



Instituto Superior Minero Metalúrgico de Moa  
"Antonio Núñez Jiménez"  
Departamento de Informática

# *Trabajo de diploma* *para optar por el título de* *Ingeniería en Informática*

*Sistema para la Gestión de los contratos y el  
control de los medios informáticos en la  
empresa Agropecuaria Sagua de Tánamo.  
(SIGCORA)*

**Autor(es):** *José Angel Mendoza Matilla*

**Tutor(es):** *MSc. Edgar Núñez Torres  
Ing. Yadira Arguelles Blanco  
Ing. Ricardo Salazar Pupo*

*Moa, Cuba  
Julio, 2015*

***Declaración de autoría:***

Declaro que soy el único autor del trabajo: Sistema para la Gestión de los contratos y el control de los medios informáticos en la empresa Agropecuaria Sagua de Tánamo y certifico su propiedad a favor del Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, el cual podrá hacer uso del mismo con la finalidad que estime conveniente.

Para que así conste firmo la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año 2015.

\_\_\_\_\_  
José Ángel Mendoza Matilla

Firma del Autor

\_\_\_\_\_  
MSc. Edgar Núñez Torres

Firma de Tutor

\_\_\_\_\_  
Ing. Yadira Arguelles Blanco

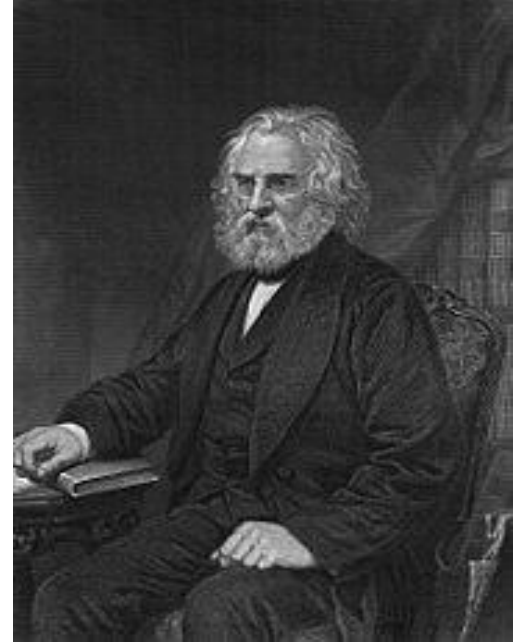
Firma de Tutor

\_\_\_\_\_  
Ing. Ricardo Salazar Pupo

Firma de Tutor

*La clave del éxito depende sólo de lo que  
podamos hacer de la mejor manera posible.*

*Henry Wadsworth Longfellow*



# *Dedicatoria*

*Dedico el presente trabajo en especial a mi esposa, padres, abuelos y familiares los cuales influyeron de forma positiva en la realidad que estoy viviendo. A mis compañeros de aula y en especial los de cuarto donde viví situaciones inolvidables.*

*En lo personal doy gracias a mis tutores que fueron pilar fundamental en la culminación de este proyecto y en mi formación. Y a la revolución por forjar los grandes profesores que me atendieron con paciencia estos cinco largos años.*

## Agradecimientos

*Primero quisiera agradecer a quienes me dieron la vida, mis padres y Dios que sin ellos nada de esto hubiera sido posible.*

*A toda mi familia por darme su ayuda en todo momento. A mi tutor Ricardo que siempre me tendió la mano cuando corría con un problema. A todos mis amigos que hoy están felices por verme en esta situación única, en especial a Ronal que me apoyo en todo momento.*

*A mi tutora Yadira y Edgar del ISMM los cuales me aconsejaron y me ayudaron en la confección de la tesis.*

*A mis amistades y compañeros de curato Abel, Iván y Edilberto que han aportado su granito de arena en mi desarrollo como profesional.*

*A mis compañeros de aula Info. 2010 que compartieron conmigo 5 años de dura lucha y buenos momentos, a mis profesores que de forma directa e indirecta influyeron en mi formación como mejor persona y profesional.*

*A todos mis más grandes y sinceros agradecimientos.*

## *Resumen*

En el municipio Sagua de Tánamo la producción de software brinda soluciones a muchos problemas relacionados con las industrias. Este trabajo presenta una aplicación Web capaz de optimizar la Gestión de los Contratos y el Control de los medios Informáticos en la Empresa Agropecuaria Sagua de Tánamo. Estos procesos en la actualidad presentan dificultades debido a que se desarrollan de forma descentralizada y manual y muchas veces no hay posibilidad de acceso a registros generados en tiempo pasados y que en algún momento del presente se necesitan consultar, proporcionando demoras por parte de los especialistas en la entrega y recepción de información. El sistema fue desarrollado mediante las fases de la metodología XP y con el apoyo de tecnologías y herramientas como el lenguaje de programación PHP y EXT-JS, Sistema Gestor de Base de Datos MySQL y el servidor Web Apache.

## *Abstract*

In the municipality of Sagua Tánamo production software it provides solutions to many problems related industries. This paper presents an able to optimize the management of contracts and the control of information technology in Agropecuaria Sagua de Tánamo Web application. These processes have difficulties today because they are developed in a decentralized and manual and often no possibility of access to records generated in recent time and that at some point of this is to access, providing delays by specialists delivering and receiving information. The system was developed by the phases of the XP methodology and supported by technologies and tools like the programming language PHP and EXT-JS, Manager MySQL Database System and the Apache Web server.

# Índice

<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA .....</b>	<b>5</b>
1.1 INTRODUCCIÓN .....	5
1.2 ESTADO DEL ARTE .....	5
1.2.1 Contrato.....	5
1.3 FLUJO ACTUAL DE LOS PROCESOS .....	6
1.4 ANÁLISIS CRÍTICO DE LA EJECUCIÓN DE LOS PROCESOS .....	7
1.5 REGLAS DEL NEGOCIO A CONSIDERAR .....	7
1.6 SISTEMAS INFORMÁTICOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN .....	8
1.6.1 OK Inventario 2.1.5 .....	8
1.6.2 FullTrust .....	9
1.6.3 SATEKO (Manejo Integral del Escritorio) Servicios profesional de cómputo .....	9
1.6.4 Herramienta para el control de los medios informáticos de la Empresa Cdte. “Ernesto Che Guevara”. .....	10
1.6.5 Sistema Informático para el Control de Inventario de Equipos de Cómputo en la Empresa Agropecuaria Sagua de Tánamo.....	10
1.7 TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	11
1.7.1 Lenguajes de programación .....	11
1.7.2 Sistemas gestores de base de datos.....	12
1.7.3 EMS SGL Manager para MySQL .....	14
1.7.4 ¿Qué es una aplicación web? ¿Por qué se utiliza? .....	15
1.7.5 Servidores para aplicaciones Web .....	18
1.7.6 Paquete de tecnología Web .....	20
1.7.7 EXT-JS.....	20
1.8 METODOLOGÍAS EXISTENTES .....	22
1.8.1 Metodologías Tradicionales .....	23
1.8.2 Metodologías Ágiles .....	24
1.9 ESTILO ARQUITECTÓNICO:.....	29
1.9.1 Arquitectura en 3 capas: .....	29
1.9.2 Justificación de la arquitectura a utilizar.....	31
1.10 HERRAMIENTAS CASE.....	31
1.11 CONCLUSIONES .....	33
<b>CAPÍTULO 2 REQUISITOS .....</b>	<b>34</b>
<b>Sistema para la Gestión de los contratos y el control de los medios informáticos en la Empresa Agropecuaria Sagua de Tánamo. (SIGCORA)</b>	



2.1	INTRODUCCIÓN .....	34
2.2	ACTORES DEL SISTEMA.....	34
	TABLA 2.2 DEFINICIÓN DE ACTORES DEL SISTEMA .....	34
2.3	DEFINICIÓN DE LOS REQUISITOS FUNCIONALES.....	34
2.4	HISTORIAS DE USUARIOS .....	38
2.5	ESTIMACIÓN DE ESFUERZO POR HU.....	40
2.5.1	<i>Planificación de Iteraciones</i> .....	42
2.5.2	<i>Plan de duración de las Iteraciones</i> .....	43
2.6	DIAGRAMA DE CASOS DE USO DEL SISTEMA .....	46
2.7	DEFINICIÓN DE LOS REQUISITOS NO FUNCIONALES .....	48
2.7.1	<i>Requisitos No Funcionales</i> .....	48
2.8	CONCLUSIONES .....	50
	<b>CAPÍTULO 3 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA .....</b>	<b>51</b>
3.1	INTRODUCCIÓN .....	51
3.2	DIAGRAMA DE CLASES DEL DISEÑO .....	51
	MODELO DE DISEÑO.....	51
3.2.1	<i>Gestionar Contrato</i> .....	52
3.2.2	<i>Gestionar Máquina</i> .....	53
3.3	DIAGRAMAS DE SECUENCIAS .....	54
3.4	DIAGRAMA DE DESPLIEGUE .....	55
3.5	DIAGRAMAS DE COMPONENTES .....	56
3.6	IMPLEMENTACIÓN DE LOS CASOS DE USOS CRÍTICOS.....	57
3.6.1	<i>Codificación</i> .....	57
	<i>Patrones de eventos con ExtJS</i> .....	58
3.7	MODELO DE DATOS .....	61
3.8	PRUEBA.....	63
3.8.1	<i>Pruebas de Aceptación</i> .....	63
3.9	CONCLUSIONES .....	64
	<b>CAPÍTULO 4 ESTUDIO DE FACTIBILIDAD .....</b>	<b>66</b>
4.1	INTRODUCCIÓN .....	66
4.2	ESTUDIO DE FACTIBILIDAD .....	66
4.2.1	<i>Factibilidad Técnica</i> .....	66
4.2.2	<i>Factibilidad Económica</i> .....	67
4.2.3	<i>Efectos Económicos</i> .....	68
4.2.4	<i>Fichas de Costo</i> .....	69
4.2.5	<i>Factibilidad Operativa</i> .....	73
	<b>Sistema para la Gestión de los contratos y el control de los medios informáticos en la</b> <b>Empresa Agropecuaria Sagua de Tánamo.</b> <b>(SIGCORA)</b>	

4.3 CONCLUSIONES DEL CAPÍTULO .....	74
<b>CONCLUSIONES GENERALES.....</b>	<b>75</b>
<b>RECOMENDACIONES .....</b>	<b>76</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>77</b>
<b>BIBLIOGRAFÍA .....</b>	<b>79</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>82</b>
<b>ANEXO 1. HISTORIA DE USUARIO .....</b>	<b>I</b>
<b>ANEXO 2. INTERFACES DEL SISTEMA.....</b>	<b>I</b>
<b>ANEXO 3. TARJETAS DE TAREAS .....</b>	<b>I</b>
<b>ANEXO 4. PRUEBAS DE ACEPTACIÓN POR HU .....</b>	<b>I</b>
<b>ANEXO 5. DIAGRAMAS DE SECUENCIAS.....</b>	<b>I</b>

## Índice de tabla

Tabla 2.2 Definición de actores del sistema .....	35
Tabla 2.3.1: Requisitos Funcionales.....	39
Tabla 2.4: Historia de Usuarios .....	40
Tabla 2.4.1: HU Gestionar Persona.....	41
Tabla 2.5: Estimación de Esfuerzo por HU .....	43
Tabla 2.5.1: Duración de Iteraciones.....	46
Tabla 3.8.1: Prueba de Aceptación a la HU Gestionar Personas.....	67
Tabla 4.2: Descripción de Requerimientos .....	67
Tabla 4.2.2: Ficha de Costo en CUC.....	70
Tabla 4.2.3: Ficha de Costo en MN .....	71

## Índice de figuras

Figura 1.9: Arquitectura de tres niveles .....	32
Figura 2.6 Diagrama del caso de uso del sistema para actor Administrador .....	49
Figura 2.7 Diagrama del caso de uso del sistema para actor Jurídico .....	50
Figura 2.8 Diagrama del caso de uso del sistema para los actores .....	51
Figura 3.2.1 Diagrama de clase de Diseño Gestionar Contrato .....	55
Figura 3.2.2 Diagrama de clase de Diseño Gestionar Maquina .....	56
Figura 3.3.1 Diagrama de Secuencia Insertar Contrato .....	59
Figura 3.3.2 Diagrama de Secuencia Insertar Maquina .....	60
Figura 3.4.1 Diagrama de Despliegue .....	61
Figura 3.5.1 Diagrama de Componente Gestionar Contrato .....	62
Figura 3.5.2 Diagrama de Componente Gestionar Maquina .....	63
Figura 3.6.1: Código del caso de uso Mostrar listado de contratos .....	64
Figura 3.6.2: código del caso de uso Mostrar listado de contratos .....	67
Figura 3.6.3: Vistas de la acción insertar Contrato .....	67
Figura 3.7: Modelo Físico de Datos .....	68
Figura 4.2: Factibilidad del Software .....	74

## INTRODUCCIÓN

La influencia de la evolución de las tecnologías para el mundo pone en evidencia la importancia de la relación hombre/máquina como eslabón fundamental en el desarrollo económico y social de los países, logrando integrarse en casi todas las esferas de la vida, acercándose la dependencia y necesidad, en muchas ocasiones, de optimizar los procesos cotidianos con el objetivo de viabilizar la gestión de la información.

Desde el Triunfo de la Revolución el Comandante en Jefe Fidel Castro aseguró que el futuro del país tenía que ser necesariamente un futuro de hombres de ciencia. Para hacer realidad estas aspiraciones se incrementa el número de centros e instituciones que hacen énfasis en el desarrollo las Tecnologías de la Información y las Comunicaciones (TIC) en Cuba.

El alcance que tienen hoy las TIC, se hace evidente con la incorporación de la informática a las diferentes esferas productivas que es ya una necesidad. Las formas de utilización de estas tecnologías en las esferas productivas del modelo económico en que se desenvuelven, traen consigo el ahorro de recursos, tiempo y facilidad de acceso a la información sin importar la distancia.

Contribuyendo al proceso de informatización del país se forman ingenieros informáticos en distintas universidades del sistema educacional cubano, capaces de resolver las más disímiles tareas, desarrollando múltiples aplicaciones informáticas orientadas a resolver las necesidades existentes en el país. Las herramientas desarrolladas son aplicadas a empresas o entidades.

En el municipio Sagua de Tánamo la producción de software brinda soluciones a muchos problemas relacionados con las industrias. Teniendo en cuenta la importancia del trabajo que se realiza en la Empresa Agropecuaria de Sagua de Tánamo se ha propuesto informatizar parte de los procesos que en ella se desarrollan.

Los procesos de control de los medios informáticos y gestión de los contratos de la entidad presentan dificultades, ya que muchas veces no hay posibilidad de

acceso a registros generados en tiempo pasados y que en algún momento del presente se necesitan consultar. Los datos recogidos la mayoría de las veces suelen perderse o extraviarse durante el proceso, los cuales pudieran aportar información de interés en el momento de emitir alguna valoración, proporcionando demoras por parte de los especialistas en la entrega y recepción de información.

Esta situación repercute de forma negativa sobre la empresa y no cumple a cabalidad los objetivos reales de la entidad.

Con el presente trabajo se pretende dar solución a la situación anteriormente expuesta, definiéndose como **Problema científico**: ¿Cómo favorecer el proceso de control de los medios informáticos y la gestión de los contratos en la Empresa Agropecuaria de Sagua de Tánamo?

Para darle solución al problema planteado se define como **Objeto de estudio**: informatización del proceso de control de los medios informáticos y gestión de los contratos de la entidad. Delimitando como **Campo de acción**: Aplicación informática del proceso de control de medios informáticos y gestión de los contratos en la Empresa Agropecuaria de Sagua de Tánamo.

**Objetivo general**: Implementar una aplicación web que favorezca el proceso de control de inventario de los medios informáticos y la gestión de los contratos en la empresa Agropecuaria de Sagua de Tánamo, aplicando herramientas avanzadas de informatización acorde a la tecnología con que cuenta la empresa, que permita la seguridad y confiabilidad de la información.

Tiene como **Idea a defender**: El desarrollo de una aplicación web que facilite la organización y el manejo cómodo de la información, en el menor tiempo posible, favorecerá el proceso de control de inventario de los equipos informáticos y la gestión de los contratos en la Empresa Agropecuaria de Sagua de Tánamo.

Se plantearon las siguientes **tareas específicas**:

- 1) Elaboración del Marco Teórico.
- 2) Caracterizar los fundamentos teóricos relacionados con la informatización de los procesos de gestión de los contratos y el control de los medios informáticos.
- 3) Estudio de las aplicaciones informáticas similares en el mundo y en Cuba para identificar procesos afines con el software a desarrollar.
- 4) Selección de las tecnologías y metodologías para el desarrollo de la aplicación.
- 5) Definición de las funcionalidades que va a contener la aplicación.
- 6) Diseño de una arquitectura que satisfaga los objetivos planteados.
- 7) Implementar aplicación informática para el Proceso de gestión y control de inventarios de equipos informáticos y los contratos.
- 8) Realización de pruebas a la aplicación informática desarrollada.
- 9) Realizar el análisis de factibilidad a la aplicación informática.
- 10) Confeccionar el Manual de Usuario.

Entre los métodos empleados para el desarrollo de estas tareas científicas podemos mencionar:

### **Métodos Teóricos:**

**Histórico-lógico:** Se utilizó para la búsqueda de antecedentes del software, las herramientas utilizadas, así como para determinar la forma en que se realiza el control de los medios informáticos y la gestión de los contratos en la Empresa Agropecuaria de Sagua de Tánamo.

**Análisis y síntesis:** Este método se utiliza en los fundamentos teóricos, en el procesamiento de la información y en la descomposición de cada uno de los requerimientos del sistema informático.

### **Métodos empíricos:**

**Observación:** Fue empleada en la caracterización de la ejecución de las actividades vigentes en la empresa, así como para evaluar el desarrollo de la investigación.

**Entrevista:** Se empleó en la determinación de los requerimientos funcionales del sistema que se va implementar. Se llevó a cabo un diálogo con los especialistas en la materia.

**El trabajo consta de introducción, cuatro capítulos, conclusiones generales, recomendaciones, bibliografía y anexos:**

**Introducción:** Se describe la problemática existente, el problema científico, objeto de estudio, campo de acción, objetivo general, idea a defender y tareas específicas.

**Capítulo 1-Fundamentación Teórica:** Brinda una descripción de los conceptos relacionados al objeto de estudio, el objetivo general y las tareas específicas, así como un análisis de los sistemas existentes vinculados al campo de acción, también se abordan temas específicos relacionados con la metodología y las herramientas a utilizar para el desarrollo del software.

**Capítulo 2-Planificación:** Se pone en práctica la utilización de la metodología abordada en el Capítulo 1 y detalles específicos en cada una de sus fases, la elaboración de las tarjetas de Clases, Responsabilidades y Colaboración (CRC) además de los requisitos de sistema para el desarrollo del proyecto y propuesta de interfaz de usuario.

**Capítulo 3-Diseño, implementación y prueba:** 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.

**Capítulo 4-Estudio de Factibilidad:** Se hace un análisis de los esfuerzos necesarios para la creación del sistema, su factibilidad y un gráfico de comparación de ventajas de la realización del proceso de la forma anterior y de la forma optimizada.



## Capítulo 1 Fundamentación Teórica

### *1.1 Introducción*

En el capítulo que a continuación se muestra, se abordan aspectos generales relacionados al objeto de estudio y los conceptos que se tratan en el mismo. Se hace un estudio de las aplicaciones informáticas existentes dedicadas a este fin. También se precisan las herramientas y tecnologías que son empleadas para dar cumplimiento al objetivo propuesto.

### *1.2 Estado del arte*

#### **1.2.1 Contrato**

El **contrato**, es un acuerdo de voluntades, verbal o escrito, manifestado en común entre dos, o más, personas con capacidad (partes del contrato), que se obligan en virtud del mismo, regulando sus relaciones relativas a una determinada finalidad o cosa, y a cuyo cumplimiento pueden compelerse de manera recíproca, si el contrato es bilateral, o compelerse una parte a la otra, si el contrato es unilateral.

Es el contrato, en suma, un acuerdo de voluntades que genera «derechos y obligaciones relativos», es decir, sólo para las partes contratantes y sus causahabientes. Pero, además del acuerdo de voluntades, algunos contratos exigen, para su perfección, otros hechos o actos de alcance jurídico, tales como efectuar una determinada entrega (*contratos reales*), o exigen ser formalizados en documento especial (*contratos formales*), de modo que, en esos casos especiales, no basta con la sola voluntad.

De todos modos, el contrato, en general, tiene una connotación patrimonial, incluso parcialmente en aquellos celebrados en el marco del derecho de familia,

y es parte de la categoría más amplia de los negocios jurídicos. Es función elemental del contrato originar *efectos jurídicos* (es decir, obligaciones exigibles), de modo que a aquella relación de sujetos que no derive en *efectos jurídicos* no se le puede atribuir cualidad contractual.(23)

## ***1.3 Flujo actual de los procesos***

En el proceso de control de los medios informáticos interviene la persona designada en el departamento de contabilidad para el control de los medios básicos de la empresa en conjunto con el administrador redes, con el objetivo de realizar el control de inventario. El administrador revisa las máquinas con todos sus complementos y el contador el equipo, como unidad de la entidad. Este se realiza cada tres meses.

El proceso de gestión de los contratos se realiza a través del departamento jurídico con el campesino. Una vez que el campesino entrega el expediente previamente elaborado y aprobado por la delegación de la agricultura se procede a elaborar el contrato, el cual debe tener el número de expediente, los datos personales del campesino y la actividad que va a desempeñar, entre otros elementos.

El campesino ya con el contrato realizado se convierte en usufructuario, pasa por donde el jurídico que le firma el modelo en conjunto con la económica y pasa por la caja y paga la bienichuria, dejando copia en el jurídico, en la caja y él mantiene el original, como constancia de que el pago se realizó.

El jurídico hace una lista con los datos personales y los del contrato de todos los usufructuarios con el objetivo de facilitar la búsqueda de la información a la hora redactar los informes solicitados por la administración o de hacer algún cambio en algún contrato.

Estos cambios se hacen en los siguientes casos:

Cuando un campesino se acerca a la empresa con el objetivo de vender la propiedad a la entidad o realizar un traspaso a un familiar en caso de enfermedad

Un familiar del dueño de la propiedad que fallece.

El jurídico busca en la lista que se encuentra en formato Excel el nombre y el número de expediente de la persona, con estos datos localiza el expediente y procede a hacer los cambios pertinentes.

En caso de que el campesino no desee seguir con el contrato o la empresa verifica que el campesino no está cumpliendo con lo establecido en el contrato se procede a darlo por concluido.

Por otra parte la administración solicita al jurídico las siguientes informaciones:

Usufructuarios por zona,

Contratos vencidos y por vencer

Cantidad de contratos.

## ***1.4 Análisis crítico de la ejecución de los procesos***

En la empresa Agropecuaria de Sagua de Tánamo la gestión de los contratos se hace muy complicada a la hora de tomar los datos, por lo que hay un alto nivel de error a la hora de la toma de los mismos por el gran volumen de documentación que se genera. Existe gran dificultad al momento de trabajar con esta información en el instante de dar reportes, entorpeciendo en gran parte el tiempo de respuesta de los especialistas.

Lo explicado anteriormente ocurre también con el control de los medios informáticos, entorpeciendo el proceso a la hora de trabajar con esta información ya que se encuentra descentralizada y con alto nivel de riesgo en cuanto a la pérdida o deterioro de la documentación. Esto provoca que existan problemas con los reportes que se deben de entregar cuando soliciten por parte de la dirección de la entidad y otros organismos rectores de la actividad.

## ***1.5 Reglas del negocio a considerar***

- Solo el personal jurídico puede realizar los contratos.
- Los contratos tienen un tiempo de vida de hasta 10 años y se mantienen archivados por otros 5 por ser documentos auditables..

- Los campesinos no podrán tener activos más de un contrato.
- El administrador no puede realizar el cambio de ningún componente y periféricos de la maquina ya que lo realiza el personal de Copextel.

## *1.6 Sistemas informáticos existentes vinculados al campo de acción*

En la Empresa Agropecuaria no existe ningún sistema informático destinado a la gestión de información de los equipos de cómputo y el control de los contratos. Se ha realizado una investigación acerca de algún software que realice estas funciones dentro y fuera del país arrojando los siguientes resultados.

### **1.6.1 OK Inventario 2.1.5**

OK-Inventario es un software innovador, versátil y personalizable diseñado para proveer soluciones para la contabilidad, gestión empresarial, control de almacén de inventario, gestión de ventas y gestión de documentación para las pequeñas y medianas empresas comerciales y de fabricación.

Nuestro software le ayudará a rastrear la información esencial sobre su empresa y hacer el trabajo de contabilidad mucho más fácil. La interfaz del programa es muy simple y fácil de usar. Sólo le tomará dos o tres días de entrenamiento antes de aprender los elementos clave del programa y empezar a usarlo para su negocio.

OK-inventario ayuda a administrar todos los negocios y los procesos de contabilidad de su empresa. Sus principales funcionalidades están dirigidas a **(2)**:

- Gestión de ventas y suministros.
- Finanzas.
- Gestión de almacenes e inventario.

## 1.6.2 FullTrust

FullTrust es un software diseñado especialmente para gestionar pequeños y medianos comercios, cubriendo áreas tales como el control de inventarios, la gestión de clientes, presupuestos, pedidos y ventas, entre otros. FullTrust incluye un terminal de punto de venta listo para ser usado con lectores de código de barra. FullTrust posee una serie de prestaciones adicionales no presentes en otros paquetes de software similares, como por ejemplo una utilidad de para gestionar ventas a crédito, una utilidad para el diseño e impresión de etiquetas de código de barra y también un módulo para la gestión de gastos del negocio.

Una de las principales diferencias de FullTrust en relación a otros programas similares es que FullTrust usa PostgreSQL como motor de bases de datos. **(3)**

## 1.6.3 SATEKO (Manejo Integral del Escritorio) Servicios profesional de cómputo

Presta servicio para el manejo de sistema de cómputo y otros elementos de escritorio, tales como impresoras, fax, arañas de audio-conferencia, para facilitar y agilizar las respuesta a sus requerimientos cotidianos desde una plataforma unificada, mejorando los tiempos de atención, disminuyendo los gastos operativos y liberado, con el fin de hacer más eficiente y rentable su Organización. **(22)**

Alcance de servicio que presta:

- ❖ Back-up diario del PC
- ❖ Envío de información a servidores
- ❖ Back-up In Cloud (almacenamiento de datos en la nube privada o pública)
- ❖ Manejo de actualizaciones en los equipos del cliente
- ❖ Instalación de parche y actualizaciones

### **1.6.4 Herramienta para el control de los medios informáticos de la Empresa Cdte. “Ernesto Che Guevara”.**

Es una propuesta de herramienta informática basada en la tecnología web para el control de los medios informáticos de la empresa del Níquel Ernesto Che Guevara de Moa. La misma nunca se puso en explotación ya que no se llegó a su implementación. Las funcionalidades que se tienen concebida no satisfacen a la Empresa Agropecuaria Sagua de Tánamo.

### **1.6.5 Sistema Informático para el Control de Inventario de Equipos de Cómputo en la Empresa Agropecuaria Sagua de Tánamo.**

Este software está basado en la tecnología web, el mismo es para el control de los medios informáticos de la Empresa Agropecuaria Sagua de Tánamo. Las funcionalidades implementadas son las siguientes (5):

- Hoja de Usuario
- Registrar Usuario
- Listar Usuario
- Listar Inventario
- Buscar Inventario
- Registrar Monitor
- Registrar CPU
- Registrar Conjunto
- Exportar Datos a PDF
- Autenticar usuario

A pesar de haberse creado la herramienta informática, no se llegó a utilizar porque no cumplía con las necesidades de la empresa en cuestión, ya que no cumple en su totalidad con el proceso de control de los medios informáticos. Además por política de la empresa se rediseñó la estrategia de desarrollo de aplicaciones web por lo que la desarrollada anteriormente no estaba acorde.

De forma general las aplicaciones antes estudiadas son completísimos software que realizan múltiples funcionalidades como: generar resultados estadísticos, gestionar ventas y suministros, finanzas, etc.; pero tienen grandes desventajas y es que no son libres ni gratuitos por lo que esto influye a la hora de mejorar su funcionalidades, que al no poseer su código fuente es imposible trabajar en la mejora local. También se pone a prueba el factor del costo de los mismos, ya que son muy caros y su forma de adquisición es por vía internet, por la cual hay que pagar licencias y llaves.

## *1.7 Tendencias y tecnologías actuales*

### **1.7.1 Lenguajes de programación**

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en el conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente.

**Pueden ser divididos en dos grupos:**

**Del lado del cliente:**

**Java Script:** Esto es un lenguaje interpretado por lo que no requiere compilación. Fue creado por la empresa Netscape Communication. Es similar al Java, aunque no es un lenguaje orientado a objetos (POO). La mayoría de los navegadores en sus últimas versiones interpretan JavaScript. El código de JavaScript puede ser integrado a nuestras páginas Web.

**Del lado del servidor**

**PHP**

Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de líneas de comando o en la creación de otros tipos de

programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas QT (Quasar Toolkit).

PHP es un acrónimo recursivo que significa Hypertext Pre-processor (inicialmente PHP Tools o Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdorf en 1994. **(8)**

## ¿Por qué PHP?

### Ventajas de PHP

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de bases de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial en la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientadas a Objetos.
- Biblioteca nativa de funciones sumamente amplias e incluidas.
- No requiere definición de tipo de variables.
- Tiene manejo de excepciones.

### 1.7.2 Sistemas gestores de base de datos

“Un Sistema Gestor de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una Base de Datos (BD), el cual es un software de propósito general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones. Pueden ser de propósito general o específico”. Existen varios SGBD, como son: SQL Server, Postgres, MySQL, Oracle. **(9)**



## MySQL

Es un sistema de gestión de base de datos relacional, multi plataforma, multi-hilo y multi-usuario, 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 del 2008. Por un lado lo ofrece GNU GPL (General Public License), pero entidades que quieran incorporarlo en productos privados pueden comprar a la empresa una licencia que les permita ese uso. Está desarrollando en su mayor parte ANSI C (Estándar desarrollado por el Instituto Nacional Estadounidense de Estándares).

Al contrario de proyectos como Apache donde el software es desarrollado por una comunidad publica y el copyright del código está en el poder del autor individual. MySQL está poseído y patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el licenciamiento anteriormente abordado. Además de las ventas de licencias privadas, 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. (10)

### **MySQL cuenta con muchas ventajas, entre las que se destacan:**

- **Alto rendimiento:** Es muy rápido con respecto a otros gestores de bases de datos.
- **Bajo coste:** Está disponible de manera gratuita, bajo una licencia de código abierto, o por un precio reducido en forma de licencia comercial si resultara necesario para su aplicación.
- **Facilidad de configuración y aprendizaje:** Las bases de datos más modernas utilizan SQL. Si ha utilizado otros RDBMS, no debería tener problemas para adaptarse a este sistema. MySQL resulta además más sencillo de configurar que otros productos similares.
- **Portabilidad:** Se puede utilizar en una gran cantidad de sistemas Unix diferentes así como bajo Microsoft Windows.

- **Accesibilidad a código fuente:** Como en el caso de PHP, puede obtener y modificar el código fuente de MySQL.

### 1.7.3 EMS SQL Manager para MySQL.

Es una herramienta de alto rendimiento para la administración de MySQLDatabase Server y el desarrollo. Funciona con cualquier versión de MySQL desde la 3.23 a la más reciente y soporta todas las últimas características de MySQL, incluyendo disparadores, vistas, procedimientos almacenados y funciones, claves foráneas InnoDB, los datos Unicode y así sucesivamente.

- SQL Manager permite crear, editar todos los objetos de base de datos MySQL
- El diseño de bases de datos MySQL visualmente
- Ejecutar scripts de SQL
- Importar y exportar datos de bases de datos MySQL
- Gestión de usuarios y sus privilegios
- Tiene muchas otras características útiles para la eficiente administración de MySQL

### ¿Por qué EMS SQL Manager para MySQL?

Se propone para la realización de esta aplicación la utilización del Manager para MySQL porque brinda una rápida y sencilla administración y navegación de las bases de datos, además de gestión de seguridad y fácil creación de visuales.

Existen otras herramientas que no pierden su importancia y su vigencia en desarrollo de software, pero no son mencionadas en este documento debido a que el software no se acomoda a la Intranet de la Empresa Agropecuaria Sagua de Tánamo y se hace uso de las herramientas antes mencionadas, por política de la empresa.

## 1.7.4 ¿Qué es una aplicación web? ¿Por qué se utiliza?

Una aplicación Web es un sistema informático que los usuarios usan accediendo a un servidor Web a través de los protocolos de Internet. Las aplicaciones Web son populares por su practicidad del navegador Web como cliente ligero.

La facilidad de actualizar y mantener las aplicaciones Web sin distribuir e instalar software en miles de estaciones de trabajo es otra razón de su creciente popularidad.

“Una aplicación Web esta comúnmente estructurada como una aplicación en tres capas. En su forma más común el navegador es la primera capa, un motor usando alguna tecnología de Web dinámica (Ej: CGI (Common Gateway Interface) PHP (Hypertext Pre-processor), ASP (Active Server Pages), Java Servlets o Perl (Practical Extracting and Reporting Language) es la capa del medio, y algún tipo de gestor de bases de datos como última capa”.(6)

En tiempos recientes se ha usado la estrategia de generalizar esta arquitectura mediante la adición de piezas de hardware que permitan balancear la carga de los servidores Web y de las aplicaciones.

Las aplicaciones Web son de un desarrollo poco costoso, sencillo y rápido. “Presentan acceso ubicuo, sin necesidad de distribución e idealmente con pocos requerimientos técnicos. Con datos centralizados y fácil integración de datos múltiples fuentes”. (6)

### **Ventajas de la Web:**

- **Compatibilidad multiplataforma:** Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargable. Varias tecnologías incluyendo Java, Flash, ASP (Active Server Pages) y Ajax (Asynchronous JavaScript And XML) permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.
- **Actualización:** Las aplicaciones basadas en Web están siempre actualizadas con el último lanzamiento sin requerir que el usuario tome acciones pro-activas, y sin necesitar llamar la atención del usuario o

interferir con sus hábitos de trabajo, pues no se hace necesario iniciar nuevas descargas y/o procedimientos de instalación (algunas veces imposibles cuando usted está trabajando dentro de grandes organizaciones).

- **Inmediatez de acceso:** Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online y está listo para trabajar sin importar cuál es su configuración o su hardware.
- **Menos requerimientos de su memoria:** Las aplicaciones basadas en web tienen menos demandas de memoria RAM de parte del usuario final que los programas instalados localmente. Al residir y correr en los servidores del proveedor, esas aplicaciones basadas en Web usan en muchos casos la memoria de las computadoras donde ellas corren, dejando más espacio para correr múltiples aplicaciones sin incurrir en frustrantes deterioros en el rendimiento.
- **Menos Bugs:** Las aplicaciones son menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con las aplicaciones basadas en web, todos utilizan la misma versión, y todos los bugs pueden ser corregidos tan pronto como son descubiertos.
- **Precio:** Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y marketing requerido por el software descargable tradicional. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables y no totalmente gratuitas, mientras que ofrecen componentes adicionales y servicios Premium como una opción.
- **Los datos también son online:** El hecho de que el manejo de los datos sea realizado de forma remota libra al usuario de la responsabilidad en la protección de los mismos y al mismo tiempo logra que los recursos sean accesibles en cualquier momento.

- **Múltiples usuarios concurrentes:** las aplicaciones basadas en Web pueden ser utilizadas por múltiples usuarios al mismo tiempo. No hay más necesidad de compartir pantallas o enviar instantáneas cuando múltiples usuarios pueden ver o incluso editar el mismo documento de manera conjunta.
- **Los datos son más seguros:** Si bien la ruptura de discos no va a desaparecer, es probable que los usuarios escuchen mucho menos del tema. A medida que las compañías se hagan cargo del almacenamiento de los datos del usuario, granjas de almacenamiento de datos redundantes, altamente fiables, los usuarios van a tener mucho menos riesgo de perder sus datos debido a una ruptura de discos impredecible o a un virus de la computadora. Las compañías que proveen aplicaciones basadas en Web van a brindar amplios servicios de resguardo de datos ya sea como una parte integral del servicio básico o como una opción paga.
- **Desarrollar aplicaciones en el lenguaje que usted quiera:** Una vez que las aplicaciones han sido separadas de computadoras locales y sistemas operativos específicos, pueden también ser escritas en prácticamente cualquier lenguaje de programación. Debido a que las aplicaciones Web esencialmente son una colección de programas más que un simple programa, ellas podrían ser escritas en cualquier lenguaje de programación existente. (7)

## **Desventajas de la Web (20):**

- Las aplicaciones web requieren navegadores web totalmente compatibles para funcionar. Incluso muchas veces requieren las extensiones apropiadas y actualizadas para operar.
- Muchas veces requieren una conexión a internet para funcionar, si la misma se interrumpe, no es posible utilizarla más. De todas maneras, en

ocasiones, pueden ser descargadas e instaladas localmente para su uso offline.

- Muchas no son de código abierto, perdiendo flexibilidad.
- La aplicación web desaparece si así lo requiere el desarrollador o si el mismo se extingue. Las aplicaciones tradicionales, en general, pueden seguir usándose en esos casos.
- El usuario, en general, no tiene libertad de elegir la versión de la aplicación web que quiere usar. Un usuario podría preferir usar una versión más antigua, hasta que la nueva sea probada.
- En teoría, el desarrollador de la aplicación web puede rastrear cualquier actividad que el usuario haga. Esto puede traer problemas de privacidad.
- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones Web (mediante formularios principalmente) son muy escasas.

## 1.7.5 Servidores para aplicaciones Web

“El servidor Web es un programa que corre sobre el servidor que escucha las peticiones HTTP (Hyper Text Markup Language) que le llegan y las satisface. Dependiendo del tipo de petición, el servidor Web buscará una página Web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición”. (11)

Entre los servidores de aplicaciones web más usados se encuentran: Apache, Zope, IIS, Tomcat. Se selecciona Apache para el desarrollo del sistema propuesto por las siguientes características:

Sustancialmente, es un proyecto nacido para crear un servidor de Web estable, fiable y veloz para plataformas Unix. Apache nace, por una parte, de un código ya existente y de una serie de parches (patch) para mejorar su fiabilidad y sus características; de ahí su nombre. Tiene amplia aceptación en la

red: desde 1996, es el servidor HTTP más usado en el mundo y alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo. **(13)**

Otras particularidades que nos brinda este servidor son:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables y están ahí para que se instalen cuando se necesiten. Otra cosa importante es que cualquiera que posea alguna experiencia en la programación de C o Perl (Practical Extracting and Reporting Language) puede escribir un módulo para realizar una función determinada.
- Trabaja con Perl, PHP y otros lenguajes de Script. Perl se destaca en el mundo del Script y Apache utiliza su parte del pastel de Perl tanto con soporte CGI (Commun Gateway Interfaces), como con Java y páginas JSP (Paginas de Servidor Java). Teniendo todo el soporte que se necesita para tener páginas dinámicas.
- Permite personalizar la respuesta ante los errores que se puedan dar en el servidor. Es posible configurarlo para que se ejecute un determinado script cuando ocurra un error en concreto. **(12)**

## **Justificación de servidor Web a utilizar**

Para la creación del software se utilizó como Servidor Web Apache el cual es el complemento perfecto para páginas dinámicas desarrolladas con PHP y MySQL. Comparte con estos muchas de sus características, como son la gratuidad (también se tratan de software libre), su popularidad, su sencillez de manejo y su versatilidad, ya que podemos instalarlo sobre Linux o sobre Windows. Es por ello que es el utilizado en este proyecto.

## 1.7.6 Paquete de tecnología 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 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 Sistemas Operativos: Microsoft Windows, GNU/Linux, Solaris, y MacOS X”. **(14)**

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 existan otras herramientas como son los lenguajes de programación Web, gestores de bases de datos y servidores Web, aunque no se mencionan y se describen, no pierden su vigencia, funcionalidad e importancia en esta investigación solo se hace referencia:

- A los lenguajes de programación, JavaScript y PHP (Personal Home Page).
- El gestor de bases de datos, MySQL.
- El paquete de tecnología Web, XAMPP. **(14)**

## 1.7.7 EXT-JS

En el mercado actualmente existen múltiples librerías de Javascript que permiten realizar todo tipo de maravillas en el navegador web. Ext-JS permite que con pocas líneas de código sea posible realizar interfaces amigables para los usuarios. Es la librería Javascript más avanzada para el desarrollo rápido de aplicaciones con una apariencia totalmente novedosa y una arquitectura flexible que permite construir aplicaciones complejas. **(15)**



Esta librería incluye:

- Componentes UI (Interfaz de Usuarios) de alto rendimiento y personalizables.
- Modelo de componentes extensibles.
- Un API (Application Programming Interface) fácil de usar.
- Licencias Open Source (Código Abierto) y comerciales. **(15)**

## Facilidades que Brinda EXT-JS

Ext-JS permite construir aplicaciones RIA, acrónimo de Rich Internet Applications (Aplicaciones Ricas en Internet). Lo que RIA intenta proveer es aquello de lo que siempre ha adolecido la web, una experiencia de usuario muy parecida o igual a la que se tiene en las aplicaciones de escritorio.

Las aplicaciones web tradicionales tienen problemas como la recarga continua de las páginas cada vez que el usuario pide nuevo contenido, o la poca capacidad multimedia, para lo cual se han hecho necesarios plug-ins<sup>1</sup> externos. **(15)**

Encaja dentro de este esquema como un motor que permite crear aplicaciones RIA mediante Javascript. Si se enmarca a Ext-JS dentro del desarrollo RIA, éste sería el render de la aplicación que controla el cliente y que se encarga de enviar y obtener información del servicio.

Una de las grandes ventajas de utilizar Ext-JS es que permite crear aplicaciones complejas aprovechando así sus componentes predefinidos así como un manejador de layouts similar al que provee Java Swing, gracias a esto provee una experiencia consistente sobre cualquier navegador, evitando el tedioso

---

<sup>1</sup>(**plug-in**). Es un módulo de hardware o software que añade una característica o un servicio específico a un sistema más grande.

problema de validar que el código escrito funcione bien en cada uno (Firefox, Internet Explorer, Safari, Google Chrome). **(15)**

Usar un motor de render como Ext-JS, permite tener además estos beneficios:

- **Existe un balance entre Cliente – Servidor:** La carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo.
- **Comunicación asíncrona:** En este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un clic o una acción del usuario, dándole la libertad de cargar información sin que el cliente se dé cuenta.
- **Eficiencia de la red:** El tráfico de red puede disminuir al permitir que la aplicación elija qué información desea transmitir al servidor y viceversa, sin embargo la aplicación que haga uso de la pre-carga de datos puede revertir este beneficio por el incremento del tráfico.

## *1.8 Metodologías existentes*

En la actualidad el desarrollo de software está estrechamente vinculado a las actividades realizadas para llevarlo a cabo, en un proyecto de desarrollo de software la metodología define Quién debe hacer Qué, Cuando y Como debe realizarlo. Una metodología es un proceso que rige el desarrollo del proyecto. No existe una metodología de desarrollo de software universal, las características de cada proyecto exigen que el proceso sea confiable. Es de vital importancia saber cuál es la metodología a utilizar teniendo en cuenta las características y las propiedades del producto a realizar.

Las metodologías existentes en la actualidad se dividen en dos grandes grupos teniendo en cuenta sus características: Las tradicionales: RUP (Rational Unified Process), MSF (Microsoft Solution Framework), así como las ágiles XP: (Extreme Programming), SCRUM. Las primeras basadas en el uso exhaustivo de documentación durante todo el ciclo de vida del proyecto,

mientras que las segundas ponen vital importancia en la capacidad de respuestas a los cambios, la confianza en las habilidades del equipo de trabajo y al mantener una buena comunicación con el cliente. **(19)**

## 1.8.1 Metodologías Tradicionales

Persiguen la consecución de un proyecto cuya planificación está definida. Contando con los medios adecuados para controlar su ejecución, corregir posibles desviaciones y documentar lo realizado.

### Proceso Unificado de Desarrollo (RUP)

RUP es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del UML y trabajo de muchas metodologías utilizadas por los clientes.

El ciclo de vida de RUP se caracteriza por:

- Dirigido por Casos de Uso (CU)
- Centrado en la arquitectura
- Iterativo e incremental

Es práctico dividir el trabajo en partes más pequeñas o miniproyectos, cada uno de ellos es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. Cada iteración se realiza de forma planificada es por eso que se dice que son miniproyectos. **(19)**

### Ventajas de RUP:

- Es sencillo, ya que sigue los pasos intuitivos necesarios a la hora de desarrollar el software.
- Seguimiento detallado en cada una de las fases.
- No se necesita interacción con el cliente en todo el ciclo de desarrollo.
- Gran número de documentación que proporciona un control estricto de todo lo que pasó a lo largo del proyecto.

- Se define el alcance del proyecto desde un principio.

## **Desventajas de RUP:**

- La evaluación de riesgos es compleja.
- El cliente deberá ser capaz de describir y entender a un gran nivel de detalle para poder acordar un alcance del proyecto con él.

### **1.8.2 Metodologías Ágiles**

La comunidad de programadores se planteó utilizar un nuevo paradigma, radicalmente diferente a las metodologías tradicionales. Metodologías más ágiles, capaces de lidiar con ese entorno caótico que es el desarrollo de software, para así pasar a tener el caos controlado y poder llegar a resultados prácticos más fácilmente.

#### **XP (Extreme Programming)**

Es un enfoque de la ingeniería de software formulado por Kent Beck, autor del primer libro sobre la materia, *Extreme Programming Explained. Embrace Change* (1999).

Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que estos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad, que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos.

Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos. Se puede considerar la programación extrema como la adición de las mejores metodologías de desarrollo de acuerdo a lo que se pretenda realizar con el proyecto y aplicarlo de manera dinámica durante el ciclo de vida del software.

La programación extrema se basa en la simplicidad, la comunicación y el reciclado continuo de código, para algunos no es más que aplicar una pura lógica. Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clímax 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.

## **Procesos de XP**

**El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:**

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

“En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración. El ciclo de vida de XP consiste en seis fases: Exploración, Planificación para la Entrega, Iteraciones, Producción, Mantenimiento y Muerte del proyecto”. **(16)**

**Las seis prácticas esenciales por donde se debe empezar:**

- Pruebas

- Refactorización
- Programación en pares
- Juego de planificación
- Liberaciones pequeñas
- Integraciones sistemáticas

## Fases de la metodología XP

### Fase 1: Planificación

- Se escriben historias de usuarios, 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 forma que creen un test de aceptación para historia de usuarios (user story) y permitan 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 usuarios, 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 usuarios a desarrollar y las tareas de desarrollo.
- Se cambia el proceso cuanto sea necesario, para adaptarlo al proyecto.

### Fase 2: 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 de código.
- Se escriben tarjetas de clases-responsabilidades-colaboración (CRC)

para cada objeto, que permita abstraerse al pensamiento estructurado y que el equipo de desarrollo completo participe en el diseño.

## Fase 3: 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. 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.
- El código se ajustará a unos estándares de codificación, asegurando la consistencia y facilidad de comprensión y refactorización del mismo.
- 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 esta 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 solo habrá una pareja de programadores que integre el código.
- Se integra código y se lanza dicha integración de manera frecuente, evitando divergencias en el 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 extras de trabajo.

## Fase 4: 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 error.
- Se realizan pruebas de aceptación frecuentemente, publicando los resultados de las mismas. Estas pruebas son generalmente a partir de las historias de usuarios 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 la correspondiente historia de usuario se ha completado.

## ¿Por qué XP?

Actualmente XP es el método ágil más documentado (hay una colección de libros con XP Series de Addison Wesley) y extendido. Existe una gran comunidad de desarrolladores XP. Otra de las ventajas de XP es que no es necesario adoptarlo en forma completa, sino que pueden utilizarse varias de sus prácticas en forma independiente. Esto hace que el costo de su implementación sea mucho más accesible que el de otras metodologías.

Un estudio a la bibliografía, muestra las ventajas y desventajas que tiene XP y que exponemos en los puntos siguientes.

## Ventajas de XP

- Puede ser implementado en forma parcial (elegir sólo algunas de las prácticas).
- Puede ser implementado en forma gradual.
- Puede adaptarse a las necesidades de cualquier equipo de desarrollo. De hecho, Kent Beck recomienda a los equipos que lo adapten a sus necesidades.
- Exige que se establezca una comunicación más fluida con el cliente y que este tenga mayor participación en el proceso de desarrollo. La consecuencia de esto es que el cliente se involucre más en el desarrollo del producto.



- Actualmente es la metodología ágil más extendida y documentada. Se realizan pruebas constantemente del sistema. **(16)**

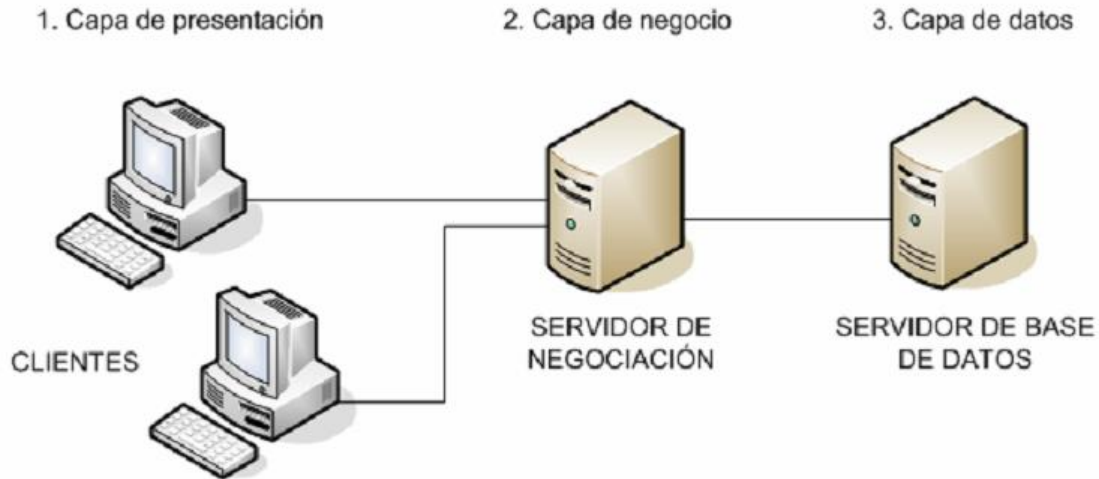
## ***1.9 Estilo Arquitectónico:***

La Arquitectura es el esqueleto o base de una aplicación. Representa la organización fundamental de un sistema. Desde los pequeños programas hasta los sistemas más grandes poseen una estructura y un comportamiento que los hace clasificables según su "arquitectura". En la Web es muy común la utilización de la arquitectura 3-capas, n-capas, MVC, entre otras.

### **1.9.1 Arquitectura en 3 capas:**

Arquitectura en capas: es donde se define como organizar el modelo de diseño a través de capas, que pueden estar físicamente distribuidas, lo que quiere decir que los componentes de una capa solo pueden hacer referencia a componentes en capas inmediatamente inferiores. Este patrón es importante porque simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no son consistentes de ningún detalle o interfaz de las superiores.

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios, de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.



**Figura 1.9: Arquitectura de tres niveles**

## Capas o Niveles

**Capa de presentación:** es la que ve el usuario (hay quien la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

**Capa de negocio:** es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.

**Capa de datos:** es donde residen los datos y es la encargada de acceder a ellos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único ordenador (no es lo típico). Si bien lo más usual es que haya una multitud de ordenadores en donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio.

## 1.9.2 Justificación de la arquitectura a utilizar

Se propone como estilo arquitectónico a utilizar en el proyecto la **arquitectura en tres capas** debido a las ventajas que esta brinda:

### **Ventajas de esta Arquitectura**

La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio solo 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 niveles; simplemente es necesario conocer la API (Application Program Interface) que existe entre niveles. En el diseño de sistemas informáticos actual se suele usar las arquitecturas multi-nivel 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).

## *1.10 Herramientas Case*

**Embarcadero ER-Studio:** 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. 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. (17)

## **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.(18)

**Rational Rose Enterprise Edition:** Está basada en el Lenguaje Unificado de Modelación (UML), que permite crear los diagramas que se van generando durante el proceso de Ingeniería en el Desarrollo del Software.(4)

## **Funciones**

Incluye también estas funciones:

1. Soporte a modelos de análisis, ANSI C++, RoseJ y Visual C++ según el documento "Design Patterns: Elements of Reusable Object – Oriented Software".
2. Los componentes del modelo se pueden controlar independientemente, lo que permite una gestión y un uso de modelos más granular.
3. Soporte para compilación y descompilación de las construcciones más habituales de Java 1.5.
4. Generación de código en lenguaje Ada, ANSI C++, C++, CORBA, Java y Visual Basic, con funciones configurables de sincronización entre los modelos y el código.
5. Soporte para enterprise Java Beans 2.0.
6. Funciones de análisis de calidad de código.
7. Complemento de modelado Web que incluye funciones de visualización, modelado y herramientas para desarrollar aplicaciones Web.

8. Modelado en UML para diseñar bases de datos, que integra los requisitos de datos y aplicaciones mediante diseños lógicos y analíticos.
9. Creación de definiciones de tipo de documentos DTD en XML.
10. Integración con otras herramientas de desarrollo de IBM Rational.
11. Integración con cualquier sistema de control de versiones compatibles con SSC, como IBM Rational ClearCase.
12. Posibilidad de publicar en las Web modelos e informes para mejorar la comunicación entre los miembros del equipo.

### ***1.11 Conclusiones***

Para dar solución al problema que da origen a esta investigación, se brindan detalles de las herramientas y lenguajes que fueron utilizadas para la confección del problema, se puede decir que está creada la base para la realización del sistema propuesto. Guiando el desarrollo de este software con la metodología XP.

## Capítulo 2 Requisitos

### *2.1 Introducción*

En este Capítulo, se pone en práctica la fase de planeación y requisitos en la cual se determinan y detallan a fondo las necesidades del cliente. Se plantean los requisitos funcionales y no funcionales de Sistema propuesto, los mismos serán implementados mediante el empleo de historias de usuarios (HU). Para lo cual será necesaria una estimación del esfuerzo en la realización de las mismas y el establecimiento del plan de iteraciones del sistema y su duración, el plan de entrega del proyecto y se creará una descripción de las interfaces de usuarios.

### *2.2 Actores del Sistema*

**Tabla 2.2 Definición de actores del sistema**

Nombre del actor	Descripción
<b>Administrador</b>	<b>Persona encargada de asesorar, dar seguimiento al estado de la aplicación y es el encargado de controlar los medios informáticos.</b>
<b>Personal Jurídico</b>	<b>Personal encargado de gestionar los contrato de la entidad</b>

### *2.3 Definición de los requisitos funcionales*

**Tabla 2.3.1: Requisitos Funcionales**

Requisitos Funcionales
<b>1. Insertar Usuarios</b>
<b>2. Mostrar Usuarios</b>
<b>3. Modificar Usuarios</b>

4. Eliminar Usuarios
5. Insertar Quejas
6. Mostrar Quejas
7. Eliminar Quejas
8. Insertar Persona
9. Mostrar Persona
10. Modificar Persona
11. Eliminar Persona
12. Insertar Departamento
13. Mostrar Departamento
14. Modificar Departamento
15. Eliminar Departamento
16. Insertar PC
17. Mostrar PC
18. Modificar PC
19. Eliminar PC
20. Insertar Componente
21. Insertar Periféricos
22. Mostrar MotherBoard
23. Modificar MotherBoard

<b>24. Eliminar MotherBoard</b>
<b>25. Mostrar Disco</b>
<b>26. Modificar Disco</b>
<b>27. Eliminar Disco</b>
<b>28. Mostrar RAM</b>
<b>29. Modificar RAM</b>
<b>30. Eliminar RAM</b>
<b>31. Mostrar CPU</b>
<b>32. Modificar CPU</b>
<b>33. Eliminar CPU</b>
<b>34. Mostrar Lector</b>
<b>35. Modificar Lector</b>
<b>36. Eliminar Lector</b>
<b>37. Mostrar Fuente</b>
<b>38. Modificar Fuente</b>
<b>39. Eliminar Fuente</b>
<b>40. Mostrar Monitor</b>
<b>41. Modificar Monitor</b>
<b>42. Eliminar Monitor</b>
<b>43. Mostrar Teclado</b>



44. Modificar Teclado
45. Eliminar Teclado
46. Mostrar Mouse
47. Modificar Mouse
48. Eliminar Mouse
49. Mostrar Bocina
50. Modificar Bocina
51. Eliminar Bocina
52. Insertar USB
53. Mostrar USB
54. Modificar USB
55. Eliminar USB
56. Insertar Contrato
57. Mostrar Contrato
58. Modificar Contrato
59. Eliminar Contrato
60. Mostrar Tierra
61. Modificar Tierra
62. Eliminar Tierra
63. Mostrar Usufructuario

64. Modificar Usufructuario
65. Eliminar Usufructuario
66. Graficar PC*Dpto
67. Graficar USB*Capacidad
68. Insertar Contrato Entidad
69. Mostrar Contrato Entidad
70. Modificar Contrato Entidad
71. Eliminar Contrato Entidad
72. Mostrar Usufructuario con deudas
73. Mostrar Usufructuario sin deudas
74. Mostrar Cantidad Usufructuario * consejo popular
75. Mostrar Cantidad Usufructuario * zona
76. Mostrar Contratos vencidos
77. Mostrar Contratos vigentes
78. Graficar Contrato

### ***2.4 Historias de Usuarios***

Las Historias de Usuarios (HU), son la técnica utilizada en XP para la descripción de los requisitos del software. Son el resultado directo del intercambio entre cliente - desarrollador a través de reuniones programadas donde las tormentas de ideas (brainstorming) arrojan no solo los requerimientos. Además 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.

### Modelo de Plantilla de HU

Tabla 2.4: Historia de Usuarios

Historia de Usuarios	
<b>Número:</b> No. Historia de Usuario	<b>Usuario:</b> Usuario entrevistado para obtener la información
<b>Nombre:</b> Nombre para identificar la HU	
<b>Prioridad en el Negocio:</b> <b>Importancia:</b> Alta ,Media, Baja	<b>Riesgo en el Desarrollo:</b> Dificultad: Alta, Media, Baja
<b>Puntos estimados:</b> Estimada de 1 a 3 puntos	<b>Iteración Asignada:</b> Iteración a la que corresponda
<b>Programador Responsable:</b> Nombre del Programador	
<b>Descripción:</b> Breve descripción de lo que se realizará en la Historia de Usuario	
<b>Observaciones:</b> Algunas observaciones de interés	

Tabla 2.4.1: HU Gestionar Persona

Historia de Usuarios	
<b>Número: No. 1</b>	<b>Usuario:</b> Administrador
<b>Nombre:</b> Gestionar Persona	
<b>Prioridad en el Negocio:</b> Alta	<b>Riesgo en el Desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración Asignada:</b> 1
<b>Programador Responsable:</b> José Angel Mendoza	
<b>Descripción:</b> El usuario insertará los datos de la persona, una vez insertados se podrá, mostrar, modificar y eliminar.	
<b>Observaciones:</b> Confirmado con el cliente	

**Insertar Persona**

CI:   
Nombres:   
Apellidos:   
Militante:   
Sexo: ☒ Hombre ☐ Mujer  
Estado Civil:   
Dirección Particular:   
Hijos:

**Modificar Persona**

	CI	Nom...	Apelli...	Sexo	Est Civil	Dir P...	Hijos	Milita...
1	9105...	abel	nico	Masc...	Divor...	hjdjk...	0	PCC
2	9105...	Jose	Mend...	ewq...	Casado	Mayari	1	UJC
3	8807...	Luky	Luky	Masc...	Soltero	Sabala	0	No
4	8807...	Ricardo	Salazar	Hom...	Soltero	Mace...	0	PCC
5	8807...	Ricar...	Salazar	masc...	Soltero	mace...	10	No

[Para ver las Historias de Usuarios ver \[Anexo1: Historia de Usuario\]](#)

## 2.5 Estimación de Esfuerzo por HU

Para el buen desarrollo del sistema propuesto se realizó una estimación para cada HU definidas arrojando los siguientes resultados

### Estimación de Esfuerzo por HU

Tabla 2.5: Estimación de Esfuerzo por HU

Historia de Usuario	Puntos por Estimación
Gestionar Persona	0.71 Semana
Gestionar Impresora	0.71 Semana
Gestionar Maquina	0.71 Semana
Gestionar MotherBoard	0.71 Semana
Gestionar Disco Duro	0.71 Semana
Gestionar Memoria RAM	0.71 Semana
Gestionar Micro/CPU	0.71 Semana
Gestionar Lector DVD/CD	0.71 Semana
Gestionar Fuente	0.71 Semana
Gestionar Monitor	0.71 Semana
Gestionar Teclado	0.71 Semana

<b>Gestionar Mouse</b>	<b>0.71 Semana</b>
<b>Gestionar Bocinas</b>	<b>0.71 Semana</b>
<b>Gestionar UPS</b>	<b>0.71 Semana</b>
<b>Gestionar USB</b>	<b>0.71 Semana</b>
<b>Gestionar Departamento</b>	<b>0.71 Semana</b>
<b>Gestionar Usuario</b>	<b>0.71 Semana</b>
<b>Gestionar Contrato</b>	<b>0.71 Semana</b>
<b>Gestionar Contrato Entidad</b>	<b>0.71 Semana</b>
<b>Gestionar Usufructuario</b>	<b>0.71 Semana</b>
<b>Gestionar Tierra</b>	<b>0.71 Semana</b>
<b>Insertar Componentes</b>	<b>0.71 Semana</b>
<b>Insertar Periféricos</b>	<b>0.71 Semana</b>
<b>Mostrar Usufructuario con deudas</b>	<b>0.71 Semana</b>
<b>Mostrar Usufructuario sin deudas</b>	<b>0.71 Semana</b>
<b>Mostrar Cantidad Usufructuario * consejo popular</b>	<b>0.71 Semana</b>
<b>Mostrar Cantidad Usufructuario * consejo popular</b>	<b>0.71 Semana</b>
<b>Mostrar Contratos vencidos</b>	<b>0.71 Semana</b>
<b>Mostrar Contratos vigentes</b>	<b>0.71 Semana</b>
<b>Graficar PC*Dpto</b>	<b>0.42 Semana</b>
<b>Graficar Total USB * Capacidad</b>	<b>0.42 Semana</b>
<b>Graficar Contrato * tipo</b>	<b>0.42 Semana</b>

**Total de Semanas****21.85 Semanas**

### 2.5.1 Planificación de Iteraciones

A partir de las HU antes expuestas y la estimación del esfuerzo propuesto para la realización de las mismas, se realiza la planificación de la etapa de implementación del sistema, apoyándose en el tiempo se ha intentado concentrar las funcionalidades relacionadas en una 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 releasse, lo cual debe ser consistente con el contenido de una o dos iteraciones.

En relación con lo antes tratado se decide realizar al sistema 5 iteraciones, las cuáles se explican de forma detallada 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 ya que de estas depende el desarrollo de las posteriores.

Al concluir dicha iteración se contará con todas las funcionalidades descritas en las HU 1, 2, 3, 4, 5, 6, 7,8 ,9 y 10, las cuales hacen alusión a gestionar el personal , impresora, departamento, quejas, máquina y algunas piezas de esta.

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 11, 12, 13, 14, 15, 16 y 17. Las mismas brindan las funcionalidades de gestión de piezas de máquina. La versión que se obtenga de esta iteración en unión con la anterior, le facilitará al cliente una oportunidad para ver si cumple con las necesidades antes acordadas.

### Tercera Iteración

Esta iteración tiene como propósito llevar a cabo el desarrollo de la HU 18, 19, 20, 21, 22, 23 y 24. Las cuáles proporcionan las funcionalidades de gestionar los Contratos y todavía componentes de la máquina.

### Cuarta Iteración

Esta iteración tiene como objetivo llevar a cabo la realización de las HU 30, 31 y 32. Las cuáles son funciones del administrador para crear usuarios y ver las quejas o sugerencias de los mismos. Además de graficar datos específicos pedidos por el cliente.

### Quinta Iteración

Esta es la iteración que tiene la función desarrollar las HU 24, 25, 26, 27, 28 y 29. Las encargadas de mostrar los datos específicos pedidos por el cliente.

#### 2.5.2 Plan de duración de las Iteraciones

Como parte del ciclo de vida de un proyecto guiado por la metodología XP, se crea el plan de duración de iteraciones que se realizaran en la vida del proyecto. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden de implementación de las HU.

#### Plan de Duración de Iteraciones

Tabla 2.5.1: Duración de Iteraciones

Iteración	Historias de Usuarios	Duración Total
1ra Iteración	Gestionar Persona	4 Semanas
	Gestionar Impresora	
	Gestionar Departamento	
	Gestionar Maquina	
	Gestionar MotherBoard	
	Gestionar Quejas	
	Insertar Componentes	

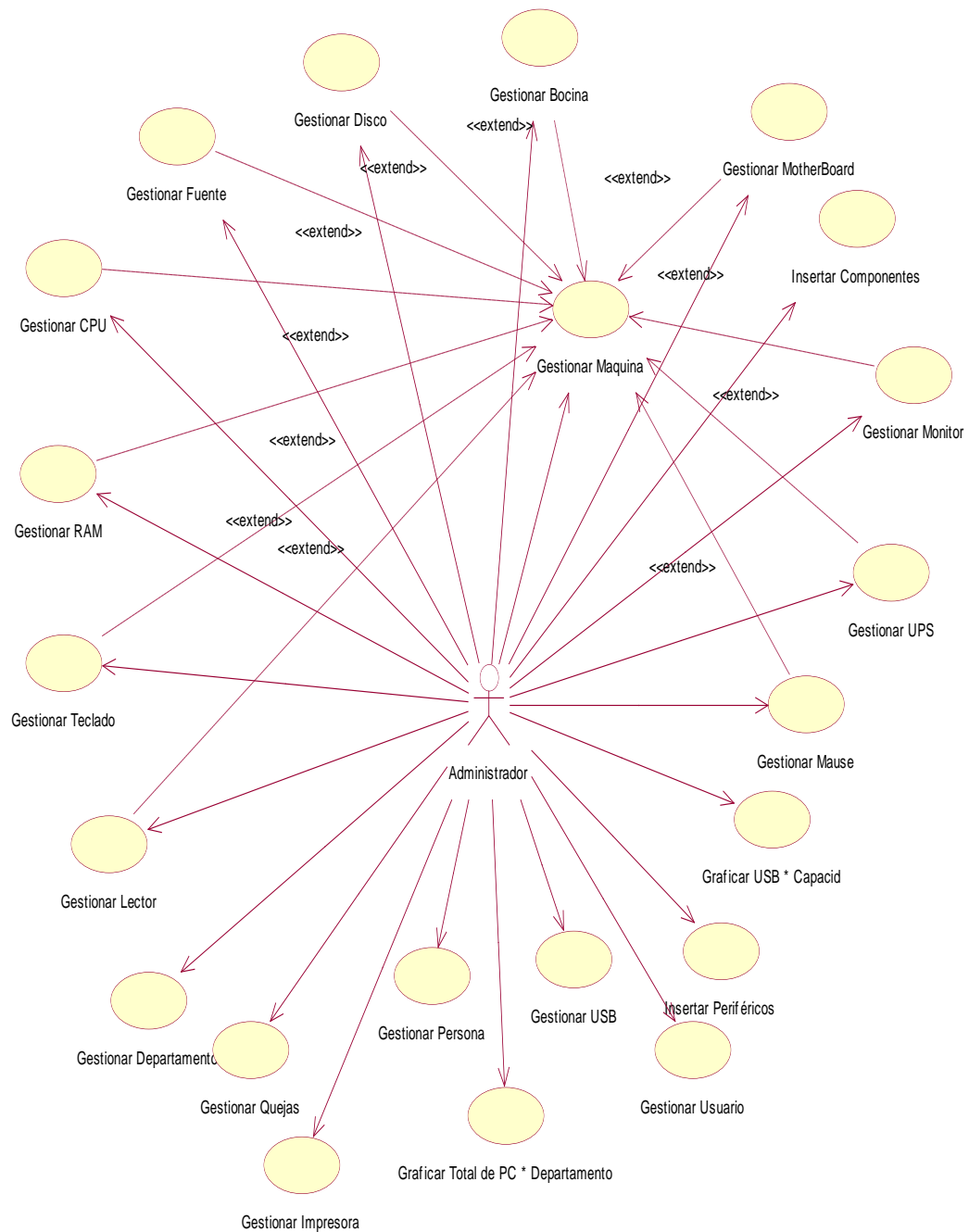
	Insertar Periféricos	
	Gestionar RAM	
	Gestionar Usuario	
2da Iteración	Gestionar Disco Duro	3 Semanas
	Gestionar Micro/CPU	
	Gestionar Lector DVD/CD	
	Gestionar Fuente	
	Gestionar Monitor	
	Gestionar Bocinas	
	Gestionar UPS	
3ra Iteración	Gestionar Mouse	5 Semanas
	Gestionar USB	
	Gestionar Contrato	
	Gestionar Contrato Entidad	
	Gestionar Usufructuario	
	Gestionar Teclado	
	Gestionar Tierra	
4 Iteración	Mostrar Contrato * Tipo	3 Semanas
	Mostrar Usufructuario con deudas	
	Mostrar Usufructuario sin deudas	
5 Iteración	Graficar Cantidad Usufructuario por consejo popular	4 Semanas
	Graficar Cantidad Usufructuario por zona	



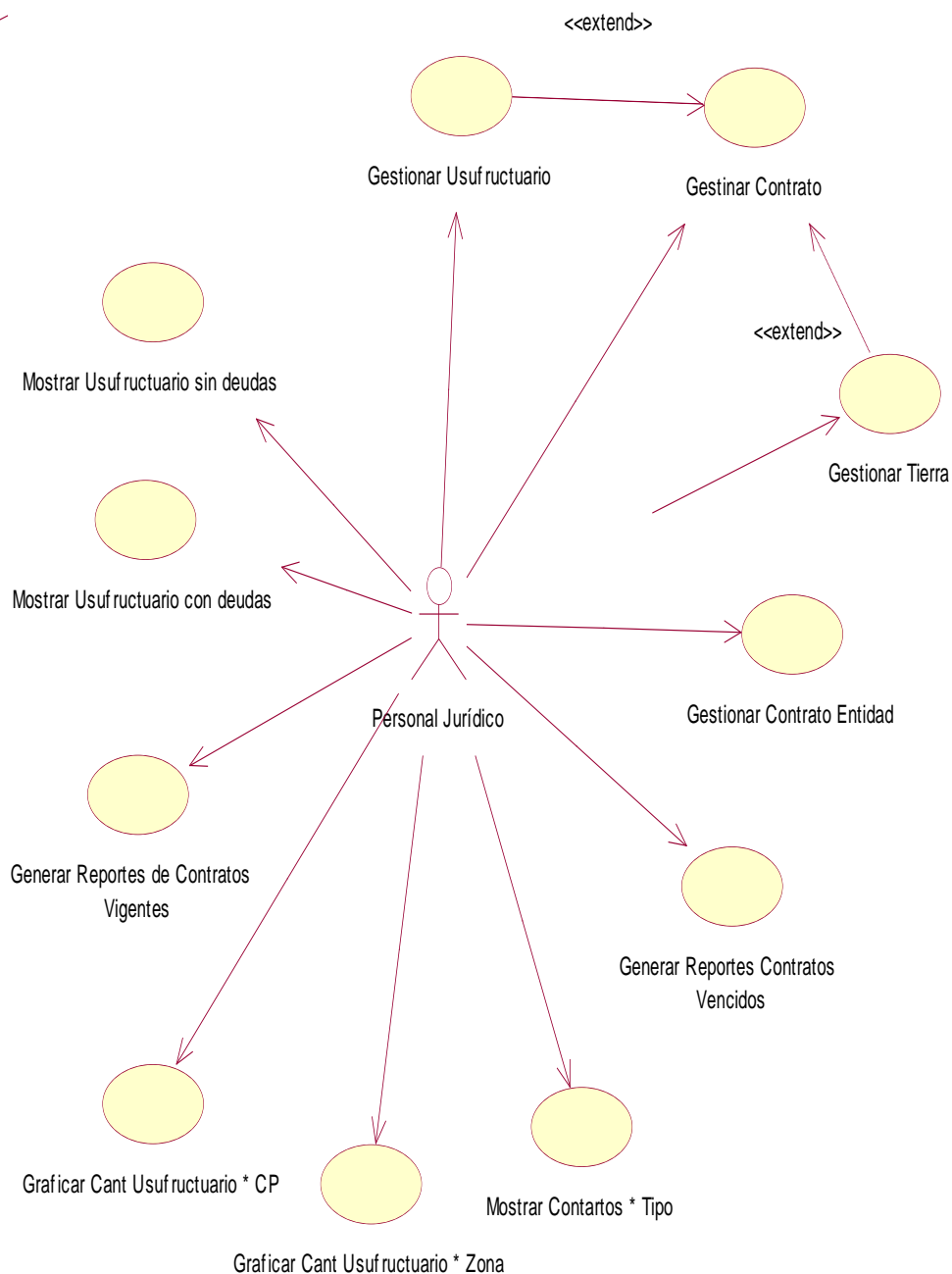
	Generar reporte de Contratos vencidos	
	Generar reporte de Contratos vigentes	
	Graficar Total PC * departamento	
	Graficar Total USB * capacidad	

[Para ver las Tarjetas de Ingeniería ver \[Anexo 4: Tarjetas de Ingeniería\]](#)

## 2.6 Diagrama de casos de uso del sistema



**Figura 2.6 Diagrama del caso de uso del sistema para actor Administrador**



**Figura 2.7 Diagrama del caso de uso del sistema para actor Jurídico**

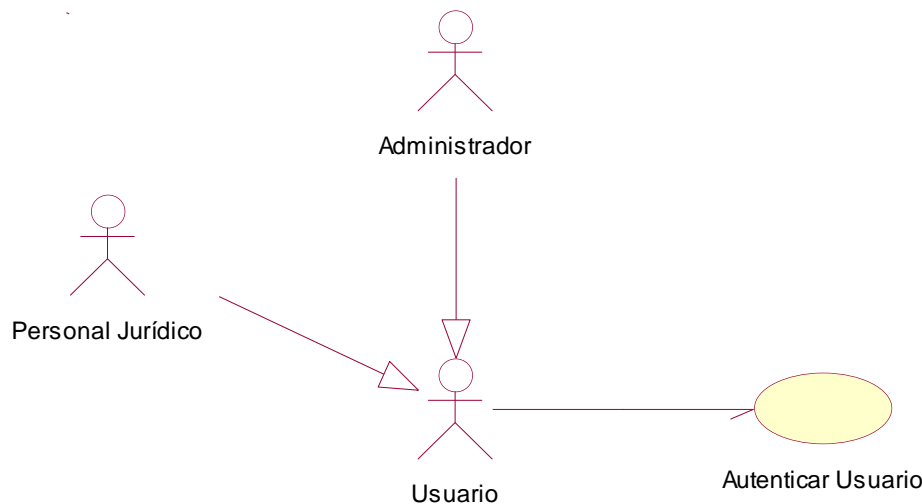


Figura 2.8 Diagrama del caso de uso del sistema para los actores

## 2.7 Definición de los requisitos no funcionales

### 2.7.1 Requisitos No Funcionales

#### Una navegación sencilla

##### Usabilidad:

.

- **Facilidad de uso por parte de los usuarios:** el sistema debe presentar una interfaz amigable que permita una interacción fácil con el mismo, para todo tipo de usuarios.
- **Especificación de la terminología utilizada:** el sistema debe adaptarse al lenguaje y técnicas usados por los clientes en la rama para una mayor comprensión.
- **Menús:** el sistema debe presentar menús laterales y en barras que permitan el acceso rápido a las opciones que el sistema brinda a los usuarios.

**Rendimiento:** Los tiempos de respuestas no son determinantes.

**Portabilidad:** Necesidad de que el sistema sea multiplataforma.

**Seguridad:**

- Identificar al usuario antes de que pueda entrar al sistema
- Garantizar que la información sea editada solo por quién tiene privilegios
- Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel del usuario activo.
- Verificaciones sobre acciones irreversibles como eliminaciones.

**Software:**

- Navegador compatible o superior con Internet Explorer 4, Mozilla Firefox 4.0
- Apache 2.0
- PHP 5.0
- MySQL 5.2
- Ext-JS 3.2.0

**Requisitos para la documentación de usuarios en línea y ayuda del sistema:**

- Manual de usuario: el sistema debe presentar un manual de usuario
- Mediante el expediente del proyecto se garantiza la documentación actualizada del grupo de desarrollo: se precisa que la documentación del sistema este actualizada en todos los aspectos, fases de trabajo y ciclos de vida del proyecto propiciando así un respaldo legal.

**Interfaces de Comunicación:**

La comunicación entre cliente servidor se realizará mediante el protocolo de seguridad HTTP.

### **2.8 Conclusiones**

En este capítulo se fundamentaron los aspectos referentes a la fase de planeación y diseño, teniendo en cuenta la participación y peticiones del cliente, se elaboraron las HU y se estimó el esfuerzo necesario para la realización de cada una de ellas. Se conformó un plan de lanzamiento de las iteraciones teniendo en cuenta que para pasar de una iteración a otra debe haberse cumplido la anterior. Dándose por terminada esta fase de la metodología y con la base para pasar a la siguiente etapa de desarrollo.

### Capítulo 3 Descripción de la solución propuesta

#### 3.1 Introducción

En el presente capítulo se realizan las etapas diseño e implementación de la solución propuesta para el problema planteado, haciendo uso de la metodología XP antes mencionada. Para ello se define la arquitectura de la herramienta, así como su proceso de funcionamiento en virtud de cumplir con los requisitos funcionales de la misma. Se representa la modelación de los diagramas fundamentales. Una vez completada la codificación del software se procede a aplicar las pruebas, para examinar la estructura externa y garantizar la calidad del mismo.

#### 3.2 Diagrama de clases del diseño

##### Modelo de Diseño

El modelo de diseño es un modelo de objetos que describe la realización de los CU, y sirve como una abstracción del modelo de implementación y el código fuente. Es usado como una entrada inicial en las actividades de implementación y prueba.

**Contiene:** protocolo, capsula, realización de CU, señales, eventos, subsistema de diseño, paquetes de diseño, interfaces, clases del diseño, clases de prueba, diseño de pruebas.

**Propósito:** el modelo de diseño es una abstracción de la implementación del sistema. Es usado para concebir un documento del diseño del sistema de SW. Es abarcador, compuesto por artefactos que engloban todas las clases del diseño, subsistemas, paquetes, colaboraciones, y las relaciones entre ellos.

## 3.2.1 Gestionar Contrato

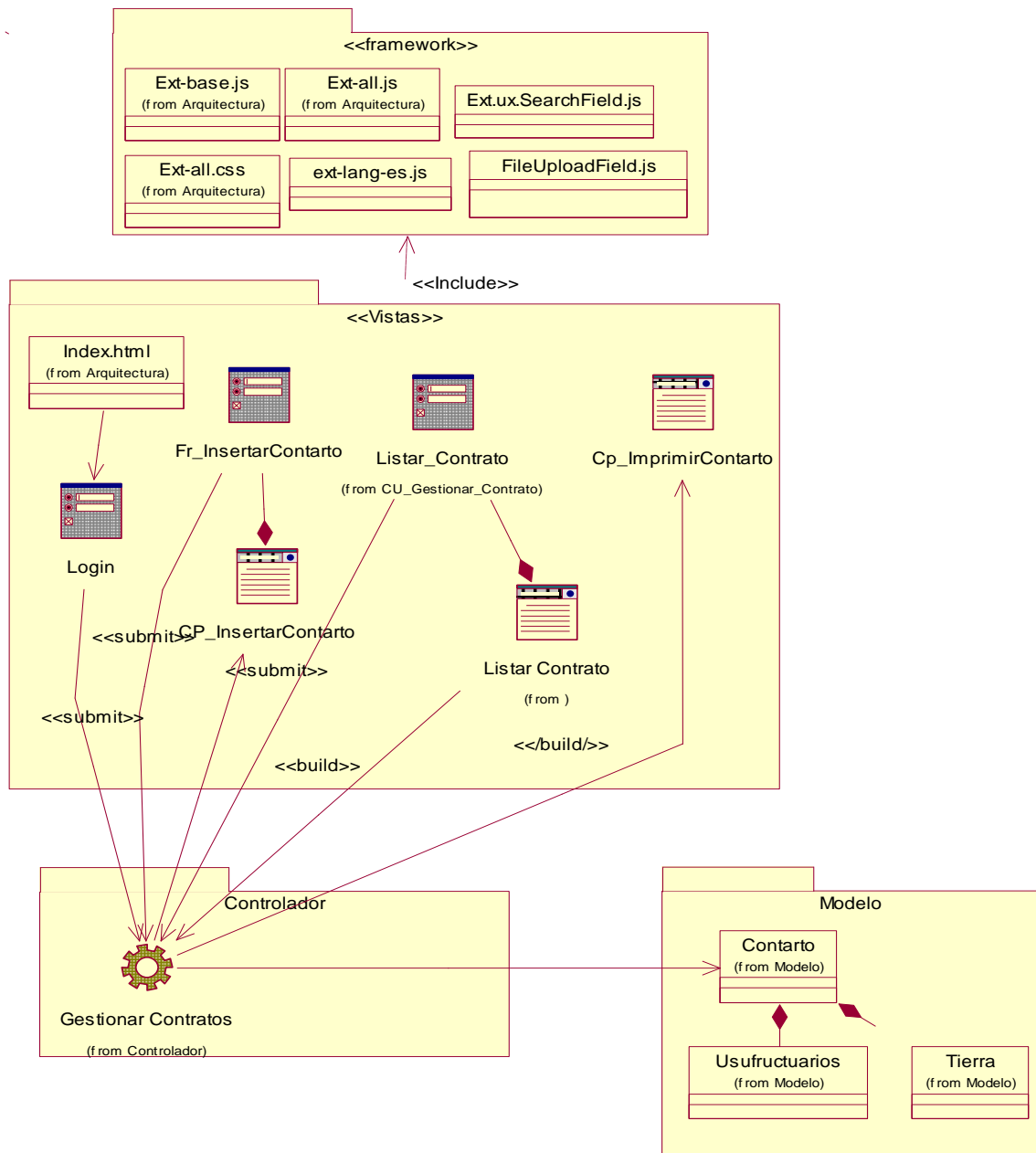


Figura 3.2.1 Diagrama de clase de Diseño Gestionar Contrato



## 3.2.2 Gestionar Máquina

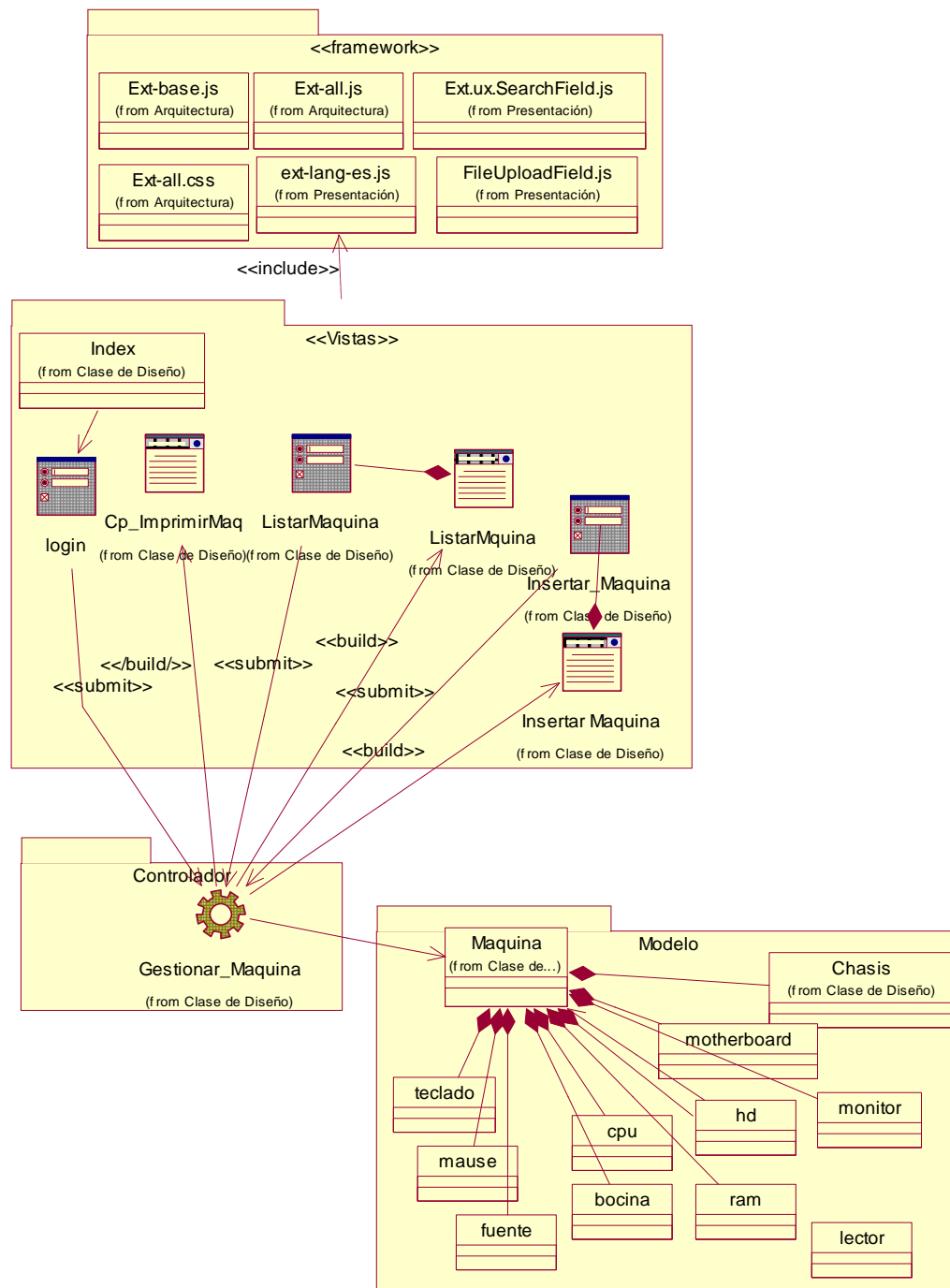
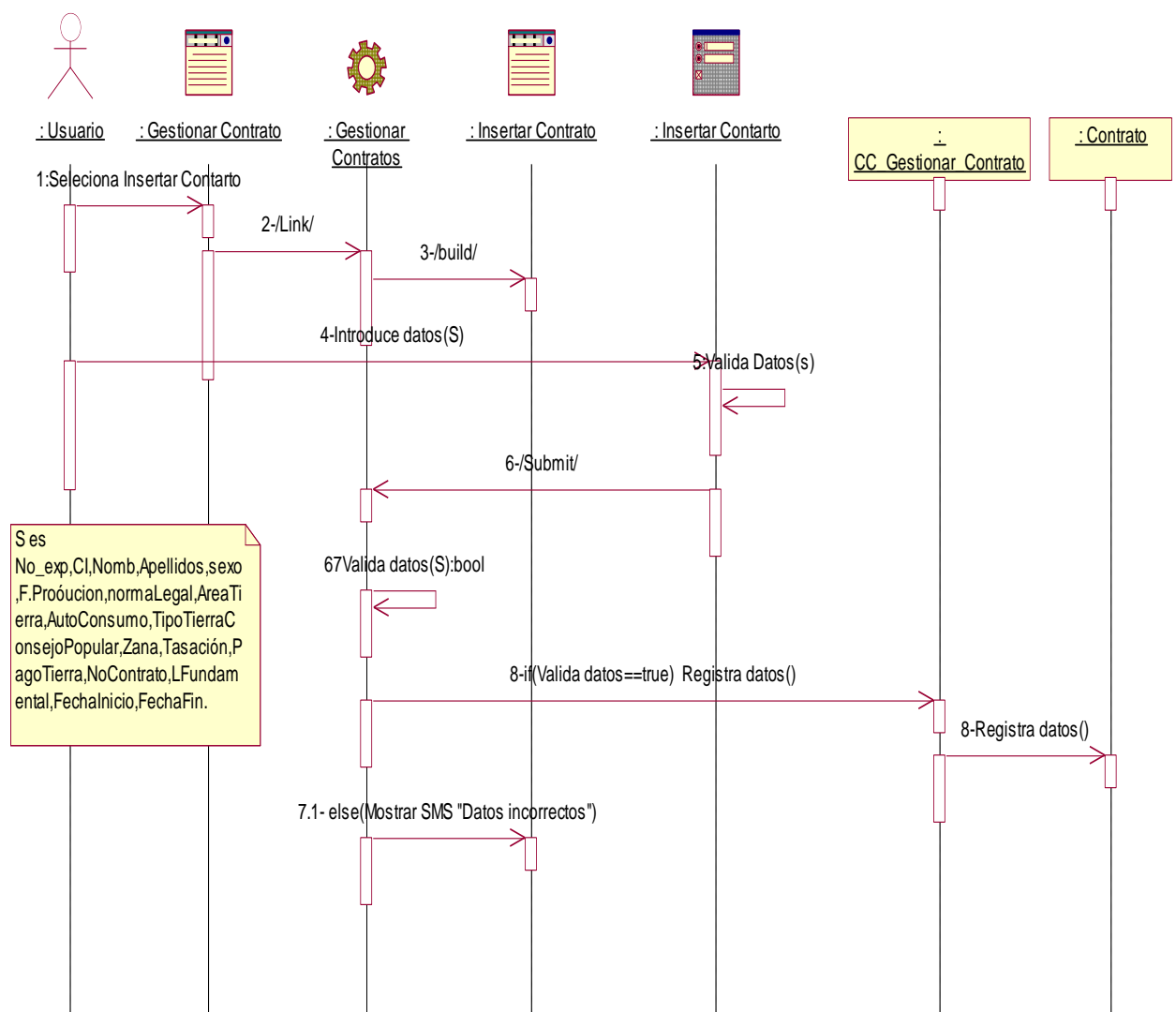


Figura 3.2.2 Diagrama de clase de Diseño Gestionar Maquina

## 3.3 Diagramas de secuencias

Un diagrama de secuencia destaca la ordenación temporal de los mensajes. Los diagramas que a continuación se presentan muestran la interacción entre objetos de los casos de uso gestión de los contratos y control de los medios informáticos respectivamente mediante el envío y recepción mensajes en orden de su sucesión, ofreciendo una señal clara del flujo de control a lo largo del tiempo.



**Figura 3.3.1 Diagrama de Secuencia Insertar Contrato**

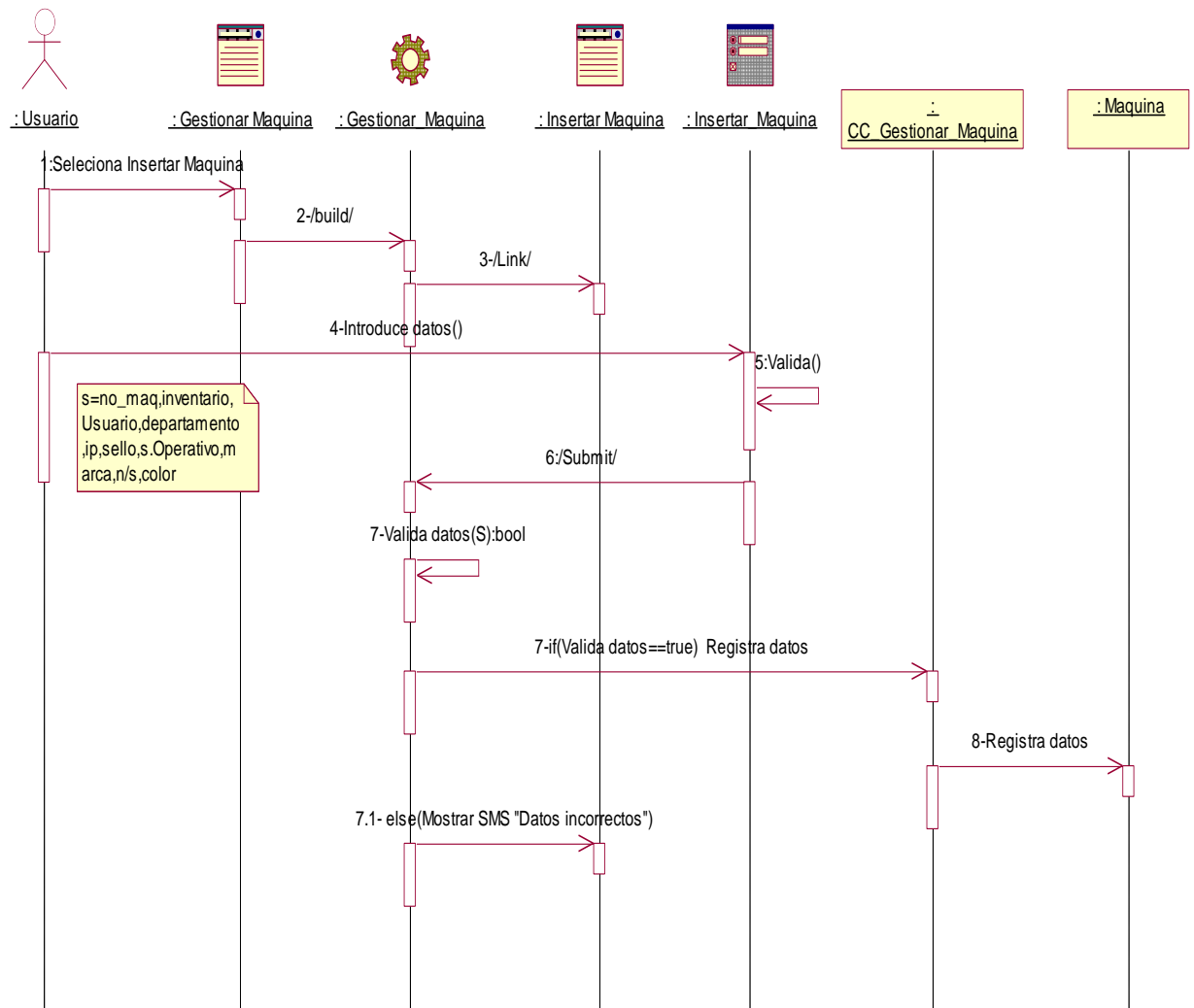
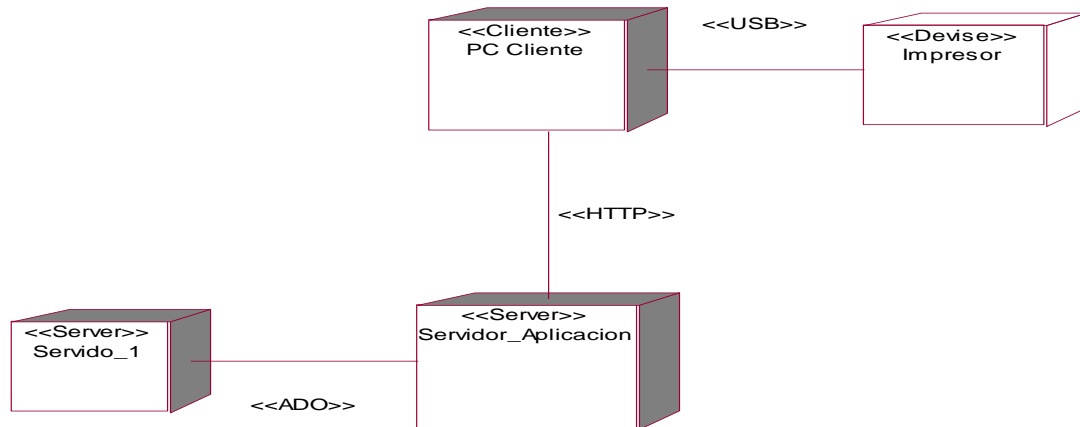


Figura 3.3.2 Diagrama de Secuencia Insertar Maquina

## 3.4 Diagrama de despliegue

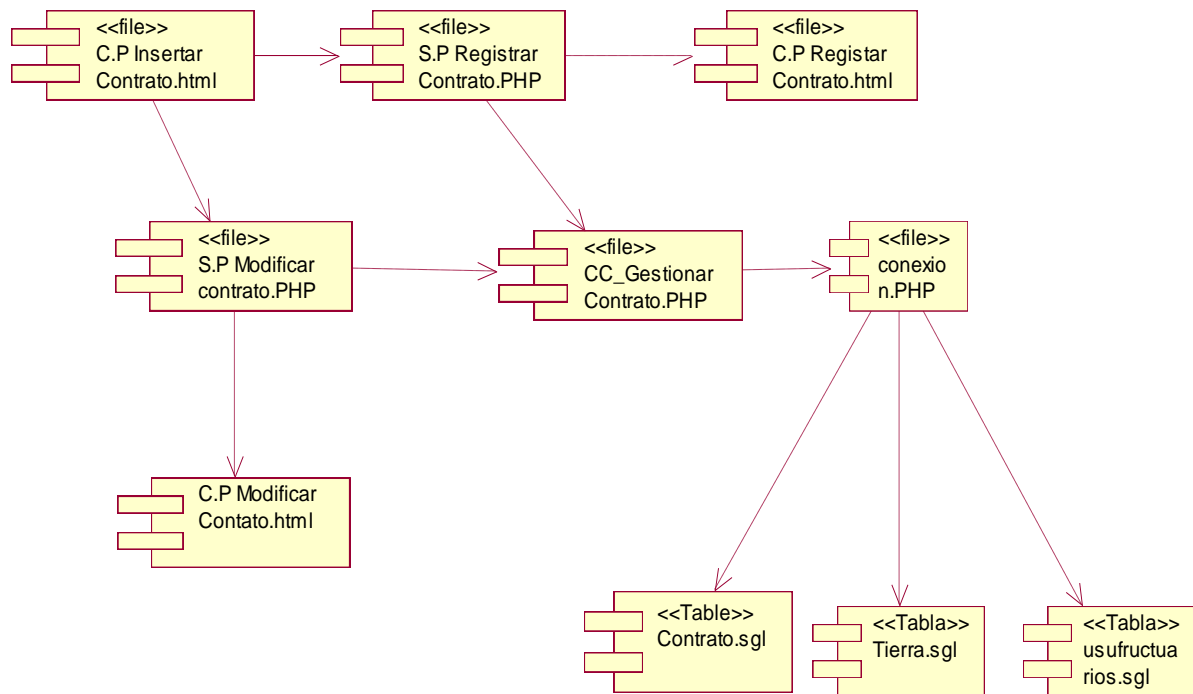
El Modelo de Despliegue se utiliza para capturar los elementos de configuración del procesamiento y las conexiones entre esos elementos. También se utiliza para visualizar la distribución de los componentes de la aplicación en los nodos físicos.



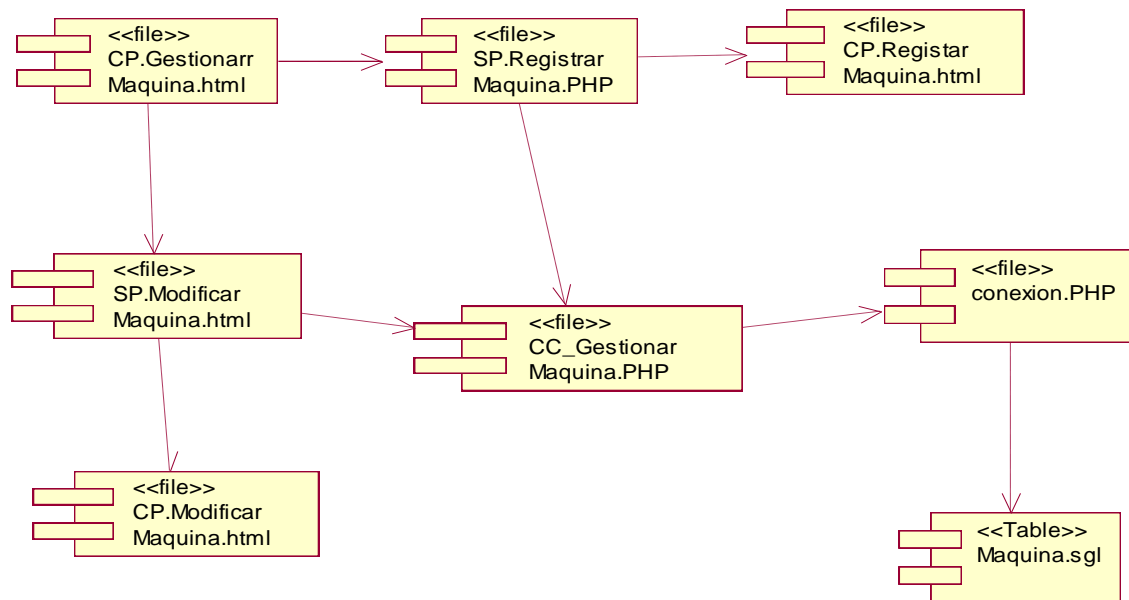
**Figura 3.4.1 Diagrama de Despliegue**

## 3.5 Diagramas de componentes

El diagrama de componentes el empaquetamiento físico de los elementos del modelo de diseño.



**Figura 3.5.1 Diagrama de Componente Gestionar Contrato**



**Figura 3.5.2 Diagrama de Componente Gestionar Maquina**

## 3.6 Implementación de los casos de usos críticos

### 3.6.1 Codificación

La codificación de componentes es una actividad crítica desarrollada durante la disciplina de Implementación con el propósito de completar una parte del sistema. En esta actividad el programador escribe código fuente, reutiliza código, compila e implementa los elementos del modelo de diseño.

A continuación se muestra un fragmento de código del caso de uso Mostrar listado de contratos.

```
var g_ManteArchivo = new Ext.grid.EditorGridPanel({
    stripeRows: true,
    enableColLock: true,
    loadMask: true,
    store: store,
    columns: [
        new Ext.grid.RowNumberer(),

        { header: "Numero Expediente",
          dataIndex: 'no_exp', // nombre del campo de la base de datos
          editor: nf_chapa, // llama al componente tipo TextField Creado para que podamos editar el campo
          sortable: true, // Ordenable
          width: 100
        },
        { header: "Linea Fundada",
          dataIndex: 'linea_fund', // nombre del campo de la base de datos
          editor: lin, // llama al componente tipo TextField Creado para que podamos editar el campo
          sortable: true, // Ordenable
          width: 90,
        },
        { header: "Fecha Inicio",
          dataIndex: 'fecha_inicio_contrato', // nombre del campo de la base de datos
          editor: tf_fecha, // llama al componente tipo TextField Creado para que podamos editar el campo
          //vtype: 'daterange',
          sortable: true, // Ordenable
          width: 90
        },
        { header: "Fecha Fin",
          dataIndex: 'fecha_fin_contrato', // nombre del campo de la base de datos
          editor: tf_fecha1, // llama al componente tipo TextField Creado para que podamos editar el campo
          //vtype: 'daterange',
          sortable: true, // Ordenable
          width: 100
        },
    ],
});
```

**Figura 3.6.1: Código del caso de uso Mostrar listado de contratos**

## Patrones de eventos con ExtJS

En la programación con JavaScript y ExtJS se aplica el paradigma más común para programar Interfaces de usuarios, la Programación Orientada a Eventos. La programación dirigida por eventos es un paradigma de programación en el que tanto la estructura como la ejecución de los programas van determinados por los sucesos que ocurran en el sistema, definidos por el usuario o que ellos mismos provoquen. **(25)**

Para entender la programación dirigida por eventos, podemos oponerla a lo que no es: mientras en la programación secuencial (o estructurada) es el programador el que define cuál va a ser el flujo del programa, en la programación dirigida por

eventos será el propio usuario, o lo que sea que esté accionando el programa. Aunque en la programación secuencial puede haber intervención de un agente externo al programa, estas intervenciones ocurrirán cuando el programador lo haya determinado, y no en cualquier momento como puede ser en el caso de la programación dirigida por eventos.

El creador de un programa dirigido por eventos debe definir los eventos que manejarán su programa y las acciones que se realizarán al producirse cada uno de ellos, lo que se conoce como el administrador de evento. Los eventos soportados estarán determinados por el lenguaje de programación utilizado, por el sistema operativo e incluso por eventos creados por el mismo programador.

En la programación dirigida por eventos, al comenzar la ejecución del programa se llevarán a cabo las inicializaciones, que en ExtJS podemos conocerlos como controladores (o “listeners”) y demás código inicial y a continuación la aplicación quedará esperando hasta que se produzca algún evento. Cuando alguno de los eventos esperados por el sitio tenga lugar, el programa pasará a ejecutar el código del correspondiente administrador de evento, todos controlados por el motor de Observable de ExtJS. Por ejemplo, si el evento consiste en que el usuario ha hecho clic en el botón Guardar de un formulario, se ejecutará el código del administrador de evento, que será el que haga que la película se muestre por pantalla.

En ExtJS, la programación dirigida por eventos es la base de lo que llamamos interfaz de usuario, aunque puede emplearse también para desarrollar interfaces entre componentes de Software o módulos del núcleo.

### **Detección de eventos**

En contraposición al modelo clásico, la programación orientada a eventos permite interactuar con el usuario en cualquier momento de la ejecución de la aplicación. Esto se consigue debido a que los programas creados bajo esta arquitectura se componen por un bucle exterior permanente encargado de recoger los eventos, y distintos procesos que se encargan de tratarlos. Habitualmente, este “Observador”

permanece oculto al programador que simplemente se encarga de tratar los eventos mediante los controladores.

A modo de conclusión los eventos son mensajes, el mensaje para ejecutar una función, generado por alguna parte del programa en donde intervino un usuario que lo provocó, a esto se le llama programación Orientada a Eventos. Un evento notifica a otra parte del programa, por medio de un controlador, en ExtJS llamado “listener”, a que haga algo. Los eventos son generados como respuestas a acciones del usuario, aunque no exclusivamente, también pueden ser por cambios de la misma aplicación. Cabe la pena concluir que la fuente de eventos es completamente independiente al motor que controla esos posibles eventos, lo que hace suponer que los eventos ocurren a pesar de que nadie los este “escuchando”

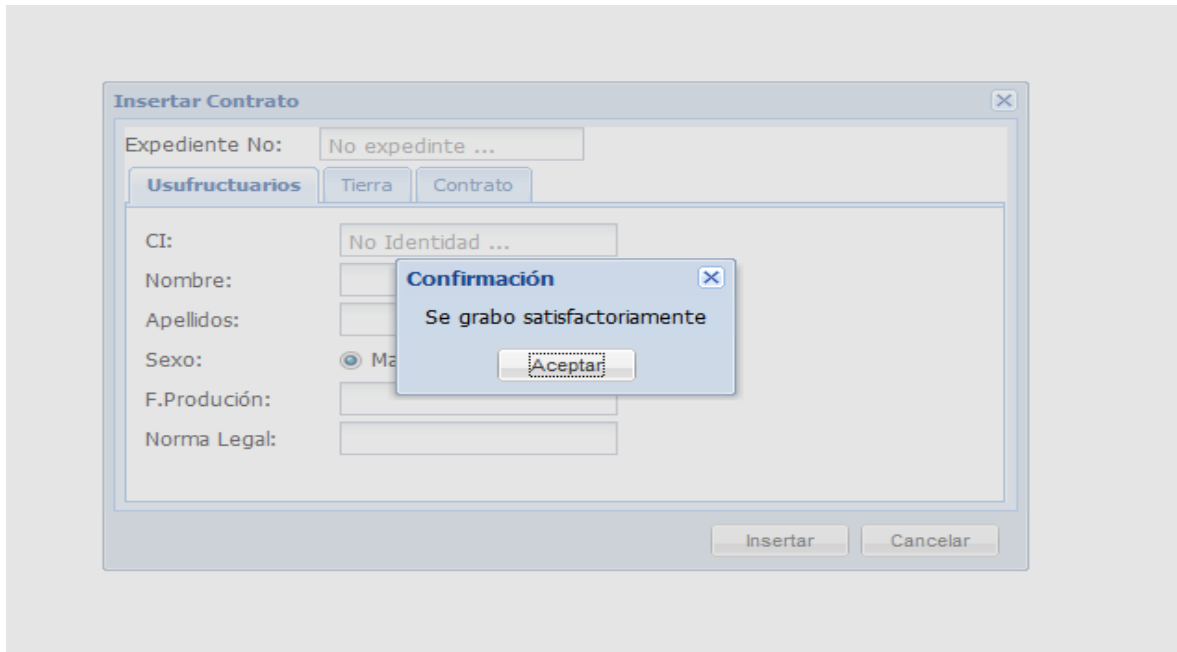
```

buttons:[ {
  xtype: 'button',
  text: 'Insertar',
  handler: function() {
    if (Ext.getCmp('formulario').getForm().isValid()) {
      Ext.getCmp('formulario').getForm().submit({
        method: 'POST',
        params: { task: 'GRABAR' },
        waitTitle: 'Validando datos',
        waitMsg: 'Enviando datos..',
        success: function(form, action) {
          var data = Ext.util.JSON.decode(action.response.responseText);
          Ext.Msg.alert('Confirmación', data.message.reason);
          Ext.getCmp('formulario').getForm().reset();
        },
        failure: function(form, action) {
          var data = Ext.util.JSON.decode(action.response.responseText);
          Ext.Msg.show({
            title: 'Error',
            msg: data.errors.reason,
            buttons: Ext.Msg.OK,
            icon: Ext.MessageBox.ERROR
          });
        }
      });
    }
  }
},
],

```

**Figura 3.6.2: código del caso de uso Mostrar listado de contratos**





**Figura 3.6.3: Vistas de la acción insertar Contrato**

## 3.7 Modelo de Datos

Se presentan las tablas con que interactúa la aplicación propuesta en la investigación, se recogen y modelan todos los datos que dispone el sistema.

Como resultado del modelo de datos se obtuvieron diecinueve tablas, relacionadas entre sí, que responden a las solicitudes de recuperación y almacenamiento de datos necesarios para el funcionamiento del software.

El modelo consta de

Las mismas cumplen con las restricciones de la segunda forma normal (2NF) y ningún atributo no-primario de las tablas es dependiente de una clave primaria, por lo que queda demostrado que está normalizada en la tercera forma normal (3NF).

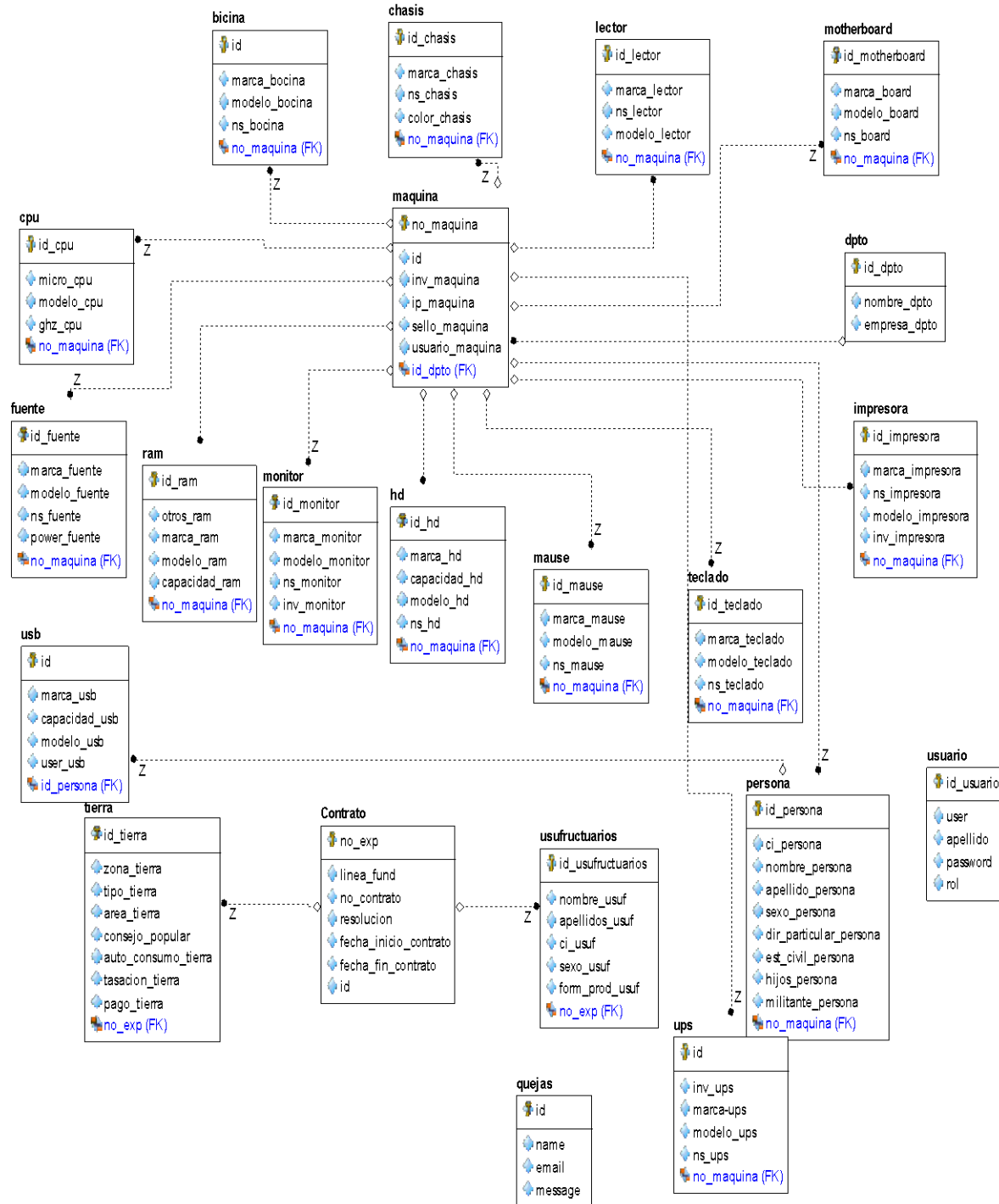


Figura 3.7: Modelo Físico de Datos

## 3.8 Prueba

Una de las características de la metodología XP es el proceso de pruebas. Al realizar pruebas al software tanto como sea posible se aumenta la calidad de los sistemas, reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección.

### 3.8.1 Pruebas de Aceptación

XP propone la realización de pruebas unitarias, encargadas de verificar el código y diseñadas por los programadores. Las pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida por el cliente.

#### Prueba de Aceptación a la HU Gestionar Personas

Tabla 3.8.1: Prueba de Aceptación a la HU Gestionar Personas

Prueba de Aceptación
HU: 1
Nombre: Gestionar Personas
Descripción: La especialista debe suministrarle los datos al sistema de una persona para luego mostrarlo, modificarlo y eliminarlo.
Condiciones de ejecución: Se deben introducir los datos
Entrada/Pasos ejecución: <ul style="list-style-type: none"> <li>• Insertar Persona</li> <li>• Mostrar Persona</li> <li>• Modificar Persona</li> <li>• Eliminar Persona</li> </ul>
Resultado: Gestiona correctamente la persona
Evolución de la prueba: Aceptada

[Para ver las Pruebas de Aceptación \[Anexo 5: Pruebas de Aceptación\]](#)

### *3.9 Conclusiones*

En este capítulo se realizó la fase de Desarrollo y Pruebas, se ejecutaron las iteraciones a partir de la distribución de tareas de las HU. Se realizó la prueba de aceptación a las HU según sus funcionalidades a cumplir. Cumpliendo así satisfactoriamente con todos los requisitos exigidos por parte del cliente.

Se presentó además el Modelo de Datos de la aplicación obtenida, logrando una visión detallada de sus atributos y las relaciones entre sus clases. Demostrando su normalización. Además se realizó los diagramas de clases de diseño, secuencia, y componente de los caso de uso más crítico igualmente el esquema despliegue.

### Capítulo 4 Estudio de factibilidad

#### *4.1 Introducción*

En la actualidad es imprescindible que a la hora de desarrollar un proyecto, se realice como es debido el estudio de su factibilidad. En general ningún producto informático está libre de posibles riesgos en la concepción del proyecto. Por tanto es necesario minimizar recursos humanos, materiales y financieros; de ahí que es de vital importancia estimar la relación costo–beneficio, así como el esfuerzo, capital humano y el tiempo de desarrollo que se emplea en la ejecución de estos. Este análisis permitió determinar las posibilidades de diseñar el Sistema propuesto y su puesta en marcha, los aspectos fueron clasificados en tres áreas, las cuáles se describen a continuación:

- Factibilidad Técnica
- Factibilidad Económica
- Factibilidad Operativa

#### *4.2 Estudio de Factibilidad*

##### **4.2.1 Factibilidad Técnica**

La Factibilidad Técnica consistió en realizar una evaluación de la tecnología existente en la empresa, este estudio destinado a recolectar información sobre los componentes técnicos que posee la organización y la posibilidad de hacer uso de los mismos para desarrollar y poner en marcha el Sistema.

De acuerdo a la tecnología necesaria para la implantación del Sistema Informático para la Gestión de los Contratos y el Control de los medios Informáticos en la Empresa Agropecuaria de Sagua de Tánamo se evaluó bajo dos enfoques: Software y Hardware.

##### **Hardware**

La siguiente tabla muestra los requerimientos mínimos que debe cumplir el servidor donde va a radicar el Sistema.

**Tabla 4.2: Descripción de Requerimientos**

Requerimientos Mínimos	Disponibles
<b>Procesador (Celeron) 1.6 MHz</b>	Procesador (i3)Core 3.1 MHz
<b>256 Mb de Memoria RAM</b>	4 GB de Memoria RAM
<b>Disco Duro 20 GB</b>	Disco Duro 500 GB
<b>Tarjeta de Red</b>	Tarjeta de Red
<b>Monitor, Teclado, Mouse</b>	Monitor, teclado, Mouse
<b>Unidad de Protección UPS</b>	Unidad de Protección UPS

Evaluando el Hardware existente y tomando en cuenta la configuración mínima necesaria, la institución no requiere de inversión inicial para adquirir nuevos equipos, ni tampoco para actualizar los equipos existentes. Los mismos satisfacen los requerimientos establecidos tanto para el desarrollo y puesta en marcha del Sistema propuesto.

### **Software**

En cuanto al software, la entidad cuenta con todas las aplicaciones que se emplearon para el desarrollo del proyecto y puesta en funcionamiento del Sistema. Por tal motivo no requiere inversión para la adquisición de los mismos.

Las estaciones de trabajo operan bajo ambiente Windows, para el uso general de las estaciones en actividades diversas se debe poseer las herramientas de escritorio y navegadores que existen actualmente en el mercado.

Como resultado del análisis técnico se determinó que en los momentos actuales, la empresa cuenta con la infraestructura tecnológica (Hardware y Software) adecuados para la implantación y puesta en marcha del Sistema propuesto.

### **4.2.2 Factibilidad Económica**

Se presenta un estudio que arrojó los resultados de factibilidad económica del desarrollo del sistema de Gestión de los Contrato y el control de los medios Informáticos.

Se determinaron los recursos para desarrollar, implantar y mantener en operación el Sistema; haciendo una evaluación, donde se puso de manifiesto el equilibrio existente entre los costos del Sistema y los beneficios que se liberaron de este.

Lo que permite observar de una manera más precisa y rápida los beneficios del proyecto.

### 4.2.3 Efectos Económicos

Los efectos Económicos se dividen en tres grupos

- Efectos Directos
- Efectos Indirectos
- Efectos Externos

#### Efectos directos

##### Positivos

El personal relacionado con el software:

El administrador podrá llevar a cabo un mejor control del inventario de los equipos informáticos, hará la recopilación de los datos con un mínimo margen de error. Tendrá la oportunidad de generar reportes con mayor rapidez haciendo huso del sistema el cual le facilitará y dará mayor calidad a su trabajo.

El departamento jurídico con esta aplicación eficaz y eficiente de los procesos la gestión de los contratos en la empresa, podrá obtener información actualizada, antigua, resumida y detallada. Por lo que mejorara la calidad del trabajo de los mismos.

##### Negativos

El navegador compatible o superior con Internet Explorer 4 y Mozilla Firefox 4. Se recomienda estos navegadores dado que versiones inferiores no cuentan con los estándares de diseño del proyecto.

#### Efectos Indirectos

Debido a que el software no está construido con finalidad de venta sus efectos económicos indirectos no son de un valor significativo en el mercado.

### Efectos Externos

El software desarrollado tendrá efectos sobre otras áreas de la entidad ya que la información con que se trabaja agilizará el trabajo de los especialistas.

#### 4.2.4 Fichas de Costo

Para determinar el costo económico del proyecto se elaboran las fichas de costo correspondientes a la moneda libremente convertible y nacional.

#### Costo en CUC

Tabla 4.2.2: Ficha de Costo en CUC

Ficha de Costo	
Precios	
<b>Costos de MonedasCUC</b>	
<b>Costos Directos</b>	
Compra de Equipos de cómputo	00.0
Alquiler de equipos de cómputo	00.0
Compra de licencia de Software	00.0
Materiales directos	00.0
Subtotal	00.0
<b>Costos Indirectos</b>	
Formación del personal que elabora el proyecto	00.0
Gastos en llamadas telefónicas	00.0
Gastos para el mantenimiento del centro	00.0
Know How	00.0



Gasto de consumo de energía eléctrica	15.52
Gastos en representación	00.0
Subtotal	15.52
Gastos en Distribución y Venta	00.0
Participación en ferias o exposiciones	00.0
Gastos en Transporte	00.0
Compra de materiales de propaganda	00.0
Subtotal	15.52
Total	15.52

## Costo en MN

Tabla 4.2.3: Ficha de Costo en MN

Ficha de Costo	
Precios	
Costos de Monedas MN	
Costos Directos	
Salario del personal que laborará en el proyecto	600.0
Depreciación de equipos	85.68
Seguridad Social	00.0
Vacaciones $100 \times 9.09\% = 9.09$	00.0

<b>Gastos en llamadas telefónicas</b>	00.0
<b>Impuesto por fuerza de trabajo</b>	685.68
<b>Subtotal</b>	
<b>Costos Indirectos</b>	
<b>Know How</b>	00.0
<b>Subtotal</b>	685.68
<b>Total</b>	685.68

El análisis de costo-beneficio se basa en un principio muy simple:

Compara los beneficios y los costos de un proyecto particular y si los primeros exceden a los segundos entrega un elemento de juicio inicial, que indica su aceptabilidad. Mientras que el análisis costo-efectividad sigue la misma lógica, compara los costos con las potencialidades de alcanzar más eficientemente los objetivos no expresables en moneda; si no en productos. Para esta técnica es imprescindible definir una variable directa que haga variar los costos.

Teniendo en cuenta que el costo para este proyecto es despreciable, tomaremos como costo, el tiempo en minutos empleado para realizar las actividades de gestión de los contratos y la de control de los medios informáticos en la empresa y la variable seria, complejidad de las pruebas que se desarrollan durante este proceso.

#### **Valores de la variable (Forma Anterior)**

##### **1. Control de inventario de los quipos de cómputo**

- Toma de los datos de los equipos de cómputo (15 min)
- Obtener información de la cantidad de inventario (35 min)
- Realizar el control de inventarios de los componentes y periféricos(115 min)
- Conocer la cantidad de máquinas por departamento (20 min)
- Saber la cantidad de USB por capacidad que existe en la empresa

( 35 min)

### 2. Gestión de los Contrato

- Toma de datos del contrato (35 min)
- Información de los usufructuario con deudas(60 min)
- Información de los usufructuario sin deudas(60 min)
- Información de los contratos vigentes(45 min)
- Información de los contratos vencidos(45 min)
- Informar Cantidad de usufructuario por zona(30 min)
- Informar Cantidad de usufructuario por CP(45 min)
- Información de la Cantidad de contratos(60 min)

### **Valores de la variable (Forma Propuesta):**

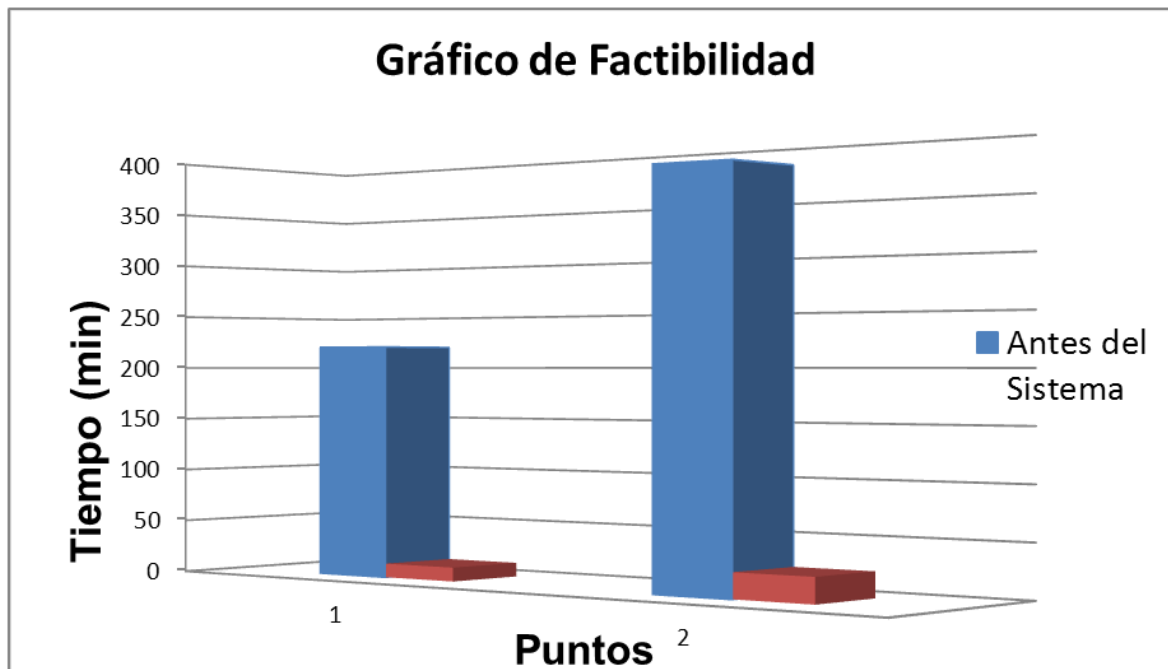
### 3. Control de inventario de los quipos de cómputo

- Registrar maquina (10 min)
- Información de la cantidad de inventario (0.5 min)
- Control de inventarios de los componentes y periféricos(1.5 min)
- Ver Cantidad de máquinas por departamento (0.5 min)
- Saber la cantidad de USB por capacidad que existe en la empresa ( 0.5 min)

### 4. Gestión de los Contrato

- Registrar contrato (20 min)
- Información de los usufructuario con deudas(0.5 min)
- Información de los usufructuario sin deudas(0.5 min)
- Información de los contratos vigentes(0.5 min)
- Información de los contratos vencidos(0.5 min)
- Cantidad de usufructuario por zona(0.5 min)
- Cantidad de usufructuario por CP(0.5 min)

El gráfico que se presenta a continuación muestra el comportamiento de estas variables, teniendo en cuenta las formas de realización de las actividades que componen el proceso.



**Figura 4.2: Factibilidad del Software**

Como se observa en el gráfico anterior el Sistema propuesto muestra una mayor eficiencia y adaptabilidad a las nuevas circunstancias descritas anteriormente, motivo por el cual queda demostrado lo factible de la aplicación desarrollada.

## 4.2.5 Factibilidad Operativa

La Factibilidad Operativa permite predecir si se pone en marcha el Sistema propuesto. Aprovechando los beneficios que ofrece a todos los usuarios involucrados con el mismo; ya sea los que interactúan de forma directa como los que se benefician indirectamente.

Por otro lado el correcto funcionamiento del software en cuestión, siempre estará supeditado a la capacidad de los trabajadores encargados de dicha tarea.

La necesidad y deseo de un cambio en el sistema actual, expresada por los trabajadores, condujo a la aceptación del proyecto, que de una manera más rápida y amigable cubra todos sus requerimientos y expectativas.

Basándose en las entrevistas y conversaciones sostenidas con el personal involucrado se demostró que estos no representan ninguna oposición al cambio, por lo que el software es Factible Operacionalmente.

Con la finalidad de garantizar el buen funcionamiento del sistema y que este impactará de forma positiva a los usuarios, el mismo fue desarrollado en forma estándar a los sistemas existentes en la institución. Presentando una interfaz amigable al usuario, lo que se traduce en una herramienta de fácil manejo y comprensión.

### ***4.3 Conclusiones del Capítulo***

En este capítulo se realizó el estudio del costo real en que se incurrió durante el diseño e implementación del Sistema propuesto, mediante la metodología Costo Efectividad (Beneficios). Se analizaron todos los factores directos, indirectos, externos e intangibles. Así como se calculó el costo de ejecución del producto final mediante la Ficha de Costo, demostrándose así la conveniencia del software.

## Conclusiones Generales

El Sistema informático se desarrolló siguiendo la metodología XP. El software está provisto de un ambiente cómodo, fácil de entender, cumple con los estándares de diseño y utiliza técnicas modernas de programación.

Para ello se siguieron los siguientes pasos:

- Se elaboró el marco teórico metodológico que fundamenta la investigación, permitiendo el análisis del Proceso de Gestión de los contratos y control de los medios informáticos en la empresa Agropecuaria Sagua de Tánamo.
- Se analizaron los sistemas informáticos existentes relacionados con el objeto de estudio.
- Se realizó un estudio y selección de las tecnologías y herramientas a utilizar, para el diseño e implementación de la aplicación.
- Se desarrolló el producto final, que consiste en el Diseño e implementación de una aplicación Web para la Gestión de los contratos y el control de los medios informáticos en la empresa Agropecuaria Sagua de Tánamo.
- Se efectuó un estudio de la factibilidad técnica, económica, y operativa del software en cuestión; arrojando resultados satisfactorios para el proyecto.

Por lo antes expuesto se concluye que los tareas propuestas fueron cumplidas satisfactoriamente.

## Recomendaciones

Se recomienda a las entidades desarrolladoras de software la reutilización de esta aplicación web como vía de:

- ❖ Contribuir con el ahorro de recursos que necesita el país.
- ❖ Disminuir el tiempo de respuesta al usuario.
- ❖ Cumplir con el paradigma de independencia tecnológica.
- ❖ Facilitar el proceso de integración entre las aplicaciones que se desarrollen.
- ❖ Crear una comunidad que participe en el desarrollo de tecnologías propias.

## Referencias Bibliográficas

1. **Sánchez, Iliana Díaz y Rodríguez, MarcosA. Martínez.** *Herramienta para el control de los medios informáticos de la Empresa* . Moa : s.n., 2008.
2. Free Vista Files. *Free Vista Files*. [En línea] 23 de febrero de 2010. [Citado el: 25 de mayo de 2015.] <http://www.freevistafiles.com/Software-de-gesti-n-negocio-inventario+Ok-Inventario-Gratuita-software-de-gesti-n-de-negocio.html>.
3. **SOFTONIC INTERNACIONAL S.A.** SOFTONIC . *SOFTONIC* . [En línea] 8 de Abril de 2015. [Citado el: 25 de 5 de 2015.] <http://fulltrust.softonic.com/>.
4. EcuRed. *EcuRed*. [En línea] 2001. [Citado el: 25 de Mayo de 2015.] <http://www.rational.com.ar/herramientas/roseenterprise.html>.
5. **Molina, Ernesto Yariel Gámez.** *Sistema informatico para el control de inventario de equipos de cómputuo en la empresa Agropecuaria Saqua de Tánamo*. Moa : s.n., 2012.
6. **Subiros Muños, Dariel Raúl.** *Desarrollo de una interfaz gráfica de usuario para el preprocesador meteorológico AERMET. Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”*. Moa : s.n., 2009.
7. Masternewmedia.org. [En línea] 2005. [Citado el: 3 de Abril de 2013.]
8. **KI Networks: Achour, Adnane.** [En línea] 2005. [Citado el: 3 de Abril de 2013.] <http://researchnetworks.ki.se/converis/person/262>.
9. (MarcadorDePosición1)9. **CAVSI.** ¿Qué es un sistema gestor de Bases de Datos o SGBD? [En línea] 2008. [Citado el: 3 de Abril de 2013.] <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
10. **Vegas.** 2002. 11.
11. MySQL. [En línea] [Citado el: 10 de Abril de 2013.] <http://Fsinemed.com%2Frecursos%2Fdocs%2FMySQL.pdf>.
12. **Linux., Á.t.d.** 2009.
13. **Santiago, I.S.C.B.M.** *Programación web II*. noviembre 2009.



14. XAMPP . [En línea] 14 de Febrero de 2012. [Citado el: 15 de Abril de 2013.] <http://myu-charly.blogspot.com/>.
15. **Frederick, S., Ramsay, Colin y Blades, Steve 'Cutter'**. *Learning Ext JS*. 2008.
16. **K, Beck**. *Extreme Programming Explained. Embrace Change*. s.l. : Pearson Education, 1999.
17. ER/Studio. [En línea] [Citado el: 10 de Abril de 2013.] <http://www.monografias.com/trabajos14/modelodebase/modelodebase.shtml>.
18. EmbarcaderoER/Studio. [En línea] [Citado el: 10 de Abril de 2013.] <http://bureaudeprensa.com/es/>.
19. **Zulueta Torres, Agustín**. *Modulo para la Extracción, Pre procesamiento, Descripción y Almacenaje en formato XML, de la información recuperada por el Sistema Automatizado de Información Virtual del ISMMM*. 2010.
20. Ventajas y Desventajas de las Aplicaciones Web. 5 de 1 de 2010. [http://www.alegsa.com.ar/Respuesta/ventajas\\_y\\_desventajas\\_de\\_la\\_s\\_aplicaciones\\_web.htm](http://www.alegsa.com.ar/Respuesta/ventajas_y_desventajas_de_la_s_aplicaciones_web.htm) (último acceso: 24 de 6 de 2015).
21. La Güeb de Joaquín UML -Tarjetas de CRC. Disponible en: <http://jms32.eresmas.net/tacticos/UML/UML04/UML0402.html>
22. **SAS, SATTEK**. SATTEK. *SATTEK*. [En línea] 2012. [Citado el: 15 de Abril de 2015.] [http://www.sattek.co/servicios\\_computo.html](http://www.sattek.co/servicios_computo.html).
23. Baudrit Carrillo, D.: Derecho Civil IV, Volumen I. Teoría General del Contrato. 3era edición, 2000
24. Norbert Wiener. Cibernética y sociedad (The Human Use of Human Beings: Cybernetics and Society) (1950)
25. *EXTMX*. [En línea] Josue Hernandez, 15 de Enero de 2015. [Citado el: 10 de Junio de 2015.] <http://werxLtd.com/wp/2009/08/13/primitive-oop-with-extjs/>.

## Bibliografía

**Sánchez, Iliana Díaz y Rodríguez, MarcosA. Martínez.** *Herramienta para el control de los medios informáticos de la Empresa*. Moa : s.n., 2008.

Free Vista Files. *Free Vista Files*. [En línea] 23 de febrero de 2010. [Citado el: 25 de mayo de 2015.] <http://www.freevistafiles.com/Software-de-gesti-n-negocio-inventario+Ok-Inventario-Gratuita-software-de-gesti-n-de-negocio.html>.

**SOFTONIC INTERNACIONAL S.A.** SOFTONIC . *SOFTONIC* . [En línea] 8 de Abril de 2015. [Citado el: 25 de 5 de 2015.] <http://fulltrust.softonic.com/>.

EcuRed. *EcuRed*. [En línea] 2001. [Citado el: 25 de Mayo de 2015.] <http://www.rational.com.ar/herramientas/roseenterprise.html>.

**Molina, Ernesto Yariel Gámez.** *Sistema informático para el control de inventario de equipos de cómputo en la empresa Agropecuaria Saqua de Tánamo*. Moa : s.n., 2012.

**Subiros Muños, Dariel Raúl.** *Desarrollo de una interfaz gráfica de usuario para el preprocesador meteorológico AERMET. Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”*. Moa : s.n., 2009.

Masternewmedia.org. [En línea] 2005. [Citado el: 3 de Abril de 2013.]

**KI Networks: Achour, Adnane.** [En línea] 2005. [Citado el: 3 de Abril de 2013.] <http://researchnetworks.ki.se/converis/person/262>.

(MarcadorDePosición1)9. **CAVSI.** ¿Qué es un sistema gestor de Bases de Datos o SGBD? [En línea] 2008. [Citado el: 3 de Abril de 2013.] <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.

**Vegas.** 2002. 11.

MySQL. [En línea] [Citado el: 10 de Abril de 2013.] <http://Fsinemed.com%2Frecursos%2Fdocs%2FMySQL.pdf>.

**Linux., Á.t.d.** 2009.

**Santiago, I.S.C.B.M.** *Programación web II*. noviembre 2009.

XAMPP . [En línea] 14 de Febrero de 2012. [Citado el: 15 de Abril de 2013.]  
<http://myu-charly.blogspot.com/>.

**Frederick, S., Ramsay, Colin y Blades, Steve 'Cutter'.** *Learning Ext JS*. 2008.

**K, Beck.** *Extreme Programming Explained. Embrace Change*. s.l. : Pearson Education, 1999.

ER/St tudio. [En línea] [Citado el: 10 de Abril de 2013.]  
<http://www.monografias.com/trabajos14/modelodebase/modelodebase.shtml>.

EmbarcaderoER/Studio. [En línea] [Citado el: 10 de Abril de 2013.]  
<http://bureaudeprensa.com/es/>.

**Zulueta Torres, Agustín.** *Modulo para la Extracción, Pre procesamiento, Descripción y Almacenaje en formato XML, de la información recuperada por el Sistema Automatizado de Información Virtual del ISMMM*. 2010.

**Graham, Paul.** *The Other Road Ahead* . [En línea] 2011. [Citado el: 5 de Ma de 2013.]

La Güeb de Joaquín UML -Tarjetas de CRC. Disponible en:  
<http://jms32.eresmas.net/tacticos/UML/UML04/UML0402.html>

**SAS, SATTEK.** *SATTEK. SATTEK*. [En línea] 2012. [Citado el: 15 de Abril de 2015.] [http://www.sattek.co/servicios\\_computo.html](http://www.sattek.co/servicios_computo.html).

Baudrit Carrillo, D.: *Derecho Civil IV, Volumen I. Teoría General del Contrato*. 3era edición, 2000

Norbert Wiener. Cibernética y sociedad (The Human Use of Human Beings: Cybernetics and Society) (1950)

*EXTMX*. [En línea] Josue Hernandez, 15 de Enero de 2015. [Citado el: 10 de Junio de 2015.] <http://werxLtd.com/wp/2009/08/13/primitive-oop-with-extjs/>.

[http://www.ecured.cu/index.php/Rational\\_Rose\\_Enterprise\\_Edition](http://www.ecured.cu/index.php/Rational_Rose_Enterprise_Edition)  
<http://fulltrust.softonic.com/>

**Jacobson, Ivar.** *El proeceso unificado de desarrollo de sotware, volumenII*. Habana : Félix Varela, 2004.

**García, Rosa María Mato.** *Sistema de base de datos* . Habana : Félix Varela, 2006.

Frederick, S., Ramsay, Colin y Blades, Steve 'Cutter'. *Learning Ext JS*. 2008.

Santiago, I.S.C.B.M., *Programación web II*. noviembre 2009.

## Glosario de términos

**Aplicación:** Es el programa que el usuario activa para trabajar en el ordenador. Existen muchos programas de ordenador que pueden clasificarse como aplicación. Generalmente se les conoce como Software.

**Aplicación WEB:** Una aplicación Web es un sistema informático que los usuarios usan accediendo a un servidor Web a través de los protocolos de Internet. Las aplicaciones Web son populares por su practicidad del navegador Web como cliente ligero.

**API:** Una API (del inglés Application Programming Interface - Interfaz de Programación de Aplicaciones) es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

**Metodologías de Desarrollo:** Se define como un conjunto de filosofías, etapas, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de sistemas de información.

**Multiplataforma:** Es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas o sistemas operativos.

**Programación Extrema(XP):** Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.

**RUP:** El Proceso Unificado Rational (RUP) es una metodología de desarrollo para la programación orientada a objetos. Según Rational (diseñadores de Rose Rational y el Idioma Modelado Unificado UML), RUP está como un mentor en línea que mantiene pautas, plantillas, y ejemplos de todos los aspectos y fases de desarrollo del programa.

**Software Libre:** Es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente, aunque conserve su carácter de libre, puede ser vendido comercialmente.

**Servidor Web:** Un programa que corre sobre el servidor que escucha las peticiones HTTP (Hyper Text Markup Language) que le llegan y las satisface. Dependiendo del tipo de petición, el servidor Web buscará una página Web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición

**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).

**Ext-JS:** Es una librería Javascript que permite construir aplicaciones RIA (Rich Internet Applications).

**HU:** Historia de Usuarios son las tablas creadas para la realización de las interfaces del sistema.

**HL:** Historia Laboral de un trabajador

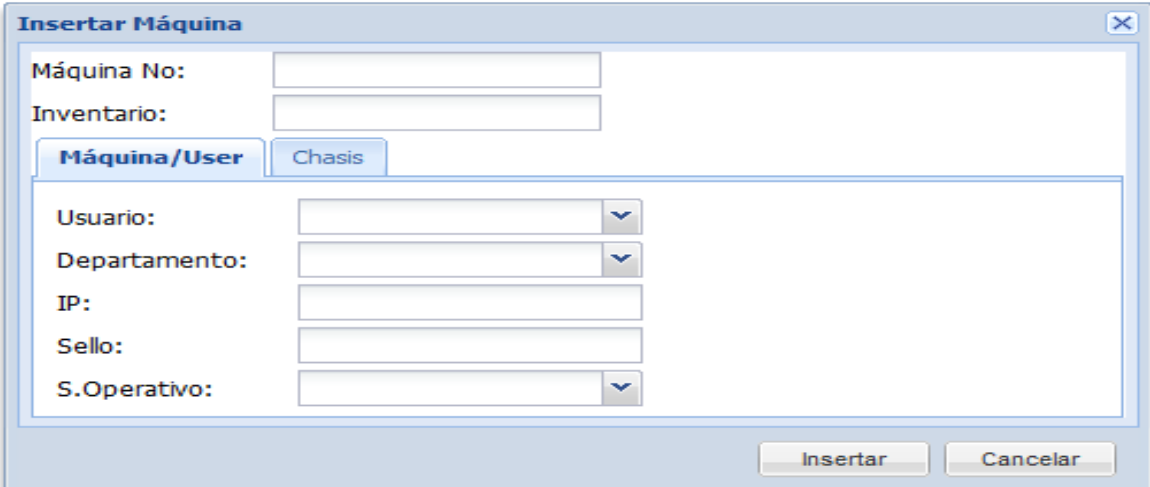
**IU:** Interfaces de Usuarios

**Open Source:** Licencias Open Source (Código Abierto) es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

## Anexo 1. Historia de Usuario

**Tabla 1: HU Gestionar Maquina**

Historia de Usuarios	
Número: No. 4	Usuario: Administrador
Nombre: Gestionar Maquina	
Prioridad en el Negocio: Alta	Riesgo en el Desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 1
Programador Responsable: José Angel Mendoza	
Descripción: El usuario insertará los datos, una vez insertados se podrá, mostrar, modificar y eliminar.	
Observaciones: Confirmado con el cliente	

**Tabla 2: HU Insertar Periféricos**

Historia de Usuarios	
Número: No. 6	Usuario: Administrador
Nombre: Insertar Periféricos	
Prioridad en el Negocio: Alta	Riesgo en el Desarrollo: Alto
Puntos estimados: 3	Iteración Asignada: 1
Programador Responsable: José Angel Mendoza	
Descripción: El usuario insertará los datos de: Monitor, Teclado, Mause,	

**Bocinas, UPS.**

**Observaciones: Confirmado con el cliente**

**Insertar Componentes**

No. Máquina:

**Monitor** | Teclado | Mouse | Bocinas | UPS

Marca:

Modelo:

Pulgadas:

N/S:

Inventario:

Insertar Cancelar

**Tabla 15: HU Gestionar Monitor**

### Historia de Usuarios

**Número: No. 15**

**Usuario:Administrador**

**Nombre: Gestionar Monitor**

**Prioridad en el Negocio: Media**

**Riesgo en el Desarrollo: Alto**

**Puntos estimados: 2**

**Iteración Asignada: 2**

**Programador Responsable: José Angel Mendoza**

**Descripción: El usuario podrá ya con los datos insertados: mostrar, modificar y eliminar una vez que los datos estén insertados.**

**Observaciones: Confirmado con el cliente**

**Modificar Monitor**

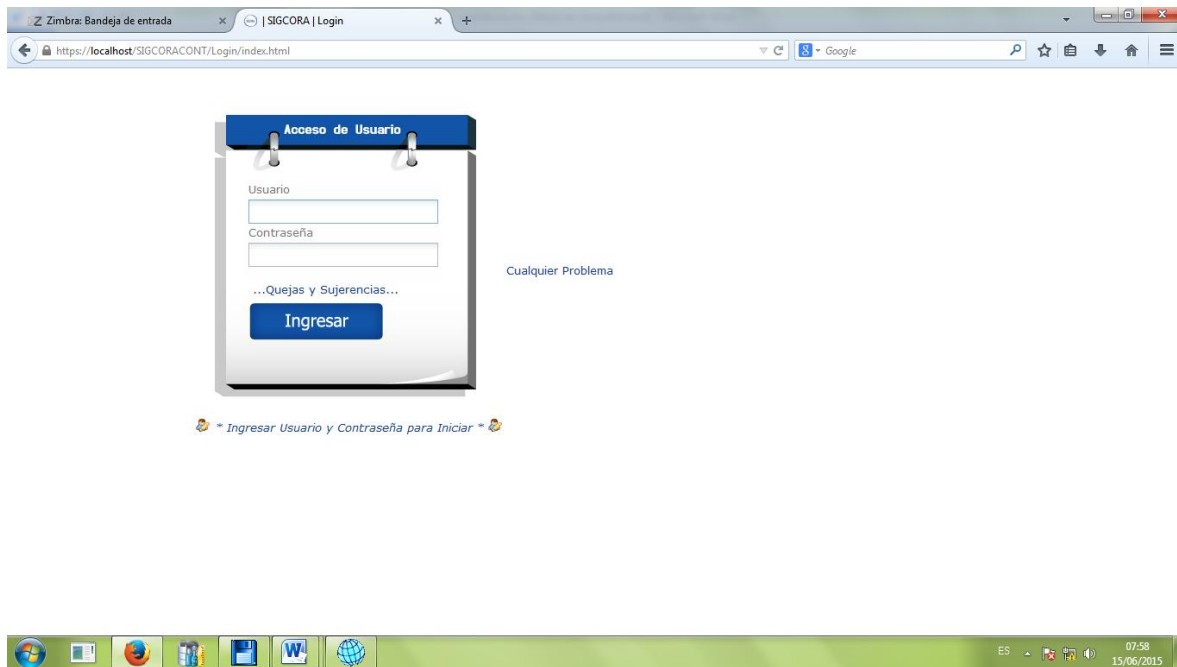
Eliminar | Buscar...

	Marca	Modelo	Pulgadas	Numero S...	Numero In...	Numero M...
1	sony	asa14	18	1256	1258789	12

Página  de 1
 
 1 - 1 de 1 Archivos



## Anexo 2. Interfaces del Sistema

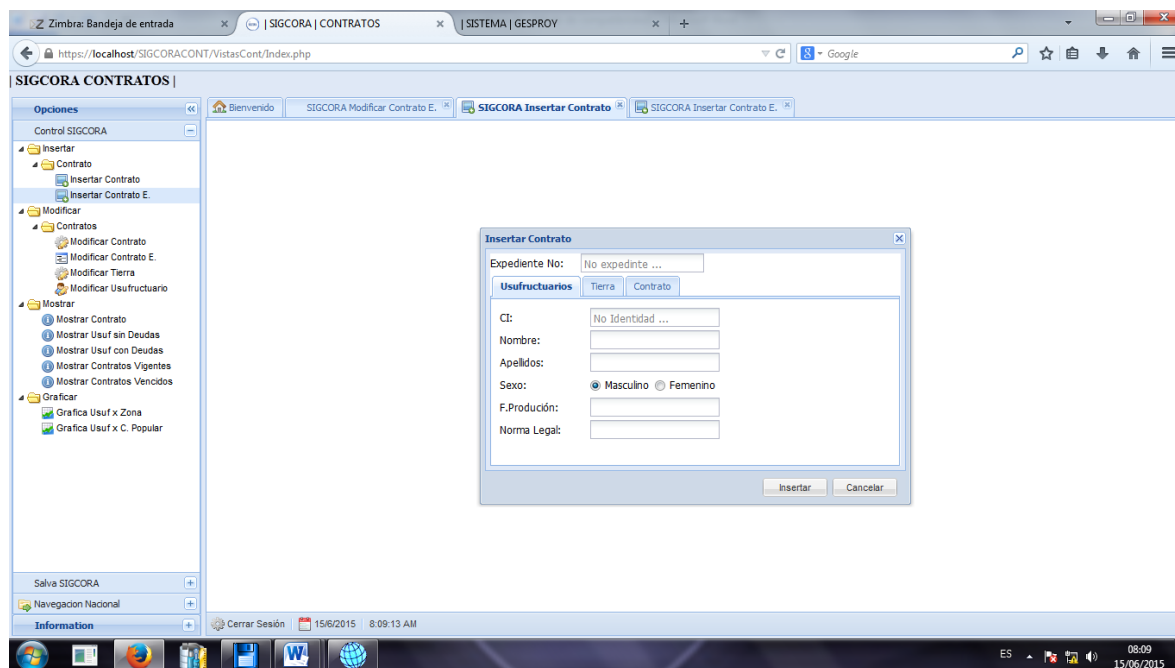


*Figura 1: Login de usuarios*



*Figura 2: Página de Inicio*

**Sistema para la Gestión de los contratos y el control de los medios informáticos en la empresa Agropecuaria Sagua de Tánamo.  
(SIGCORA)**



*Figura 3: Insertar un Contrato*

### Anexo 3. Tarjetas de Tareas

Tarea de Ingeniería	
Número tarea: 1	Número de HU: 1
Nombre de Tarea: Insertar Persona	
Tipo de tarea: Desarrollo	Puntos estimados:0.3
Fecha Inicio: 26/2/15	Fecha Fin: 1/3/15
Programador Responsable: José Angel Mendoza Matilla	
Descripción: Esta tarea permite insertar una persona en la BD.	

Tabla 1: Tarjeta de Tareas Insertar Persona

Tarea de Ingeniería	
Número tarea: 2	Número de HU: 1
Nombre de Tarea: Mostrar Persona	
Tipo de tarea: Desarrollo	Puntos estimados:0.3
Fecha Inicio: 26/2/15	Fecha Fin: 1/3/15
Programador Responsable: José Angel Mendoza Matilla	
Descripción: Esta tarea permite visualizar los datos de la persona o personas insertadas en la BD.	

Tabla 2: Tarjeta de Tareas Mostrar Persona

Tarea de Ingeniería	
Número tarea: 4	Número de HU: 1
Nombre de Tarea: Modificar Persona	
Tipo de tarea: Desarrollo	Puntos estimados:0.3
Fecha Inicio: 2/3/15	Fecha Fin: 5/3/15
Programador Responsable: José Angel Mendoza Matilla	
Descripción: Esta tarea permite modificar los datos de la persona o personas insertadas en la BD.	

Tabla 3: Tarjeta de Tareas Modificar Persona

## Anexo 4. Pruebas de Aceptación por HU

Tabla 2: Prueba de Aceptación Gestionar Impresora

Prueba de Aceptación
HU: 5
Nombre: Gestionar Componentes
Descripción: La especialista debe suministrarle los datos al sistema, para luego mostrarlo, modificarlo y eliminarlo.
Condiciones de ejecución: Se deben introducir lo datos
Entrada/Pasos ejecución: <ul style="list-style-type: none"> <li>Insertar Componentes</li> </ul>
Resultado: Gestiona correctamente el Componentes
Evolución de la prueba: Aceptada

Tabla 1: Prueba de Aceptación Gestionar Componentes

Prueba de Aceptación
HU: 3
Nombre: Gestionar Departamento
Descripción: La especialista debe suministrarle los datos al sistema, para luego mostrarlo, modificarlo y eliminarlo.
Condiciones de ejecución: Se deben introducir lo datos
Entrada/Pasos ejecución: <ul style="list-style-type: none"> <li>Insertar Departamento</li> <li>Mostrar Departamento</li> <li>Modificar Departamento</li> <li>Eliminar Departamento</li> </ul>
Resultado: Gestiona correctamente la Departamento
Evolución de la prueba: Aceptada

Tabla 2: Prueba de Aceptación Gestionar Departamento

## Anexo 5. Diagramas de Secuencias

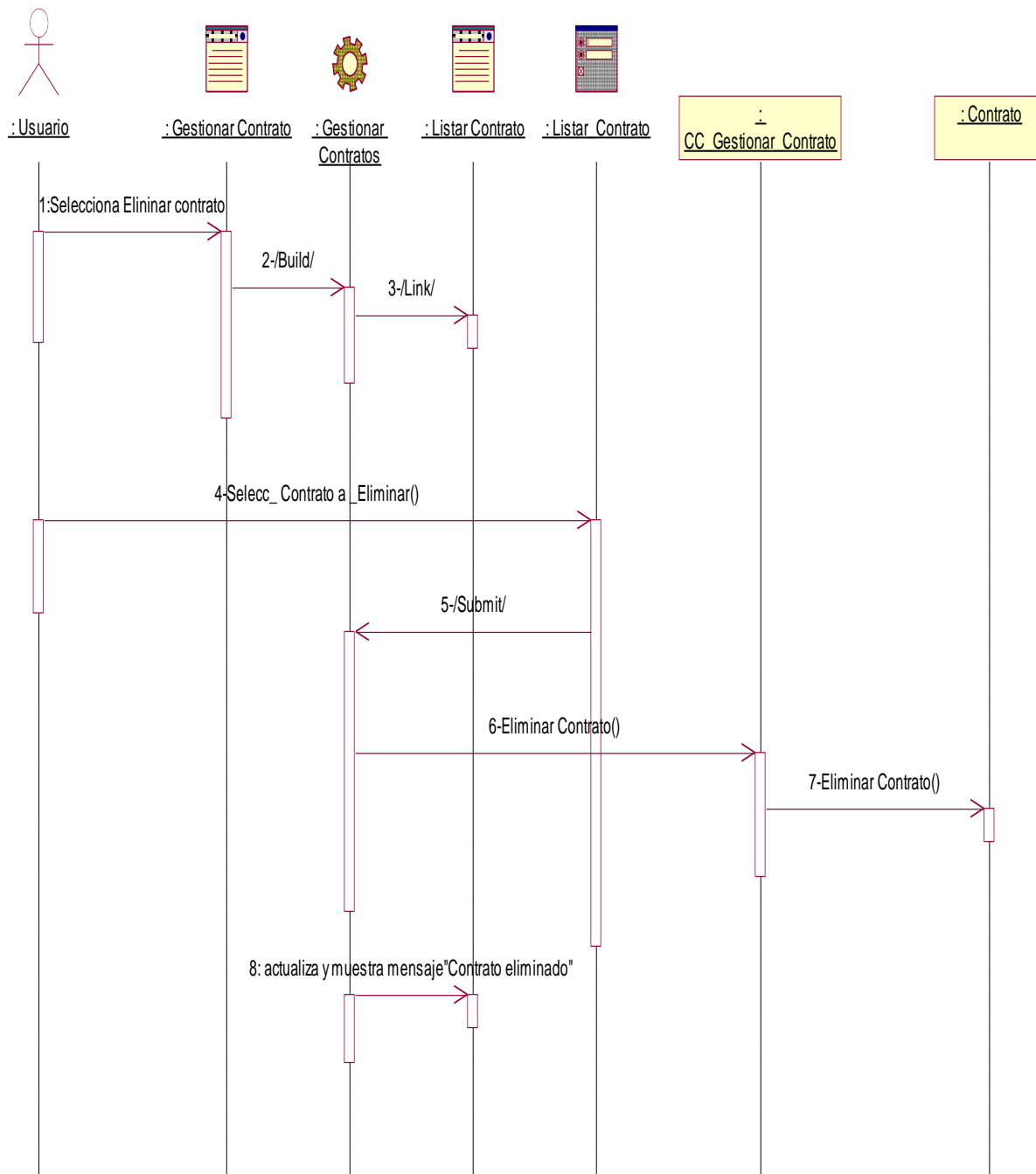


Figura 3.2 Diagrama de Secuencia Eliminar Contrato

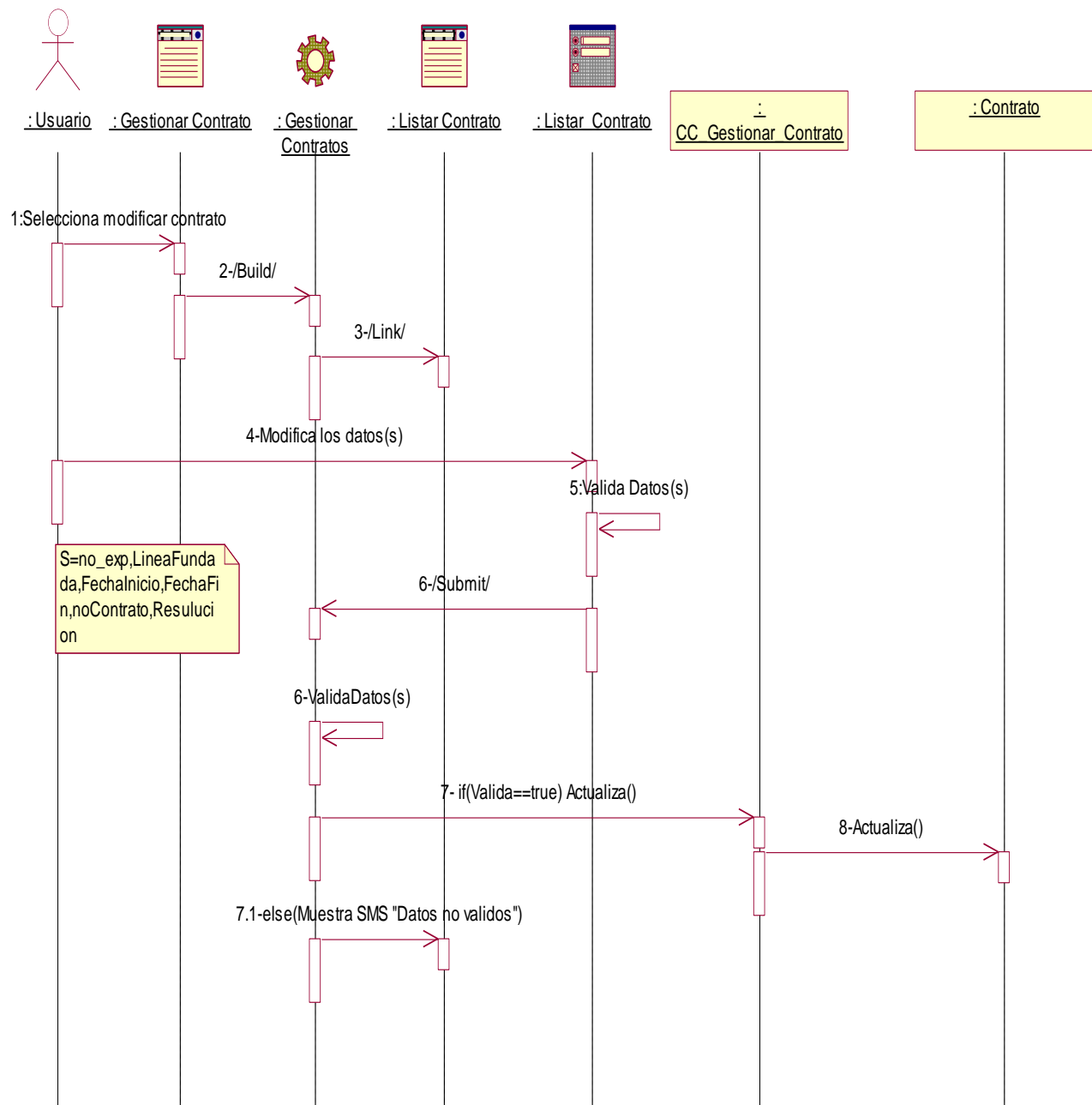


Figura 3.3 Diagrama de Secuencia Modificar Contrato

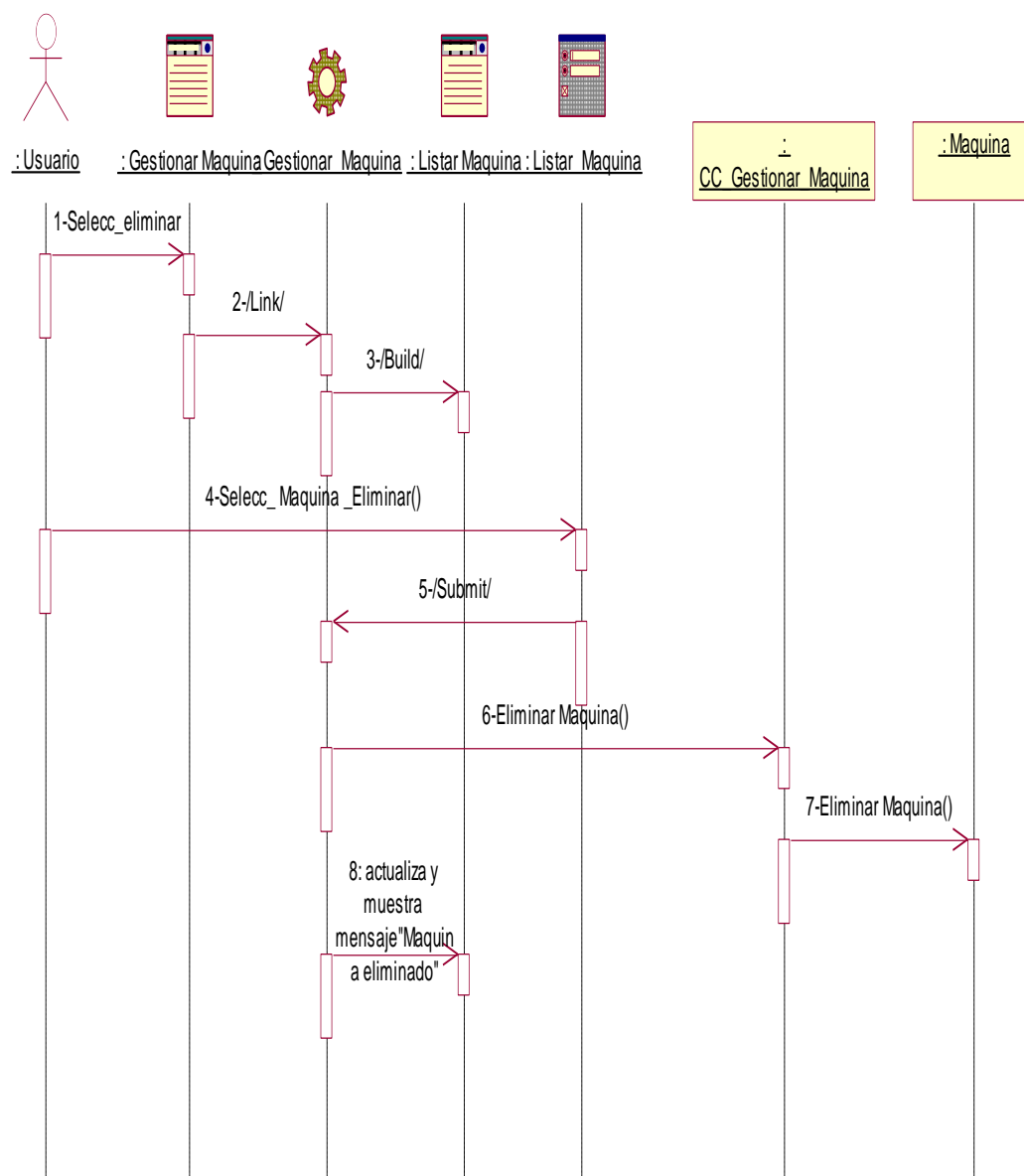


Figura 3.2 Diagrama de Secuencia Eliminar Maquina

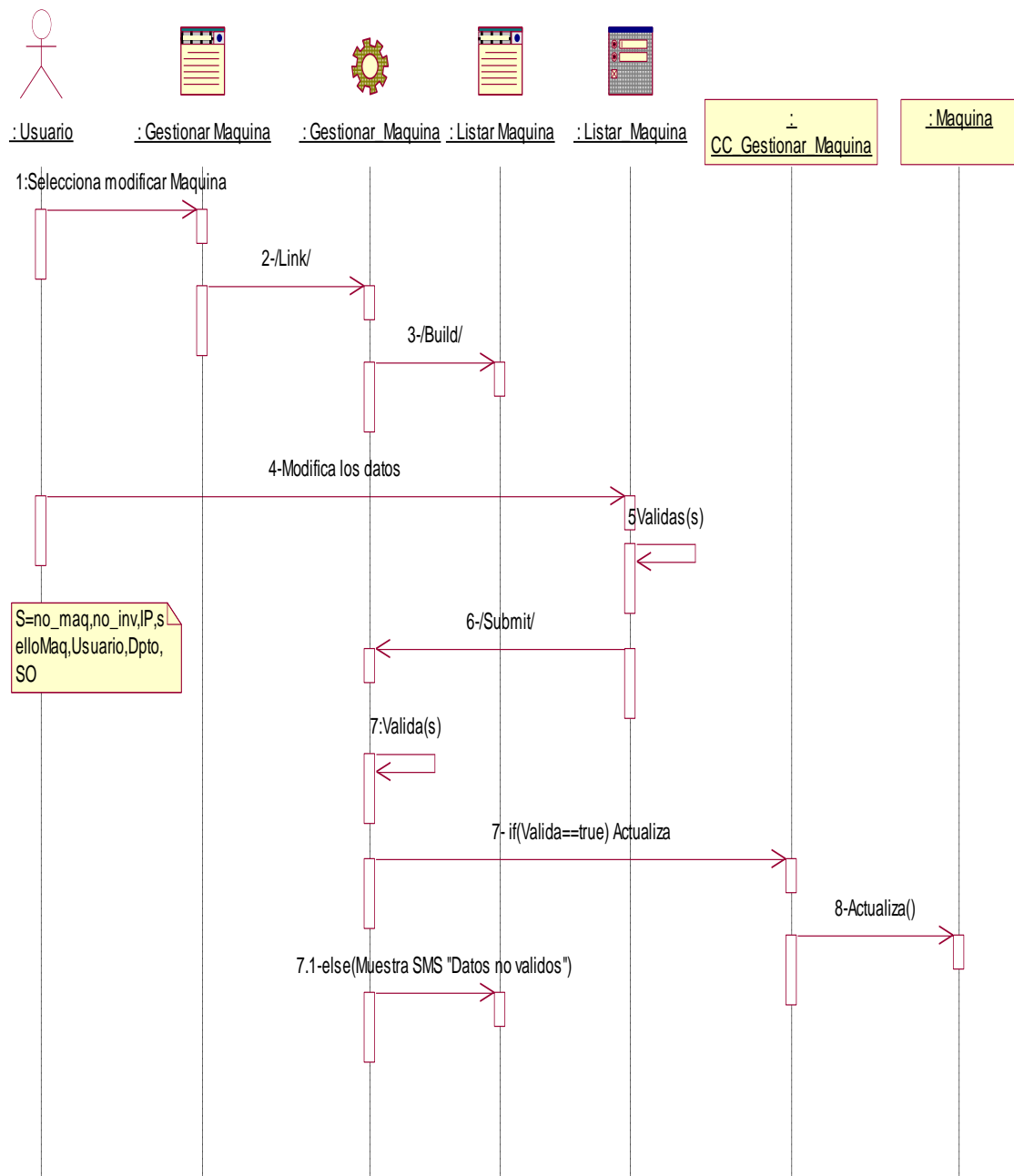


Figura 3.3 Diagrama de Secuencia Modificar Maquina