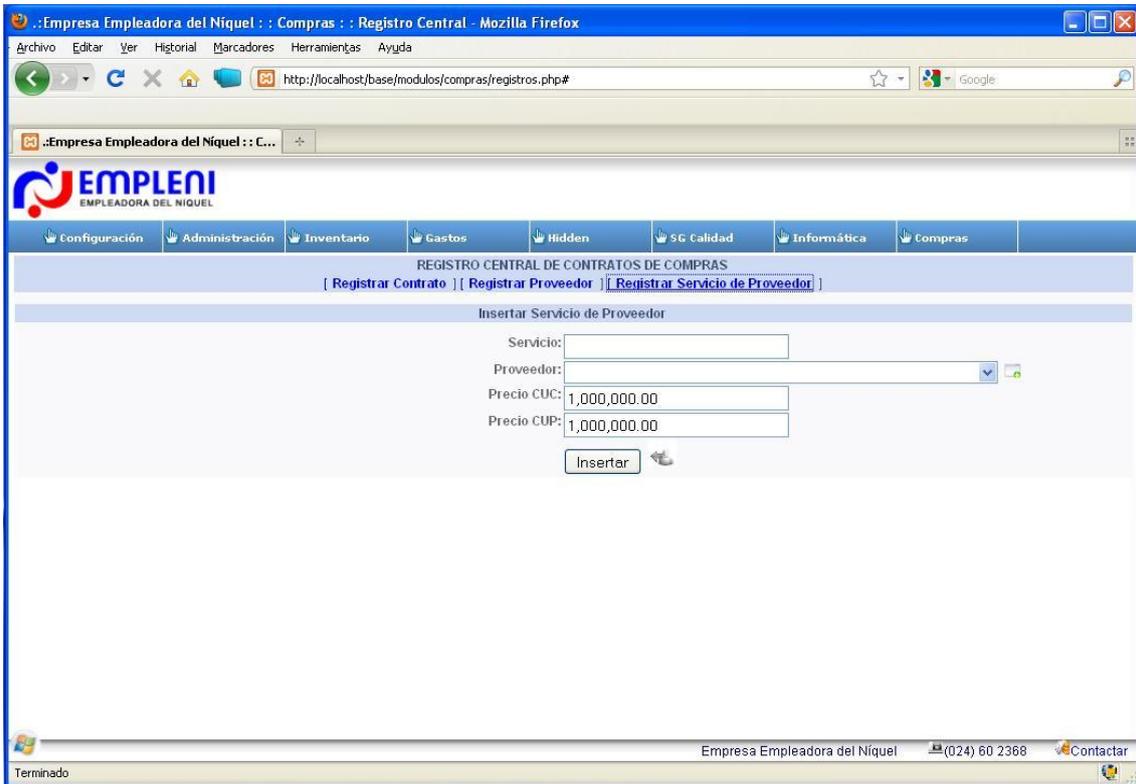


Figura 1.11 Interfaz: Insertar Servicio de Proveedor.



---

**ANEXO 2: HISTORIAS DE USUARIO****HU No.1: Gestión de contratos**

Historia de usuario	
<b>Número:</b> 1	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Gestión de la información de los contratos.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> El usuario debe insertar los datos de los contratos, en caso de que requieran de suplemento, también debe ser insertada esta información, además debe ser almacenada la información de los contratos en el registro central de contratos. Una vez añadidos los contratos, se podrá mostrar y actualizar la información de los mismos.	
<b>Observaciones:</b> Confirmado con el cliente.	

Tabla 2.1 HU No.1: Gestión de contratos

**HU No.2: Gestión de responsables de los contratos**

Historia de usuario	
<b>Número:</b> 2	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Gestión de responsables de los contratos.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Siempre que se inserte un contrato, el usuario debe asignarlo a un técnico en gestión comercial para su control y seguimiento.	
<b>Observaciones:</b> Confirmado con el cliente.	

Tabla 2.2 HU No.2: Gestión de responsables de los contratos

**HU No.3: Gestión de proveedores**

Historia de usuario	
<b>Número:</b> 3	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Gestión de proveedores.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> El usuario debe introducir la información de los proveedores para conocer los productos que oferta y si se aprueban o no para su contratación.	
<b>Observaciones:</b> Confirmado con el cliente.	

Tabla 2.3 HU No.3: Gestión de proveedores

**HU No.4: Gestión de servicios de proveedores**

Historia de usuario	
<b>Número:</b> 4	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Gestión de servicios de proveedores.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Consiste en insertar la información relacionada con los servicios que prestan los diferentes proveedores con el objetivo de su contratación futura.	
<b>Observaciones:</b> Confirmado con el cliente.	

Tabla 2.4 HU No.4: Gestión de servicios de proveedores

**HU No.5: Gestión de facturación**

Historia de usuario	
<b>Número:</b> 5	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Gestión de facturación.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Se insertan los datos de las facturas con el objetivo de tener un control de las mismas.	
<b>Observaciones:</b> Confirmado con el cliente.	

Tabla 2.5 HU No.5: Gestión de facturación

**HU No.6: Subir documentos de los contratos**

Historia de usuario	
<b>Número:</b> 6	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Subir documentos de los contratos.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> El usuario debe subir los documentos en formato digital de los contratos.	
<b>Observaciones:</b> Confirmado con el cliente.	

Tabla 2.6 HU No.6: Subir documentos de los contratos

**HU No.7: Subir documentos legales de proveedores**

Historia de usuario	
<b>Número:</b> 7	<b>Usuario:</b> Especialista en Gestión Comercial
<b>Nombre:</b> Subir documentos legales de proveedores.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Jorge Mario	
<b>Descripción:</b> El usuario debe subir los documentos legales de cada proveedor.	
<b>Observaciones:</b>	

Tabla 2.7 HU No.7: Subir documentos legales de proveedores

**ANEXO 3: TARJETAS CRC****Tarjeta CRC No.1 Gestión de contratos**

<b>Nombre de la clase:</b> Gestión_Contratos	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar datos de contrato.	Insertar_Contrato
Modificar datos de contrato.	Modificar_Contrato
Agregar suplemento a contratos.	Agregar_Suplemento
Listar contratos y mostrar su estado de ejecución.	Listar_Contrato
Actualizar contratos.	Actualizar_Contrato
Archivar información de los contratos en el registro central.	Registrar_Contrato
Mostrar información de contratos.	Mostrar_Registro

Tabla 2.8 Tarjeta CRC No.1

**Tarjeta CRC No.2: Insertar contrato**

<b>Nombre de la clase:</b> Insertar_Contrato	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar los datos de un contrato.	

Tabla 2.9 Tarjeta CRC No.2

**Tarjeta CRC No.3: Modificar contrato**

<b>Nombre de la clase:</b> Modificar_Contrato	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Modificar datos de contrato.	

Tabla 2.9 Tarjeta CRC No.3

**Tarjeta CRC No.4: Agregar suplemento**

<b>Nombre de la clase:</b> Agregar_Suplemento	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Agregar suplemento a contratos.	

Tabla 2.10 Tarjeta CRC No.4

**Tarjeta CRC No.5: Listar contrato**

<b>Nombre de la clase:</b> Listar_Contrato	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Listar contratos y mostrar su estado de ejecución.	

Tabla 2.11 Tarjeta CRC No.5

**Tarjeta CRC No.6: Actualizar contrato**

<b>Nombre de la clase:</b> Actualizar_Contrato	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Actualizar contratos.	

Tabla 2.12 Tarjeta CRC No.6

**Tarjeta CRC No.7: Registrar contrato**

<b>Nombre de la clase:</b> Registrar_Contrato	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Archivar información de los contratos en el registro central.	

Tabla 2.13 Tarjeta CRC No.7

**Tarjeta CRC No.8: Mostrar registro**

<b>Nombre de la clase:</b> Mostrar_Registro	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Mostrar información de contratos.	

Tabla 2.13 Tarjeta CRC No.8

**Tarjeta CRC No. 9: Gestión de responsables de contratos**

<b>Nombre de la clase:</b> Gestión_Responsables	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar responsable de contrato.	Insertar_Responsable
Modificar responsable de contrato.	Modificar_Responsable
Eliminar responsable de contrato.	Eliminar_responsable
Mostrar responsables de contrato.	Mostrar_responsable

Tabla 2.14 Tarjeta CRC No.9

**Tarjeta CRC No.10: Insertar responsables de contrato**

<b>Nombre de la clase:</b> Insertar_Responsable	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Asigna a un especialista de la UEB de compras como responsable de un contrato.	

Tabla 2.15 Tarjeta CRC No.10

**Tarjeta CRC No.11: Modificar responsables de contrato**

<b>Nombre de la clase:</b> Modificar_Responsable	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Modifica la información del responsable de un contrato.	

Tabla 2.16 Tarjeta CRC No.11

**Tarjeta CRC No.12: Eliminar responsables de contrato**

<b>Nombre de la clase:</b> Eliminar_Responsable	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Elimina persona responsable de un contrato.	

Tabla 2.17 Tarjeta CRC No.12

**Tarjeta CRC No.13: Mostrar responsables de contrato**

<b>Nombre de la clase:</b> Eliminar_Responsable	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Muestra los responsables de contratos.	

Tabla 2.18 Tarjeta CRC No.13

**Tarjeta CRC No.14: Gestión de proveedores**

<b>Nombre de la clase:</b> Gestión_Proveedores	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar información de proveedor	Insertar_Proveedor
Modificar información de proveedor	Modificar_Proveedor
Eliminar información de proveedor	Eliminar_Proveedor
Mostrar información de proveedor	Mostrar_Proveedor

Tabla 2.19 Tarjeta CRC No.14

**Tarjeta CRC No.15: Insertar datos de proveedor**

<b>Nombre de la clase:</b> Insertar_Proveedor	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Inserta un proveedor determinado	

Tabla 2.20 Tarjeta CRC No.15

**Tarjeta CRC No.16: Modificar datos de proveedor**

<b>Nombre de la clase:</b> Modificar_Proveedor	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Modifica la información de un proveedor	

Tabla 2.21 Tarjeta CRC No.16

**Tarjeta CRC No.17: Eliminar datos de proveedor**

<b>Nombre de la clase:</b> Eliminar_Proveedor	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Elimina un proveedor determinado	

Tabla 2.22 Tarjeta CRC No.17

**Tarjeta CRC No.18: Eliminar datos de proveedor**

<b>Nombre de la clase:</b> Eliminar_Proveedor	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Elimina un proveedor determinado	

Tabla 2.23 Tarjeta CRC No.18

**Tarjeta CRC No.19: Mostrar proveedor**

<b>Nombre de la clase:</b> Mostrar_Proveedor	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Muestra la información de los proveedores.	

Tabla 2.24 Tarjeta CRC No.19

**Tarjeta CRC No.20: Gestión de servicios de proveedores**

<b>Nombre de la clase:</b> Gestión _ Servicios	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar servicio de proveedor.	Insertar_Servicio
Modificar servicio de proveedor.	Modificar_Servicio
Eliminar servicio de proveedor.	Eliminar_Servicio
Mostrar servicio de proveedor.	Mostrar_Servicio

Tabla 2.25 Tarjeta CRC No.20

**Tarjeta CRC No.21: Insertar servicio de proveedor**

<b>Nombre de la clase:</b> Insertar_Servicio	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar servicio de un proveedor determinado.	

Tabla 2.26 Tarjeta CRC No.21

**Tarjeta CRC No.22: Modificar servicio de proveedor**

<b>Nombre de la clase:</b> Modificar_Servicio	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Modificar información de un servicio determinado.	

Tabla 2.27 Tarjeta CRC No.22

**Tarjeta CRC No.23: Modificar servicio de proveedor**

<b>Nombre de la clase:</b> Eliminar_Servicio	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Eliminar un servicio determinado.	

Tabla 2.28 Tarjeta CRC No.23

**Tarjeta CRC No.24: Mostrar servicio de proveedor**

<b>Nombre de la clase:</b> Eliminar_Servicio	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Muestra los servicios de proveedores.	

Tabla 2.29 Tarjeta CRC No.24

**Tarjeta CRC No.25 Gestión de facturación**

<b>Nombre de la clase:</b> Gestión_Facturación	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar datos de factura.	Insertar_Factura
Modificar datos de factura.	Modificar_Factura
Mostrar datos de factura.	Mostrar_Factura

Tabla 2.30 Tarjeta CRC No.25

**Tarjeta CRC No.26: Insertar datos de factura**

<b>Nombre de la clase:</b> Insertar_Factura	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Insertar el importe cuc ó cup de las facturas.	

Tabla 2.31 Tarjeta CRC No.26

**Tarjeta CRC No. 27: Modificar datos de factura**

<b>Nombre de la clase:</b> Modificar_Factura	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Modificar datos de las facturas.	

Tabla 2.32 Tarjeta CRC No.27

**Tarjeta CRC No. 28: Mostrar datos de factura**

<b>Nombre de la clase:</b> Mostrar_Factura	
<b>Tipo de la clase:</b> Utilitaria	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Mostrar datos de las facturas.	

Tabla 2.33 Tarjeta CRC No.28

**Tarjeta CRC No.29 Subir documentos de los contratos**

<b>Nombre de la clase:</b> Subir_Documento	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Subir los documentos de los contratos.	

Tabla 2.34 Tarjeta CRC No.29



**Tarjeta CRC No.30: Subir documentos legales de proveedores**

<b>Nombre de la clase:</b> Subir_Legales	
<b>Tipo de la clase:</b> Lógica del negocio	
<b>Responsabilidades:</b>	<b>Colaboradores:</b>
Subir los documentos legales de proveedores.	

Tabla 2.35 Tarjeta CRC No.30



ANEXO 4: MODELO DE DATOS

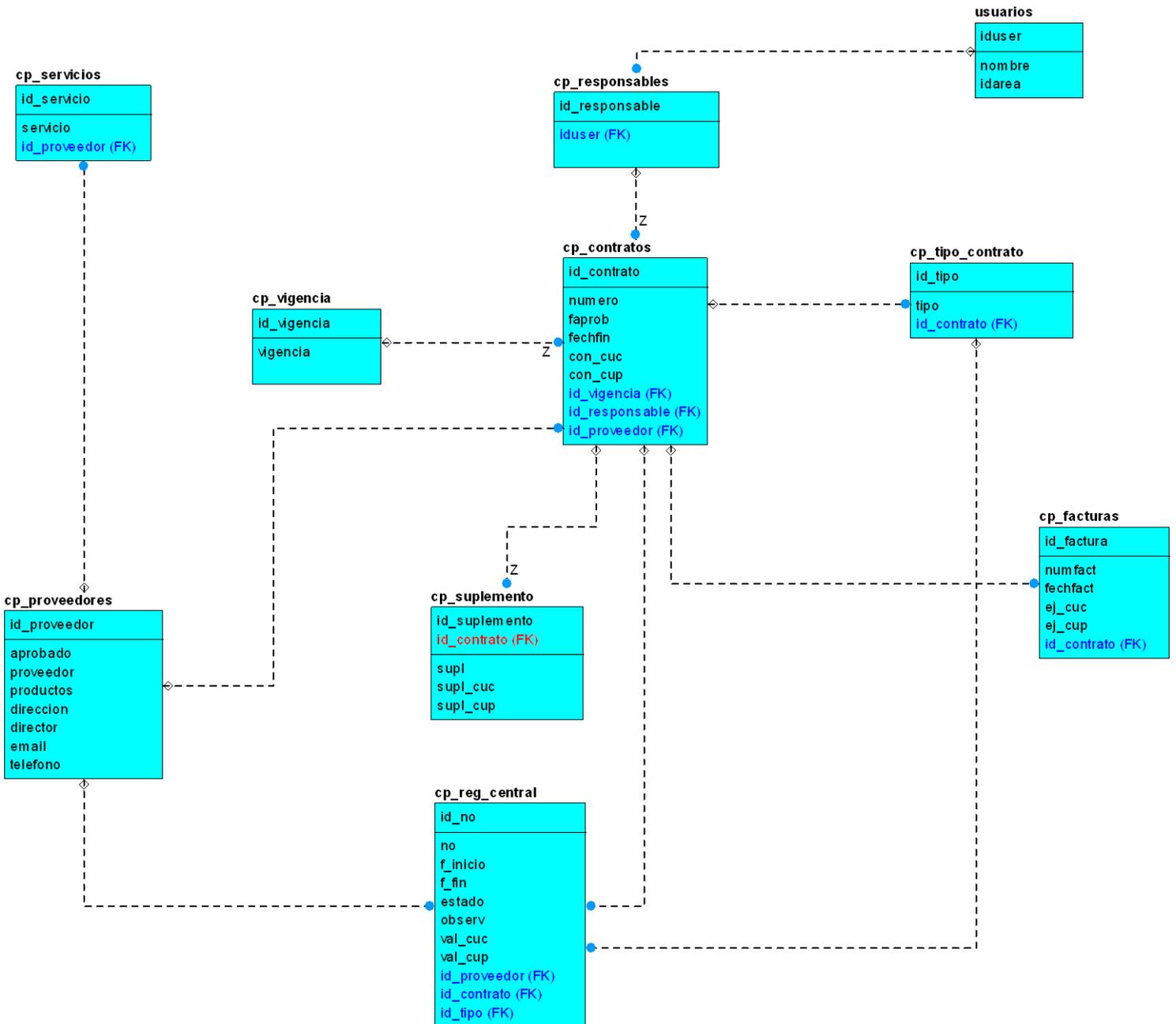


Fig. 3.1 Modelo de datos

**ANEXO 5: TARJETAS DE TAREAS****Tarjeta de tarea No.1: Insertar los datos de los contratos**

Tarea ingeniería	
<b>Número tarea:</b> 1	<b>Número historia:</b> 1
<b>Nombre tarea:</b> Insertar los datos de los contratos.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 1/02/2010	<b>Fecha fin:</b> 3/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea facilita introducir los datos de los contratos.	

Tabla 3.3 Tarjeta de tarea No. 1

**Tarjeta de tarea No. 2: Modificar los datos de los contratos**

Tarea de ingeniería	
<b>Número tarea:</b> 2	<b>Número historia:</b> 1
<b>Nombre tarea:</b> Modificar los datos de los contratos.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 4/02/2010	<b>Fecha fin:</b> 6/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea permite que el usuario pueda modificar los datos de los contratos una vez que son insertados.	

Tabla 3.4 Tarjeta de tarea No. 2

**Tarjeta de tarea No. 3: Agregar suplemento a contrato**

Tarea de ingeniería	
<b>Número tarea:</b> 3	<b>Número historia:</b> 1
<b>Nombre tarea:</b> Agregar suplemento a contrato.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 3
<b>Fecha inicio:</b> 7/02/2010	<b>Fecha fin:</b> 9/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea facilita que cuando un contrato esté por vencerse, pueda renovarse su tiempo de vida a través de la agregación de un suplemento al mismo.	

Tabla 3.5 Tarjeta de tarea No. 3

**Tarjeta de tarea No. 4: Mostrar estado de los contratos**

Tarea de ingeniería	
<b>Número tarea:</b> 4	<b>Número historia:</b> 1
<b>Nombre tarea:</b> Mostrar estado de los contratos.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 10/01/2010	<b>Fecha fin:</b> 13/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Muestra el estado de ejecución de los contratos, la vigencia, importe, caducidad, etc.	

Tabla 3.6 Tarjeta de tarea No. 5

**Tarjeta de tarea No. 5: Insertar responsable de contrato**

Tarea de ingeniería	
<b>Número tarea:</b> 5	<b>Número historia:</b> 2
<b>Nombre tarea:</b> Insertar responsable de contrato.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 14/02/2010	<b>Fecha fin:</b> 16/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea posibilita asignarle a un técnico en gestión comercial un contrato para su seguimiento y control.	

Tabla 3.7 Tarjeta de tarea No. 5

**Tarjeta de tarea No. 6: Modificar responsable de contrato**

Tarea de ingeniería	
<b>Número tarea:</b> 6	<b>Número historia:</b> 2
<b>Nombre tarea:</b> Modificar responsable de contrato.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 17/02/2010	<b>Fecha fin:</b> 18/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Modificar un responsable de contrato, viabiliza designar a otra persona de la UEB de compras como responsable de un contrato en caso de que su responsable anterior no pueda llevar su seguimiento por cualquier razón.	

Tabla 3.8 Tarjeta de tarea No. 6

**Tarjeta de tarea No. 7: Eliminar responsable de contrato**

Tarea de ingeniería	
<b>Número tarea:</b> 7	<b>Número historia:</b> 2
<b>Nombre tarea:</b> Eliminar responsable de contrato.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 17/02/2010	<b>Fecha fin:</b> 18/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea facilita eliminar a un responsable de contrato.	

Tabla 3.9 Tarjeta de tarea No. 7

**Tarjeta de tarea No. 8: Mostrar responsables de contrato**

Tarea de ingeniería	
<b>Número tarea:</b> 8	<b>Número historia:</b> 2
<b>Nombre tarea:</b> Eliminar responsable de contrato.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 17/02/2010	<b>Fecha fin:</b> 18/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea facilita mostrar los responsables de los contratos.	

Tabla 3.10 Tarjeta de tarea No. 8

**Tarjeta de tarea No. 9: Insertar información de proveedor**

Tarea de ingeniería	
<b>Número tarea:</b> 9	<b>Número historia:</b> 3
<b>Nombre tarea:</b> Insertar información proveedor.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 22/02/2010	<b>Fecha fin:</b> 24/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea facilita insertar la información de los proveedores.	

Tabla 3.11 Tarjeta de tarea No. 9

**Tarjeta de tarea No. 10: Modificar información de proveedor**

Tarea de ingeniería	
<b>Número tarea:</b> 10	<b>Número historia:</b> 3
<b>Nombre tarea:</b> Modificar información de proveedor.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 25/02/2010	<b>Fecha fin:</b> 26/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> La información de los proveedores puede ser modificada en caso de que presente algún error.	

Tabla 3.12 Tarjeta de tarea No. 10

**Tarjeta de tarea No. 11: Eliminar información de proveedor**

Tarea de ingeniería	
<b>Número tarea:</b> 11	<b>Número historia:</b> 3
<b>Nombre tarea:</b> Eliminar información de proveedor.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 25/02/2010	<b>Fecha fin:</b> 26/02/2010
<b>Programador responsable:</b> Jorge Mario basulto Aguilera.	
<b>Descripción:</b> Esta tarea posibilita eliminar la información de un proveedor.	

Tabla 3.13 Tarjeta de tarea No. 11

**Tarjeta de tarea No. 12: Mostrar información de proveedores**

Tarea de ingeniería	
<b>Número tarea:</b> 12	<b>Número historia:</b> 3
<b>Nombre tarea:</b> Eliminar información de proveedores.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 25/02/2010	<b>Fecha fin:</b> 26/02/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea posibilita mostrar la información de los proveedores.	

Tabla 3.14 Tarjeta de tarea No. 12

**Tarjeta de tarea No. 13: Insertar servicio de proveedor**

Tarea de ingeniería	
<b>Número tarea:</b> 13	<b>Número historia:</b> 4
<b>Nombre tarea:</b> Insertar servicio.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 1/03/2010	<b>Fecha fin:</b> 3/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea permite insertar los servicios de proveedores.	

Tabla 3.15 Tarjeta de tarea No. 13

**Tarjeta de tarea No. 14: Modificar servicio de proveedor**

Tarea de ingeniería	
<b>Número tarea:</b> 14	<b>Número historia:</b> 4
<b>Nombre tarea:</b> Modificar servicio de proveedor.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 4/03/2010	<b>Fecha fin:</b> 5/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea permite modificar la información de un servicio.	

Tabla 3.16 Tarjeta de tarea No. 14

**Tarjeta de tarea No. 15: Eliminar servicio de proveedor**

Tarea de ingeniería	
<b>Número tarea:</b> 15	<b>Número historia:</b> 4
<b>Nombre tarea:</b> Eliminar servicio de proveedor.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 4/03/2010	<b>Fecha fin:</b> 5/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Permite eliminar un servicio de proveedor.	

Tabla 3.17 Tarjeta de tarea No. 15

**Tarjeta de tarea No. 16: Mostrar servicios de proveedores**

Tarea de ingeniería	
<b>Número tarea:</b> 16	<b>Número historia:</b> 4
<b>Nombre tarea:</b> Eliminar servicio de proveedor.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 4/03/2010	<b>Fecha fin:</b> 5/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Permite conocer los servicios que brindan los proveedores con el propósito de su posible contratación.	

Tabla 4.8 Tarjeta de tarea No. 16

**Tarjeta de tarea No. 17: Insertar datos de factura**

Tarea de ingeniería	
<b>Número tarea:</b> 17	<b>Número historia:</b> 5
<b>Nombre tarea:</b> Insertar datos de facturas.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 8/03/2010	<b>Fecha fin:</b> 10/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Permite insertar los datos de las facturas.	

Tabla 3.19 Tarjeta de tarea No. 17

**Tarjeta de tarea No. 18: Modificar datos de factura**

Tarea de ingeniería	
<b>Número tarea:</b> 18	<b>Número historia:</b> 5
<b>Nombre tarea:</b> Modificar datos de facturas.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 11/03/2010	<b>Fecha fin:</b> 13/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Permite modificar los datos de las facturas.	

Tabla 3.20 Tarjeta de tarea No. 18

**Tarjeta de tarea No. 19: Mostrar los datos de las facturas**

Tarea de ingeniería	
<b>Número tarea:</b> 19	<b>Número historia:</b> 5
<b>Nombre tarea:</b> Mostrar los datos de las facturas.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 14/03/2010	<b>Fecha fin:</b> 16/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto aguilera	
<b>Descripción:</b> Permite eliminar los datos de las facturas.	

Tabla 3.21 Tarjeta de tarea No. 19

**Tarjeta de tarea No. 20: Subir contrato**

Tarea de ingeniería	
<b>Número tarea:</b> 20	<b>Número historia:</b> 6
<b>Nombre tarea:</b> Subir contrato.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 22/03/2010	<b>Fecha fin:</b> 24/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea permite subir a la aplicación los documentos de los contratos.	

Tabla 3.22 Tarjeta de tarea No. 20

**Tarjeta de tarea No. 21: Subir documento legal**

Tarea de ingeniería	
<b>Número tarea:</b> 21	<b>Número historia:</b> 7
<b>Nombre tarea:</b> Subir documento legal.	
<b>Tipo de tarea :</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 25/03/2010	<b>Fecha fin:</b> 27/03/2010
<b>Programador responsable:</b> Jorge Mario Basulto Aguilera.	
<b>Descripción:</b> Esta tarea permite subir los documentos legales de cada proveedor a la aplicación.	

Tabla 3.23 Tarjeta de tarea No. 2



## ANEXO 6: PRUEBAS DE ACEPTACIÓN

### Pruebas de aceptación para la HU: Gestión de contratos

<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de contratos.
<b>Nombre:</b> Prueba para comprobar la entrada de datos de contratos.
<b>Descripción:</b> Validación de entrada de datos de contrato.
<b>Condiciones de ejecución:</b> El usuario debe introducir los datos de los contratos.
<b>Entrada/Pasos ejecución:</b> El usuario intenta introducir los datos de los contratos.
<b>Resultado:</b> Se emite un mensaje de error en caso de que: <ul style="list-style-type: none"><li>• No se introduzcan los datos solicitados.</li><li>• Se intente insertar un contrato que ya existe.</li><li>• Se intente introducir un formato no válido.</li></ul> Se emite un mensaje en caso de información de éxito al insertar el contrato.
<b>Evaluación de la prueba:</b> Aceptada.

Tabla 3.25 PA: Prueba para comprobar la de entrada de datos de contratos.

<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de contratos.
<b>Nombre:</b> Prueba para comprobar la entrada de datos del suplemento.
<b>Descripción:</b> Validación de entrada de datos del suplemento.
<b>Condiciones de ejecución:</b> El usuario debe introducir los datos del suplemento.
<b>Entrada/Pasos ejecución:</b> El usuario intenta introducir los datos del suplemento.
<b>Resultado:</b> Se emite un mensaje de error en caso de que: <ul style="list-style-type: none"><li>• No se introduzcan los datos solicitados.</li><li>• Se intente agregar un suplemento ya existente.</li><li>• Se intente introducir un formato no válido.</li></ul> Se emite un mensaje en caso de información de éxito al agregar el suplemento al contrato.
<b>Evaluación de la prueba:</b> Aceptada.

Tabla 3.26 PA: Prueba para comprobar la de entrada de datos de suplemento.



<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de Contratos.
<b>Nombre:</b> Prueba para comprobar la visualización de los contratos.
<b>Descripción:</b> Selección correcta de las fechas que pasan por el filtro.
<b>Condiciones de ejecución:</b> El usuario debe seleccionar la opción listar contratos.
<b>Entrada/Pasos ejecución:</b> <ul style="list-style-type: none"><li>❖ El usuario selecciona la opción listar contratos.</li><li>❖ Luego debe seleccionar el período que desea revisar.</li></ul>
<b>Resultado:</b> <ul style="list-style-type: none"><li>• Se emite un mensaje de error en caso de que la primera fecha sea menor que la segunda.</li><li>• O son visualizados correctamente los contratos.</li></ul>
<b>Evaluación de la prueba:</b> Aceptada o denegada.

Tabla 3.27 PA: Prueba para comprobar la visualización de los contratos.

### Pruebas de aceptación para la HU: Gestión de responsables de contratos

<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de responsables de contratos.
<b>Nombre:</b> Prueba para comprobar la entrada de información de responsables de contrato.
<b>Descripción:</b> Introducción correcta de información de responsables de contrato.
<b>Condiciones de ejecución:</b> El usuario debe insertar un responsable.
<b>Entrada / Pasos ejecución:</b> El usuario intenta insertar un responsable.
<b>Resultado:</b> <ul style="list-style-type: none"><li>• Se emite un mensaje de error en caso de entrada incorrecta de datos.</li><li>• Se emite un mensaje de información en caso de éxito al asignar un responsable a un contrato.</li></ul>
<b>Evaluación de la prueba:</b> Aceptada.

Tabla 3.28 PA: Prueba para comprobar la de entrada de información de responsables de contratos.

**Pruebas de aceptación para la HU: Gestión de proveedores**

<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de proveedores
<b>Nombre:</b> Prueba para comprobar la entrada de información de proveedores.
<b>Descripción:</b> Introducción correcta de la información de proveedores.
<b>Condiciones de ejecución:</b> El usuario debe introducir la información del proveedor.
<b>Entrada / Pasos ejecución:</b> El usuario intente introducir la información del proveedor.
<b>Resultado:</b> <ul style="list-style-type: none"><li>• Se emite un mensaje de error en caso de entrada incorrecta de datos.</li><li>• Se emite un mensaje de información en caso de éxito al insertar un proveedor.</li></ul>
<b>Evaluación de la prueba:</b> Aceptada.

Tabla 3.29 PA: Prueba para comprobar la de entrada de información de proveedores.

**Pruebas de aceptación para la HU: Gestión de servicios de proveedores**

<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de servicios de proveedores
<b>Nombre:</b> Prueba para comprobar la inserción de la información de los servicios de proveedores.
<b>Descripción:</b> Validación de la entrada correcta de información.
<b>Condiciones de ejecución:</b> El usuario debe introducir los datos del servicio.
<b>Entrada/Pasos ejecución:</b> El usuario intenta introducir un servicio.
<b>Resultado:</b> <ul style="list-style-type: none"><li>• Se emite un mensaje de error en caso de entrada incorrecta de datos.</li><li>• O uno de información de éxito al insertar el servicio de proveedor.</li></ul>
<b>Evaluación de la prueba:</b> Aceptada.

Tabla 3.30 PA: Prueba para comprobar la inserción de la información de los servicios de proveedores.

**Prueba de aceptación para la HU: Gestión de facturación**

<b>Prueba de aceptación</b>
<b>HU:</b> Gestión de facturación.
<b>Nombre:</b> Prueba para comprobar la inserción del monto de las facturas.
<b>Descripción:</b> Validación de entrada de datos de facturas.
<b>Condiciones de ejecución:</b> Deben existir contratos insertados.
<b>Entrada/Pasos ejecución:</b> El usuario intenta introducir los datos de facturas.
<b>Resultado:</b> Se emite un mensaje de error en caso de que: <ul style="list-style-type: none"><li>• No se introduzcan los datos requeridos.</li><li>• Entrada incorrecta de datos.</li><li>• Se intente insertar una factura que ya existe.</li><li>• O uno de información de éxito al insertar el monto de la factura.</li></ul>
<b>Evaluación de la prueba:</b> Aceptada

Tabla 3.31 PA: Prueba para comprobar la inserción del monto de las facturas.

**Prueba de aceptación para la HU: Subir documentos de los contratos**

<b>Prueba de aceptación</b>
<b>HU:</b> Subir documentos de los contratos.
<b>Nombre:</b> Prueba de validación al subir un documento.
<b>Descripción:</b> Extensión correcta y existencia de documento.
<b>Condiciones de ejecución:</b>
<b>Entrada/Pasos ejecución:</b> El usuario intenta subir un documento.
<b>Resultado:</b> Se emite un mensaje de error en caso de que: <ul style="list-style-type: none"><li>• La extensión del documento no sea válida.</li><li>• El documento ya existe.</li></ul>
<b>Evaluación de la prueba:</b> Aceptada

Tabla 3.31 PA: Prueba para comprobar la inserción del monto de las facturas



ANEXO 7: AVAL



Ave. Puerto s/n reparto Rolo Monterrey, Moa. Holguín CUBA Teléfono: (024) 604366 Pizarra: (024) 602368 Fax: (024) 602280 email: [empleni@empleni.moa.minbas.cu](mailto:empleni@empleni.moa.minbas.cu)

AVAL

Moa, 17 de junio del 2009

"Año 52 del Triunfo de la Revolución"

Por este medio hago saber que el Trabajo de Diploma titulado: "Sistema de Gestión Integral de la Empresa Empleadora del Níquel. Módulo Gestión de Contratos de Compras", elaborado por el estudiante Jorge Mario Basulto Aguilera, del Instituto Superior Minero Metalúrgico de Moa Dr. Antonio Núñez Jiménez, cumple con el objetivo por el cual fue confeccionado: Diseñar un sistema automatizado para la gestión eficiente de la información de los contratos en la UEB Compras en la Empresa Empleadora del Níquel "EMPL ENI", que facilita el control de la ejecución de los mismos con los proveedores que tiene nuestra empresa en esa área, ahorrando a la entidad aproximadamente 5,000.00 CUP y 3,000.00 CUC por concepto de compra de este tipo de software.

María Isabel González Bañobre  
Directora UEB de Compras



Ing. Yodexy Mosqueda Naranjo  
Esp. Principal Grupo de Redes

