

UNIVERSIDAD AUSTRAL DE CHILE
SEDE PUERTO MONTT
ESCUELA DE INGENIERIA EN COMPUTACION



Sistema Control de Herramientas, Insumos y Materiales en
Bodegas de Empresas Schwerter de Puerto Montt

Seminario de Titulación para
optar al título de Ingeniero en
Computación

PROFESORA PATROCINANTE:
Sra. Viviana Alvarado Espinoza.

LUIS EDUARDO ULE MOLINA
PUERTO MONTT – CHILE

2013



Universidad Austral de Chile

Escuela de Ingeniería en Computación

Los Pinos s/n, Balneario Pelluco
Sede Puerto Montt
Casilla 1327 · Fono: 56 65 260990
Fax: 56 65 277156
Email: ecomputa@uach.cl
www.uach.cl

Puerto Montt, 29 de julio de 2013

COMUNICACIÓN INTERNA N° 116/13

DE : Sra. Sandra Ruiz Aguilar
DIRECTORA ESCUELA DE INGENIERIA EN COMPUTACION

A : Sra. Sandra Burgos H - DIRECTORA ACADEMICA (S) SEDE PUERTO MONTT
Sra. Angélica Barrientos -DEPARTAMENTO REGISTRO ACADEMICO SEDE PTO. MONTT

C.c : Luís Eduardo Ule Molina
Viviana Alvarado Espinoza
Mónica Gallardo Vargas
Sandra Ruiz Aguilar

MOTIVO:

Informar a usted, las calificaciones obtenidas por el alumno de Ingeniería en Computación Sr. Luís Eduardo Ule Molina Rut 12,752,599-4, en su informe de Titulación "Sistema de Manejo de Artículos en Bodega de Herramientas, Insumos y Materiales (HIM) en Empresas Nilson Schwerter y Empresas Asociadas"

Viviana Alvarado Espinoza	6,2
Mónica Gallardo Vargas	5,5
Sandra Ruiz Aguilar	5,0
Promedio Seminario	5,57

Sin otro particular, le saluda atentamente,



SRA/mva

PUERTO MONTT, 29 de julio, 2013

De : Sra. Viviana Alvarado Espinoza
PROFESORA PATROCINANTE

A : Sra. Sandra Ruiz Aguilar
DIRECTORA ESCUELA INGENIERÍA EN COMPUTACIÓN

MOTIVO:

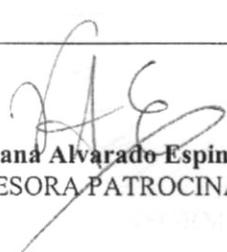
Informar a Usted la calificación obtenida por el alumno **Luis Eduardo Ule Molina** en su Seminario de Titulación "**Sistema de Manejo de Artículos en Bodega de Herramientas, Insumos y Materiales (HIM) en Empresas Nilson Schwerter y Empresas Asociadas**":

NOTA: 6,2 (seis coma dos)

JUSTIFICACION:

- Desarrollo metodológico adecuado
- Pocas observaciones en ortografía y redacción
- Faltó documentar detalles de implementación

OTRAS OBSERVACIONES:


Viviana Alvarado Espinoza
PROFESORA PATROCINANTE

PUERTO MONTT, 29 de Julio de 2013.

De : Sr. Mónica Gallardo Vargas
PROFESORA INFORMANTE

A : Sra. Sandra Ruiz Aguilar
DIRECTORA ESCUELA INGENIERÍA EN COMPUTACIÓN

MOTIVO:

Informar a Usted la calificación obtenida por el alumno **Luís Eduardo Ule Molina** en su Seminario de Titulación "Sistema de Manejo de Artículos en Bodega de Herramientas, Insumos y Materiales (HIM) en Empresas Nilson Schwerter y Empresas Asociadas":

NOTA:

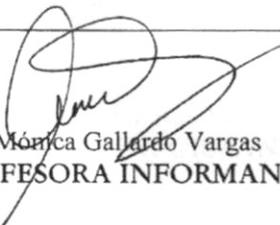
5,5

JUSTIFICACION:

El documento no muestra claramente el desarrollo del proyecto usando la metodología

OTRAS OBSERVACIONES:

No se ajusta completamente al formato del Seminario


Mónica Gallardo Vargas
PROFESORA INFORMANTE

PUERTO MONTT, 29 julio 2013

De : Sra. Sandra Ruiz Aguilar
PROFESORA INFORMANTE

A : Sra. Sandra Ruiz Aguilar
DIRECTORA ESCUELA INGENIERÍA EN COMPUTACIÓN

MOTIVO:

Informar a Usted la calificación obtenida por el alumno **Luis Eduardo Ule Molina** en su Seminario de Titulación "**Sistema de Manejo de Artículos en Bodega de Herramientas, Insumos y Materiales (HIM) en Empresas Nilson Schwerter y Empresas Asociadas**":

NOTA: 5.0

JUSTIFICACION:

La descripción del sistema no fue muy adecuada. Falta mayor detalle técnico.

OTRAS OBSERVACIONES:

Informe tiene varios errores que hay que corregir.


Sandra Ruiz Aguilar
PROFESORA INFORMANTE

Dedicada en especial al amor incondicional de mis pequeñas hijas
Anabelle y Bianca.

Agradecimientos

Quisiera dar las gracias a mis padres José y Gloria, que con su eterno apoyo han logrado llevarme a la etapa que hoy me encuentro, en donde con buenos y malos momentos han podido ayudarme a cumplir con uno de los objetivos más importantes en mi vida profesional.

Agradecer a mis hermanos Claudio Alejandro y María Jesús, también a Claudia por su valiosa compañía y amor entregado, todos ellos con su incondicional cariño supieron ayudarme en momentos cuando las cosas se ponían difíciles.

También agradecer a mis amigos de mi etapa universitaria, que con su apoyo, compañía y solidaridad me ayudaron a cumplir con éxito las etapas de mi vida universitaria, en especial a Lerou, Aguila, Otarola, España, Guelet, Moraga, Marta, Carola, Silva y a Carlitos Ojeda.

A mis profesores de la universidad, por darse molestia de ayudarme siempre que se los solicité. Además, agradecer a mi profe guía Viviana Alvarado, quien ha sido de gran ayuda para completar con éxito esta última etapa de mi carrera universitaria.

*“No nos atrevemos a muchas cosas porque son difíciles,
pero son difíciles porque no nos atrevemos a hacerlas”.*

Lucio Anneo Séneca

INDICE

Síntesis en castellano

Síntesis en inglés - Abstract

1	INTRODUCCIÓN	1
2	OBJETIVOS	3
2.1	OBJETIVO GENERAL.....	3
2.2	OBJETIVOS ESPECÍFICOS.....	3
3	PLANTEAMIENTO DEL PROBLEMA	4
3.1	ANTECEDENTES.....	4
3.1.1	<i>Definición del Problema</i>	4
3.1.2	<i>Esfuerzos Anteriores</i>	6
3.1.3	<i>Solución Propuesta</i>	6
3.2	JUSTIFICACIÓN	8
3.2.1	<i>Situación sin proyecto</i>	8
3.2.2	<i>Situación con proyecto</i>	9
3.3	DELIMITACIÓN.....	10
4	METODOLOGÍA	11
4.1	ITERACIONES.....	13
5	RECURSOS	16
5.1	HARDWARE	16
5.2	SOFTWARE	17
6	ANÁLISIS DEL SISTEMA	19
6.1	RECOPIACIÓN DE REQUERIMIENTOS.....	19
6.1.1	<i>Modelo de Casos de Uso</i>	26
6.1.2	<i>Breve Descripción de los Casos de Uso</i>	29
6.1.3	<i>Herramientas de trabajo</i>	31
6.2	DEFINICIÓN DEL SISTEMA.....	32

6.3	MÓDULOS DEL SISTEMA.....	33
6.4	SEGURIDAD DEL SISTEMA.....	34
6.5	USUARIOS DEL SISTEMA.....	34
6.6	MODELO DE 3 CAPAS.....	35
6.7	ENTORNO TECNOLÓGICO – HARDWARE – SOFTWARE.....	39
7	DISEÑO DEL SISTEMA.....	42
7.1	DISEÑO DE LA BASE DE DATOS.....	42
7.1.1	<i>Diseño Conceptual.....</i>	<i>43</i>
7.1.2	<i>Diseño Lógico de la Base de Datos.....</i>	<i>61</i>
7.1.3	<i>Diseño Físico de la Base de Datos.....</i>	<i>74</i>
7.1.4	<i>Diseñar la Representación Física.....</i>	<i>77</i>
7.1.5	<i>Requerimientos y Mecanismos de Seguridad.....</i>	<i>82</i>
7.2	DISEÑO DE LA APLICACIÓN.....	84
7.2.1	<i>Diseño de Transacción.....</i>	<i>85</i>
7.2.2	<i>Diseño de Interfaz de Usuario.....</i>	<i>86</i>
8	DESARROLLO DEL SISTEMA E IMPLEMENTACIÓN.....	97
8.1	IMPLEMENTACIÓN DE LA BASE DE DATOS.....	97
8.1.1	<i>Script de creación de la Base de Datos Bodegas_HIM.....</i>	<i>98</i>
8.2	IMPLEMENTACIÓN DE INTERFAZ DEL SISTEMA.....	106
9	PRUEBAS DEL SISTEMA.....	107
9.1	PRUEBA DE UNIDAD.....	108
9.2	PRUEBA DE INTEGRACIÓN.....	109
9.3	PRUEBA DE VALIDACIÓN.....	109
9.4	PRUEBA DE SISTEMA.....	110
10	IMPLANTACIÓN.....	112
10.1	INSTALACIÓN DE LA APLICACIÓN.....	112
10.2	ENTRENAMIENTO DE USUARIOS.....	112
11	CONCLUSIONES Y RECOMENDACIONES.....	113
12	BIBLIOGRAFÍA.....	115

13	ANEXO	116
13.1	PRUEBA DE CAJA NEGRA.....	116
13.2	PRUEBA DE CAJA BLANCA.....	121

TABLAS

<i>Tabla 1: Definición del equipo de trabajo</i>	8
<i>Tabla 2: Toma de requerimientos - Reunión 1</i>	22
<i>Tabla 3: Toma de requerimientos - Reunión 2</i>	24
<i>Tabla 4: Toma de requerimientos - Reunión 3</i>	24
<i>Tabla 5: Resumen de Requerimientos</i>	26
<i>Tabla 6: Entidades del Sistema</i>	49
<i>Tabla 7: Identificación de las Relaciones</i>	53
<i>Tabla 8: Identificación y Asociación de atributos de la entidad Bodega</i>	54
<i>Tabla 9: Identificación y Asociación de atributos de la entidad Documento</i>	55
<i>Tabla 10: Identificación y Asociación de atributos de la entidad articulo</i>	56
<i>Tabla 11: Identificación y Asociación de atributos de la entidad persona</i>	57
<i>Tabla 12: Identificación de Claves Candidatas y Primarias</i>	59
<i>Tabla 13: Fragmento Tabla N° 7 - Relaciones N:N del capítulo 7.2.1.2</i>	63
<i>Tabla 14: Fragmento Código Script crea Tabla Documento</i>	77
<i>Tabla 15: Fragmento Código Script creación de las Tablas</i>	105
<i>Tabla Anexo 1: Menús existentes del Sistema</i>	117
<i>Tabla Anexo 2: Resultados pruebas de Menús del Sistema</i>	121

FIGURAS

<i>Figura 1: Diagrama de Metodología XP</i>	12
<i>Figura 2: Diagrama de Casos de Uso – Modelo Plan de Negocio</i>	27
<i>Figura 3: Diagrama de Casos de Uso – Modelos de Casos de Uso</i>	28
<i>Figura 4: Diagrama capa de datos</i>	37
<i>Figura 5: Diagrama capa de datos</i>	40
<i>Figura 6: Entorno Tecnológico</i>	40
<i>Figura 7: Diagrama Modelo Conceptual de Datos</i>	60
<i>Figura 8: Diagrama Modelo Lógico de Datos</i>	62
<i>Figura 9: Relación N:N entidad Bodega-Articulo</i>	64
<i>Figura 10: Relación N:N entidad Localidad - Proveedor</i>	64
<i>Figura 11: Relación N:N entidad Documento – Articulo</i>	65
<i>Figura 12: Diagrama del Modelo Físico de la Base de Datos</i>	79
<i>Figura 13: Diagrama de Navegación de Menús</i>	89
<i>Figura 14: Ventana de Inicio del Sistema</i>	91
<i>Figura 15: Ventana Principal del Sistema.</i>	92
<i>Figura 16: Ventana del Mantenedor de Maestro de Usuarios</i>	93
<i>Figura 17: Ventana de Menús del Mantenedor de Maestros del Sistema.</i>	94
<i>Figura 18: Ventana de Selección y Modificación de Usuarios.</i>	95
<i>Figura 19: Ventana Mensaje de Grabación de Registros.</i>	96
<i>Figura 20: Representación de la Base de Datos.</i>	97
<i>Figura 21: Interfaz de Sistema – Búsqueda y Eliminación de Movimientos</i>	106

Síntesis

En Chile uno de los rubros más dinámicos es el de la Construcción, es por esta razón que en lo que a pequeña y mediana escala se refiere, siempre los problemas de control en sus herramientas y materiales de construcción, son los mismos, ya que este control debe recaer en el detallado control de artículos que pueden ser vitales para concretar el correcto desarrollo de las distintas etapas que se puede subdividir una obra de construcción.

Las grandes empresas constructoras han resuelto este problema entregando a empresas de menor tamaño la ejecución de estas etapas, creando la figura comercial de subcontrato, de esta manera dejan en otras manos el control y adquisición de las herramientas, insumos y materiales de construcción que se utilizarán en las distintas etapas que pueda tener una construcción, las que son denominadas como obras.

En el presente Seminario de Titulación se aborda este problema basándose en los distintos casos existentes en las empresas constructoras propiedad del Ingeniero Civil Sr. Nilson Schwerter, en donde se plantea el cómo se analizó, evaluó y creó un Sistema de Control capaz de permitir, a los administradores, encargados de adquisiciones y encargados de bodegas, el uso

de un mejor control de sus herramientas, insumos y materiales (HIM) que son utilizadas en la ejecución de las diversas obras que estas empresas ejecutan.

El Sistema de Control permite la creación de obras, asignarles un stock de artículos HIM, controlar de mejor manera los traslados de artículos HIM, gestionar los préstamos o usos de herramientas, controlar los insumos y materiales consumidos en el desarrollo de las distintas etapas de las obras en ejecución y lo más importante, poder identificar a los actores participantes en estas etapas.

En este documento se describe en detalle el procedimiento de cada paso para el desarrollo de la aplicación, desde la toma de requerimientos y construcción de base de datos, hasta la puesta en marcha del producto.

Para el desarrollo de este seminario de titulación el alumno tesista recurrió a un método de desarrollo llamado "Extreme Programming" que permitió la ágil construcción del producto. Además de esto, se utilizaron algunas herramientas de desarrollo de softwares, algunas de ellas son Power Builder 12.5, Power Designer 15, SQL Server 2008 R2, entre otros.

Abstract

In Chile, one of the most dynamic is the Construction, is for this reason that when it comes to small and medium scale refers always control problems are the same, and that fall in simplifying and controlling articles detailed concrete can be vital to the proper development of the different stages that can be subdivided a building.

The large construction companies have solved this problem by delivering to smaller companies implementing these steps, creating commercial figure subcontract, so leave in other hands the control and acquisition of tools, supplies and materials to be used at various stages that can be subdivided construction which are known as works.

In this titling seminar this problem based on the various existing cases construction companies owned by Mr. Nilson Schwerter Civil Engineer, where he raises the how was analyzed, evaluated and created a control system capable of allowing to managers, procurement officers and managers of warehouses, the use of better control of your tools, supplies and materials (HIM) that are used in the execution of the various works that these companies run.

The control system allows the creation of works, assign a stock of items HIM, better control shipments of items HIM, manage loans or uses of tools, control inputs and materials consumed in the development of the different stages of the works in progress and most importantly, to identify the actors involved in these stages.

This document describes in detail the procedure for each step in the development of the application, from requirements gathering and database construction, to launch the product.

For the development of this seminar the student thesis degree turned to a development method called "Extreme Programming" that allowed the agile product construction. In addition, we used some software development tools, some of them are PowerBuilder 12.5, Power Designer 15, SQL Server 2008 R2, among others.

1 Introducción

En la ciudad de Puerto Montt, el Ingeniero Civil Sr. Nilson Schwerter posee tres empresas constructoras, en donde trabaja como persona natural utilizando su RUT personal, además de Sociedad Constructora Schwerter y Nilson Schwerter SPA, en las cuales desarrolla la misma actividad comercial. Estas empresas ejecutan obras de construcción en diversos lugares de la zona sur de Chile, con lo cual, deben tener la capacidad de controlar y administrar los stocks de las herramientas, insumos y materiales necesarios para la ejecución de las obras existentes.

Las Empresas Schwerter necesitan implementar un sistema capaz de facilitar el control y gestión de sus herramientas, insumos y materiales (HIM) utilizados en las diversas obras, de tal manera que este sistema posea la cualidad de ser confiable y seguro en cuanto a sus reportes y manejo de información y tenga la sencillez de ser operado en obra por personal de bajo manejo computacional.

En la actualidad, los administradores para controlar los stocks, se valen de los distintos capataces o encargados de bodega ubicados en cada lugar donde se ejecutan las obras, los que en su mayoría utilizan planillas en formato Excel

o planillas llevadas en formulario manual. Es importante mencionar también, que el nuevo sistema debe permitir la actualización vía archivos exportables/importables, ya que en la mayoría de las ocasiones los lugares de realización de obras no cuentan con acceso a internet.

Cabe hacer presente que entre los grandes problemas existentes hoy en día, son los registros de entrega de herramientas al personal, los ingresos de artículos HIM mediante adquisición directa en obra y los traslados de herramientas o materiales entre las propias obras sin pasar por el control de la bodega central de las empresas Schwerter, esto ocurre, debido a que en ocasiones existen artículos que pueden ser críticos en el cumplimiento de las distintas etapas de construcción o por hechos imprevistos ocurridos en las obras que se realizan.

En las hojas posteriores el alumno tesista, explicará de manera detallada los procesos que se siguieron para conseguir la realización de un sistema que pueda servir de real apoyo en el control de los artículos HIM en las distintas bodegas con la que cuenta y contarán las empresas Schwerter.

2 Objetivos

2.1 Objetivo general

Implementar un Sistema de control del flujo y stock de herramientas, insumos y materiales en las empresas constructoras Schwerter, en donde los usuarios puedan registrar los movimientos de artículos, controlar el stock y centralizar la información, en las distintas obras que se ejecutan en cualquier lugar que estas se desarrollen.

2.2 Objetivos específicos

- Gestionar el stock de los materiales e insumos y los movimientos asociados a ellos.

- Controlar el stock de herramientas existentes en obra.

- Importar y exportar información de obras ubicadas en lugares que no tienen acceso a internet.

- Obtener diversos informes como gastos por mes, ingresos de artículos por fecha, entre otros.

3 Planteamiento del Problema

3.1 Antecedentes

3.1.1 Definición del Problema

Nilson Schwerter, Ingeniero Civil de profesión, posee 3 empresas constructoras que dependen de él. Estas empresas ejecutan obras de construcción en diversos lugares de la zona sur de Chile, con lo cual, deben tener la capacidad de controlar y administrar los stocks de las herramientas, insumos y materiales necesarios para la ejecución de las obras existentes. Las Empresas Schwerter necesitan implementar un sistema capaz de facilitar el control y gestión de sus herramientas, insumos y materiales (HIM) utilizados en las diversas obras, de tal manera que este sistema posea la cualidad de ser confiable y seguro en cuanto a sus reportes y manejo de información y tenga la sencillez de ser operado en obra por personal de bajo manejo computacional.

Hoy en día los encargados de administrar se valen, de los distintos capataces o encargados de bodega ubicados en cada lugar donde se ejecutan las obras, los que en su mayoría utilizan métodos propios para organizar su información de stocks de artículos HIM que se encuentran en sus distintas bodegas, las que se encuentran en los diversos lugares de ejecución de sus obras.

Se hace presente que en la mayoría de los distintos lugares donde se encuentran las obras, no existe conexión a internet, debido a que en ocasiones estas se desarrollan en lugares muy distantes de centros urbanos.

También es importante señalar que el sistema debe tener la capacidad de poder ser operado por personal de bajo manejo computacional y con actualizaciones producidas por la exportación e importación de archivos para suplir la falta de conexión remota a internet.

Los problemas existentes hoy en día se centran principalmente en la falta de control de los registros de entrega de herramientas al personal, los ingresos de artículos HIM que se generan mediante adquisición directa en obra y los traslados de herramientas o materiales entre las propias obras, sin necesariamente contar con el control de la bodega central de las Empresas Schwerter, esto se debe, a que en ocasiones existen artículos que pueden ser

críticos en el cumplimiento de las distintas etapas de construcción o por hechos imprevistos ocurridos en las obras que se realizan.

3.1.2 Esfuerzos Anteriores

En la resolución de este problema, es que se ha estudiado la posibilidad de incorporar algún sistema informático que cumpla los requisitos antes expuestos, pero las soluciones estudiadas resultaron ser o demasiado complejas y costosas o no muy simples de adaptar al nivel de usuarios computacionales básicos, con lo cual, la mejor solución es generar y adaptar un sistema propio, lo que además significaría poder resolver un problema generalizado en la industria de este rubro.

3.1.3 Solución Propuesta

El alumno tesista construirá un Sistema centralizado de Manejo de artículos en Bodegas de Herramientas, Insumos y Materiales para las empresas del Ingeniero Civil Sr. Nilson Schwerter, el cuál gestionará el control y automatización del manejo eficiente y eficaz de toda la amplia gama de artículos con los que debe contar una empresa constructora para poder llevar a cabo sus obras en los distintos lugares que éstas se encuentren.

El sistema contará con un servidor a nivel local, el que se ubicará en las oficinas centrales de la empresa constructora, el que será administrado por un usuario de perfil de administrador del sistema, quien en conjunto con el alumno tesista serán los responsables de crear la carga inicial de la información en el sistema, además cargará en los equipos de los clientes la información que deba contar una nueva bodega en obra, también creará el registro de usuarios y guiará a los encargados de bodegas en el uso del sistema, en donde ellos deberán registrar cada movimiento de artículos HIM producido en las bodegas asignadas a sus cargo y llevarán un riguroso registro de los préstamos de las herramientas a los trabajadores autorizados para el uso de éstas.

EQUIPO DE TRABAJO

- El equipo de trabajo y personas que utilizarán el sistema y aportarán en la contribución explícita del alumno para resolver el problema.

Persona	Labor
Desarrollador	El alumno tesista, será quien se encargará de recoger los requerimientos para el sistema, hacer el modelamiento y diseño de la Base de Datos, crear las distintas interfaces de Usuario, programar las funcionalidades y realizar las pruebas de integración y pruebas de software del sistema, con el apoyo teórico y en ocasiones práctico de la profesora

	Patrocinante.
Usuarios	El personal que utilizará este sistema serán todos los encargados de las distintas bodegas de la empresa constructora, junto con el jefe de adquisiciones quien podrá planificar los flujos y prioridad de los artículos HIM a adquirir. El jefe de administración quien podrá hacer consultas acerca de los consolidados de artículos HIM con los que cuenta la empresa constructora para evaluar requerimientos futuros. Además el ingeniero calculista podrá utilizar este sistema para obtener información si se cumple cabalmente con la programación definida antes del inicio de cada obra.

Tabla 1: Definición del equipo de trabajo

3.2 Justificación

3.2.1 Situación sin proyecto

Si el proyecto no se logra finalizar, la empresa constructora y más específicamente la administración de éstas seguirían sin tener un registro y control claro ni actualizado de las necesidades de herramientas, insumos y materiales (HIM) con los que se cuenta y/o debe contar en las distintas obras

que la empresa ejecuta, junto con esto los jefes de obra o capataces de estas obras seguirán llevando control en formularios manuales y la posterior transcripción en planillas Excel, lo que conlleva a errores en el ingreso de datos o inexactitudes, tanto en cantidades como en fechas.

Sin la aplicación que desarrollará el alumno en su proyecto de tesis no existiría una actualizada programación de los pedidos de herramientas, insumos y materiales que hacen falta en las obras o etapas de estas obras que se ejecuten.

3.2.2 Situación con proyecto

Con la aplicación e implantación del sistema que desarrollará el alumno dentro de su proyecto de tesis y la utilización por parte de los distintos entes involucrados, las Empresas Constructoras, propiedad de Nilson Schwerter, podrán contar con mayor confiabilidad y oportunidad del real stock de sus herramientas, insumos y materiales con los que cuentan, tanto en su bodega central como en sus bodegas en las distintas obras que se ejecutan.

Con el desarrollo de este proyecto de tesis, las empresas también podrán controlar de mejor manera la utilización y lugares en donde se encuentran

algunos de sus activos fijos, como lo son las herramientas, que no necesariamente deben ser adquiridas en cada obra que se realiza.

3.3 Delimitación

Dentro del desarrollo del Sistema de Control HIM el alumno tesista no abarcará la individualización exclusiva de todas las herramientas, insumos y materiales, ya que esto implicaría identificar de manera innecesaria cada artículo o cada lote de artículos, además de identificar cada herramienta con algún código único, lo que no es necesario puesto que las herramientas que se utilizan siempre deben estar en excelentes condiciones para cumplir con el objetivo por el cual fueron adquiridas.

4 Metodología

Para desarrollar el Sistema de Control de HIM el alumno tesista ha escogido una de las metodologías ágiles de programación, en donde una de sus principales características es que esta es una metodología “liviana” que pasa por alto la utilización de elaborados casos de uso, la exhaustiva definición de requerimientos y la producción de una extensa documentación.

La elección de esta metodología apunta también a sus otras características que vale la pena mencionar, estas son:

- Metodología creada a base de prueba y error.
- Énfasis en el desarrollo del software más que una buena documentación.
- Empieza en pequeño y añade funcionalidad con retroalimentación continua.
- No introduce funcionalidades antes de que sean necesarias.
- El cliente o el usuario se convierte en miembro activo del mismo equipo de diseño del sistema.
- Su utilidad es medida con cuatro valores:

- ✓ Simplicidad: La simplicidad consiste en desarrollar sólo el sistema que realmente se necesita. Implica resolver en cada momento sólo las necesidades actuales.
- ✓ Comunicación: XP pone en comunicación directa y continua a clientes y desarrolladores. El cliente se integra en el equipo para establecer prioridades y resolver dudas. De esta forma ve el avance día a día, y es posible ajustar la agencia y las funcionalidades de forma consecuente
- ✓ Feedback: una metodología basada en el desarrollo incremental iterativo de pequeñas partes, con entregas y pruebas frecuentes y continuas, proporcionan un flujo de retroinformación valioso para detectar los problemas o desviaciones.
- ✓ Coraje valentía: implica saber tomar decisiones difíciles, reparar un error cuando se detecta.

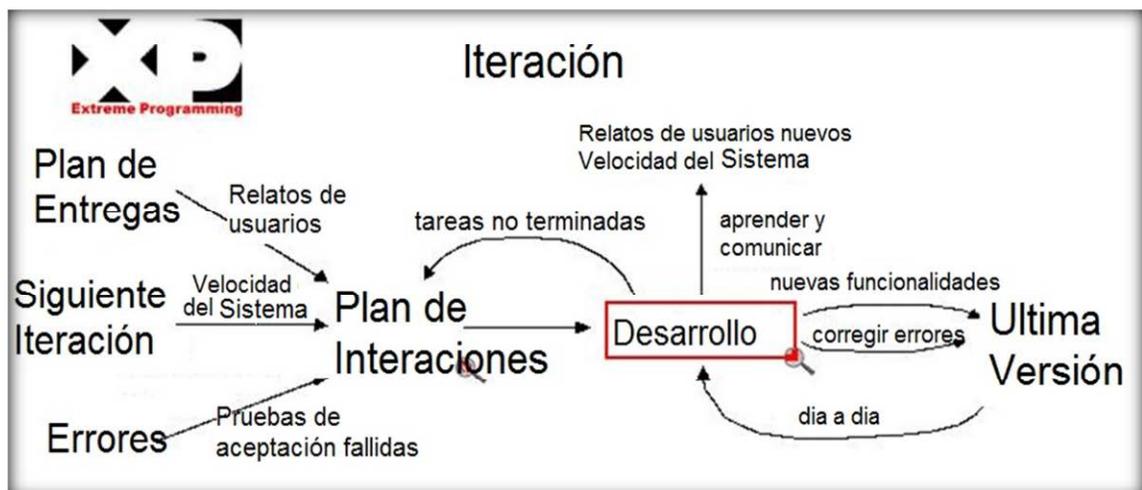


Figura 1: Diagrama de Metodología XP

Como la metodología XP no indica cómo diseñar una base de datos, para realizar esta actividad, se utilizará la Metodología “Sistemas de Bases de Datos”, la que se describe en el libro de [Thomas Conolly & Carolyn Begg, Año 2005]. Este modelo está dividido en 3 etapas: desarrollo del Modelo Conceptual, Lógico y Físico, para luego finalizar con la generación del código de la Base de Datos. Este modelo es uno de los más utilizados, ya que proporciona una terminología adecuada para las primeras fases de diseño y un método para discriminar entre entidad y atributo de entidad.

4.1 Iteraciones

Para el desarrollo de este proyecto, el alumno tesista realizará tres iteraciones. A continuación, se explica brevemente de que se tratará cada una de ellas:

1. La primera iteración será de vital importancia para generar una buena base del proyecto. Se comenzará con una reunión entre la Jefa Administrativa, la Jefa de Adquisiciones (clientes) de la empresa constructora y el alumno, quienes plantearán y detallarán los requerimientos del sistema. En esta reunión el alumno captará la idea central del problema, junto con un modelamiento inicial de la solución y la definición del alcance del Sistema de

Control HIM, generada por el alumno tesista. Se realizará el estudio del proyecto y del modelamiento inicial de éste, se probarán algunas posibles soluciones pudiendo ir detectando errores los que se irán corrigiendo en las etapas siguientes. Una vez finalizada esta iteración se deberá contar con un esquema lo suficientemente robusto para ya ser evaluado por el cliente, ya sea la Jefa de Administración y/o la Jefa de Adquisiciones.

2. En esta segunda parte, luego del estudio del esquema anteriormente generado por la parte demandante del sistema y luego del cumplimiento de una nueva reunión para avanzar en los detalles del sistema se verán los avances del proyecto para ir realizando los cambios se sean necesarios y afinando los aciertos, previo visto bueno del cliente. Además, se deben estudiar e incorporar posibles nuevas funcionalidades si son necesarias, tomando en cuenta el tiempo que pueda conllevar estas posibles modificaciones. Una vez finalizada esta etapa, se tiene una nueva versión del sistema de control HIM, la cual se entrega a las pruebas del cliente previa revisión detallada de las modificaciones realizadas, con lo que se da por finalizada esta segunda iteración.

3. En esta última de las iteraciones se concentrarán los esfuerzos en solucionar pequeños problemas detectados en la segunda iteración y enfocándose a lo ya desarrollado, cabe hacer presente que jamás se dejan

de lado las críticas emitidas por el cliente. Se afinará cada funcionalidad del sistema. El tiempo de estudiar la solución ya ha finalizado, sólo se aplicarán mejoras y enfatizar en concretar los requerimientos. Finalizada esta iteración se debe entregar el Sistema de Control HIM en un estado final y funcional.

5 Recursos

A continuación se detallan los recursos utilizados:

5.1 Hardware

Para el proceso de desarrollo del Sistema el alumno tesista contará con el siguiente Hardware:

- Estación de Trabajo
 - *Procesador: Intel Pentium Dual-Core 1,6 Ghz.*
 - *Memoria RAM: 2 Gb*
 - *Disco Duro: 160 Gb*
 - *Red: PCI-E Lan-Express AS 802.11g*

Para el proceso de implementación y producción del proyecto, se trabajará con un servidor, en el cual se instalará la Base de Datos y será el equipo que contará con respaldo eléctrico y respaldo de datos controlado y periódico.

- Equipo Servidor
 - *Procesador:* Intel(R) Dual Core 2 Ghz.
 - *Memoria RAM:* 4 Gb
 - *Disco Duro:* 320 Gb
 - *Red:* Ethernet RTL 8139 C

- Equipos Clientes, serán del tipo
 - *Procesador:* Intel(R) Dual Core 1,6 Ghz.
 - *Memoria RAM:* 2 Gb
 - *Disco Duro:* 120 Gb

5.2 Software

El siguiente será el software que utilizarán los distintos equipos computacionales que participarán en el Sistema:

Características del Software:

- Estación de Desarrollo
 - *Windows 7 Professional 32 Bits*

- *NetFramework 3.5*
- *SQL Server 2008 R2*
- *Power Designer 15*
- *Power Builder 12*

➤ **Equipo Servidor**

- *Windows 7 Professional 32 Bits*
- *NetFramework 3.5 o superior*
- *SQL Server 2008 R2*

➤ **Equipos Clientes, serán del tipo**

- *Windows 7 Home Basic o superior*
- *NetFramework 3.5 o superior*

6 Análisis del Sistema

La etapa de análisis contempla el estudio de las actividades a realizar, definir los alcances del sistema, los límites y las definiciones de éste, junto con la recolección y la detallada examinación de la información que proporcionará los requerimientos que se deberán cumplir.

6.1 Recopilación de requerimientos

El trabajo a desarrollar se centrará en lograr un sistema con un grado aceptable de eficiencia y eficacia. Bajo esta premisa se han definido las actividades escogidas en la metodología adoptada en el transcurso del seminario, detallada con anterioridad en el Capítulo de “Metodología”, abordada en el Capítulo Cuarto.

El trabajo consiste principalmente en abordar el sistema enfocándose en la problemática de controlar los inventarios de herramientas, insumos y materiales (HIM), en el lugar que éstos se encuentren, ya sea Bodega Central o alguna bodega de las distintas obras que puedan existir.

Ante este escenario, las actividades diferencian el trabajo en dos frentes de acción que se desarrollan en paralelo en las etapas de análisis y diseño. La implementación, pruebas e implantación se realizan sobre el Sistema de Control de obras.

Los requerimientos han sido recopilados a través de una serie de reuniones preestablecidas entre el alumno tesista y las jefas de administración, y jefa de adquisiciones, quienes para este proyecto serán “El Cliente”. Estas reuniones son de vital importancia en la metodología de desarrollo XP. Se trata de reuniones en las que se reúnen los integrantes del equipo, en donde el Cliente debe explicitar de manera clara y precisa todos sus requerimientos. Con esta primera reunión se toma una primera lista de requerimientos y se acuerda la próxima reunión para analizar el grado de avance del proyecto.

El alumno tesista debe ser capaz de analizar e interpretar lo solicitado, creando pequeños bosquejos para luego modelar el diseño más consolidado y funcional del sistema.

En una nueva reunión se presentan los avances en donde se escuchan las críticas y sugerencias, con la posibilidad de analizar posibles modificaciones o mejoras al sistema. Por otra parte, el Cliente es capaz de requerir nuevas

funcionalidades, donde se replican los pasos previos, se ofrecen soluciones y se escoge la mejor alternativa para llevarlas adelante.

Se realizaron tres reuniones en total, en donde en dos de ellas se dio énfasis a la toma de requerimientos y presentación de avances. La toma de requerimientos fue de fácil asimilación por el alumno tesista ya que, entre sus conocimientos previos, este alumno ha desarrollado actividades en la industria de la construcción, lo que permitió con mayor facilidad captar en un corto periodo los requerimientos del Cliente.

Cabe hacer mención que este es uno de los pasos más importantes dentro del análisis del sistema para poder entender bien y con claridad lo que el Cliente solicita.

A continuación, se describen las reuniones más importantes, las que permitieron obtener los requerimientos.

Toma de Requerimientos – Reunión 1

1.	<p>Se solicita el detalle de los problemas que se quieren solucionar con el Sistema de Control HIM</p> <p>Se explica que con los métodos usados por la empresa jamás consiguen de manera oportuna la información de los stocks.</p> <p>Se explica que los administradores de las empresas, necesitan poder contar con información más exacta y oportuna.</p> <p>Se indica que se necesita contar con el stock real de las herramientas y saber el número de ellas en las distintas obras.</p> <p>Se necesitan obtener diversos informes, como ingresos y egresos de artículos, junto con el control de los traslados que se originan entre y hacia las obras.</p>
2.	<p>Identificación de los Usuarios (Clientes)</p> <p>Jefa Administrativa, quien define los requerimientos iniciales del sistema.</p>
3.	<p>Administración básica del sistema.</p> <p>Se definen los distintos perfiles del uso del sistema.</p>
4.	<p>Interfaz de Usuario</p> <p>Se definen formas de ingreso al sistema y el estilo de interfaz de usuario para ingreso de movimientos</p>
5.	<p>Informes.</p> <p>Se definen los datos que deben contener los informes prioritarios.</p>

Tabla 2: Toma de requerimientos - Reunión 1

Toma de Requerimientos – Reunión 2

1.	<p>Se escuchan las opiniones de Usuarios</p> <p>Se indica que además es necesario llevar el control de la caja chica que se asigna a las obras, en donde el descontrol que se produce es debido a que se adquieren productos en las distintas obras y ese control no pasa por la bodega central.</p> <p>Se definen los datos que irán en los archivos exportables/importables de los movimientos.</p> <p>Se explica que en muchas ocasiones no se cuenta con internet en los lugares donde se desarrollan la mayoría de las obras, por lo que el sistema debe suplir este problema.</p> <p>Se muestra primer prototipo del sistema en donde se indica que el sistema va bien encaminado y está cumpliendo los objetivos para los cuales fue concebido.</p>
2.	<p>Identificación de los Usuarios (Clientes)</p> <p>Se incorpora la Jefa de Adquisiciones, quien avala los requerimientos y agrega otros requerimientos.</p>
3.	<p>Administración básica del sistema.</p> <p>Se mantiene la definición de los distintos perfiles de usuarios del sistema.</p>
4.	<p>Interfaz de Usuario</p> <p>Se aprueban las formas de ingreso al sistema y el estilo de interfaz de</p>

	ingreso de movimientos.
5.	Informes.
	Aun no se generan informes en esta etapa para ser revisados por los usuarios.

Tabla 3: Toma de requerimientos - Reunión 2

Toma de Requerimientos – Reunión 3

1.	Se escuchan las opiniones de usuarios
	Se hacen las numerosas pruebas por los usuarios, quienes solicitan facilitar el acto de exportar e importar archivos.
2.	Identificación de los Usuarios (Clientes)
	Se incorporan los usuarios finales como algunos capataces y encargados de bodegas.
3.	Administración básica del sistema.
	Se crean todos los usuarios iniciales que poseerá el sistema
4.	Interfaz de Usuario
	Se aprueban las interfaces de usuario del sistema.
5.	Informes.
	Se revisan y corrigen los informes que se han generado.

Tabla 4: Toma de requerimientos - Reunión 3

Al finalizar las reuniones realizadas en la iteración correspondiente según la metodología de Extreme Programming (XP), se entregaron avances reales, concretos y sólidos, tanto a modo de esquema como elementos de uso funcional que el Cliente y los usuarios pudieron ir probando, junto con ello se mejoraron detalles observados y se realizó la nueva toma de requerimientos.

A continuación, se detalla un resumen con los requerimientos más importantes captados en las primeras reuniones.

Resumen de Requerimientos		
ID	Detalle	Tipo
1.	Identificar a los usuarios del sistema.	Funcional
2.	Identificar y asignar privilegios para cada usuario según su cargo y funcionalidad dentro de la Empresa.	Funcional
3.	Administración de la información asociada a los Maestros	Funcional
4.	Registro de Movimientos	Funcional
5.	Eliminación de Movimientos	Funcional
6.	Consultas de Stocks	Funcional
7.	Exportar Archivo de Movimientos	Funcional
8.	Importar Archivo de Movimientos	Funcional
9.	Registrar Prestamos y Devoluciones de Herramientas	Funcional

10.	Restringir el acceso al sistema permitiendo sólo el personal autorizado	Seguridad
11.	Computadores clientes capaces de correr NetFramework 3.5 o Superior.	Implantación

Tabla 5: Resumen de Requerimientos

6.1.1 Modelo de Casos de Uso

Utilizando el lenguaje de modelamiento de sistemas de software, llamado UML¹, ha sido posible crear una representación gráfica de los casos de uso del sistema, las que se muestran a continuación.

En el Modelo del Plan de Negociación de los procesos que se grafica a continuación, se describen las acciones requeridas para la ejecución del proyecto, incluyendo los registros de movimientos, como el control en la recepción y en las salidas, generación de archivos, y administración de los maestros que serán necesarios para asegurar que todos los outputs del proceso sean controlados.

¹ UML: Unified Modeling Language o Lenguaje Unificado de Modelado. Es un lenguaje de modelamiento de sistemas software.

Modelo del Plan de Negocio

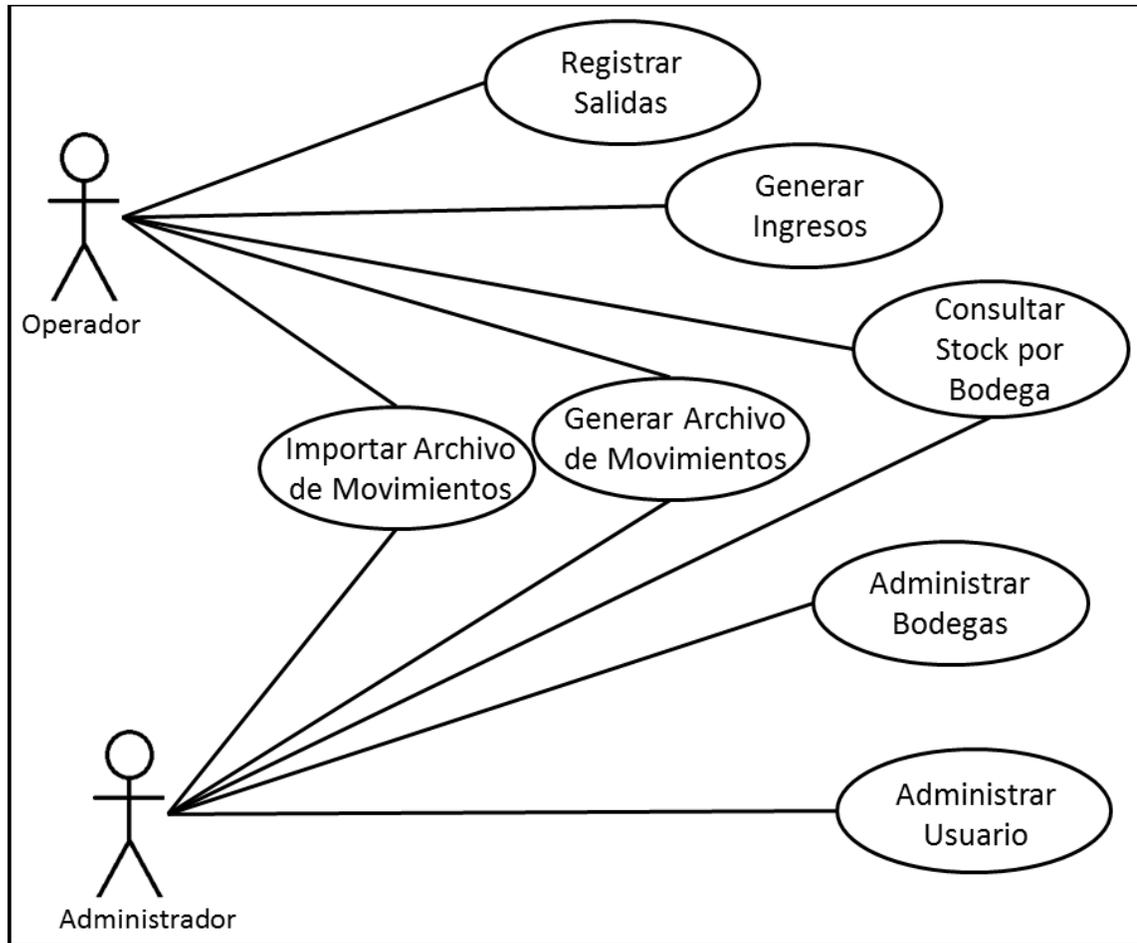


Figura 2: Diagrama de Casos de Uso – Modelo Plan de Negocio

En el Modelo de Casos de Uso que se muestra a continuación el alumno tesista, en su calidad de desarrollador del software y los clientes llegan a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema.

El Modelo de Casos de Uso sirve como acuerdo entre clientes y desarrolladores, el que proporciona la entrada fundamental para el análisis, el diseño y las pruebas.

Modelo de Casos de Uso

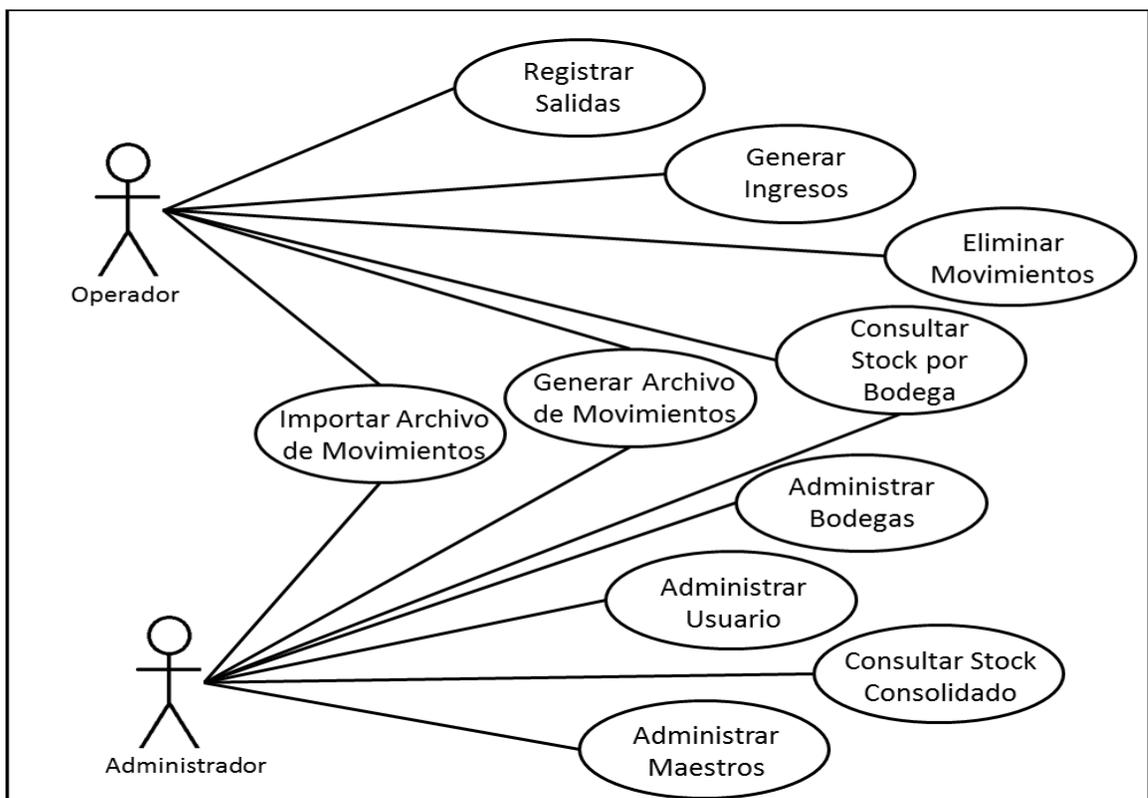


Figura 3: Diagrama de Casos de Uso – Modelos de Casos de Uso

6.1.2 Breve Descripción de los Casos de Uso

1. **Generar Ingresos:** Esta acción se realizará cuando el encargado de la bodega reciba artículos en su respectiva bodega, ya sean estos ingresos producidos por compra directa en obra, por traslado desde otra bodega o por entrega directa del proveedor a solicitud de la Jefa de Adquisiciones de la empresa.
2. **Registrar Salidas:** Esta acción se realizará cuando el encargado de la bodega tenga que hacer una entrega de algún artículo existente en su bodega, esta acción se produce por un préstamo de herramientas, reciba artículos en su respectiva bodega, ya sean estos ingresos producidos por compra directa en obra, por traslado desde otra bodega o por entrega directa del proveedor a solicitud de la Jefa de Adquisiciones de la empresa.
3. **Eliminar Movimientos:** Esta acción se ejecutará cuando se hayan ingresados datos erróneos y se desee anular el movimiento, ya sea de ingreso o de salida de artículos.

4. **Consultar Stock por Bodega:** En esta acción se podrán hacer las consultas de stock por algún artículo determinado o por el total de artículos existente en la bodega.

5. **Generar Archivo de Movimientos:** Esta operación se realizará cada vez que se necesite exportar el la información con los movimientos de entrada, salida y los stocks de los artículos existentes en la bodega de trabajo. (Local)

6. **Importar Archivo de Movimientos:** Esta operación se realizará cada vez que se necesite cargar o importar la información con los movimientos de entrada, salida y los stocks de los artículos existentes en una bodega ajena. Diseñada principalmente para que el Administrador del Sistema, obtenga información de otras bodegas.

7. **Administrar Bodegas:** Esta operación está definida para crear, eliminar, cambiar estado (baja o alta) a las bodegas que se van incorporando a la empresa producto de nuevas obras.

8. **Administrar Usuarios:** Esta operación está definida para crear, eliminar, cambiar estado (baja o alta) a los usuarios que utilizarán el

sistema y otros que se valdrán del mismo para la ejecución de las obras de la empresa constructora.

9. **Administrar Maestros:** Esta operación está definida para administrar la información contenida en otras tablas en general, tales como artículo, proveedores, empresa, comuna, localidad, entre otras.

10. **Consultar Stock Consolidado:** Esta operación se realizará cuando el administrador desee consultar el stock total del universo de bodegas, el cual se deberá hacer una vez se hayan importado los archivos de movimientos de las bodegas que se desee incluir en este consolidado.

6.1.3 Herramientas de trabajo

Al finalizar la primera reunión con el cliente y tras el detallado estudio y análisis de cada uno de los requerimientos, el alumno expone las herramientas de desarrollo a utilizar y se le explica al cliente que éstas son herramientas que el alumno domina y son con las que se trabajará en el sistema.

Más en detalle y sin ir a la especificación técnica con el cliente, se expone el desglose de las herramientas de software utilizadas para el desarrollo del sistema de control HIM, las que para el diseño de los modelos Conceptuales y Físicos de datos se utiliza Power Designer versión 15, esta herramienta ayuda a generar el script con el que se genera la base de datos, la que se montará sobre SQL Server 2008 2da. revisión, luego para el proceso de programación del sistema se trabajará con Power Builder 12.5, principalmente por ser muy cómodo en el acceso a las tablas de datos a través de los DataWindow, también se integra de manera robusta y confiable con NetFramework. Se trata de un marco de trabajo de Microsoft que permite una programación ágil gracias a la integración de variadas tecnologías. Por esto, la utilización de la herramienta Power Builder 12.5 que trabaja con este NetFramework 3.5 o superior.

6.2 Definición del sistema

En este proyecto de tesis el alumno desarrollará una aplicación, la que posee como principal objetivo, implementar un sistema de control del flujo de movimientos y stock de herramientas, insumos y materiales en las diversas bodegas con las que cuentan las empresas constructoras Schwerter.

6.3 Módulos del Sistema

El sistema está compuesto por los siguientes módulos:

- **Módulo de Movimientos:** En este módulo se realizará la ejecución de los registros de los ingresos y salidas de artículos de las bodegas, procesos definidos anteriormente, los que el encargado de cada bodega se encargará de registrar.
- **Módulo de Consultas de Stock:** Este módulo ha sido creado para realizar las consultas de stocks existentes en la bodega de trabajo, se hace presente que sólo un usuario con perfil de Administrador del sistema podrá obtener una consulta de stock consolidado del universo de bodegas de cada empresa.
- **Módulo de Administración de Maestros:** El objetivo de este módulo es la administración de la información contenida en todas las tablas maestras del sistema.
- **Módulo de Exportación e Importación de Archivos:** Este módulo ha sido diseñado con el objeto de agrupar el proceso de exportar e importar hacia

y desde un archivo plano los movimientos generados junto con los stocks existentes en la bodega.

6.4 Seguridad del Sistema

La seguridad del sistema estará limitada por el acceso a las estaciones de trabajo, las cuales estarán previamente autorizadas por el administrador en donde se cargará el sistema considerando el destino de dicha estación, luego de eso, se creó una interfaz de inicio de sesión para la cual el usuario debe ingresar su nombre de usuario y una contraseña válida, este usuario y su respectiva contraseña han sido creados previamente por el administrador del sistema.

6.5 Usuarios del Sistema

Los usuarios del sistema serán de tres tipos principalmente, estos tipos de perfil son:

- Perfil Administrador: Quien podrá acceder a modificar o eliminar registros maestros, crear, eliminar o modificar datos de usuarios y realizar las consultas de stocks consolidados en todas las bodegas del universo de bodegas existentes, este usuario tendrá un perfil

plenipotenciario ya que podrá incluso autorizar traslados y dar de baja artículos HIM.

- Perfil Operador: Es quien estará a cargo de ingresar los movimientos existentes en una bodega, así como hacer consultas a nivel local del stock existente en la bodega asignada.

- Perfil Jefe de Obra: Este tipo de usuarios deberá ingresar al sistema cuando se necesite realizar un traslado o dar de baja artículos existentes en la bodega de trabajo, ya que la responsabilidad de su cargo lo faculta a tomar decisiones de este tipo.

6.6 Modelo de 3 Capas

Cabe señalar que en el caso de la instalación del Sistema de Control HIM en las bodegas de las obras, no se realizarán conexiones de red, dejando en un solo equipo todas a las aplicaciones requeridas.

Básicamente el sistema consistirá en tres módulos que interactúan entre sí, distribuidos en tres capas claramente visibles:

1. Capa de Presentación.
2. Capa de Negocios.
3. Capa de Servicios de Datos.

Mediante este tipo y cantidad de capas el sistema permitirá un uso fácil para todos los usuarios a partir de la información que entrará al sistema, la capa de presentación será la encargada de interactuar con el usuario y la aplicación, la capa de negocios generará los cálculos o procesos de negocios y la capa de servicios de datos se encargará de manipular y ejecutar las reglas de datos relacional.

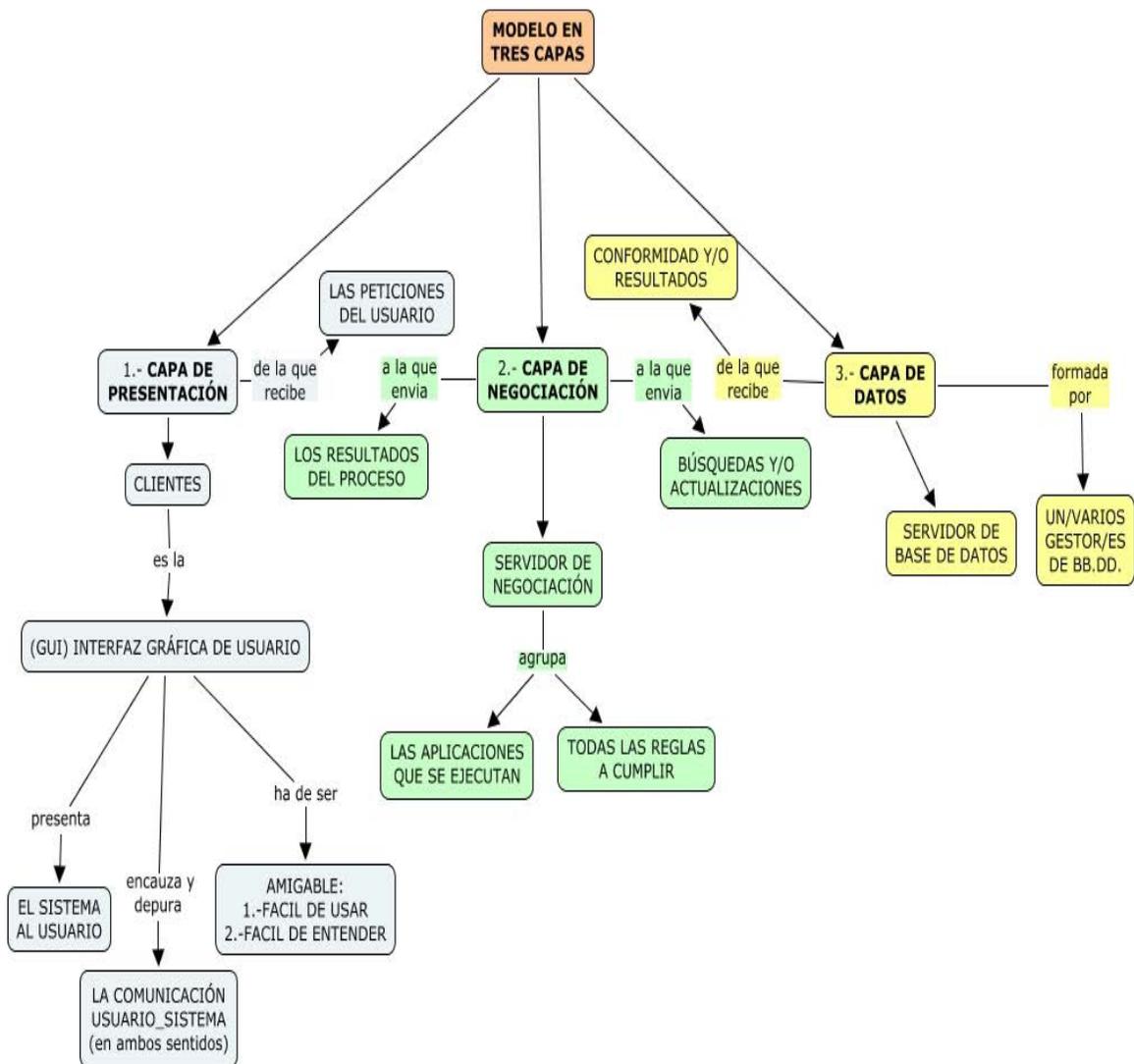


Figura 4: Diagrama capa de datos

En resumen el sistema de control deberá almacenar la información de los artículos que ingresan y salen de las bodegas y avisos o informes de los niveles de stocks.

- **Capa de Presentación:** Es la que se encarga de que el sistema interactúe con el usuario y viceversa, muestra el sistema al usuario, le presenta la información y obtiene la información del usuario en un mínimo de proceso. Es conocida como interfaz gráfica y debe tener la característica de ser amigable, o sea, entendible y fácil de usar para el usuario. Esta capa se comunica únicamente con la capa intermedia o de negociación.
- **Capa de Negociación:** Es donde residen las funciones que se ejecutan, se reciben las peticiones del usuario, se procesa la información y se envían las respuestas tras el proceso. Se denomina capa de negociación o capa de lógica del negocio, porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de acceso a datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.
- **Capa de Datos:** Esta capa es la encargada de almacenar los datos del sistema y de los usuarios. Su función es almacenar y devolver datos a la capa de negocio. En una arquitectura de tres capas, esta capa es la única que puede acceder a los mismos. En este caso está

formada SQL Server 2008 r2, potente gestor de bases de datos, localizado en el servidor y en las estaciones de clientes.

6.7 Entorno Tecnológico – Hardware – Software

Se requirió de un entorno tecnológico estable y seguro para el buen funcionamiento del sistema de control HIM, en este caso se utilizará el Hardware y Software identificados en el Capítulo Nro. 5, añadiendo además que este sistema está siendo construido de manera de simplificarlo para usuarios de bajo manejo computacional, por lo cual, el entorno tecnológico debe ser, no muy complejo, pero sí, muy estable y seguro como se indica al comienzo de este punto.

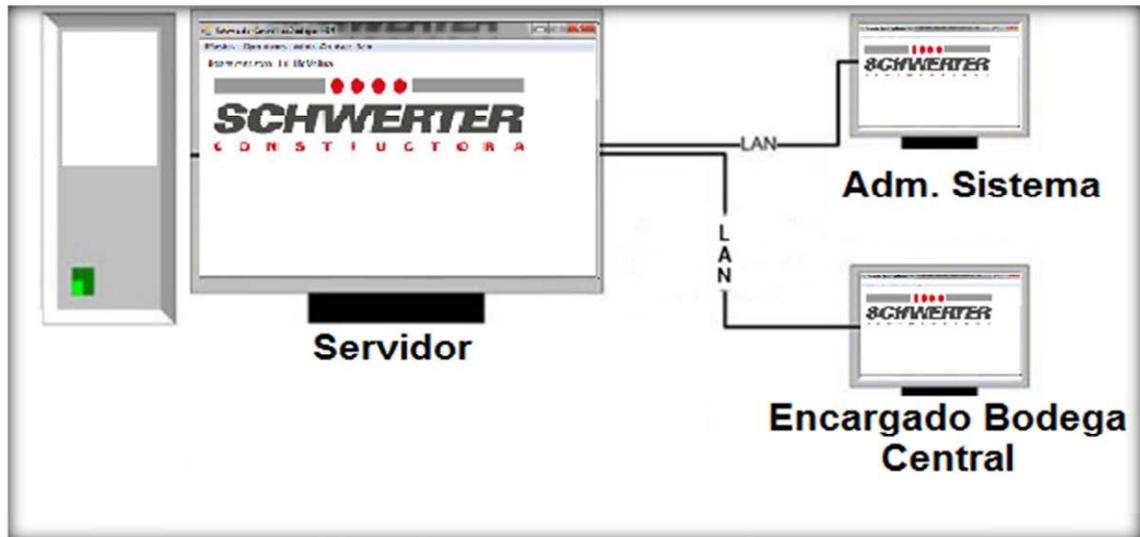


Figura 5: Diagrama capa de datos



Figura 6: Entorno Tecnológico

Hay que considerar que en este proyecto se dan 2 casos de acceso a la información, en donde en el primero (Ilustración 5), se establece un servidor el que permanece en las instalaciones de la empresa constructora, administrado a

nivel local por el administrador del Sistema y siendo accedido por el encargado de bodega de la Bodega Central y en el segundo caso (Figura N° 6), se instalan todos los componentes en una sola estación de trabajo, ya que como se ha explicado, este sistema debe ser capaz de funcionar de manera autónoma y en lugares de difícil acceso a una red proveedora de Internet.

7 Diseño del Sistema

7.1 Diseño de la Base de Datos

Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada. Puesto que un diseño correcto es esencial para lograr los objetivos fijados para la base de datos, parece lógico emplear el tiempo que sea necesario en un buen diseño ya que, se necesita que la base de datos pueda ser modificable y posea una fácil adaptación a algún futuro cambio de requerimientos

En general, el objetivo del diseño de una base de datos es generar un conjunto de relaciones que permitan almacenar la información con un mínimo de redundancia, pero que a la vez faciliten la recuperación de la información.

El diseño de la Base de Datos adoptado en este seminario de tesis, tiene su origen en el Libro “Sistemas de Bases de Datos” [THOMAS M. CONNOLLY y CAROLYN E. BEGG; Año 2005], el cual involucra los procesos de: diseño conceptual, diseño lógico y diseño físico de una base de datos.

7.1.1 Diseño Conceptual

El objetivo del diseño conceptual es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar esta información.

El diseño conceptual parte de las especificaciones de requisitos de usuario y su resultado es una descripción de alto nivel de la estructura de la base de datos, independientemente del SGBD o DBMS que se vaya a utilizar para manipularla.

Para modelar conceptualmente es posible utilizar varios Modelos de Datos, un modelo práctico para ilustrar el diseño conceptual es el modelo entidad relación.

En este caso se mostrará el modelo conceptual para una mayor claridad:

- Entidades.
- Relaciones.
- Atributos.

- Dominio de Atributos.
- Claves Candidatas y Claves Primarias.

7.1.1.1 Identificación de Entidades

Una entidad es una "cosa" u "objeto" del mundo real, con existencia independiente y distinguible de los demás objetos. Cada entidad tiene un conjunto de propiedades y valores que la identifican de forma unívoca.

Teniendo este concepto claro podemos describir las siguientes entidades que se obtuvieron en el proceso de análisis del sistema.

Nº	Nombre Entidad	Descripción	Ocurrencia
1	bodega	Almacena las bodegas que existan en el sistema, las cuales han sido creadas previamente por el administrador del sistema.	Una bodega pertenece sólo a una obra
2	empresa	Contiene el registro de las empresas constructoras de	Una empresa es propietaria de al

		trabajo del sistema.	menos una o más obras.
3	obra	Almacena las obras que existan en el sistema.	Una obra pertenece a una sola empresa.
4	cargo	Almacena los cargos de las personas.	Un cargo lo ostenta uno o más personas.
5	comuna	Almacena las comunas de los proveedores.	Una comuna corresponde a uno o más proveedores.
6	central	Es de tipo bodega cumple con descripción de una bodega.	Sólo existe una central por cada empresa constructora.
7	almacena	Contiene la información de artículos relacionado a una bodega.	Le corresponde uno o más artículos asociados a una o más bodegas.

8	origen	Almacena los posibles orígenes de bodega al generar un movimiento.	Un origen puede tener uno o más documentos.
9	destino	Almacena los posibles destinos de bodega al generar un movimiento.	Un destino puede poseer uno o más documentos.
10	usuario	Contiene la información de los usuarios del sistema de la empresa constructora.	Un usuario digita uno o varios documentos.
11	localidad	Contiene la información de las localidades de los proveedores y bodegas del sistema de la empresa constructora.	Una localidad pertenece a una sola comuna y en ella reside uno o varios proveedores.
12	articulo	Contiene la información de los artículos que se almacenan en las bodegas del sistema de la empresa constructora.	Un artículo se almacena en una o más bodegas.

13	contiene	Almacena la información de los artículos y movimientos que se registran en un documento.	Contiene almacena uno o varios artículos de una o varias bodegas en uno o varios movimientos.
14	proveedor	Contiene la información de los proveedores de las empresas.	Un proveedor reside en una o varias localidades.
15	documento	Almacena la información de los movimientos digitados por los usuarios.	Un documento contiene uno o varios artículos en una o varias bodegas.
16	unidad_medida	Guarda la información de las unidades de medida con las que trabaja el sistema.	Un artículo se asocia a sólo una unidad de medida.
17	persona	Guarda la información de las personas que forman parte del sistema.	Una persona se asocia a un único cargo.

18	consumidor	Almacena la información de todas las personas que solicitan productos de las bodegas.	Un consumidor acepta un único documento por uno o varios artículos que consume.
19	autorizador_salida_traslado	Almacena la información de todas las personas que pueden autorizar un traslado de artículos entre las bodegas del sistema.	Un autorizador_salida_traslado a través de un único documento puede autorizar traslados de uno o varios artículos entre uno o varias bodegas del sistema.
20	solicitante_ingreso_traslado	Almacena la información de todas las personas que pueden solicitar el traslado de artículos entre las bodegas del sistema.	Un solicitante_ingreso_traslado a través de un único documento puede

			solicitar traslados de uno o varios artículos entre uno o varias bodegas del sistema.
21	comprador	Almacena la información de todas las personas que están autorizadas para comprar artículos para las bodegas del sistema.	Un comprador puede realizar compras de uno o varios artículos para una o varias bodegas.
22	transportista	Almacena la información de todas las personas se encuentran como transportistas autorizados por la empresa constructora.	Un transportista puede realizar el traslado de uno o varios artículos entre una o varias bodegas.

Tabla 6: Entidades del Sistema

7.1.1.2 Identificación de las Relaciones

En una base de datos relacional, las relaciones permiten evitar los datos redundantes.

Una relación se establece haciendo coincidir los datos de las columnas de clave, normalmente las columnas con el mismo nombre en ambas tablas. En la mayor parte de los casos, la relación hace coincidir la clave principal de una tabla, que proporciona un identificador único para cada fila, con una entrada de la clave externa de la otra tabla.

Hay tres tipos de relaciones entre tablas. El tipo de relación que se crea depende de cómo se definen las columnas relacionadas.

- **Relación de uno a muchos:** La relación de uno a varios es el tipo más común. En este tipo de relación, una fila de la tabla A puede tener muchas filas coincidentes en la tabla B, pero una fila de la tabla B sólo puede tener una fila coincidente en la tabla A. Una relación uno a varios se crea si sólo una de las columnas relacionadas es una clave principal o tiene una restricción PK.

- **Relación varios a varios:** En una relación varios a varios, una fila de la tabla A puede tener muchas filas coincidentes en la tabla B y viceversa. Este tipo de relación se crea definiendo una tercera tabla, denominada de combinación, cuya clave principal está compuesta de las claves externas de ambas tablas A y B.

- **Relación uno a uno:** En una relación uno a uno, una fila de la tabla A no puede tener más de una fila coincidente en la tabla B y viceversa. Una relación uno a uno se crea si las dos columnas relacionadas son las claves principales o tiene restricciones PK. Este tipo de relación no es común porque la mayor parte de la información relacionada de esta manera estaría toda en una tabla.

Acto seguido, se muestra una tabla en la que aparecen las relaciones de las entidades más relevantes del sistema de control HIM, junto a su **cardinalidad**, la manera correcta de leerla es de izquierda a derecha.

Nº	Entidad	Relación	Entidad	Cardinalidad	Existencia
1	bodega	crea	documento	1:N	M:M
2	bodega	Se_ubica	localidad	N:1	M:M
3	bodega	almacena	articulo	N:N	M:M

4	Empresa	Es_propietario	Obra	1:N	M:M
5	Comuna	Pertenece	Localidad	1:N	M:M
6	Localidad	Reside	Proveedor	N:N	M:M
7	documento	Incluye	Proveedor	1:N	0:M
8	Usuario	Digita	Documento	1:N	M:M
9	documento	Tiene	Origen	N:1	0:M
10	documento	Posee	Destino	N:1	M:M
11	documento	Contiene	articulo	N:N	M:M
12	Unidad_m edida	Se_asocia	articulo	1:N	M:M
13	comprador	firma	documento	1:N	M:0
14	Solicitante _ingreso_tr aslado	solicita	documento	1:N	M:0
15	Autorizado r_salida_tr aslado	autoriza	documento	1:N	M:0
16	transportist a	recibe	documento	1:N	M:0
17	cargo	desempeña	comprador	1:N	M:M
18	consumido	acepta	documento	1:N	M:0

	r				
19	cargo	ostenta	Autorizador _salida_ tras lado	1:N	M:M

Tabla 7: Identificación de las Relaciones

7.1.1.3 Identificación y asociación de atributos con una Entidad o Relación

En esta parte, la metodología indica que se deben identificar los atributos para cada una de las entidades o relación.

En la siguiente tabla, se muestra los principales atributos de las entidades más representativas del modelo, como lo son: *bodega*, *documento*, *artículo* y *persona*.

Bodega			
Atributos	Descripción	Tipo	Valor Nulo
ID_BODEGA	Código único que identifica una bodega	Numeric(18)	No

ID_LOCALIDAD	Código Unico que representa a la localidad	Int	No
NOMBRE_BODEGA	Nombre de la bodega con la cual será identificada en la interfaz	Varchar(40)	No
ESTADO	Estado de actividad de la bodega (activo o pasivo)	char(1)	No
TELEFONO_BODEGA	Teléfono de contacto asignado a la bodega	Varchar(11)	Si

Tabla 8: Identificación y Asociación de atributos de la entidad Bodega

Documento			
Atributos	Descripción	Tipo	Valor Nulo
ID_DOCTO	Código único que identifica el documento originado	Numeric(18)	No
TIPO_MOV	Tipo de movimiento generado Ingreso o Salida	char(1)	No
TIPO_DOCTO	Tipo de documento generado	char(1)	No
FECHA_DOCTO	Fecha en la cual se genera el	DateTime	No

	documento.		
GUIA_DESPACH O	Identificador Numérico de la guía de despacho asociada a algún traslado o ingreso de artículos.	Numeric(18)	Si
PATENTE	Se almacena la patente del vehículo involucrado en el traslado de artículos a las diferentes bodegas.	Varchar(6)	Si
ID_MOV	Código identificador del movimiento de artículos	Numeric(18)	No
FECHA_MOV	Fecha en la cual se realiza el movimiento de artículos.	DateTime	Si
OBSERVACIONE S	Cadena de caracteres asignado para que los usuarios ingresen los comentarios que consideren necesarios	Varchar(300)	Si

Tabla 9: Identificación y Asociación de atributos de la entidad Documento

Articulo			
Atributos	Descripción	Tipo	Valor Nulo
ID_ARTICULO	Código único que identifica a un artículo existente en el sistema	Numeric (18)	No
DESC_ARTICULO	Descripción del detalle del articulo	Varchar (40)	No
TIPO_ARTICULO	Tipo artículo "H", "I" o "M" (Herramienta, Insumo o Material)	Char(1)	No

Tabla 10: Identificación y Asociación de atributos de la entidad articulo

Persona			
Atributos	Descripción	Tipo	Valor Nulo
RUT_PERSONA	Código único que identifica a una persona existente en el sistema, RUT Rol Único Tributario Chileno.	Numeric (9)	No
DV_PERSONA	Digito Verificador del RUT antes ingresado en el sistema	char(1)	No
NOMBRES	Cadena de caracteres que identifica los nombres de una persona ingresada en el sistema.	Varchar (40)	No

APELLIDOS	Cadena de caracteres que identifica los apellidos de una persona ingresada en el sistema.	Varchar (40)	Si
TELEFONO	Teléfono de contacto asociado a la persona antes ingresada en el sistema.	Varchar (11)	Si

Tabla 11: Identificación y Asociación de atributos de la entidad persona

7.1.1.4 Identificar las Claves Primarias y Candidatas.

Una clave primaria es un campo, o una combinación de campos, que identifica en forma única a cada registro. El objetivo de identificar las claves primarias es tener una identificación única por cada registro.

A continuación, se muestra una tabla con la identificación de la clave primaria y candidata para algunas entidades del modelo.

Claves candidatas y primarias		
Nombre Entidad	Claves Candidatas	Claves Primarias
bodega	Id_bodega	Id_bodega

empresa	Rut_empresa	Rut_empresa
obra	Id_comuna	Id_comuna
cargo	Id_cargo	Id_cargo
comuna	Id_comuna	Id_comuna
central	Id_bodega	Id_bodega
almacena	Id_bodega Id_articulo	Id_bodega Id_articulo
origen	Id_bodega	Id_bodega
destino	Id_bodega	Id_bodega
usuario	Id_usuario	Id_usuario
localidad	Id_localidad	Id_localidad
reside	Rut_proveedor Id_localidad	Rut_proveedor Id_localidad
articulo	Id_articulo	Id_articulo
contiene	Id_articulo Id_mov	Id_articulo Id_mov
proveedor	Rut_proveedor	Rut_proveedor
documento	Id_movimiento Id_bodega	Id_movimiento Id_bodega
unidad_medida	Id_unidad_medida	Id_unidad_medida

persona	Rut_persona	Rut_persona
consumidor	Rut_persona	Rut_persona
autorizador_salida_traslado	Rut_persona	Rut_persona
solicitante_ingreso_traslado	Rut_persona	Rut_persona
comprador	Rut_persona	Rut_persona
transportista	Rut_persona	Rut_persona

Tabla 12: Identificación de Claves Candidatas y Primarias

7.1.1.5 Diagrama Modelo Conceptual

Un modelo conceptual se encuentra orientado a la descripción de estructuras de datos y restricciones de integridad. Se usan fundamentalmente durante la etapa de Análisis de un problema dado y están orientados a representar los elementos que intervienen en ese problema y sus relaciones.

En la siguiente ilustración se muestra el diagrama completo del modelo conceptual de la base de datos de sistema de control HIM.

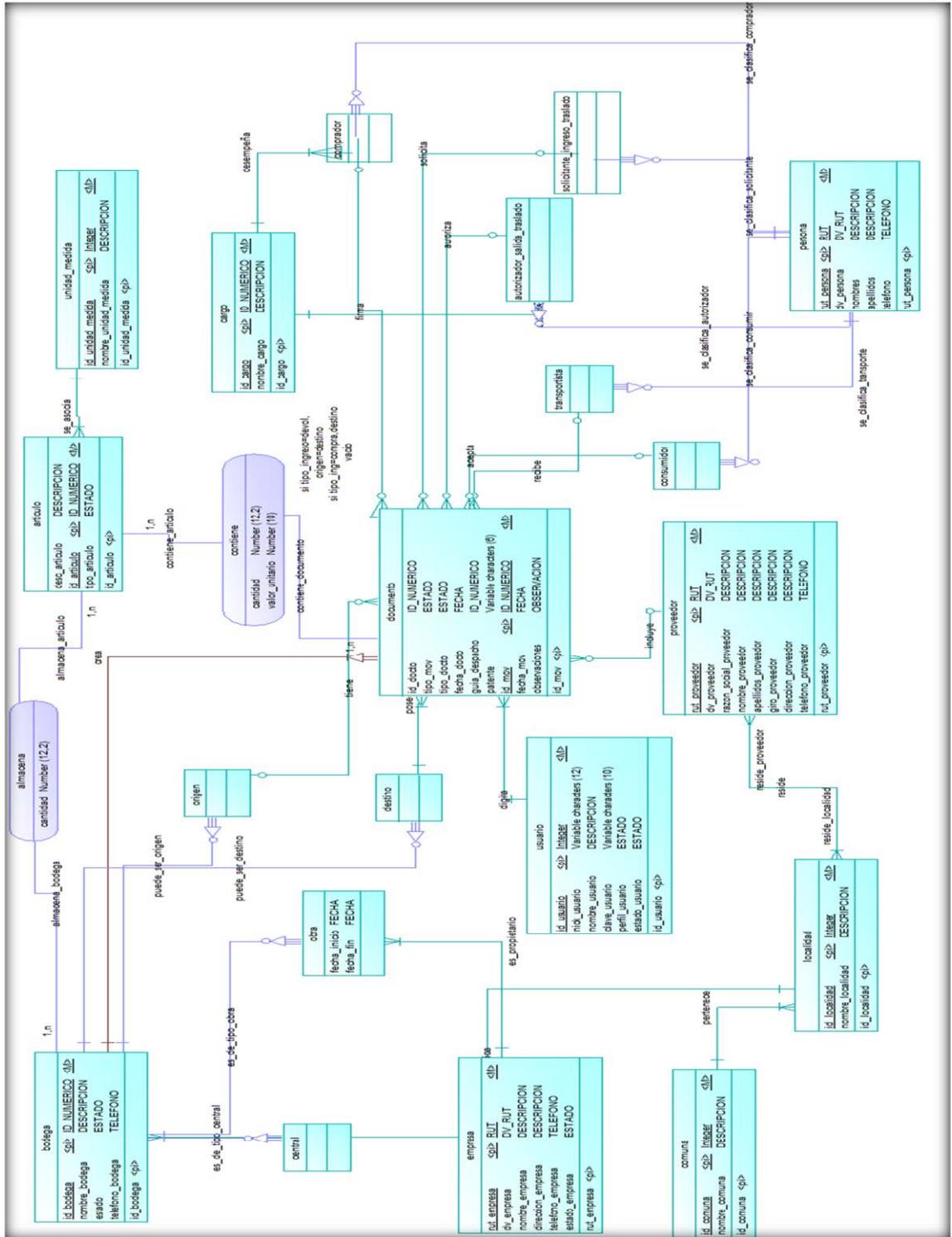


Figura 7: Diagrama Modelo Conceptual de Datos

7.1.2 Diseño Lógico de la Base de Datos

El objetivo de esta etapa es producir una descripción de la implementación de la base de datos. Esta descripción incluye las estructuras de almacenamiento y los métodos de acceso que se utilizarán para conseguir un acceso eficiente a los datos.

Esta descripción es completamente dependiente del SGBD o DBMS específico que se vaya a utilizar, en este caso será Microsoft SQL Server 2008 r2.

Partiendo del diseño conceptual obtenido en la fase anterior, se llega a un diseño lógico. Se transforman las entidades y relaciones obtenidas del modelo anterior en tablas. Para ello se debe usar la normalización.

7.1.2.1 Eliminación de Relaciones N:N

Todas las relación N:N deben ser eliminadas, reemplazándolas por una entidad intermedia y dos relaciones con las entidades originales cuya cardinalidad será 1:N y N:1 respectivamente.

Como se muestra en el capítulo 7.2.1.2 en la Tabla N°7, se generan relaciones N:N en 3 casos, éstas son:

N°	Entidad	Relación	Entidad	Cardinalidad	Existencia
3	bodega	Se_almacena	articulo	N:N	M:M
6	localidad	Se_ubica	proveedor	N:N	M:M
11	documento	Se_Contiene	articulo	N:N	M:M

Tabla 13: Fragmento Tabla N° 7 - Relaciones N:N del capítulo 7.2.1.2

CASO 1:

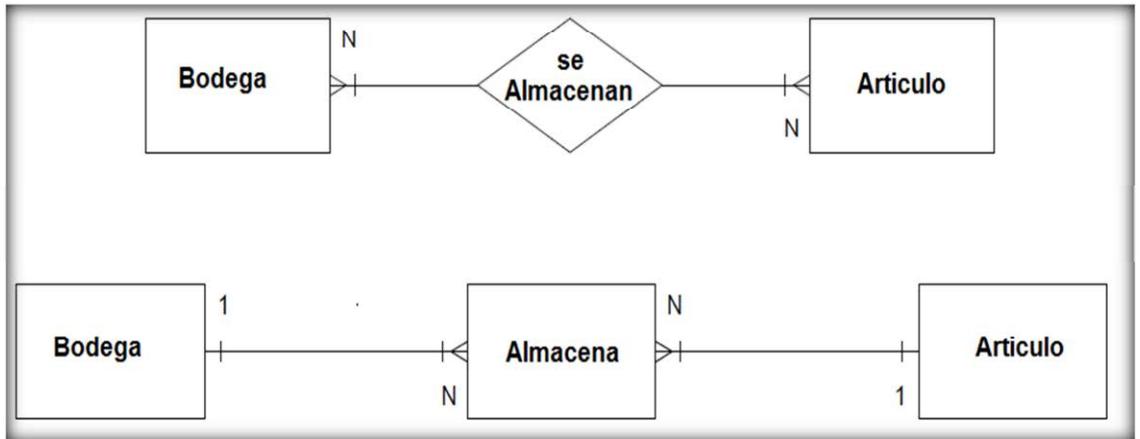


Figura 9: Relación N:N entidad Bodega-Articulo

CASO 2:

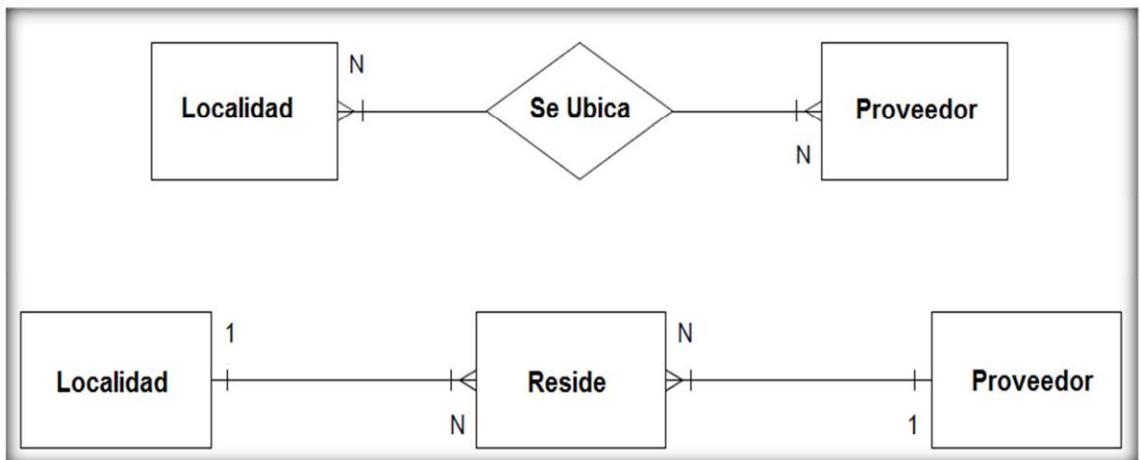


Figura 10: Relación N:N entidad Localidad - Proveedor

CASO 3:

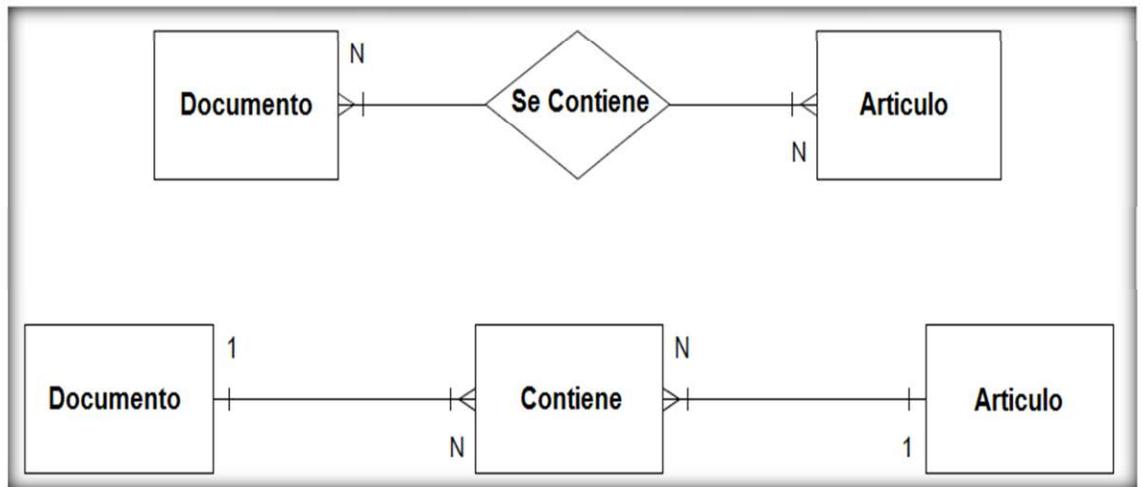


Figura 11: Relación N:N entidad Documento – Artículo

7.1.2.2 Eliminación de Relaciones con Atributos

Toda relación que posea atributos debe ser eliminada sustituyendo cada una de ellas por una nueva entidad y las relaciones binarias correspondientes de esta nueva entidad con las entidades originales.

El modelo diseñado en este seminario para el sistema a desarrollar en no presenta este tipo de relaciones.

7.1.2.3 Derivación de Relaciones al modelo Lógico

En este paso, se obtiene un conjunto de relaciones (tablas) para cada uno de los esquemas lógicos locales en donde se representen las entidades y relaciones entre entidades. Cabe mencionar que dado el carácter de multiempresa del sistema de control HIM se debió incluir en la clave primaria de algunas entidades campos adicionales que asegurasen la no duplicación de ésta.

Esto se realizará con el *Software Power Designer versión 15.0.0.2613* el cual realiza automáticamente el traspaso de claves primarias en el momento de transformar el modelo conceptual en modelo lógico.

7.1.2.4 Validación del Modelo Utilizando Normalización

La normalización es una técnica para producir un conjunto de relaciones con propiedades deseables, dados los requerimientos. Es importante señalar que la normalización ha sido concebida para prevenir anomalías de actualización e inconsistencias de los datos, no para mejorar el rendimiento.

Con la normalización se desarrolla una estructura de datos donde la redundancia es minimizada y la consistencia maximizada.

Principales ventajas de la normalización:

- Evita anomalías en inserciones, modificaciones y borrados.
- Mejora la independencia de datos.
- No establece restricciones artificiales en la estructura de los datos.

La desventaja de esta técnica es que el acceso a la base de datos puede quedar un poco más lento, afectando a su rendimiento.

Los pasos para realizar la normalización son:

- Primera Forma Normal (1FN)
- Segunda Forma Normal (2NF)
- Tercera Forma Normal (3NF)

Estos pasos son detallados a continuación.

7.1.2.4.1 Primera Forma Normal (1FN)

Una tabla está en Primera Forma Normal si:

- Todos los atributos son atómicos. Un atributo es atómico si los elementos del dominio son indivisibles, mínimos.
- La tabla contiene una llave primaria única.
- La llave primaria no contiene atributos nulos.
- No debe existir variación en el número de columnas.
- Los campos no llave deben identificarse por la llave (Dependencia Funcional)
- Debe existir una independencia del orden tanto de las filas como de las columnas, es decir, si los datos cambian de orden no deben cambiar sus significados
- Una tabla no puede tener múltiples valores en cada columna.
- Los datos son atómicos (a cada valor de X le pertenece un valor de Y y viceversa).

Esta forma normal elimina los valores repetidos dentro de una Base de Datos. En las tablas presentes dentro del modelo se observa que no existen atributos multivalorados. Por lo tanto, se puede concluir que el modelo se encuentra en la primera forma normal.

7.1.2.4.2 Segunda Forma Normal (2FN)

Una relación está en segunda forma normal si, y sólo si, está en 1FN ninguno de sus atributos no-principales son funcionalmente dependientes en una parte (subconjunto propio) de una clave primaria (Un atributo no-principal es uno que no pertenece a ninguna clave primaria).

En la derivación del modelo de datos lógico, se aprecia que cada atributo no-principal depende de la clave primaria, por lo que el modelo cumple con la segunda forma normal.

7.1.2.4.3 Tercera Forma Normal (3FN)

Una tabla está en 3NF si y sólo si las dos condiciones siguientes se cumplen:

- La tabla está en la segunda forma normal (2NF)
- Ningún atributo no-primario de la tabla es dependiente transitivamente de una clave primaria

Un atributo no-primario es un atributo que no pertenece a ninguna clave candidata. Una dependencia transitiva es una dependencia funcional $X \rightarrow Z$ en

la cual Z no es inmediatamente dependiente de X, pero sí de un tercer conjunto de atributos Y, que a su vez depende de X. Es decir, $X \rightarrow Z$ por virtud de $X \rightarrow Y$ e $Y \rightarrow Z$.

Una formulación alternativa de la definición dice: Una tabla está en 3NF si y solo si, para cada una de sus dependencias funcionales $X \rightarrow A$, por lo menos una de las condiciones siguientes se mantiene:

- X contiene A, ó
- X es una superclave, ó
- A es un atributo primario (es decir, A está contenido dentro de una clave candidata)

En el caso de la Base de Datos de este proyecto todas las tablas del modelo cumplen con la tercera forma normal, ya que todas se encuentran en la segunda forma normal y se cumple que todos los atributos no clave dependen de manera no transitiva de la clave primaria.

7.1.2.5 Definir Restricciones de Integridad

Las restricciones de integridad son reglas que se quieren imponer para proteger la base de datos, de modo que no pueda llegar a un estado inconsistente.

Existen cinco tipos de restricciones de integridad, estas son:

- Datos Requeridos
- Restricciones de Dominio
- Integridad de Entidades
- Integridad Referencial
- Restricciones de la Empresa

7.1.2.5.1 Datos Requeridos

En esta etapa se deben definir reglas que sirvan para mantener la integridad y consistencia de la base de datos.

Algunos atributos deben contener valores en todo momento, es decir, no admiten nulos. Los atributos que admiten valores nulos y los que admiten

valores nulos están definidos en el punto 7.2.1.3 Identificación y Asociación de Atributos con una Entidad o Relación

7.1.2.5.2 Restricción de Dominio

Todos los atributos tienen un dominio asociado, que es el conjunto de los valores que cada atributo puede tomar, no aceptando valores nulos, por ejemplo, en las claves primarias de cada tabla, pero también hay excepciones como lo son los atributos que complementan una información, ejemplo, en las tablas proveedor o documento, tienen atributos que admiten valores nulos, estos son: teléfono_proveedor y observaciones respectivamente.

Las tablas más importantes con especificación de atributos y datos requeridos se encuentran en la sección *Identificación y Asociación de atributos con una entidad o relación* de este documento.

7.1.2.5.3 Integridad de Entidades

El identificador de una entidad no puede ser nulo, por lo tanto, las claves primarias de las relaciones (tablas) no admiten nulos. Esto se puede validar en el punto sección 7.2.1.4 Identificación de las Claves Primarias y Candidatas.

7.1.2.5.4 Integridad Referencial

La integridad referencial protege las relaciones definidas entre las tablas cuando se crean o se eliminan filas. En SQL Server 2008 como es el caso, la integridad referencial se basa en las relaciones que existen entre claves foráneas y claves primarias, mediante restricciones FOREIGN KEY y CHECK.

La integridad referencial garantiza que los valores de clave sean coherentes en las distintas tablas. Para conseguir esa coherencia, es preciso que no haya referencias a valores inexistentes y que, si cambia el valor de una clave, todas las referencias a ella se cambien en consecuencia en toda la base de datos.

Cuando se exige la integridad referencial, SQL Server 2008 impide a los usuarios:

- Agregar o cambiar filas en una tabla relacionada si no hay ninguna fila asociada en la tabla principal.
- Cambiar valores en una tabla principal que crea filas huérfanas en una tabla relacionada.
- Eliminar filas de una tabla principal cuando hay filas relacionadas coincidentes.

7.1.2.5.5 Restricciones de la Empresa

A nivel de empresa no se determinaron restricciones a nivel de datos ni valores, sólo cabe destacar el requerimiento de restringir el acceso a la zona de administración mediante usuario y contraseña.

7.1.3 Diseño Físico de la Base de Datos

Este diseño depende del servidor, del sistema DBMS, en este caso, empleando el gestor de la Base de Datos, SQL Server 2008 r2, se implementan las tablas de la Base de Datos con sus características, organización y estructuras de almacenamiento interno.

Este nivel debe ser transparente para el usuario, además se especifica la estructura de los datos así como el modo de almacenamiento empleado. Este nivel va a depender de varios factores tanto de Hardware como de Software.

7.1.3.1 Traducir el Esquema Lógico Global

La primera fase del diseño lógico consiste en traducir el esquema lógico global en un esquema que se pueda implementar en el DBMS escogido. Las relaciones base se definen mediante el lenguaje de definición de datos del DBMS. Para ello, se utiliza la información producida durante el diseño lógico: el esquema lógico global y el diccionario de datos.

7.1.3.1.1 Diseñar las Relaciones bases para el DBMS especificado

En esta sección se deben crear las tablas y se deben especificar algunas relaciones del esquema lógico en el lenguaje del DBMS.

Con el objetivo de explicar esta sección, a continuación se desplegará una fracción del **script** generado por el Software de Diseño de la Base de Datos (Power Designer) en la creación de la tabla **documento**.

```
/*=====*/  
  
/* Table: DOCUMENTO */  
  
/*=====*/  
  
create table DOCUMENTO (
```

<i>ID_DOCTO</i>	<i>ID_NUMERICO</i>	<i>null,</i>
<i>TIPO_MOV</i>	<i>ESTADO</i>	<i>null,</i>
<i>TIPO_DOCTO</i>	<i>ESTADO</i>	<i>null,</i>
<i>FECHA_DOCTO</i>	<i>FECHA</i>	<i>null,</i>
<i>GUIA_DESPACHO</i>	<i>ID_NUMERICO</i>	<i>null,</i>
<i>PATENTE</i>	<i>varchar(6)</i>	<i>null,</i>
<i>ID_MOV</i>	<i>ID_NUMERICO</i>	<i>not null,</i>
<i>ID_BODEGA</i>	<i>ID_NUMERICO</i>	<i>not null,</i>
<i>ID_USUARIO</i>	<i>int</i>	<i>not null,</i>
<i>RUT_PROVEEDOR</i>	<i>RUT</i>	<i>null,</i>
<i>RUT_COMPRADOR</i>	<i>RUT</i>	<i>null,</i>
<i>RUT_SOLICITANTE</i>	<i>RUT</i>	<i>null,</i>
<i>RUT_AUTORIZADOR</i>	<i>RUT</i>	<i>null,</i>
<i>RUT_TRANSPORTISTA</i>	<i>RUT</i>	<i>null,</i>
<i>RUT_CONSUMIDOR</i>	<i>RUT</i>	<i>null,</i>
<i>ID_BODEGA_DESTINO</i>	<i>ID_NUMERICO</i>	<i>not null,</i>
<i>ID_BODEGA_ORIGEN</i>	<i>ID_NUMERICO</i>	<i>null,</i>
<i>FECHA_MOV</i>	<i>FECHA</i>	<i>null,</i>
<i>OBSERVACIONES</i>	<i>OBSERVACION</i>	<i>null,</i>
<i>constraint PK_DOCUMENTO</i>	<i>primary key nonclustered</i>	<i>(ID_MOV, ID_BODEGA)</i>

```
)  
go
```

Tabla 14: Fragmento Código Script crea Tabla Documento

7.1.3.1.2 Diseñar las reglas de negocio para el DBMS

Con el modelo de 3 capas definido para el desarrollo de este sistema los datos ingresan correctamente a la capa de datos en donde por medio de código se realizan las inserciones y actualizaciones a la base de datos BodegasHIM, es por esta razón que no fue necesaria la creación de *Triggers*, los cuales son procedimientos que se ejecutan cuando se cumple una condición establecida al realizar una operación (insertar, actualizar o eliminar).

7.1.4 Diseñar la Representación Física

Algunas operaciones que ayudan a gestionar con una mayor eficiencia la base de datos también pueden ser medidas, para ello hay varios factores que se deben tener en consideración:

- **Productividad de Transacciones:** Es el número de transacciones que se quiere procesar en un intervalo de tiempo.

- **Espacio en Disco:** Es la cantidad de espacio en disco que hace falta para los ficheros de la base de datos. Normalmente, el diseñador querrá minimizar este espacio.

- **Tiempo de Respuesta:** Es el tiempo que tarda en ejecutarse una transacción. Desde el punto de vista del usuario, este tiempo debería ser el mínimo posible.

En la siguiente página se muestra el diagrama físico de la base de datos.

7.1.4.1 Analizar Transacciones

Conocer las consultas y las transacciones que se van a ejecutar sobre la base de datos es necesario para realizar un buen diseño físico. Esto implica conocer tanto información cuantitativa, como cualitativa.

Por el momento no es necesario realizar un análisis a la carga de transacciones, ya que la base de datos del sistema aún maneja una gran cantidad de información.

7.1.4.2 Seleccionar Organización de Archivos

El objetivo de este paso es escoger la organización de ficheros óptima para cada relación entre las más comunes proporcionadas por los DBMS, que son del tipo Heap, HASH, ISAM y B-Tree.

En SQL Server 2008 la organización de archivos es propietaria y viene dada por defecto y no es susceptible de modificación.

7.1.4.3 Elegir índices Secundarios

Si un fichero está desordenado, cualquier índice de un solo nivel que se defina sobre él es un índice secundario. También es un índice secundario todo aquel que se define sobre un campo de un fichero ordenado que no es el campo de ordenación. Los índices secundarios no sirven para identificar los registros de la tabla, ya que distribuyen los registros existentes en un orden lógico y predefinido.

SQL Server 2000 automatiza la labor de creación de índices primarios, ya que se genera un índice de este tipo por cada clave primaria existente.

7.1.4.4 Introducción de redundancia Controlada

La poca flexibilidad de un modelo de datos puede obligar a la introducción de redundancia objeto de mejorar las prestaciones del sistema.

En el caso del sistema de control HIM no se considera que el rendimiento esperado amerite la introducción de redundancia, ya que cumple con los niveles de desempeño esperados.

7.1.4.5 Estimar los Requerimientos de Espacio en Disco

Dado que a esta altura del proyecto aún no se cuenta con información clara sobre el número de registros que podría llegar a contener en promedio cada tabla del Sistema de Control de Bodegas HIM, se hace imposible estimar el espacio de disco requerido para el almacenamiento de la base de datos, lo que hoy en día con las grandes capacidades de disco y el bajo costo que esto representa para el proyecto es un tema a considerar como una variable de muy bajo impacto.

7.1.5 Requerimientos y Mecanismos de Seguridad

Un punto importante en los sistemas de información es tratar de garantizar la coherencia de los datos, comprobando que sólo los usuarios autorizados puedan efectuar las operaciones correctas sobre la base de datos.

7.1.5.1 Diseñar Vistas de Usuario

Las vistas de usuario tienen como objetivo principal potenciar la seguridad del sistema, además de mejorar la independencia de los datos, reduciendo la

complejidad y permitiendo que los usuarios vean los datos en el formato deseado.

Las transacciones tienen permisos totales para los miembros del perfil “*Administrador*” de la base de datos.

7.1.5.2 Diseñar Reglas de Acceso

El diseño de las reglas de acceso pretende asegurar la información y su manejo tanto a nivel de base de datos como a nivel de aplicación. Es por esto que en el sistema se contemplan niveles de seguridad a nivel de base de datos y de aplicación.

- Nivel de aplicación: A nivel de aplicación se han definido dos tipos de usuarios diferentes, los potenciales clientes que tiene acceso a buscar ítems y a generar, modificar y enviar solicitudes de compra; y los usuarios administradores que tienen acceso al área de administración de la empresa y de los catálogos, para lo cual deberán suministrar un *usuario* y una *contraseña* para validar sus permisos de acceso al sistema.

- Nivel de base de datos: El plan de seguridad identifica qué usuarios pueden ver qué datos y qué actividades pueden realizar en la base de datos.

A nivel de base de datos también se manejan dos usuarios diferentes, uno es el usuario generado y requerido por SQL Server 2008 para acceder a la base de datos a través del Administrador de SQL Server, este usuario tiene completo acceso a la base de datos. El otro usuario implementado a nivel de base de datos es el usuario administrador dentro del propio sistema, el cual es dueño de la base de datos y tiene completo acceso a ésta.

7.2 Diseño de la Aplicación

En este capítulo se tiene por finalidad definir la estructura lógica y física del modelo de datos, donde se especificarán los datos y sus interrelaciones, así también se moldearán las transacciones requeridas por el usuario en el modelo de datos.

En la metodología Extreme Programming, donde se privilegia el desarrollo, se señala de manera prioritaria la construcción de diseños más bien simples, a modo de bosquejos, los que se puedan presentarse de una manera más rápida

al cliente, para que de esta forma poder conseguir una retroalimentación de las primeras impresiones de éste.

Dentro del proceso de diseño de sistemas hay que tener en cuenta los efectos que pueda producir la introducción del nuevo sistema sobre el entorno en el que va a funcionar, adecuando los criterios de diseño a las características del mismo.

En este contexto está adquiriendo una importancia creciente la adaptación de todo sistema-producto a las capacidades de las personas que van a utilizarlo, de forma que su operación sea sencilla, cómoda, efectiva y eficiente.

A continuación, se detallarán los diseños más importantes del sistema que son el de base de datos y el de interfaz.

7.2.1 Diseño de Transacción

Una transacción es un conjunto de acciones llevadas a cabo por un usuario que accede o cambia el contenido de la base de datos.

El objetivo del diseño de las transacciones será definir y documentar las características de alto nivel de las transacciones que requiere el sistema.

El diseño de transacciones que se presenta a continuación será implementado a nivel de la aplicación y no de la base de datos. Esta determinación se explica con mayor profundidad en el capítulo 8 “Desarrollo del Sistema” del presente informe de tesis.

7.2.2 Diseño de Interfaz de Usuario

El diseño de interfaz es uno de los elementos "clave" en la realización del programa. Se puede definir la interfaz como el conjunto de trabajos y pasos que seguirá el usuario, durante todo el tiempo que se relacione con el programa, detallando lo que verá y escuchará en cada momento, y las acciones que realizará, así como las respuestas que el sistema le dará.

Debido a que el elemento principal de ésta interfaz es la pantalla del computador, se tendrá especial cuidado en su organización, combinando información, elementos de interacción y la información interactiva.

Las características básicas que se quiere conseguir con esta interfaz, se podrían sintetizar en:

- Facilidad de aprendizaje y uso.

- Representación permanente de un contexto de acción (fondo).
- El objeto de interés ha de ser de fácil identificación.
- Diseño ergonómico (barra de acciones o iconos, preferentemente a la derecha).
- Las interacciones se basarán en acciones físicas sobre elementos de código visual o auditivo (iconos, imágenes, mensajes...) antes que en selecciones de tipo menú con sintaxis y órdenes.
- Las operaciones serán rápidas, incrementales y reversibles, con efectos inmediatos.
- Tratamiento del error bien cuidado y adecuado al nivel de usuario y contenidos trabajados.

Para el diseño de las interfaces de usuario la metodología elegida para diseñar el sistema fue creada por Cataldi, Lage [2000], la cual abarca los siguientes temas:

- Definir la Organización de los Menús
- Definir tipo de Iconos
- Seleccionar Efectos
- Seleccionar Textos
- Asegurar facilidad de Lectura
- Realizar el diseño de las Pantallas

- Realizar diseño de los Menú
- Definir las Actividades (búsqueda)
- Definir tipos de Ayudas didácticas(errores, mensajes)

7.2.2.1 Definir la Organización de los Menús

En primera instancia se definió qué tipo de organización se debía implementar en cada una de las opciones de los menús. La organización de los menús se definió siguiendo un esquema de organización funcional, debido a que predomina sobre los demás esquemas de organización. Dicho estándar genera un orden en el cual el usuario puede seguir con facilidad y sencillez, permitiendo una rápida ambientación por parte del usuario.

Una ventaja considerable es que el esquema funcional permite realizar las tareas en forma grupal, permitiendo al usuario seguir los procesos de forma natural, agilizando las tareas.

A continuación, se muestra el diseño del esquema de menús definido para el Sistema de Control de Bodegas HIM.

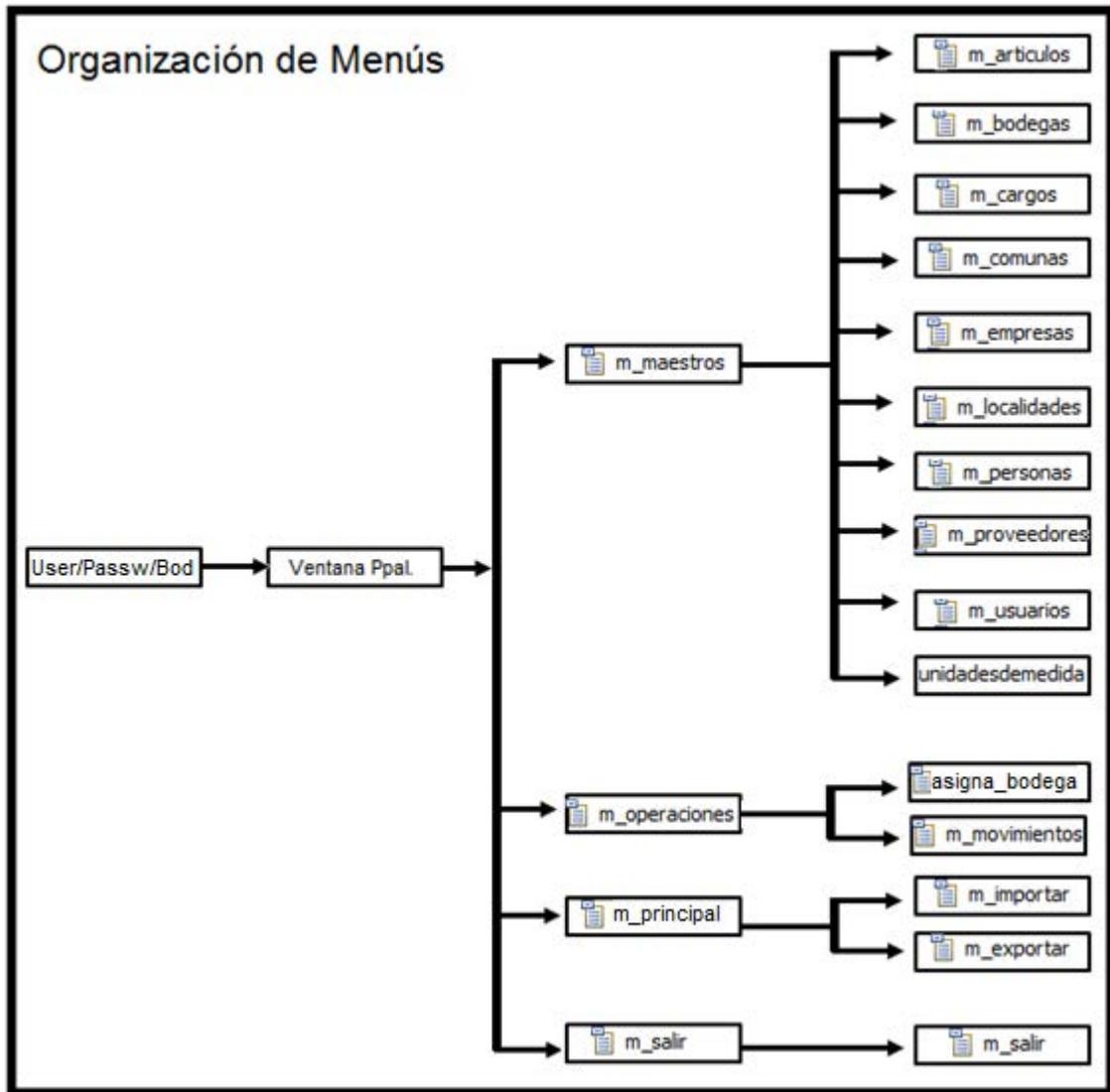


Figura 13: Diagrama de Navegación de Menús

7.2.2.2 Definir tipo de Iconos

El tipo de iconos se definió de forma que dependiendo de las funciones elegidas se asocian ciertas formas o tipos de iconos, con lo cual ayuda al usuario a determinar mediante el icono en que parte del sistema se encuentra.

Además, los iconos definidos se relacionan con el sentido que muestra el Sistema de Bodegas HIM.

7.2.2.3 Seleccionar Efectos

Se revisaron los efectos sonoros para alertar a los usuarios en los acontecimientos importantes o cruciales, dejando estandarizado cada uno de los sonidos en base al conjunto de efectos sonoros que ofrece el lenguaje de programación utilizado.

7.2.2.4 Seleccionar Textos

Con respecto a la generación de textos, estos debieron estandarizarse, dejando los botones y los textos en posiciones similares en todas las ventanas.

Se incorpora ventana de acceso al Sistema para poder ejemplificar la generación de textos y posiciones de botones del sistema.



The image shows a screenshot of a software window titled "Bienvenido al Sistema de Bodegas HIM". The window contains a login form with the following elements:

- A header text: "Ingrese usuario y contraseña".
- A text input field labeled "Usuario:".
- A text input field labeled "Contraseña:".
- A dropdown menu labeled "Bodega:".
- Two buttons at the bottom: "Ingresar" and "Salir".

Figura 14: Ventana de Inicio del Sistema

7.2.2.5 Asegurar facilidad de Lectura

Para facilitar la lectura se generaron textos de tamaño apropiado con colores sobrios y oscuros con fondos claros, los además de una apropiada ubicación. Los textos se diseñaron para llamar la atención del usuario, con la salvedad de no distraer.

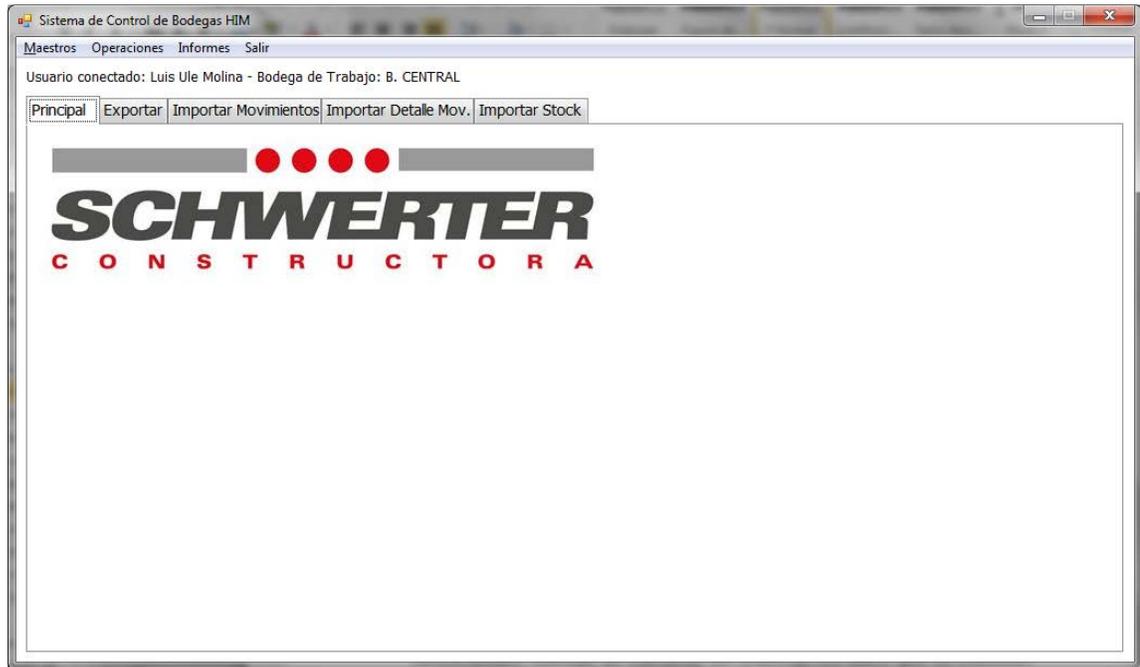


Figura 15: Ventana Principal del Sistema.

7.2.2.6 Realizar el diseño de las Pantallas

Se generó el diseño de las pantallas siguiendo un patrón de funcionalidad, además de mantener un orden de los datos que se muestran como también los que se solicitan.

A continuación, se muestra la ventana de Menús para mostrar el diseño de las pantallas con las que contará el sistema.

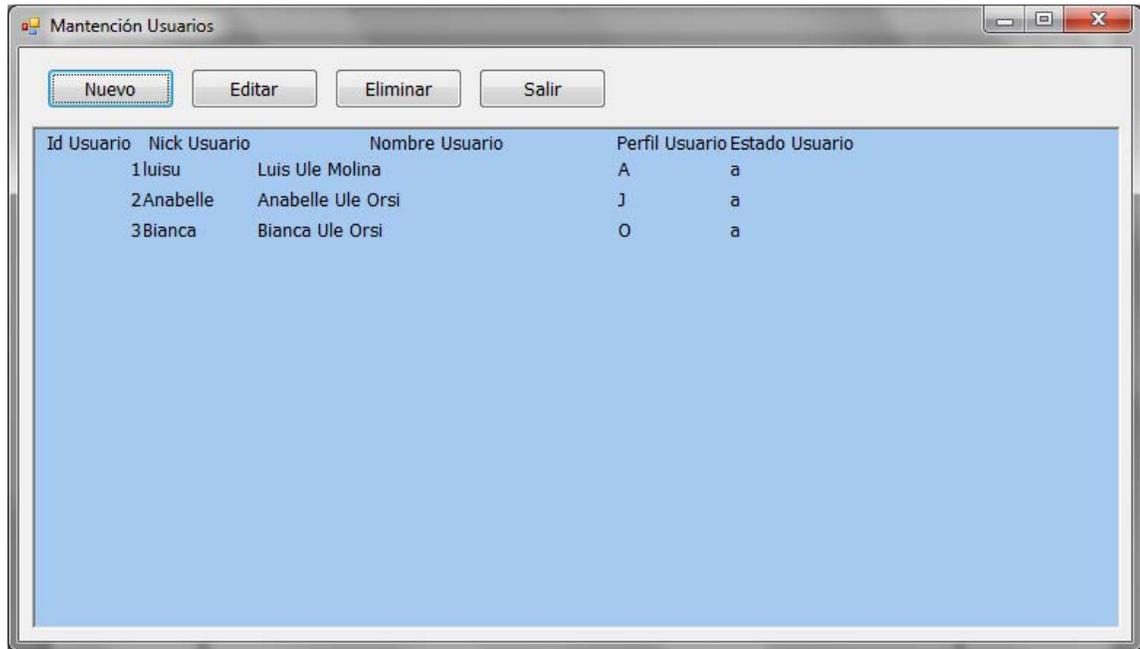


Figura 16: Ventana del Mantenedor de Maestro de Usuarios

7.2.2.7 Realizar diseño de los Menús

Se realizaron los diseños de los menús de acuerdo a la organización anteriormente definida.

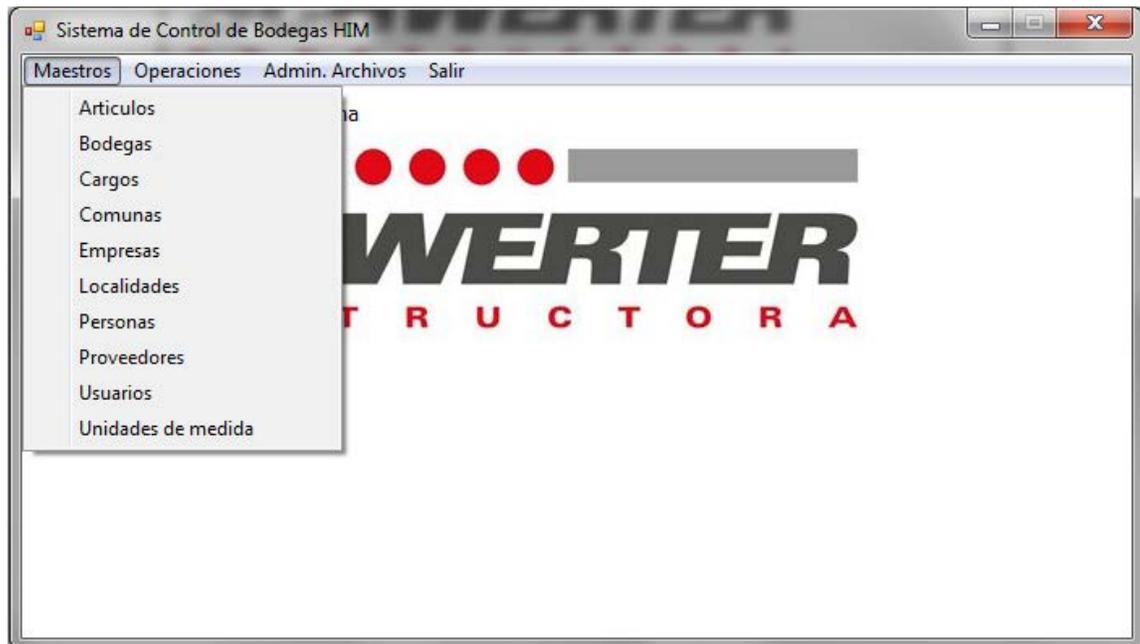


Figura 17: Ventana de Menús del Mantenedor de Maestros del Sistema.

7.2.2.8 Definir las Actividades (búsqueda)

Las actividades de búsqueda se definieron de acuerdo a las transacciones de consultas, apoyadas principalmente por las transacciones de actualización.

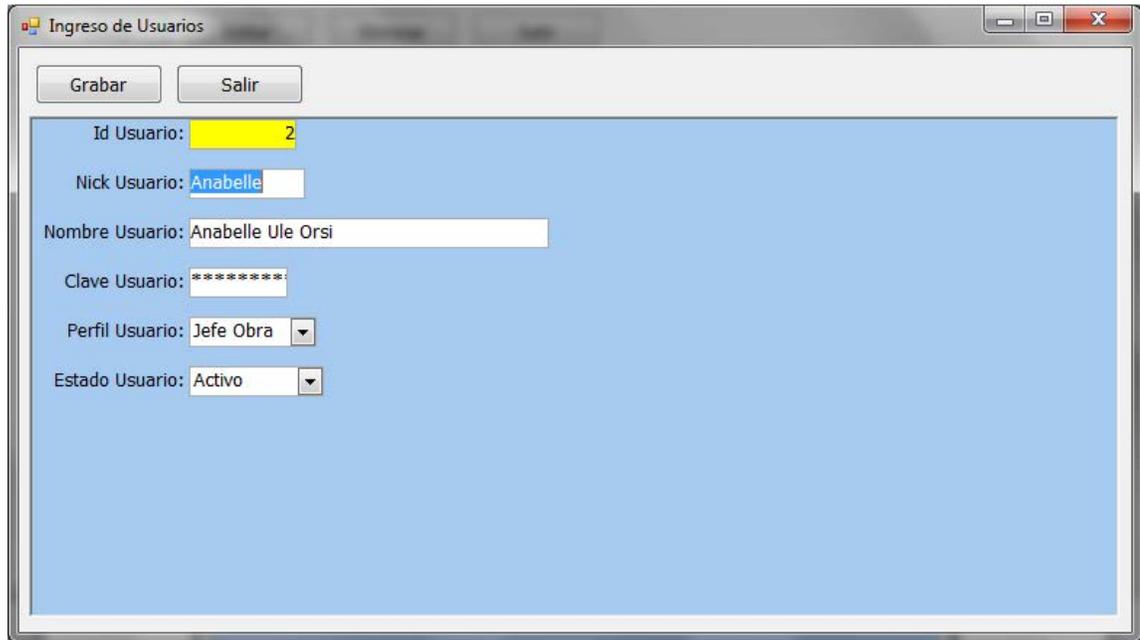


Figura 18: Ventana de Selección y Modificación de Usuarios.

7.2.2.9 Definir tipos de Ayudas didácticas (errores, mensajes)

Se definieron los tipos de ayudas generando mensajes al usuario, más allá de dar aviso al problema suscitado, se definieron para guiar al usuario con una apropiada explicación.



Figura 19: Ventana Mensaje de Grabación de Registros.

8 Desarrollo del Sistema e Implementación

8.1 Implementación de la Base de Datos

Este proceso comprende la construcción de la base de datos de Bodegas_HIM, con el código de la creación de tablas utilizadas. Además, se presenta la implementación de la interfaz de usuario, en donde se muestran las pantallas de las funciones principales.

Se explica a continuación la construcción del modelo de datos, identificando y especificando entidades, atributos y relaciones.

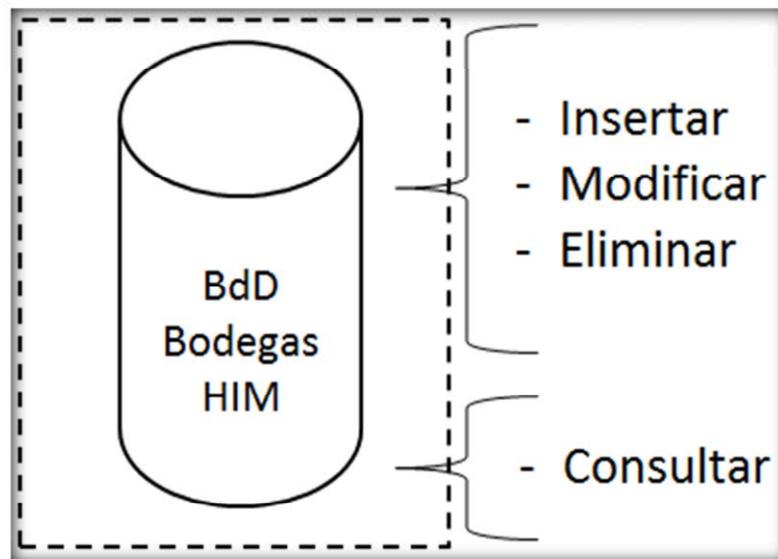


Figura 20: Representación de la Base de Datos.

La codificación, se basa en el diseño, ya que comprende la traducción a un lenguaje de programación de los pasos definidos en la documentación de diseño.

- Base de Datos
 - Transact-SQL: es el lenguaje de programación que ofrece Microsoft para poder acceder al lenguaje de consulta SQL. Ha sido usado para el desarrollo del script para la creación de la Base de Datos y su estructura interna.

8.1.1 Script de creación de la Base de Datos Bodegas_HIM

Sólo se presentará un extracto del script, el script completo se encuentra

```
/*=====*/  
/* Table: ALMACENA */  
/*=====*/  
create table ALMACENA ( ID_BODEGA          ID_NUMERICO          not null,  
                        ID_ARTICULO       ID_NUMERICO          not null,  
                        CANTIDAD          numeric(12,2)        null,
```

```

constraint PK_ALMACENA primary key (ID_BODEGA, ID_ARTICULO))go
/*=====*/
/* Index: ALMACENA_FK */
/*=====*/
create index ALMACENA_FK on ALMACENA (ID_BODEGA ASC
) go
/*=====*/
/* Index: ALMACENA2_FK */
/*=====*/
create index ALMACENA2_FK on ALMACENA (
ID_ARTICULO ASC) go
/*=====*/
/* Table: ARTICULO */
/*=====*/
create table ARTICULO ( DESC_ARTICULO DESCRIPCION null,
ID_ARTICULO ID_NUMERICO not null,
ID_UNIDAD_MEDIDA int not null,
TIPO_ARTICULO ESTADO null,
constraint PK_ARTICULO primary key nonclustered (ID_ARTICULO)) go
/*=====*/
/* Index: SE_ASOCIA_FK */

```

```
/*=====*/
create index SE_ASOCIA_FK on ARTICULO ( ID_UNIDAD_MEDIDA ASC )
go
/*=====*/
/* Table: AUTORIZADOR_SALIDA_TRASLADO */
/*=====*/
create table AUTORIZADOR_SALIDA_TRASLADO ( RUT_PERSONA
RUT not null,
ID_CARGO ID_NUMERICO not null,
constraint PK_AUTORIZADOR_SALIDA_TRASLADO primary key
nonclustered (RUT_PERSONA) ) go
/*=====*/
/* Index: OSTENTA_FK */
/*=====*/
create index OSTENTA_FK on AUTORIZADOR_SALIDA_TRASLADO (
ID_CARGO ASC ) go
/*=====*/
/* Table: BODEGA */
/*=====*/
create table BODEGA (
```

```

        ID_BODEGA      ID_NUMERICO      not null,
        ID_LOCALIDAD  int              not null,
        NOMBRE_BODEGA  DESCRIPCION      null,
        ESTADO        ESTADO        null,
        TELEFONO_BODEGA TELEFONO          null,
        constraint PK_BODEGA primary key nonclustered (ID_BODEGA) ) go
/*=====*/
/* Table: CARGO */
/*=====*/
create table CARGO (ID_CARGO      ID_NUMERICO      not null,
        NOMBRE_CARGO  DESCRIPCION      null,
        constraint PK_CARGO primary key nonclustered (ID_CARGO) ) go
/*=====*/
/* Table: CENTRAL */
/*=====*/
        create table CENTRAL (
        ID_BODEGA      ID_NUMERICO      not null,
        constraint PK_CENTRAL primary key nonclustered (ID_BODEGA) ) go
/*=====*/
/* Table: COMPRADOR */
/*=====*/

```

```

create table COMPRADOR (
    RUT_PERSONA      RUT      not null,
    ID_CARGO         ID_NUMERICO  not null,
constraint PK_COMPRADOR primary key nonclustered (RUT_PERSONA))go
/*=====*/
/* Index: DESEMPENA_FK */
/*=====*/
create index DESEMPENA_FK on COMPRADOR (
    ID_CARGO ASC ) go
/*=====*/
/* Table: COMUNA */
/*=====*/
create table COMUNA ( ID_COMUNA      int      not null,
    NOMBRE_COMUNA     DESCRIPCION  null,
constraint PK_COMUNA primary key nonclustered (ID_COMUNA)) go
/*=====*/
/* Table: CONSUMIDOR */
/*=====*/
create table CONSUMIDOR ( RUT_PERSONA      RUT      not null,
constraint PK_CONSUMIDOR primary key nonclustered (RUT_PERSONA) )
go

```

```

/*=====*/

/* Table: CONTIENE */

/*=====*/

create table CONTIENE ( ID_ARTICULO      ID_NUMERICO      not null,
                       ID_MOV          ID_NUMERICO      not null,
                       ID_BODEGA       ID_NUMERICO      not null,
                       CANTIDAD         numeric(12,2)    null,
                       VALOR_UNITARIO   numeric(10)      null,
                       constraint PK_CONTIENE primary
                       key (ID_ARTICULO, ID_MOV, ID_BODEGA)) go

/*=====*/

/* Index: CONTIENE_FK */

/*=====*/

create index CONTIENE_FK on CONTIENE (
                               ID_ARTICULO ASC ) go

/*=====*/

/* Index: CONTIENE2_FK */

/*=====*/

create index CONTIENE2_FK on CONTIENE (
                               ID_MOV ASC, ID_BODEGA ASC ) go

/*=====*/

```

```

/* Table: DESTINO */

/*=====*/

create table DESTINO (ID_BODEGA      ID_NUMERICO      not null,
                    constraint PK_DESTINO primary key nonclustered (ID_BODEGA) ) go

/*=====*/

/* Table: DOCUMENTO */

/*=====*/

create table DOCUMENTO ( ID_DOCTO      ID_NUMERICO      null,
                        TIPO_MOV        ESTADO           null,
                        TIPO_DOCTO      ESTADO           null,
                        FECHA_DOCTO      FECHA            null,
                        GUIA_DESPACHO     ID_NUMERICO       null,
                        PATENTE          varchar(6)        null,
                        ID_MOV            ID_NUMERICO       not null,
                        ID_BODEGA        ID_NUMERICO       not null,
                        ID_USUARIO       int              not null,
                        RUT_PROVEEDOR     RUT              null,
                        RUT_COMPRADOR     RUT              null,
                        RUT_SOLICITANTE   RUT              null,
                        RUT_AUTORIZADOR   RUT              null,
                        RUT_TRANSPORTISTA RUT              null,

```

```

        RUT_CONSUMIDOR    RUT          null,
        ID_BODEGA_DESTINO ID_NUMERICO not null,
        ID_BODEGA_ORIGEN ID_NUMERICO null,
        FECHA_MOV         FECHA       null,
        OBSERVACIONES     OBSERVACION null,
        constraint PK_DOCUMENTO primary key nonclustered (ID_MOV,
                ID_BODEGA) ) go

/*=====*/

        /* Index: TIENE_FK          */

/*=====*/

        create index TIENE_FK on DOCUMENTO (
                ID_BODEGA_ORIGEN ASC) go

/*=====*/

        /* Index: INCLUYE_FK          */

/*=====*/

        create index INCLUYE_FK on DOCUMENTO (
                RUT_PROVEEDOR ASC) go

```

Tabla 15: Fragmento Código Script creación de las Tablas

8.2 Implementación de Interfaz del Sistema

En la sección anterior 7.2.2 se mostraron algunas de las interfaces más características del Sistema de Control de Bodegas HIM. Cada pantalla representa parte de un módulo en particular.

Id Artículo	Cantidad	Valor Unitario	Id Mov	Bodega

Figura 21: Interfaz de Sistema – Búsqueda y Eliminación de Movimientos

9 Pruebas del Sistema

Las pruebas son un proceso individualista y el número de tipos diferentes de pruebas varía tanto como los diferentes enfoques de desarrollo, una prueba es un conjunto de actividades que se planifican por adelantado y llevan a cabo sistemáticamente.

La estrategia de prueba de software incorpora la planificación de la prueba, el diseño de un caso de prueba, la ejecución de pruebas y la agrupación y evaluación de los datos resultantes. Las pruebas se realizaron con el objetivo es demostrar que la base de datos junto a los programas de aplicación trabajan de acuerdo a los requerimientos.

Las pruebas realizadas consisten en:

- Prueba de Unidad.
- Prueba de Integración.
- Prueba de Validación
- Prueba de Sistema.

9.1 Prueba de Unidad

La prueba de unidad centra el proceso de verificación en la menor unidad del diseño del software. Aquí se prueban los caminos de control importantes, con el fin de descubrir errores dentro del ámbito de un módulo.

Se probó la Interfaz de la pantalla para asegurar que la información fluya en forma adecuada hacia y el programa que está siendo probado.

Se analizaron las estructuras de datos para asegurar que los datos mantienen su integridad durante todos los pasos de ejecución del programa.

Se probaron las condiciones límite para asegurar que el módulo funcionara correctamente dentro de los límites establecidos por el procesamiento, y finalmente se activaron los caminos básicos de la estructura de control con el fin de asegurar que las sentencias del módulo se ejecuten por lo menos una sola vez.

Complemento al capítulo de Pruebas, favor Ver Anexo 1, Página 116

9.2 Prueba de Integración

La prueba de Integración es una técnica sistemática para construir la estructura del programa mientras que al mismo tiempo, se llevan a cabo pruebas para detectar errores asociados con la interacción. El objetivo es tomar los módulos probados en unidad y estructurar un programa que esté de acuerdo con lo que dicta el diseño.

Se utilizó un tipo de integración incremental para poner a prueba el Sistema de Control de Bodegas HIM, en donde se desarrollaron módulos pequeños y funcionales que hicieron que los errores fuesen más fáciles de aislar y corregir.

9.3 Prueba de Validación

La prueba de validación se identifica con el conjunto de actividades que aseguran que el software implementa correctamente una función específica se denomina verificación. La validación se refiere a un conjunto diferente de actividades que aseguran que el software construido se ajusta a los requisitos y necesidades del cliente.

Las actividades de esta prueba se realizaron con la usuaria administradora que forma parte de los usuarios finales, quien fue la encargada de ingresar una serie de datos (usuarios, unidad de medida, artículo y proveedores, entre otros), modificarlos y eliminarlos, luego buscar los ítems que se habían ingresado y por último generar un movimiento con un caso de eso real, para comprobar luego su correcta recepción.

9.4 Prueba de Sistema

El objetivo de este tipo de pruebas es ejercitar profundamente el sistema, verificando la integración adecuada y el comportamiento de todos los elementos del sistema.

Luego de construida la aplicación y considerada aceptable para la implantación, fue cargada en su ubicación final y se efectuaron las siguientes pruebas del sistema para verificar que se hayan integrado correctamente cada uno de los elementos del sistema.

➤ Prueba de Recuperación. Se trató de forzar el fallo del sistema y para verificar su recuperación.

- Prueba de Seguridad. Se validó si el sistema manejaba correctamente la seguridad establecidas en el diseño.

- Pruebas de Rendimiento. Se comprobó el tiempo de respuesta del sistema. Y también se evaluó el comportamiento del sistema al ser usado por múltiples usuarios.

10 Implantación

En esta fase se pone en marcha el sistema para ser utilizado por el cliente, permitiendo a este administrarlo completamente.

En la implantación se desarrolló el proceso de verificar e instalar el nuevo software, entrenar a los usuarios y construir todos los archivos de datos necesarios para utilizar el Sistema de Control de Bodegas HIM

10.1 Instalación de la Aplicación

La instalación de todos los componentes del sistema, junto a sus prerequisites se realizó de manera individual en el servidor y en alguna de las estaciones de usuario presente.

10.2 Entrenamiento de Usuarios

Para el entrenamiento de los usuarios se realizaron dos sesiones. En estas sesiones se exploraron todas las funciones del sistema, explicando de manera clara la forma de llevar a cabo cada una de las diferentes acciones que el sistema permita realizar.

11 Conclusiones y Recomendaciones

Una vez finalizado el desarrollo del presente seminario de tesis se puede concluir que el Sistema de Control de Bodegas HIM, ha cumplido con el principal objetivo para el cual fue concebido.

Es importante mencionar que los conocimientos entregados por la universidad al alumno tesista han permitido que la labor de desarrollo de este seminario fuese posible, dado que apoyado en las herramientas, métodos, diagramas y diversos conocimientos más adquiridos durante la duración de la carrera del alumno fue factible concluir con éxito el Sistema de Control de Bodegas HIM.

La metodología utilizada en este seminario de titulación fue de gran ayuda, ya que enfatiza la comunicación expedita entre el desarrollador y el cliente, quien es, en definitiva quien deberá convivir a futuro con un sistema, el cual debe ser cómodo, fiable y amigable de trabajar.

La importancia de la información que entregará este sistema de control de bodegas HIM será muy valiosa para los administradores de las empresas

constructoras Schwerter, puesto que les permitirá sincronizar de mejor manera las llegadas y salidas de artículos a las bodegas de artículos HIM.

A modo de recomendación para futuras etapas de mejoramiento de este sistema se sugiere la incorporación o conexión con un futuro sistema de control de obras y etapas de obras, el que podría valerse de esta información para automatizar procesos existentes en la ejecución de una obra de construcción.

Como otra recomendación se deberá crear un manual de usuario a nivel más técnico para que el administrador del sistema pueda valerse por sí mismo en cuanto a la instalación de requisitos de software en las estaciones clientes.

La empresa no descarta la ampliación de este sistema con el objetivo de robustecer el flujo de información, para poder mejorar el control, tanto de artículos como de movimientos en las diferentes obras que se monte el sistema.

12 Bibliografía

- [INGSOFTWARE] Extreme Programing
<http://universotela.blogspot.com/2011/05/scrum-y-programacion-extrema.html>
- [XP2003] XP – Programación Extrema
<http://www.programacionextrema.org/articulos/newMethodology.es.html>
- [CONNOLLY - BEGG] Libro “Sistemas de Bases de Datos”
THOMAS M. CONNOLLY y CAROLYN E. BEGG,
4ta. Edición, Año 2005.
- [CATALDI Z.- LAGE F.] Methodology of design and development
of educational software from a pedagogical
perspective, 1ra. Edición, Año 2000.
- [MSDN Library] Biblioteca de desarrollo de Microsoft. Disponible en
<http://msdn.microsoft.com/es-ar/library>
- [<http://cybertesis.uach.cl>] Tesis Sr. Patricio Lerou Godas,
<http://cybertesis.uach.cl/tesis/uach/2005/bpmfcil617s/doc/bpmfcil617s.pdf>

13 Anexo

13.1 Prueba de Caja Negra

Este tipo de prueba se enfoca en el estudio de las funciones, entregando las entradas y recibiendo los resultados de las salidas, sin importar lo que ocurre dentro del o los procedimientos, para ello se verifican dos casos.

- Movimiento de datos (Ingreso y Actualización)
- Consultas

Para el primer caso con el tipo de ingreso se verifican que los datos ingresados en el sistema se reflejen en la base de datos.

En el segundo caso con el tipo de actualización se verifican que los datos que se proceden a actualizar se cambien en la base de datos.

En el caso de las consultas de los registros se verifica que los resultados sean los esperados.

Para definir la acción sobre la BD se utilizará la siguiente nomenclatura:

- **M: Movimiento sobre la base de datos**
- **C: Consulta sobre tablas de la base de datos**

Los resultados de las pruebas se clasificarán como:

- **S: Satisfactorio**
- **NS: No Satisfactorio**

El Sistema de Control HIM presenta los siguientes menús en la ventana de inicio principal (ver figura 15, página N° 92), los cuales se presentan a continuación y se muestran los resultados de las pruebas en la siguiente tabla:

Nombre del Menú	Denominación en el sistema
Menú Maestros	M_maestros
Menú Operaciones	M_operaciones
Menú Informes	M_informes
Menú Principal	M_principal

Tabla Anexo 1: Menús existentes del Sistema

Nombre del Menú	Descripción	Acción	Resultado
Menú Maestros			
Artículos	Permite agregar nuevos artículos	M	S
Bodegas	Permite agregar nuevas Bodegas	M	S
Cargos	Permite agregar nuevos Cargos	M	S
Comunas	Permite agregar nuevas Comunas	M	S
Empresas	Permite agregar nuevas Empresas	M	S
Localidades	Permite agregar nuevas Localidades	M	S
Personas	Permite agregar nuevas Personas	M	S
Proveedores	Permite agregar nuevos Proveedores	M	S
Usuarios	Permite agregar nuevos Usuarios	M	S
Unidad de Medida	Permite agregar nuevas Unidad de Medida	M	S
Menú Operaciones			

Asigna Bodega	Permite asociar a un usuario con una bodega	M	S
Movimientos	Ingresa información de los movimientos de artículos generados en la bodega	M	S
Menú Informes			
Consulta Stock	Realiza una consulta sobre la cantidad de artículos existentes en bodega.	C	S
Ingresos por Compra	Consulta los ingresos de artículos por concepto de compra bajo parámetros definidos por el usuario.	C	S
Ingresos por Traslado	Consulta los ingresos de artículos por concepto de traslado entre bodegas con parámetros definidos por el usuario.	C	S
Ingresos por Devolución	Consulta los ingresos de artículos por concepto de devolución bajo parámetros definidos por el usuario	C	S

Salidas por Traslado	Consulta los egresos de artículos por concepto de traslado bajo parámetros definidos por el usuario	C	S
Salidas por Consumo	Consulta los egresos de artículos por concepto de gasto o consumo bajo parámetros definidos por el usuario	C	S
Menú Principal			
Exportar	Exporta datos a un archivo plano o MS Excel para suplir la falta de internet en lugares que no exista.	C	S
Importar Movimientos	Importa datos relacionados a los movimientos generados en la bodega, desde un archivo plano o MS Excel, para suplir la falta de internet en lugares que no exista.	M	S
Importar Detalle Mov.	Importa datos relacionados a los detalles de los movimientos generados en la bodega, desde un archivo plano o MS Excel,	M	S

	para suplir la falta de internet en lugares que no exista.		
Importar Stock	Importa datos relacionados al stock de los artículos asociados a los movimientos anteriores en la bodega, desde un archivo plano o MS Excel, para suplir la falta de internet en lugares que no exista.	M	S

Tabla Anexo 2: Resultados pruebas de Menús del Sistema

13.2 Prueba de Caja Blanca

En esta etapa se realizan las pruebas de implementación, con lo cual se verifica la estructura interna del sistema.

Como es imposible mostrar todo el código que demuestra la verificación de la estructura interna del sistema sólo se mostrarán algunas secciones de código.

El siguiente ejemplo se muestra el código de verificación del ingreso en forma correcta de los caracteres en el RUT de los Proveedores y Personas.

```

//// INICIO FUNCION F_VERIFICA_RUT

integer resto, i

string factor, dv

integer suma

if len(string(rut))=8 then
    factor='32765432';
else
    factor='2765432';
end if

i=1;

suma=0;

do while i<=10
    suma=suma+integer(mid(string(rut),i,1))*integer(mid(factor,i,1))
    i=i+1
loop
    resto=MOD(suma,11)

if resto=1 then
    dv='K'
else
    if resto=0 then

```

```

        dv='0'
    else
        dv=string(11 - resto)
    end if
end if
if dv <> "" then
    return dv
else
    return ""
end if
//// FIN

```

En el siguiente código se muestra la función `f_chequea_stock`, creada con la finalidad de encontrar el stock de un artículo en una bodega.

```

// INICIO FUNCION F_CHEQUEA_STOCK
//devuelve 0 si no hay art; y la cantidad, si es que hay
decimal Id_cantidad

SELECT dbo.ALMACENA.CANTIDAD INTO :Id_cantidad
FROM dbo.ALMACENA
WHERE ( dbo.ALMACENA.ID_BODEGA = :Id_id_bodega ) AND

```

```
(dbo.ALMACENA.ID_ARTICULO = :ll_id_articulo ) using sqlca ;  
if SQLCA.SQLCode = -1 then  
    MessageBox("SQL error", "Se produjo el sgte. error al buscar stock:  
"+SQLCA.SQLErrMsgText)  
    rollback using sqlca ;  
else  
    commit using sqlca ;  
    if isnull(Id_cantidad) then  
        Id_cantidad=0  
    end if  
end if  
return(Id_cantidad)  
// FIN FUNCION F_CHEQUEA_STOCK
```