



INSTITUTO TECNOLÓGICO  
"CORDILLERA"

**CARRERA DE ANÁLISIS DE SISTEMAS**

**SISTEMATIZACIÓN PARA LOS PROCESOS DE INVENTARIO DE  
MATERIA PRIMA MEDIANTE UNA APLICACIÓN INFORMÁTICA PARA  
LA EMPRESA "BUS CA PARTES".**

**Proyecto de investigación previo a la obtención del Título de Tecnólogo en  
Análisis de Sistemas**

**Autor: Morocho Mesa Karina Estefanía**

**Tutor: Ing. Carlos Romero**

**Quito, Octubre 2014**

## DECLARATORIA

Declaro que la investigación es absolutamente original, auténtica, personal, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes. Las ideas, doctrinas resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

---

Karina Estefanía Morocho Mesa.

CC 172570017-1

## **CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL**

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, la estudiante **Morocho Mesa Karina Estefanía**, por sus propios y personales derechos, a quien en lo posterior se le denominará el “CEDENTE”; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el “CESIONARIO”. Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

**PRIMERA: ANTECEDENTE.-** a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado “**SISTEMATIZACIÓN PARA LOS PROCESOS DE INVENTARIO DE MATERIA PRIMA MEDIANTE UNA APLICACIÓN INFORMÁTICA PARA LA EMPRESA BUS CA PARTES**”, el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. b) Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

**SEGUNDA: CESIÓN Y TRANSFERENCIA.-** Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita todos los derechos patrimoniales del programa de ordenador descrito en la clausula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.). El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

**TERCERA: OBLIGACIÓN DEL CEDENTE.-** El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

**CUARTA: CUANTIA.-** La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

**QUINTA: PLAZO.-** La vigencia del presente contrato es indefinida.

**SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.-** Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se solicitará la asistencia de un Mediador del Centro de Arbitraje y Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas: a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvenición, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

**SÉPTIMA: ACEPTACIÓN.-** Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 24 días del mes de octubre del dos mil catorce.

f) \_\_\_\_\_

C.C. N° 1725700171

**CEDENTE**

f) \_\_\_\_\_

Instituto Superior Tecnológico Cordillera

**CESIONARIO**

## AGRADECIMIENTO

Agradezco a Dios por haber guiado mi camino y por permitirme dar este paso muy importante de mi vida, a mi madre por brindarme su apoyo y cariño contribuyendo con todo lo que estuvo a su alcance, a mi esposo por toda su comprensión, apoyo y amor.

Gracias a mi tutor el Ing. Carlos Romero, por el valioso aporte profesional, y humano brindado en el transcurso de este proceso, a todo el personal docente, porque gracias a ustedes soy un ser humano que piensa, que enseña y que cada día aprende de la vida algo.

Finalmente agradezco a todos aquellos que me apoyaron durante el lapso de mi carrera para realizarme profesionalmente.

## DEDICATORIA

Dedico esta tesis a Dios porque sin él no sería  
posible que haya terminado este trabajo

A mi esposo por su infinito amor y comprensión

A mi hijo por su cariño que me alentaba cada día para culminar

A mi madre por su apoyo incondicional

A mis hermanos para que no se dejen vencer y alcancen sus metas

## ÍNDICE GENERAL

### Contenido

DECLARATORIA.....	ii
CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL .....	iii
AGRADECIMIENTO.....	vii
DEDICATORIA .....	viii
ÍNDICE GENERAL.....	ix
Índice de Tablas .....	xii
Índice de Figuras .....	xiv
RESUMEN EJECUTIVO .....	xviii
ABSTRACT .....	xix
Introducción .....	xix
CAPITULO I: ANTECEDENTES .....	1
1.1 Contexto.....	1
1.2 Justificación.....	1
1.3 Definición Del Problema Central (MATRIZ T) .....	2
CAPITULO II: ANÁLISIS DE INVOLUCRADOS .....	4
2.01 Requerimientos .....	4
2.01.1 Descripción del sistema actual.....	4

2.01.2	Visión y alcance .....	4
2.01.3	Entrevistas .....	5
2.01.4	MATRIZ DE REQUERIMIENTOS.....	5
2.01.5	DESCRIPCIÓN DETALLADA. ....	6
2.02	MAPEO DE INVOLUCRADOS.....	10
2.03	MATRIZ DE INVOLUCRADOS. ....	10
CAPITULO III: PROBLEMAS Y OBJETIVOS.....		11
3.01	Árbol de Problemas.....	11
3.02	Árbol de Objetivos .....	12
3.03	Diagrama de casos de uso. ....	13
3.04	Casos de uso de realización.....	16
3.05	Especificación de Casos de uso de realización. ....	19
3.06	Diagrama de secuencia del sistema.....	21
3.07	Especificación de casos de uso. ....	24
CAPITULO IV: ANÁLISIS DE ALTERNATIVAS.....		30
4.01	Matriz de análisis de alternativas. ....	30
4.02	Matriz de Impactos de Objetivos. ....	30
4.03	Estándares para el Diseño de Clases. ....	30
4.04	Diagrama de Clases.....	33
4.05	Modelo Lógico - Físico.....	34

4.06 Diagrama de Componentes. ....	35
4.07 Diagramas de Estrategias. ....	36
4.08 Matriz de Marco Lógico.....	37
4.09 Vistas Arquitectónicas. ....	38
4.01.01 Vista Lógica. ....	38
4.01.02 Vista Física.....	39
4.01.03 Vista de Desarrollo.....	39
4.01.04 Vista de Procesos. ....	40
CAPITULO V: PROPUESTA .....	41
5.01Especificación de estándares de programación.....	41
5.02Diseño de Interfaces de Usuario. ....	41
5.03Especificación de pruebas de unidad. ....	44
5.04Especificación de pruebas de aceptación. ....	46
5.05Especificación de pruebas de carga.....	49
5.06Configuración del Ambiente mínima/ideal.....	51
CAPITULO VI: ASPECTOS ADMINISTRATIVOS .....	52
6.01 Recursos.....	52
6.02Presupuesto. ....	53
6.03Cronograma.....	54
CAPITULO VII: CONCLUSIONES Y RECOMENDACIONES .....	55

7.01 Conclusiones.....	55
7.02 Recomendaciones.....	55
ANEXOS.....	56
A.01.01 Matriz de Requerimientos.....	56
A.01.02 Mapeo de Involucrados.....	57
A.01.03 Matriz de Involucrados.....	58
A.01.04 Matriz de análisis de alternativas.....	58
A.01.05 Matriz de impacto de Objetivos.....	59
B.01.01 MANUAL DE USUARIO.....	60
B.01.02 MANUAL DE INSTALACIÓN.....	76
B.01.03 MANUAL TÉCNICO.....	103
BIBLIOGRAFÍA.....	139

### Índice de Tablas

Tabla 1. <i>Análisis de Fuerzas T.</i> .....	3
Tabla 2 <i>Entrevista realizada al Gerente de la Empresa.</i> .....	5
Tabla 3 <i>Matriz de Requerimientos.</i> .....	5
Tabla 4 <i>Descripción detallada de los requerimientos funcionales.</i> .....	6
Tabla 5 <i>Descripción detallada de los requerimientos funcionales.</i> .....	7
Tabla 6 <i>Descripción detallada de los requerimientos funcionales.</i> .....	8
Tabla 7 <i>Descripción detallada de los requerimientos funcionales.</i> .....	9

Tabla 8 <i>Descripción detallada de los requerimientos no funcionales.</i> .....	10
Tabla 9 <i>Matriz de involucrados.</i> .....	10
Tabla 10 <i>Especificación de Casos de uso de realización.</i> .....	19
Tabla 11 <i>Especificación de Casos de uso de realización.</i> .....	19
Tabla 12 <i>Especificación de Casos de uso de realización.</i> .....	20
Tabla 13 <i>Especificación de Casos de uso de realización.</i> .....	20
Tabla 14 <i>Especificación de Casos de uso de realización.</i> .....	21
Tabla 15 <i>Especificación de casos de Uso.</i> .....	24
Tabla 16 <i>Especificación de casos de Uso.</i> .....	24
Tabla 17 <i>Especificación de casos de Uso.</i> .....	25
Tabla 18 <i>Especificación de casos de Uso.</i> .....	25
Tabla 19 <i>Especificación de casos de Uso.</i> .....	26
Tabla 20 <i>Especificación de casos de Uso.</i> .....	26
Tabla 21 <i>Especificación de casos de Uso.</i> .....	27
Tabla 22 <i>Especificación de casos de Uso.</i> .....	27
Tabla 23 <i>Especificación de casos de Uso.</i> .....	28
Tabla 24 <i>Especificación de casos de Uso.</i> .....	28
Tabla 25 <i>Especificación de casos de Uso.</i> .....	29
Tabla 26 <i>Matriz de análisis de alternativas.</i> .....	30
Tabla 27 <i>Matriz de Impacto de Objetivos.</i> .....	30
Tabla 28 <i>Matriz de Marco Lógico.</i> .....	37
Tabla 29 <i>Especificación de pruebas de unidad</i> .....	44
Tabla 30 <i>Especificación de pruebas de unidad</i> .....	44

Tabla 31 <i>Especificación de pruebas de unidad</i> .....	45
Tabla 32 <i>Especificación de pruebas de unidad</i> .....	45
Tabla 33 <i>Especificación de pruebas de unidad</i> .....	45
Tabla 34 <i>Especificación de pruebas de aceptación</i> .....	46
Tabla 35 <i>Especificación de pruebas de aceptación</i> .....	47
Tabla 36 <i>Especificación de pruebas de aceptación</i> .....	47
Tabla 37 <i>Especificación de pruebas de aceptación</i> .....	48
Tabla 38 <i>Especificación de pruebas de aceptación</i> .....	48
Tabla 39 <i>Especificación de pruebas de carga</i> .....	50
Tabla 40 <i>Especificación de pruebas de carga</i> .....	51
Tabla 41 <i>Presupuesto</i> .....	53

### Índice de Figuras

Figura 1 Mapeo de Involucrados – <i>Describe los involucrados que están dentro del proyecto</i> .....	10
Figura 2 Árbol de Problemas – <i>Describe los problemas que se pueden presentar en el proceso del proyecto</i> . .....	11
Figura 3 Árbol de Objetivos – <i>Describe los objetivos planteados para el desarrollo del proyecto</i> .....	12
Figura 4 Diagrama de Casos de Uso General – <i>Describe el proceso general que se realiza en la empresa</i> . .....	13
Figura 5 Diagrama de Casos de Uso 001 - <i>Este caso de uso explica la gestión que realiza el administrador dentro de la aplicación con respecto a crear, modificar, eliminar o cambiar de usuario</i> .....	14

Figura 6 <i>Diagrama de Casos de Uso 002</i> - Este caso de uso explica la gestión que realiza el administrador, Gerente y Jefe de Bodega dentro de la aplicación con respecto al registro, modificación, eliminación o consulta de materia prima. ....	14
Figura 7 <i>Diagrama de Casos de Uso 003</i> - Este caso de uso explica la gestión que realiza el administrador, Gerente y Jefe de Bodega dentro de la aplicación con respecto al registro, modificación, eliminación o consulta de Proveedores.....	15
Figura 8 <i>Diagrama de Casos de Uso 004</i> - Este caso de uso explica el proceso para emitir los reportes.....	15
Figura 9 <i>Diagrama de Casos de Uso 005</i> – Este caso de uso explica el proceso para generar listados.....	16
Figura 10 <i>Diagrama de Casos de Realización 001</i> – Este caso de realización explica el registro y validación de datos del usuario. ....	16
Figura 11 <i>Diagrama de Casos de Realización 002</i> – Este caso de realización explica el proceso para registrar la materia prima.....	17
Figura 12 <i>Diagrama de Casos de Realización 003</i> - Este caso de realización explica el proceso para registrar proveedores.....	17
Figura 13 <i>Diagrama de Casos de Realización 004</i> - Este caso de realización explica el proceso para generar reportes.....	18
Figura 14 <i>Diagrama de Casos de Realización 005</i> - Este caso de realización explica el proceso para generar listados. ....	18
Figura 15 <i>Diagrama de Secuencia 001</i> - Este diagrama de secuencia explica el proceso para registrar usuarios. ....	21

Figura 16 <i>Diagrama de Secuencia 002</i> - Este diagrama de secuencia explica el proceso para registrar la materia prima. ....	22
Figura 17 <i>Diagrama de Secuencia 003</i> - Este diagrama de secuencia explica el proceso para registrar proveedores. ....	22
Figura 18 <i>Diagrama de Secuencia 004</i> - Este diagrama de secuencia explica el proceso para emitir reportes. ....	23
Figura 19 <i>Diagrama de Secuencia 005</i> - Este diagrama de secuencia explica el proceso para generar listados.....	23
Figura 20 <i>Diseño de tabla</i> – Describe el modelo de una tabla.....	30
Figura 21 <i>Partes de una clase</i> – Describe las partes que contiene una clase.....	32
Figura 22 <i>Diagrama de Clase</i> – Describe las tablas del diseño de la Base de Datos.	33
Figura 23 <i>Diagrama de Clases Modelo Lógico – Físico</i> - Describe las tablas del diseño de la Base de Datos en modelo lógico – físico.....	34
Figura 24 <i>Diagrama de Componentes</i> – Describe el diseño de la aplicación.....	35
Figura 25 <i>Diagrama de Estrategias</i> – Describe las estrategias que se utilizan para el desarrollo de la aplicación.....	36
Figura 26 <i>Vistas Arquitectónicas</i> – Describe de forma lógica los procesos de la aplicación. ....	38
Figura 27 <i>Vistas Arquitectónicas</i> – Describe de forma física el proceso que realizan los usuarios para el manejo de la aplicación.....	39
Figura 28 <i>Vistas Arquitectónicas</i> – Describe la vista de desarrollo de los procesos de la aplicación. ....	39

Figura 29 <i>Vistas Arquitectónicas</i> – Describe la vista de procesos de la aplicación mediante .....	40
Figura 30 <i>Fases para el Diseño de Interfaz de Usuario</i> – Esta figura explica las fases para realizar la interfaz de usuario. ....	42
Figura 31 <i>Interfaz de Logeo de Usuario</i> – Interfaz principal para ingreso de datos de usuario para entrar a la aplicación.....	42
Figura 32 <i>Interfaz de aviso de stock bajo en bodega</i> – En esta interfaz informa sobre los productos bajos en stock. ....	43
Figura 33 <i>Interfaz de Inicio</i> – Mediante esta interfaz podemos ingresar a los diferentes menús que tiene la aplicación. ....	43
Figura 34 <i>Cronograma de Actividades</i> – Describe todas las actividades realizadas para el desarrollo del proyecto. ....	54

## RESUMEN EJECUTIVO

El objetivo de este trabajo de grado consiste en la revisión de la materia prima existente en bodega, con el propósito de controlar la misma mediante el manejo adecuado y funcional del almacenaje y el aprovechamiento del espacio de la bodega.

Por esta razón es necesario establecer un sistema que controle y realice los procesos de inventario de materia prima en bodega permitiendo tener un registro de la materia prima, de sus proveedores, del kardex y de los usuarios que ingresan al sistema por motivos de seguridad.

Por otro lado nos permite confirmar que todas las transacciones realizadas en la aplicación hayan sido correctamente registradas y que el manejo físico de la materia prima en bodega coincida con los registros antes mencionados, ya que estos procesos se llevaran a cabo de forma Precisa, minimizando tiempos y será eficaz para la toma de decisiones.

## **ABSTRACT**

The aim of this work is the grade review of existing raw material warehouse, in order to control it through proper and functional storage management and use of the warehouse space.

It is therefore necessary to establish a system to monitor and perform processes raw material inventory in warehouse allowing to have a record of the raw material, its suppliers, the kardex and users entering the system for security reasons.

On the other hand allows us to confirm that all transactions in the application are correctly recorded and that the physical handling of the raw material in storage matches the aforementioned records, as these processes are carried out to the Point, minimizing time and will be effective for decision making.

## **Introducción**

El proceso de inventario se viene realizando desde tiempos remotos, pues en la antigüedad se almacenaban grandes cantidades ya sea de alimentos o de cosas que necesitaban en cada estación ya que el inventario es un método eficaz para evitar la escasez, del mismo modo en las empresas se usa para proveer materiales en el momento que los requieran.

Por otra parte se realiza inventarios por la razón económica que es la que les interesa más a las empresas.

Económicamente comprando en grandes cantidades debido a las políticas de los proveedores el costo es menor, también se ahorra en el momento de la fabricación.

## **CAPITULO I: ANTECEDENTES**

### **1.1 Contexto.**

“BUS CA PARTES” es un negocio familiar creado hace 8 años que se desenvuelve en el área de metal mecánica tiene amplia apertura en el ámbito comercial con varias carrocerías del Ecuador. Uno de sus principales objetivos es brindar el mejor servicio, buscando la perfección dentro de su campo laboral.

De acuerdo con este objetivo es importante desarrollar un aplicativo que ayude a controlar el uso de materia prima dentro de la empresa.

Presentando mejores resultados en su producción, ya que la falta de control ocasiona pérdidas debido al desperdicio de la materia prima así como también la mala distribución de la misma.

### **1.2 Justificación.**

Actualmente muchas empresas se han visto en la necesidad de buscar nuevas estrategias para mejorar sus movimientos económicos y satisfacer las necesidades tanto de la empresa como de sus clientes.

Para destacarse en el mundo comercial las empresas deben contar con un sistema de manejo de inventario permitiendo controlar sus materiales en almacén, garantizando una producción continua evitando retrasos e incumplimientos con los clientes.

Por otro lado se garantiza el control, planificación, suministro, distribución, mantenimiento, apoyo y gestión del sistema de inventario reduciendo de forma significativa sus costos de producción.

Por esta razón al notar los beneficios que ofrece la sistematización de inventario, tanto en la organización de la materia prima como en la producción hemos optado por la creación del mismo.

### **1.3 Definición Del Problema Central (MATRIZ T)**

En esta matriz se representa varios puntos como:

Situación empeorada = Que cree que falta para mejorar.

Situación actual = Que es lo que está sucediendo actualmente en la empresa.

Situación mejorada = Que se espera mejorar.

Fuerzas impulsadoras = Que ayuda a realizar el proyecto.

Fuerzas bloqueadoras = Que impide a realizar el proyecto.

I = Impacto.      PC = Patrón de cambio.

**Tabla 1. Análisis de Fuerzas T.**

ANÁLISIS DE FUERZAS T					
Situación Empeorada	Situación Actual				Situación Mejorada
La falta de organización y el desperdicio de la materia prima ocasionan retrasos en las entregas.	Falta de control en el proceso de inventarios.				Control actualizado de inventario, mejor organización, mejor distribución de materia prima, reduce costos generando ganancias y puntualidad con el cliente.
Fuerzas Impulsadoras	I	PC	I	PC	Fuerzas Bloqueadoras
Buena predisposición del Gerente Administrativo.	3	5	4	1	Mala predisposición del personal.
Existencia de herramientas tecnológicas.	4	5	3	1	Software desactualizado.
Buen nivel tecnológico por parte del encargado del inventario.	4	5	2	1	Espacio reducido para almacenamiento.
Alta necesidad de organizar.	4	5	4	1	No existe registro de materia prima.

*Notas: Matriz de análisis de fuerzas T. En esta matriz se detalla las fuerzas bloqueadoras que nos impiden lograr la situación mejorada y las fuerzas impulsadoras que nos ayudan a cumplir con la idea del proyecto.*

**1 = Bajo**

**2 = Medio Bajo**

**3 = Medio**

**4 = Medio Alto**

**5 = Alto**

## **CAPITULO II: ANÁLISIS DE INVOLUCRADOS**

### **2.01 Requerimientos**

#### **2.01.1 Descripción del sistema actual.**

La empresa carece de realizar inventarios físicos, lo que ocasiona demoras en las entregas, no existe control en los precios de compra o costos de producción, pérdida de tiempo buscando material, no hay control de la productividad.

#### **2.01.2 Visión y alcance**

Se espera mejorar el control de inventario de la materia prima, el orden, distribución en bodega y la eficiencia del personal en general.

Para esto se requiere que la aplicación trabaje de la siguiente manera:

- Funcionamiento óptimo de la aplicación.
- Reporte de existencias de materia prima.
- Reporte de salidas y entradas de materia prima.
- Requerimientos mínimos del sistema operativo.
- Mantenimiento del Sistema.
- Control del ingreso al software mediante usuarios y contraseña.

### 2.01.3 Entrevistas

**Tabla 2 Entrevista realizada al Gerente de la Empresa.**

Entrevistas		
Identificador: 001		
Preguntas	Objetivos	Análisis posterior
¿Quién tendrá acceso al sistema de inventario de materia prima?	Obtener el listado de las personas que tendrán acceso a la información.	Personas que tendrán acceso al sistema de inventario son: Gerente Administrador de bodega Administrador
¿A quién se compra la materia prima?	Obtener el nombre o la lista de proveedores.	Existe una lista de proveedores. Empresa puro acero. Hugo Guerrero. Mario Guerrero. Mundo Metal. Castillo Hermanos.
¿Qué control se ha establecido para detectar los desperdicios?	Conocer el método para detectar los desperdicios.	No se ha establecido ningún control.
¿Cuál es el proceso de recepción de la materia prima?	Conocer el proceso de recepción de la materia prima.	Se verifica que haya la misma cantidad de materia prima que aparece registrada en la factura con la que se encuentran recibiendo.
¿Se almacenan las existencias de una manera ordenada y sistemática?	Estar al tanto del proceso de almacenamiento de las existencias.	No por falta de conocimientos sobre control de inventario de materia prima.
¿Cuál es el proceso de fabricación?	Aprender sobre el proceso de fabricación.	Seleccionar la materia prima, luego se procede a realizar un cálculo de cuantas unidades salen de cada material y finalmente se procede a cortar y ensamblar todas las partes.

*Nota: Entrevista. Esta tabla detalla la entrevista realizada al Gerente para obtener el levantamiento de requerimientos.*

### 2.01.4 MATRIZ DE REQUERIMIENTOS.

**Tabla 3 Matriz de Requerimientos.**

*Ver Anexo (A.01.01)*

## 2.01.5 DESCRIPCIÓN DETALLADA.

**Tabla 4 Descripción detallada de los requerimientos funcionales.**

La aplicación debe emitir un reporte de existencias de materia prima actualizado		<b>Estado</b>	<b>Análisis</b>
<b>Creado por</b>	Karina Morocho	<b>Actualizado por</b>	Karina Morocho
<b>Fecha de creación</b>	18/06/2014	<b>Fecha de actualización</b>	23/06/2014
<b>Identificador</b>	RF001		
<b>Tipo de requerimiento</b>	Critico	<b>Tipo de requerimiento</b>	Funcional
<b>Datos de entrada</b>	Datos del producto y fecha		
<b>Descripción</b>	Una vez ingresado los datos del producto y la fecha el sistema mostrara automáticamente el reporte.		
<b>Datos de salida</b>	Reporte de existencias de materia prima		
<b>Resultados Esperados</b>	El resultado esperado es obtener información real de las existencias con las que se cuenta.		
<b>Origen</b>	Gerente		
<b>Dirigido a</b>	Encargado de bodega		
<b>Prioridad</b>	8		
<b>Requerimientos asociados</b>	Ninguno		
<b>ESPECIFICACIÓN</b>			
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. Para ejecutar el requerimiento el usuario debe estar registrado en el sistema.</li> <li>2. Una vez ingresado al sistema el usuario debe ingresar los datos del producto y la fecha.</li> </ol>		
<b>Poscondiciones</b>			
<b>Criterios de aceptación</b>	Permite que el usuario pueda realizar sus reportes de una manera fácil y con datos reales.		

*Nota: Descripción detallada de los requerimientos funcionales. Esta matriz detalla los requerimientos funcionales identificados.*

- 1-3 nivel de prioridad baja
- 4-6 nivel de prioridad media
- 7-10 nivel de prioridad alta

**Tabla 5 Descripción detallada de los requerimientos funcionales.**

La aplicación debe emitir un informe de la salida y entrada de materia prima		<b>Estado</b>	<b>Análisis</b>
<b>Creado por</b>	Karina Morocho	<b>Actualizado por</b>	Karina Morocho
<b>Fecha de creación</b>	18/06/2014	<b>Fecha de actualización</b>	23/06/2014
<b>Identificador</b>			
	RF002		
<b>Tipo de requerimiento</b>	Critico / No critico	<b>Tipo de requerimiento</b>	Funcional
<b>Datos de entrada</b>	Ingresar Fecha		
<b>Descripción</b>	El usuario debe ingresar la fecha para emitir informes de salidas y entradas de materia prima en bodega, ya que se emitirán informes de fechas pasadas o fecha actual.		
<b>Datos de salida</b>	Informe de salida y entrada de materia prima en bodega		
<b>Resultados Esperados</b>	El resultado esperado es tener un informe actualizado de los movimientos en bodega.		
<b>Origen</b>	Encargado de bodega		
<b>Dirigido a</b>	Gerente		
<b>Prioridad</b>	9		
<b>Requerimientos asociados</b>	Ninguno		
<b>ESPECIFICACIÓN</b>			
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. Para ejecutar el requerimiento el usuario debe estar registrado en el sistema.</li> <li>2. Luego de haber ingresado al sistema el usuario debe ingresar la fecha del informe que desee.</li> </ol>		
<b>Poscondiciones</b>			
<b>Criterios de aceptación</b>	Permite al encargado de bodega tener informes actualizados de los movimientos que se realizan en bodega.		

*Nota: Descripción detallada de los requerimientos funcionales. Esta matriz detalla los requerimientos funcionales identificados.*

1-3 nivel de prioridad baja  
4-6 nivel de prioridad media  
7-10 nivel de prioridad alta

**Tabla 6 Descripción detallada de los requerimientos funcionales.**

La aplicación debe contener un módulo para realizar el mantenimiento de la información.		<b>Estado</b>	<b>Análisis</b>
<b>Creado por</b>	Karina Morocho	<b>Actualizado por</b>	Karina Morocho
<b>Fecha de creación</b>	18/06/2014	<b>Fecha de actualización</b>	23/06/2014
<b>Identificador</b>	RF003		
<b>Tipo de requerimiento</b>	Critico	<b>Tipo de requerimiento</b>	Funcional
<b>Datos de entrada</b>	Usuario y password		
<b>Descripción</b>	El usuario debe tener los permisos para realizar estos mantenimientos o modificaciones de la información ingresando mediante su usuario y contraseña registrados en el sistema.		
<b>Datos de salida</b>			
<b>Resultados Esperados</b>	Con este requerimiento se mantendrá actualiza la información		
<b>Origen</b>	Encargado de bodega		
<b>Dirigido a</b>	Gerente		
<b>Prioridad</b>	9		
<b>Requerimientos asociados</b>	Ninguno		
<b>ESPECIFICACIÓN</b>			
<b>Precondiciones</b>	1. Entrar al sistema mediante usuario y password		
<b>Poscondiciones</b>			
<b>Criterios de aceptación</b>	Permite mantener la información actualizada.		

*Nota: Descripción detallada de los requerimientos funcionales. Esta matriz detalla los requerimientos funcionales identificados.*

- 1-3 nivel de prioridad baja
- 4-6 nivel de prioridad media
- 7-10 nivel de prioridad alta

**Tabla 7 Descripción detallada de los requerimientos funcionales.**

El control del sistema deberá ser mediante una cuenta con usuario y password		<b>Estado</b>	<b>Análisis</b>
<b>Creado por</b>	Karina Morocho	<b>Actualizado por</b>	Karina Morocho
<b>Fecha de creación</b>	18/06/2014	<b>Fecha de actualización</b>	23/06/2014
<b>Identificador</b>			
	RF004		
<b>Tipo de requerimiento</b>		<b>Tipo de requerimiento</b>	Funcional
<b>Datos de entrada</b>	Ingresar datos de usuario		
<b>Descripción</b>	El usuario debe ingresar sus datos para realizar su registro y así ingresar al sistema.		
<b>Datos de salida</b>	Confirmación de registro.		
<b>Resultados Esperados</b>	Este requerimiento permite controlar el ingreso de los usuarios salvaguardando la información.		
<b>Origen</b>	Administrador del sistema		
<b>Dirigido a</b>	Usuarios del sistema		
<b>Prioridad</b>	10		
<b>Requerimientos asociados</b>			
<b>ESPECIFICACIÓN</b>			
<b>Precondiciones</b>	1. Tener las autorizaciones previas al proceso de registro.		
<b>Poscondiciones</b>			
<b>Criterios de aceptación</b>	Permite mantener un control estricto del ingreso al sistema para salvaguardar la información.		

*Nota: Descripción detallada de los requerimientos funcionales. Esta matriz detalla los requerimientos funcionales identificados.*

- 1-3 nivel de prioridad baja
- 4-6 nivel de prioridad media
- 7-10 nivel de prioridad alta

**Tabla 8 Descripción detallada de los requerimientos no funcionales.**

Acceso desde internet.		<b>Estado</b>		<b>Análisis</b>	
<b>Creado por</b>	Karina Morocho	<b>Actualizado por</b>	Karina Morocho		
<b>Fecha de creación</b>	18/06/2014	<b>Fecha de actualización</b>	23/06/2014		
<b>Identificador</b>					
NRF001					
<b>Tipo de requerimiento</b>		No Critico	<b>Tipo de requerimiento</b>		No Funcional
<b>Datos de entrada</b>		URL			
<b>Descripción</b>		Ingresar la dirección de la pagina web.			
<b>Datos de salida</b>					
<b>Resultados Esperados</b>		Se puede acceder a la aplicación desde cualquier navegador de internet.			
<b>Origen</b>		Gerente			
<b>Dirigido a</b>		Sistema			
<b>Prioridad</b>		5			
<b>Requerimientos asociados</b>		Ninguno			
<b>ESPECIFICACIÓN</b>					
<b>Precondiciones</b>					
<b>Poscondiciones</b>					
<b>Criterios de aceptación</b>		Permite que los usuarios puedan utilizar la aplicación sin ningún percance.			

*Nota: Descripción detallada de los requerimientos no funcionales. Esta matriz detalla los requerimientos no funcionales identificados.*

- 1-3 nivel de prioridad baja
- 4-6 nivel de prioridad media
- 7-10 nivel de prioridad alta

## 2.02 MAPEO DE INVOLUCRADOS

Ver Anexo (A.01.02)

Figura 1 Mapeo de Involucrados – Describe los involucrados que están dentro del proyecto.

## 2.03 MATRIZ DE INVOLUCRADOS.

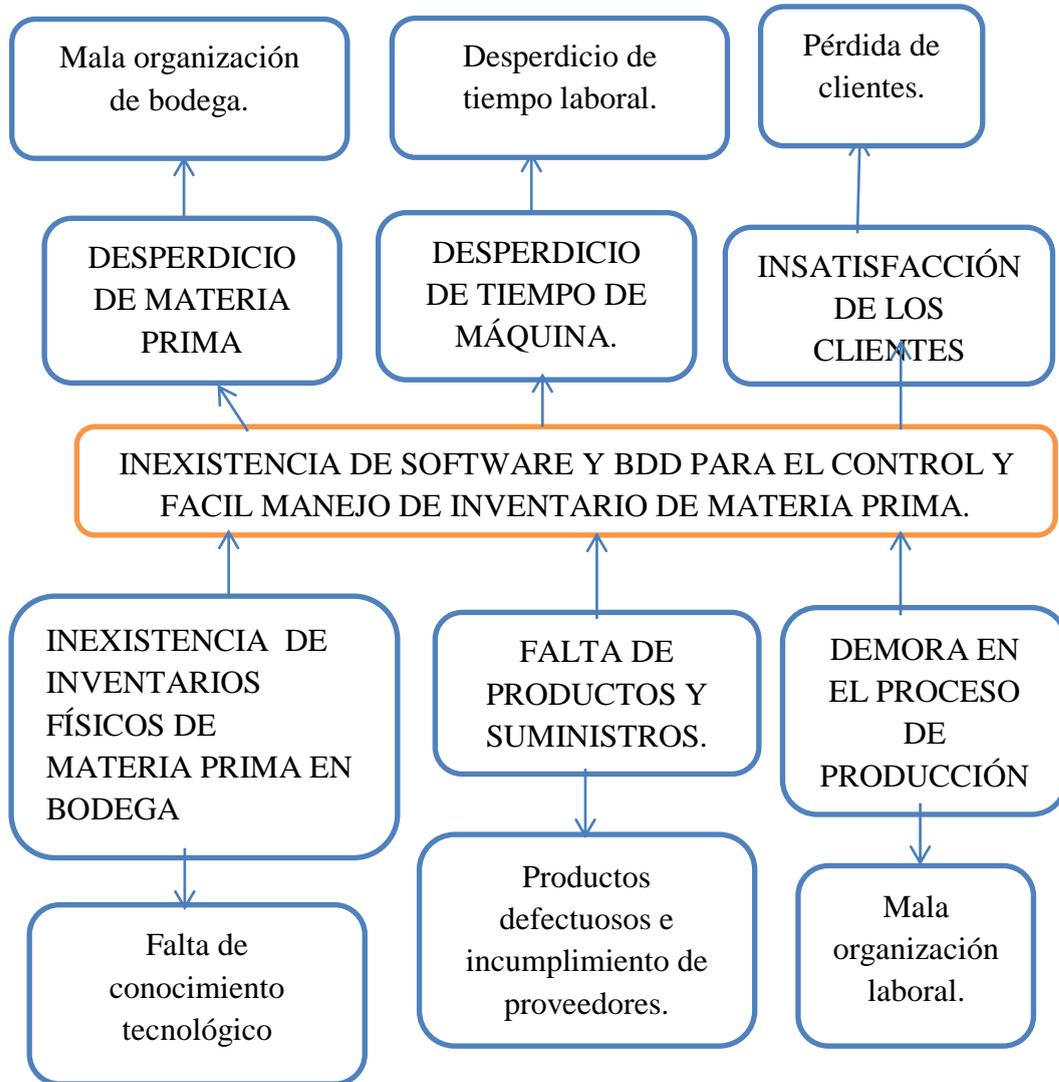
**Tabla 9 Matriz de involucrados.**

Ver Anexo (A.01.03)

## CAPITULO III: PROBLEMAS Y OBJETIVOS.

### 3.01 Árbol de Problemas

#### EFFECTOS

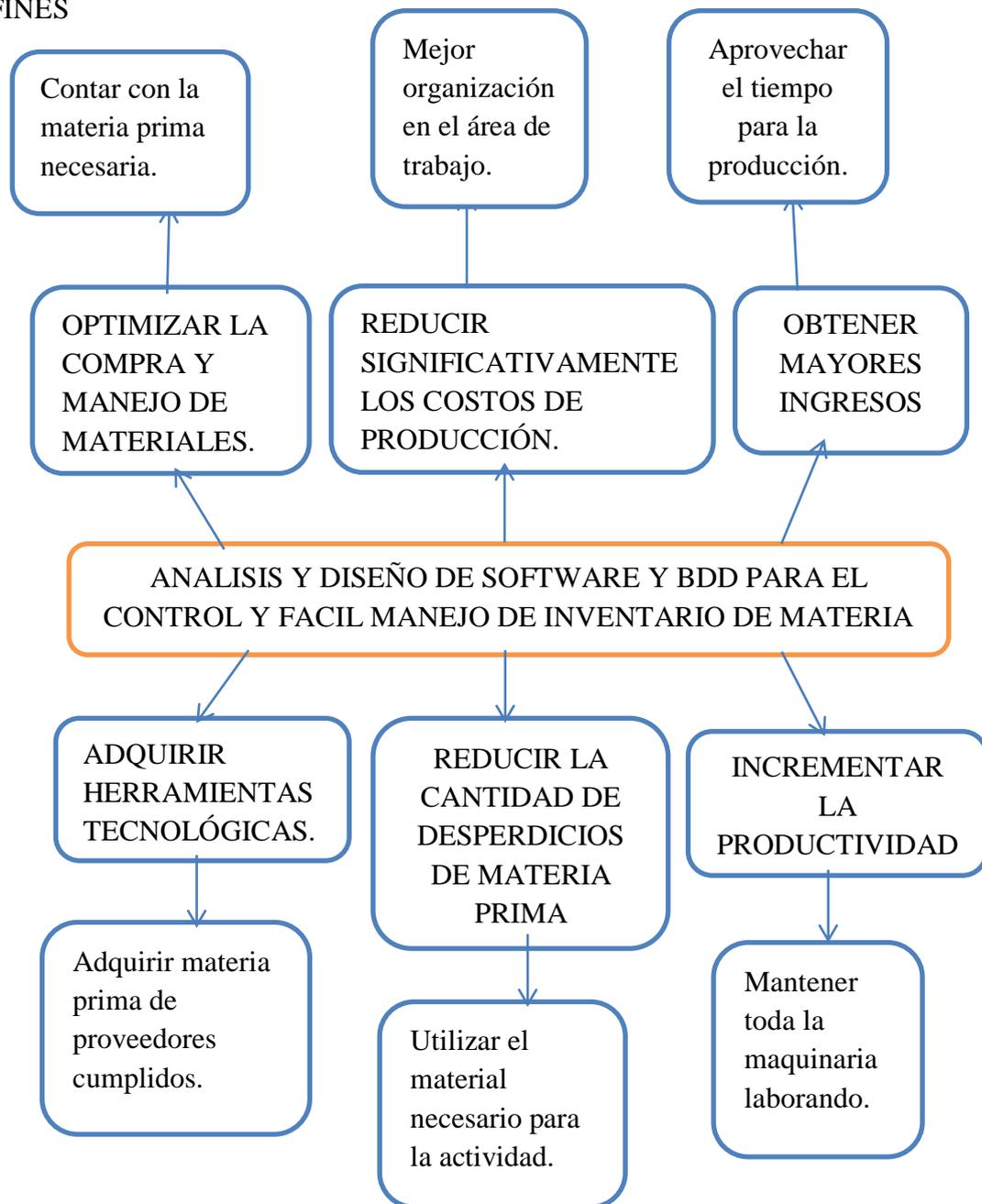


#### CAUSAS

Figura 2 Árbol de Problemas – Describe los problemas que se pueden presentar en el proceso del proyecto.

### 3.02 Árbol de Objetivos

FINES



MEDIOS

Figura 3 Árbol de Objetivos – Describe los objetivos planteados para el desarrollo del proyecto.

### 3.03 Diagrama de casos de uso.

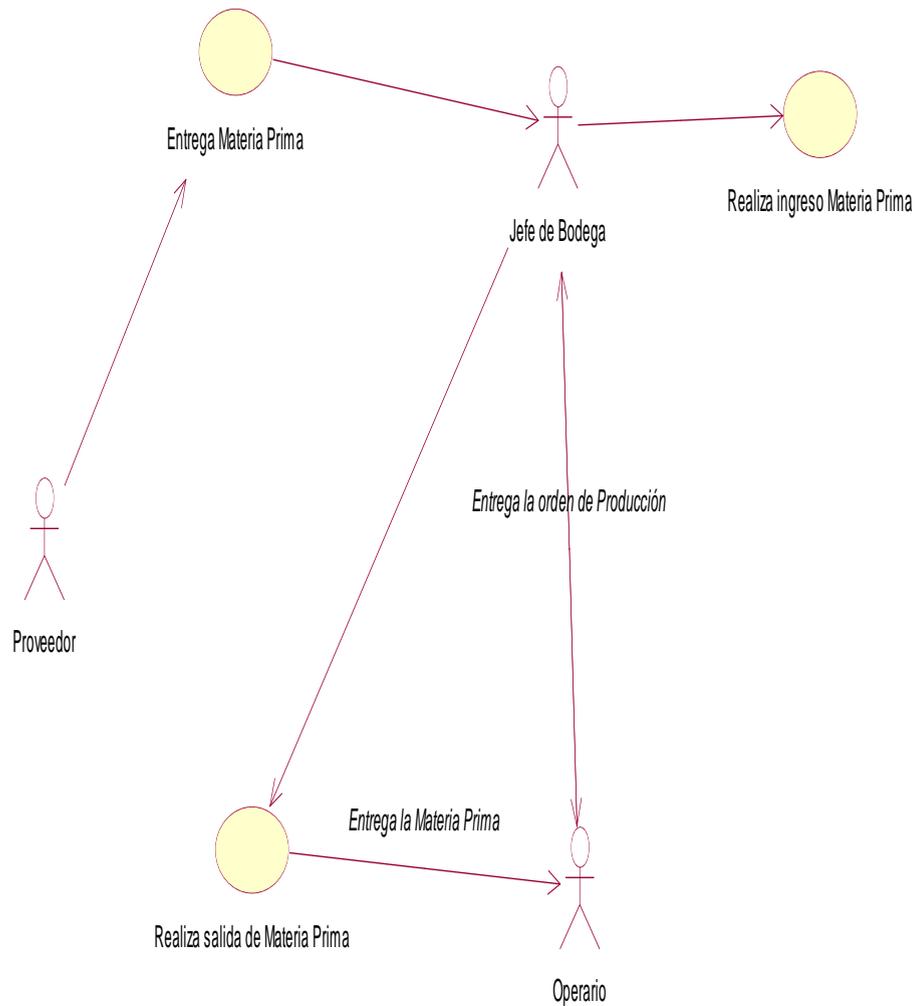


Figura 4 Diagrama de Casos de Uso General – Describe el proceso general que se realiza en la empresa.

**CU001**

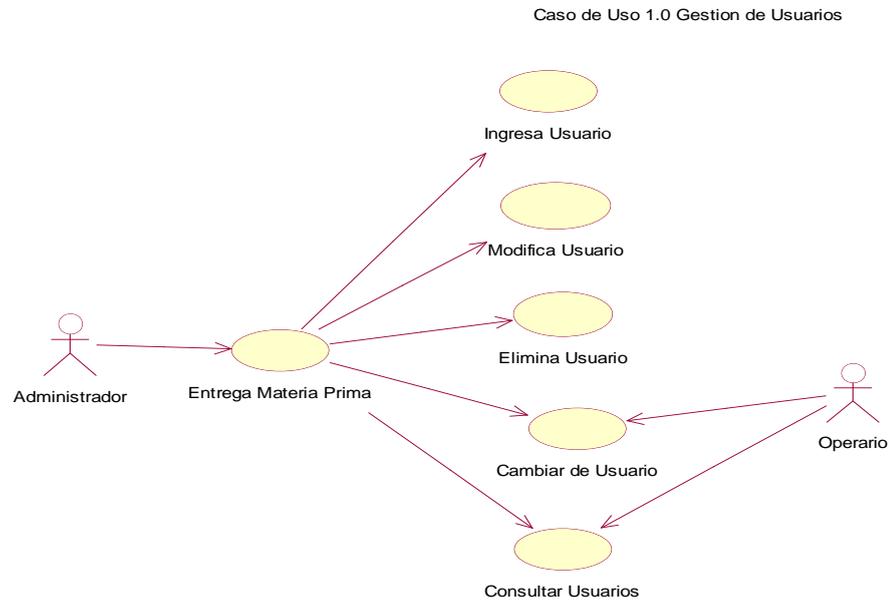


Figura 5 Diagrama de Casos de Uso 001 - Este caso de uso explica la gestión que realiza el administrador dentro de la aplicación con respecto a crear, modificar, eliminar o cambiar de usuario.

**CU002**

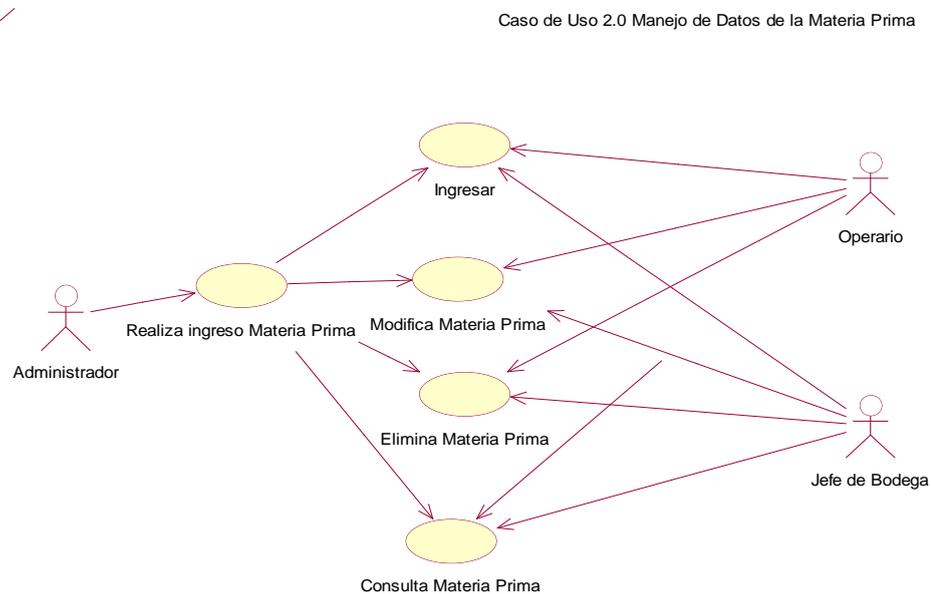


Figura 6 Diagrama de Casos de Uso 002 - Este caso de uso explica la gestión que realiza el administrador, Gerente y Jefe de Bodega dentro de la aplicación con respecto al registro, modificación, eliminación o consulta de materia prima.

**CU003**

Caso de Uso 3.0 Manejo de Datos de Proveedores

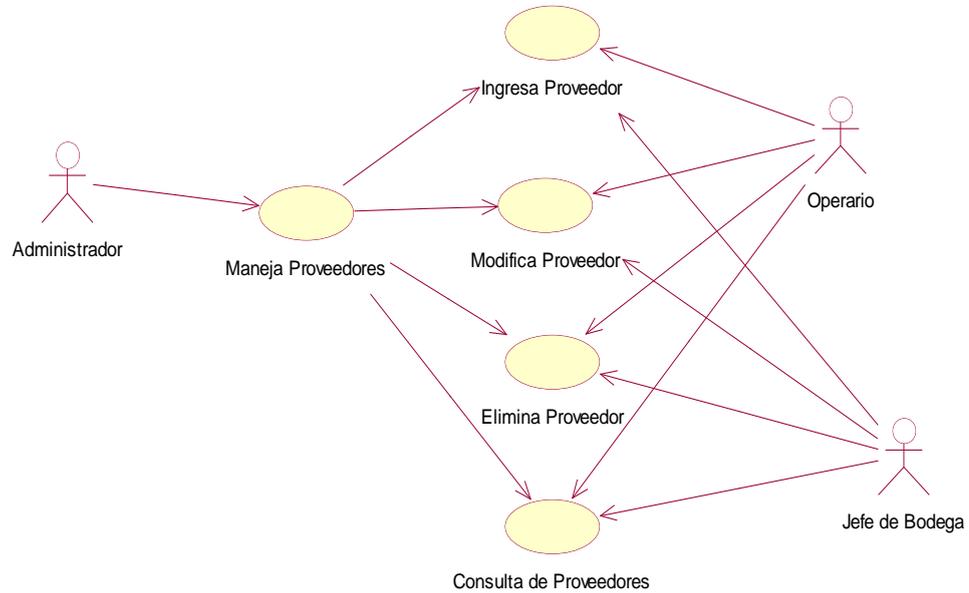


Figura 7 *Diagrama de Casos de Uso 003* - Este caso de uso explica la gestión que realiza el administrador, Gerente y Jefe de Bodega dentro de la aplicación con respecto al registro, modificación, eliminación o consulta de Proveedores.

**CU004**

Caso de Uso 4.0 Reportes de Entradas y Salidas de Materia Prima

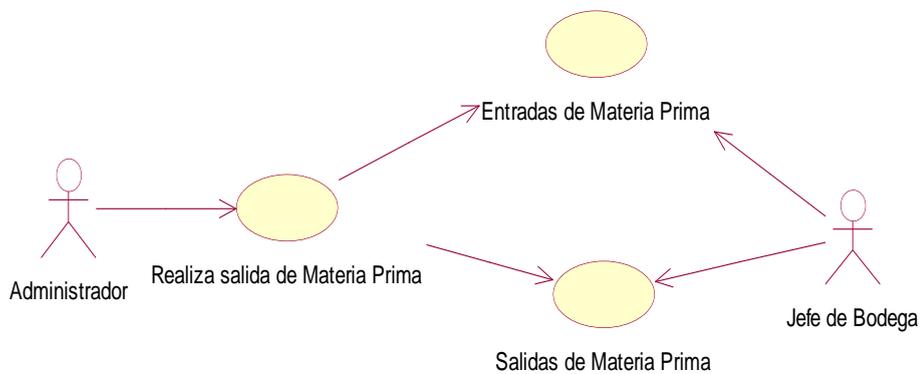


Figura 8 *Diagrama de Casos de Uso 004* - Este caso de uso explica el proceso para emitir los reportes.

**CU005**

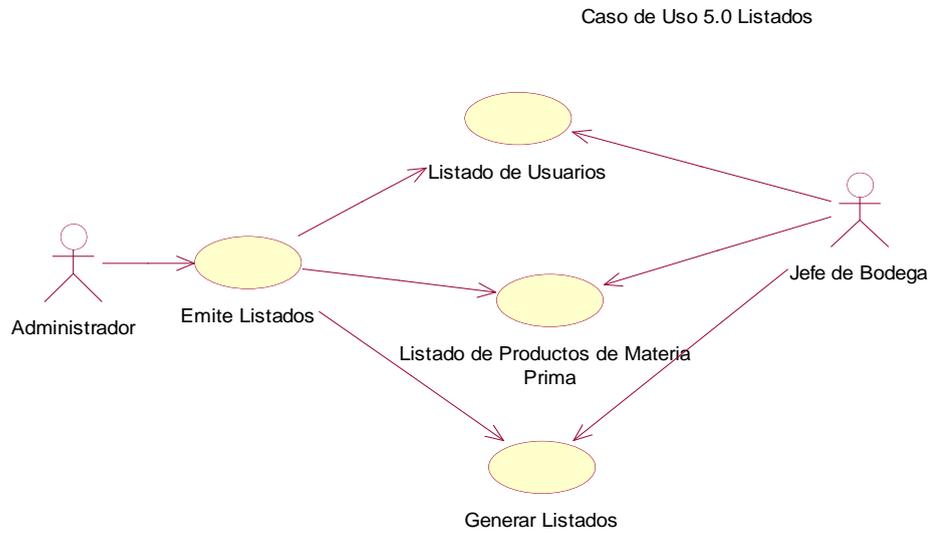


Figura 9 Diagrama de Casos de Uso 005 – Este caso de uso explica el proceso para generar listados.

**3.04 Casos de uso de realización.**

**C.U.R 001**

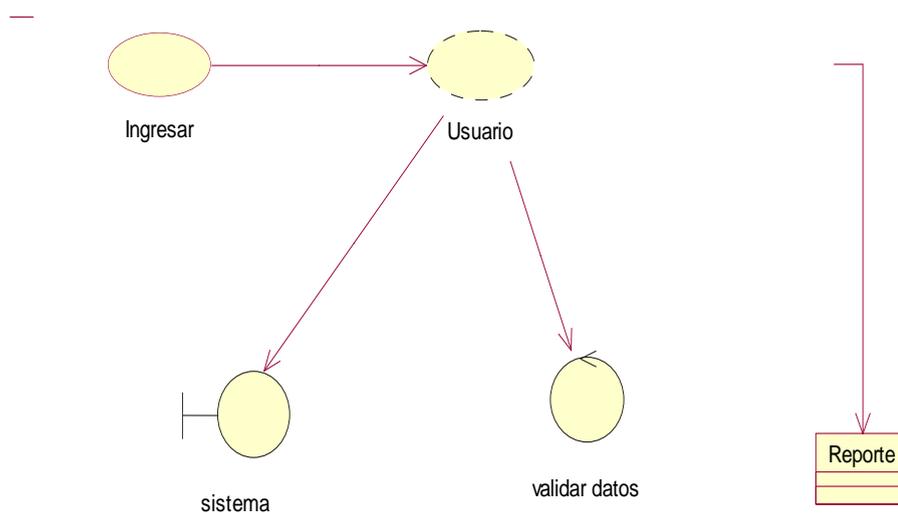


Figura 10 Diagrama de Casos de Realización 001 – Este caso de realización explica el registro y validación de datos del usuario.

### C.U.R 002

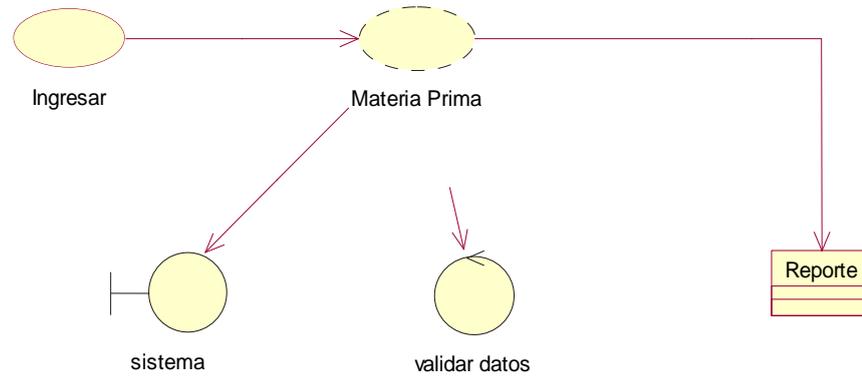


Figura 11 *Diagrama de Casos de Realización 002* – Este caso de realización explica el proceso para registrar la materia prima.

### C.U.R 003

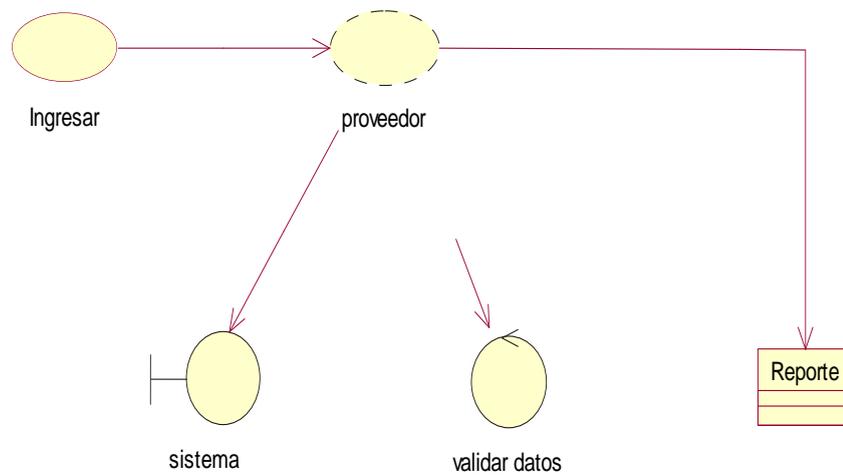


Figura 12 *Diagrama de Casos de Realización 003* - Este caso de realización explica el proceso para registrar proveedores.

**C.U.R 004**

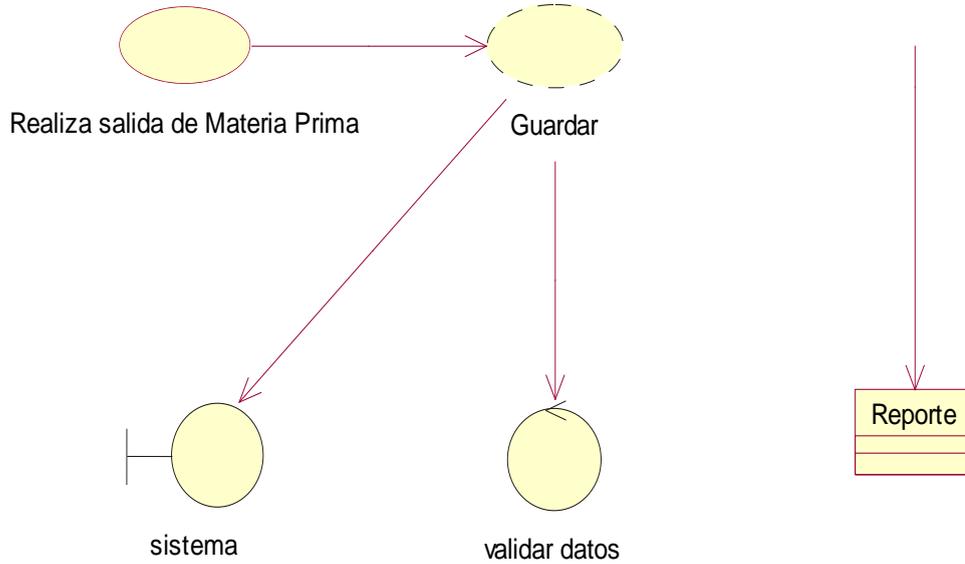


Figura 13 *Diagrama de Casos de Realización 004* - Este caso de realización explica el proceso para generar reportes.

**C.U.R 005**

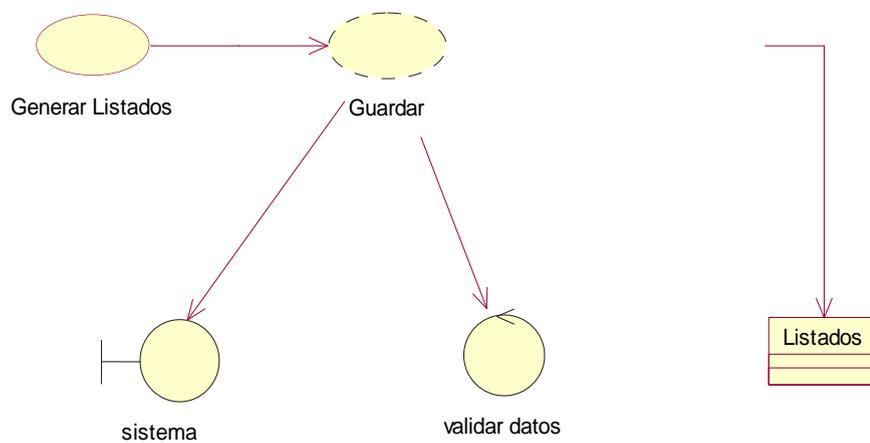


Figura 14 *Diagrama de Casos de Realización 005* - Este caso de realización explica el proceso para generar listados.

### 3.05 Especificación de Casos de uso de realización.

**Tabla 10 Especificación de Casos de uso de realización.**

<b>Nombre</b>	Ingresar Usuario
<b>Identificador</b>	C.U.R 001
<b>Responsabilidades</b>	Ingresar los datos del usuario, valida la información proporcionada, si existe ingresa a la aplicación, caso contrario procede a crear.
<b>Tipo</b>	Sistema
<b>Referencias Caso de Uso</b>	C.U 001
<b>Referencias Requisitos</b>	No tiene
<b>PRECONDICIONES</b>	
<b>De Instancia</b>	
El administrador ingresa los datos del usuario, valida la información proporcionada, si existe ingresa a la aplicación, caso contrario procede a crear.	
<b>De Relación.</b>	
<b>POSCONDICIONES</b>	
<b>De Instancia.</b>	
<b>De Relación.</b>	
<b>SALIDAS DE PANTALLA</b>	
Datos Guardados Correctamente.	

*Notas: Esta tabla describe la Especificación de los Casos de Realización en cuanto al registro de usuarios.*

**Tabla 11 Especificación de Casos de uso de realización.**

<b>Nombre</b>	Ingresar Materia Prima
<b>Identificador</b>	C.U.R 002
<b>Responsabilidades</b>	Ingresar los datos de materia prima, valida, procede a crear.
<b>Tipo</b>	Sistema
<b>Referencias Caso de Uso</b>	C.U 002
<b>Referencias Requisitos</b>	
<b>PRECONDICIONES</b>	
<b>De Instancia</b>	
Ingresar los datos de materia prima, valida la información proporcionada, si ya existe emite un error de materia prima ya creada, caso contrario procede a registrar.	
<b>De Relación.</b>	
<b>POSCONDICIONES</b>	
<b>De Instancia</b>	
<b>De Relación.</b>	
<b>SALIDAS DE PANTALLA</b>	
Datos Guardados Correctamente.	

*Notas: Esta tabla describe la Especificación de los Casos de Realización en cuanto al registro de materia prima.*

**Tabla 12 Especificación de Casos de uso de realización.**

<b>Nombre</b>	Ingresar Proveedor
<b>Identificador</b>	C.U.R 003
<b>Responsabilidades</b>	Ingresar los datos de proveedor, valida, procede a registrar.
<b>Tipo</b>	Sistema
<b>Referencias Caso de Uso</b>	C.U 003
<b>Referencias Requisitos</b>	
<b>PRECONDICIONES</b>	
<b>De Instancia</b>	
Ingresar los datos de proveedor, valida la información proporcionada, si ya existe emite un error de proveedor existente, caso contrario procede a registrar.	
<b>De Relación.</b>	
<b>POSCONDICIONES</b>	
<b>De Instancia</b>	
<b>De Relación.</b>	
<b>SALIDAS DE PANTALLA</b>	
Datos Guardados Correctamente.	
<i>Notas: Esta tabla describe la Especificación de los Casos de Realización en cuanto al registro de proveedores.</i>	

**Tabla 13 Especificación de Casos de uso de realización.**

<b>Nombre</b>	Emitir Reportes.
<b>Identificador</b>	C.U.R 004
<b>Responsabilidades</b>	Ingresar los datos de búsqueda para el reporte.
<b>Tipo</b>	Sistema
<b>Referencias Caso de Uso</b>	C.U 004
<b>Referencias Requisitos</b>	
<b>PRECONDICIONES</b>	
<b>De Instancia</b>	
Ingresar los parámetros de búsqueda, valida la información proporcionada y emite el reporte, lo imprime si lo desea.	
<b>De Relación.</b>	
<b>POSCONDICIONES</b>	
<b>De Instancia</b>	
<b>De Relación.</b>	
<b>SALIDAS DE PANTALLA</b>	
Emisión de reporte correcto.	
<i>Notas: Esta tabla describe la Especificación de los Casos de Realización en cuanto a la emisión de reportes.</i>	

**Tabla 14 Especificación de Casos de uso de realización.**

<b>Nombre</b>	Emitir Listados.
<b>Identificador</b>	C.U.R 005
<b>Responsabilidades</b>	Ingresa los datos de búsqueda.
<b>Tipo</b>	Sistema
<b>Referencias Caso de Uso</b>	C.U 005
<b>Referencias Requisitos</b>	
<b>PRECONDICIONES</b>	
<b>De Instancia</b>	
Ingresar parámetros de búsqueda, valida la información proporcionada, emite el listado y lo imprime si lo desea.	
<b>De Relación.</b>	
<b>POSCONDICIONES</b>	
<b>De Instancia</b>	
<b>De Relación.</b>	
<b>SALIDAS DE PANTALLA</b>	
Listado Correcto.	

*Notas: Esta tabla describe la Especificación de los Casos de Realización en cuanto a la emisión de listados.*

### 3.06 Diagrama de secuencia del sistema.

#### D.S 001

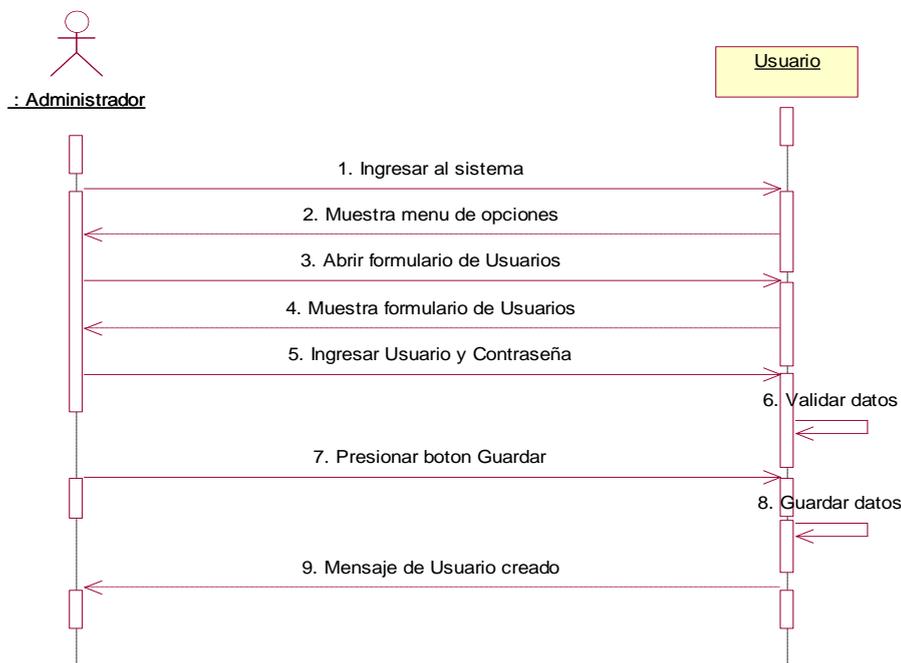


Figura 15 Diagrama de Secuencia 001 - Este diagrama de secuencia explica el proceso para registrar usuarios.

**D.S 002**

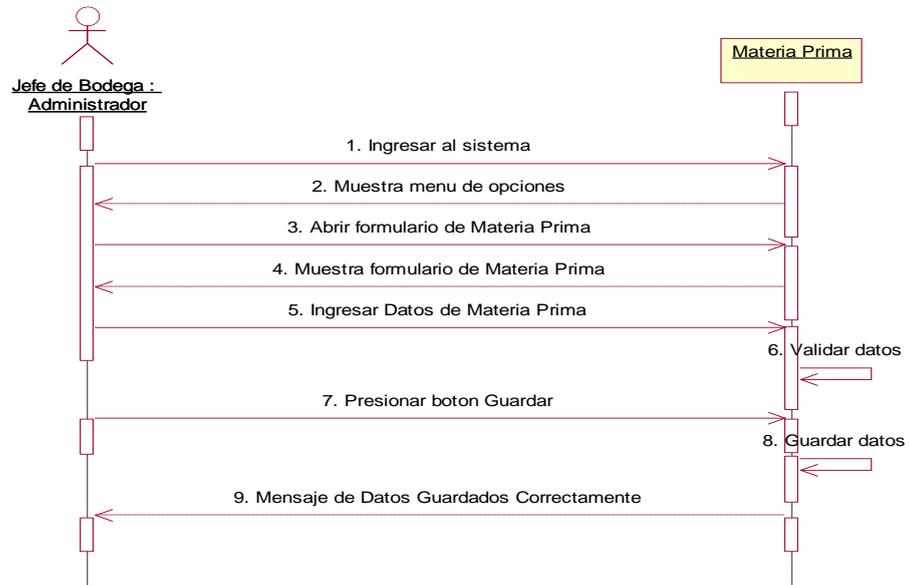


Figura 16 *Diagrama de Secuencia 002* - Este diagrama de secuencia explica el proceso para registrar la materia prima.

**D.S 003**

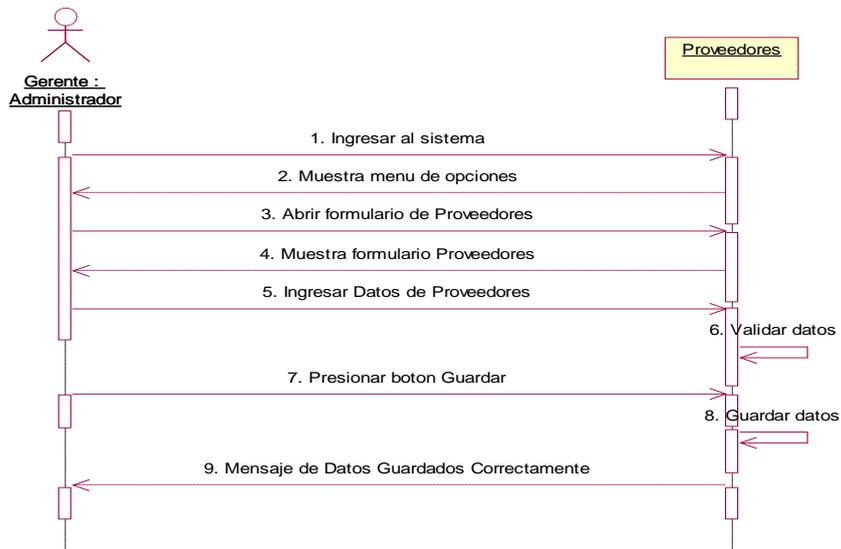


Figura 17 *Diagrama de Secuencia 003* - Este diagrama de secuencia explica el proceso para registrar proveedores.

**D.S 004**

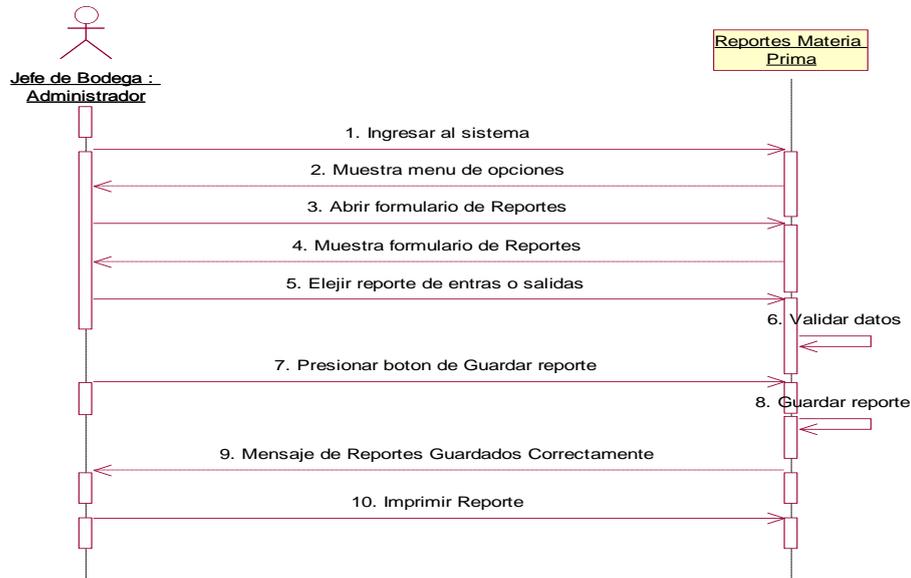


Figura 18 *Diagrama de Secuencia 004* - Este diagrama de secuencia explica el proceso para emitir reportes.

**D.S 005**

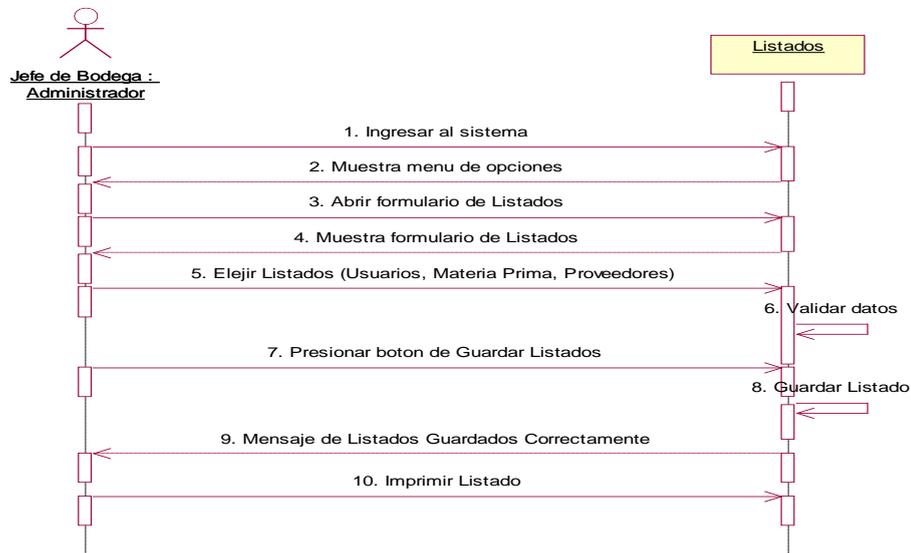


Figura 19 *Diagrama de Secuencia 005* - Este diagrama de secuencia explica el proceso para generar listados.

### 3.07 Especificación de casos de uso.

**Tabla 15 Especificación de casos de Uso.**

<b>Creación de Usuario</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Administrador
Actor Secundario:	Gerente o Jefe de Bodega.
Precondición:	Logeo en el sistema como administrador.
Descripción:	Crea usuario según el área de trabajo o departamento.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Click en la opción crear usuario</li> <li>2. Ingresar los datos del usuario como: cedula, nombre, fecha de nacimiento, estado civil, teléfono, dirección, etc.</li> <li>3. El sistema valida que los datos proporcionados sean correctos.</li> <li>4. El sistema muestra un mensaje de confirmación.</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
3a. Si los datos proporcionados son incorrectos el sistema emitirá un mensaje de error.	

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto al registro de usuarios.*

**Tabla 16 Especificación de casos de Uso.**

<b>Modificación de Usuario</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Administrador
Actor Secundario:	Gerente o Jefe de Bodega.
Precondición:	Logeo en el sistema como administrador.
Descripción:	Modifica o actualiza la información del usuario.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Ir al menú principal</li> <li>2. Click en el botón modificar</li> <li>3. Ingresar parámetros de búsqueda</li> <li>4. Modificar los datos deseados</li> <li>5. Click en el botón guardar cambios</li> <li>6. Validar los datos cambiados</li> <li>7. Mostrar mensaje de usuario modificado correctamente</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
6a. Si los datos proporcionados son incorrectos el sistema emitirá un mensaje de error.	

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la modificación de datos de los usuarios.*

**Tabla 17 Especificación de casos de Uso.**

<b>Eliminación de Usuario</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Administrador
Actor Secundario:	Gerente o Jefe de Bodega.
Precondición:	Logeo en el sistema como administrador.
Descripción:	Cambia el estado de un usuario de activo a inactivo (no se realiza eliminación ya que es necesario para la bitácora de auditoria).
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Ir al menú principal</li> <li>2. Click en el botón eliminar</li> <li>3. Ingresar parámetros de búsqueda</li> <li>4. Click en el botón eliminar o cancelar</li> <li>5. Actualizar la base de datos</li> <li>6. Mostrar mensaje de usuario eliminado</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
<ol style="list-style-type: none"> <li>4a. Si se presiona el botón eliminar se debe validar los datos</li> <li>4b. Si se presiona el botón cancelar cargar el menú principal</li> </ol>	

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la eliminación de usuarios.*

**Tabla 18 Especificación de casos de Uso.**

<b>Registro de un Producto de Materia Prima</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Jefe de Bodega
Actor Secundario:	Gerente.
Precondición:	Logeo en el sistema.
Descripción:	Registra un nuevo producto de materia prima.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Click en la opción registro de materia prima.</li> <li>2. Ingresar los datos de materia prima como: código, nombre, descripción, etc.</li> <li>3. El sistema valida que los datos proporcionados sean correctos.</li> <li>4. El sistema muestra un mensaje de confirmación.</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
<ol style="list-style-type: none"> <li>3a. Si los datos proporcionados son incorrectos el sistema emitirá un mensaje de error.</li> </ol>	

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto al registro de la materia prima.*

**Tabla 19 Especificación de casos de Uso.**

<b>Modificación de datos de Materia Prima</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Jefe de Bodega
Actor Secundario:	Gerente.
Precondición:	Logeo en el sistema.
Descripción:	Modifica o actualiza la información de Materia Prima.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Ir al menú principal</li> <li>2. Click en el botón modificar Materia Prima.</li> <li>3. Ingresar parámetros de búsqueda.</li> <li>4. Modificar los datos deseados.</li> <li>5. Click en el botón guardar cambios.</li> <li>6. Validar los datos cambiados.</li> <li>7. Mostrar mensaje de datos modificados correctamente.</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
6a. Si los datos proporcionados son incorrectos el sistema emitirá un mensaje de error.	
<i>Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la modificación de datos de la materia prima.</i>	

**Tabla 20 Especificación de casos de Uso.**

<b>Eliminación de Materia Prima (Cambio de Estado Activo e Inactivo).</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Jefe de Bodega
Actor Secundario:	Gerente.
Precondición:	Logeo en el sistema.
Descripción:	Cambia el estado de la materia prima de activo a inactivo (no se realiza eliminación ya que es necesario para la bitácora de auditoría).
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Ir al menú principal</li> <li>2. Click en el botón eliminar o estado de materia prima.</li> <li>3. Ingresar parámetros de búsqueda.</li> <li>4. Click en el botón eliminar o cancelar.</li> <li>5. Actualizar la base de datos.</li> <li>6. Mostrar mensaje de estado cambiado.</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
4a. Si se presiona el botón cambiar estado se debe validar los datos.	
4b. Si se presiona el botón cancelar cargar el menú principal.	
<i>Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la eliminación de materia prima.</i>	

**Tabla 21 Especificación de casos de Uso.**

<b>Registro de nuevo Proveedor</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Gerente
Actor Secundario:	Jefe de Bodega
Precondición:	Logeo en el sistema.
Descripción:	Se registra un nuevo Proveedor de materia prima.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Click en la opción crear o registrar proveedor</li> <li>2. Ingresar los datos del proveedor como: cedula o ruc, nombre, teléfono, dirección, etc.</li> <li>3. El sistema valida que los datos proporcionados sean correctos.</li> <li>4. El sistema muestra un mensaje de confirmación.</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
3a. Si los datos proporcionados son incorrectos el sistema emitirá un mensaje de error.	
<i>Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto al registro de proveedores.</i>	

**Tabla 22 Especificación de casos de Uso.**

<b>Modificación de Proveedor</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Gerente
Actor Secundario:	Jefe de Bodega
Precondición:	Logeo en el sistema.
Descripción:	Modifica o actualiza la información del Proveedor.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Ir al menú principal</li> <li>2. Click en el botón modificar</li> <li>3. Ingresar parámetros de búsqueda</li> <li>4. Modificar los datos deseados</li> <li>5. Click en el botón guardar cambios</li> <li>6. Validar los datos cambiados</li> <li>7. Mostrar mensaje de usuario modificado correctamente</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
6a. Si los datos proporcionados son incorrectos el sistema emitirá un mensaje de error.	
<i>Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la modificación de datos de proveedores.</i>	

**Tabla 23 Especificación de casos de Uso.**

<b>Eliminación de Proveedor (Cambio de Estado Activo e Inactivo).</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Gerente
Actor Secundario:	Jefe de Bodega.
Precondición:	Logeo en el sistema.
Descripción:	Cambia el estado del Proveedor de activo a inactivo (no se realiza eliminación ya que es necesario para la bitácora de auditoria).

Escenario Normal:

1. Ir al menú principal
2. Click en el botón eliminar o estado de Proveedor.
3. Ingresar parámetros de búsqueda.
4. Click en el botón eliminar o cancelar.
5. Actualizar la base de datos.
6. Mostrar mensaje de estado cambiado.

Poscondición:	El usuario debe tener creada una cuenta en el sistema
---------------	---

Escenario Alternativo:

- 4a. Si se presiona el botón cambiar estado se debe validar los datos.
- 4b. Si se presiona el botón cancelar cargar el menú principal.

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la eliminación de proveedores.*

**Tabla 24 Especificación de casos de Uso.**

<b>Generar Reportes (Entradas y salidas de Materia Prima)</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Jefe de Bodega.
Actor Secundario:	Gerente.
Precondición:	Logeo en el sistema.
Descripción:	Ingresar parámetros de búsqueda, verifica lo guarda.

Escenario Normal:

1. Ir al menú principal
2. Click en el botón generar reporte.
3. Ingresar parámetros de búsqueda.
4. Click en el botón guardar.
5. Mostrar mensaje de reporte correcto.

Poscondición:	El usuario debe tener creada una cuenta en el sistema
---------------	---

Escenario Alternativo:

- 4a. El usuario tiene la opción de imprimir su reporte.

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la Obtención de Reportes.*

**Tabla 25 Especificación de casos de Uso.**

<b>Emitir Listados (Usuarios, Proveedores y Materia Prima)</b>	
Fecha:	19/07/2014
Autores:	Karina Morocho
Actor Principal:	Jefe de Bodega.
Actor Secundario:	Gerente.
Precondición:	Logeo en el sistema.
Descripción:	Ingresar parámetros de búsqueda, verifica lo guarda.
Escenario Normal:	
<ol style="list-style-type: none"> <li>1. Ir al menú principal</li> <li>2. Click en el botón generar listados.</li> <li>3. Ingresar parámetros de búsqueda.</li> <li>4. Click en el botón guardar.</li> <li>5. Mostrar mensaje de listado correcto.</li> </ol>	
Poscondición:	El usuario debe tener creada una cuenta en el sistema
Escenario Alternativo:	
4a. El usuario tiene la opción de imprimir su listado.	

*Notas: Esta tabla describe la Especificación de los Casos de Uso en cuanto a la Obtención de Listados.*

## CAPITULO IV: ANÁLISIS DE ALTERNATIVAS.

### 4.01 Matriz de análisis de alternativas.

#### Tabla 26 *Matriz de análisis de alternativas.*

*Ver Anexo (A.01.04)*

### 4.02 Matriz de Impactos de Objetivos.

#### Tabla 27 *Matriz de Impacto de Objetivos.*

*Ver Anexo (A.01.05)*

### 4.03 Estándares para el Diseño de Clases.

A continuación se describen los estándares para el diseño de clases.

#### “CLASES

Es la unidad básica que encapsula toda la información de un objeto (un objeto es una instancia de una clase).

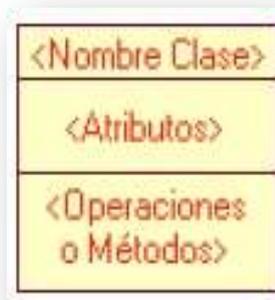


Figura 20 *Diseño de tabla* – Describe el modelo de una tabla.

## ATRIBUTOS.

Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:

**Public**  + Público: Acceso público desde cualquier lugar.

**Private**  - Privado: Acceso solo dentro de la clase.

**Protected**  # Protegido: No será accesible desde fuera de la clase pero si podrá ser accesado por métodos de la clase.

## MÉTODOS.

Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, sus características son:

**Public**  + Público: Acceso público desde cualquier lugar.

**Private**  - Privado: Acceso solo dentro de la clase.

**Protected**  # Protegido: No será accesible desde fuera de la clase pero si podrá ser accesado por métodos de la clase". (Youtube 2014)

“Partes de una clase”. (Youtube 2014)

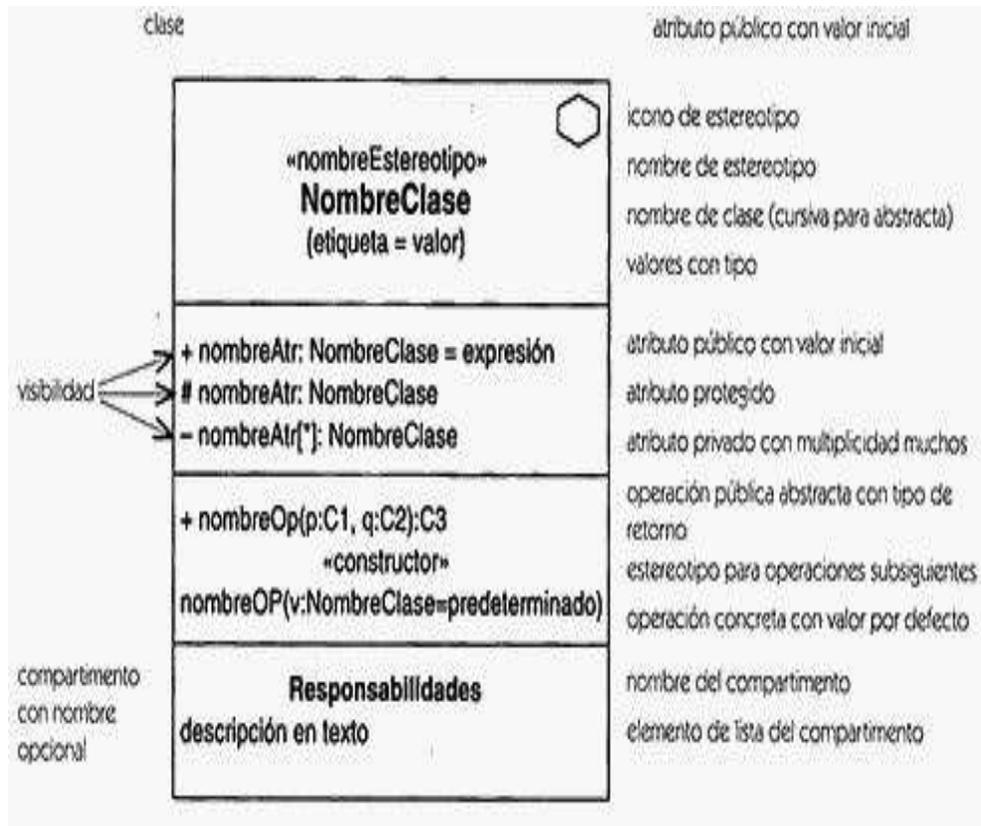


Figura 21 *Partes de una clase* – Describe las partes que contiene una clase.

#### 4.04 Diagrama de Clases.

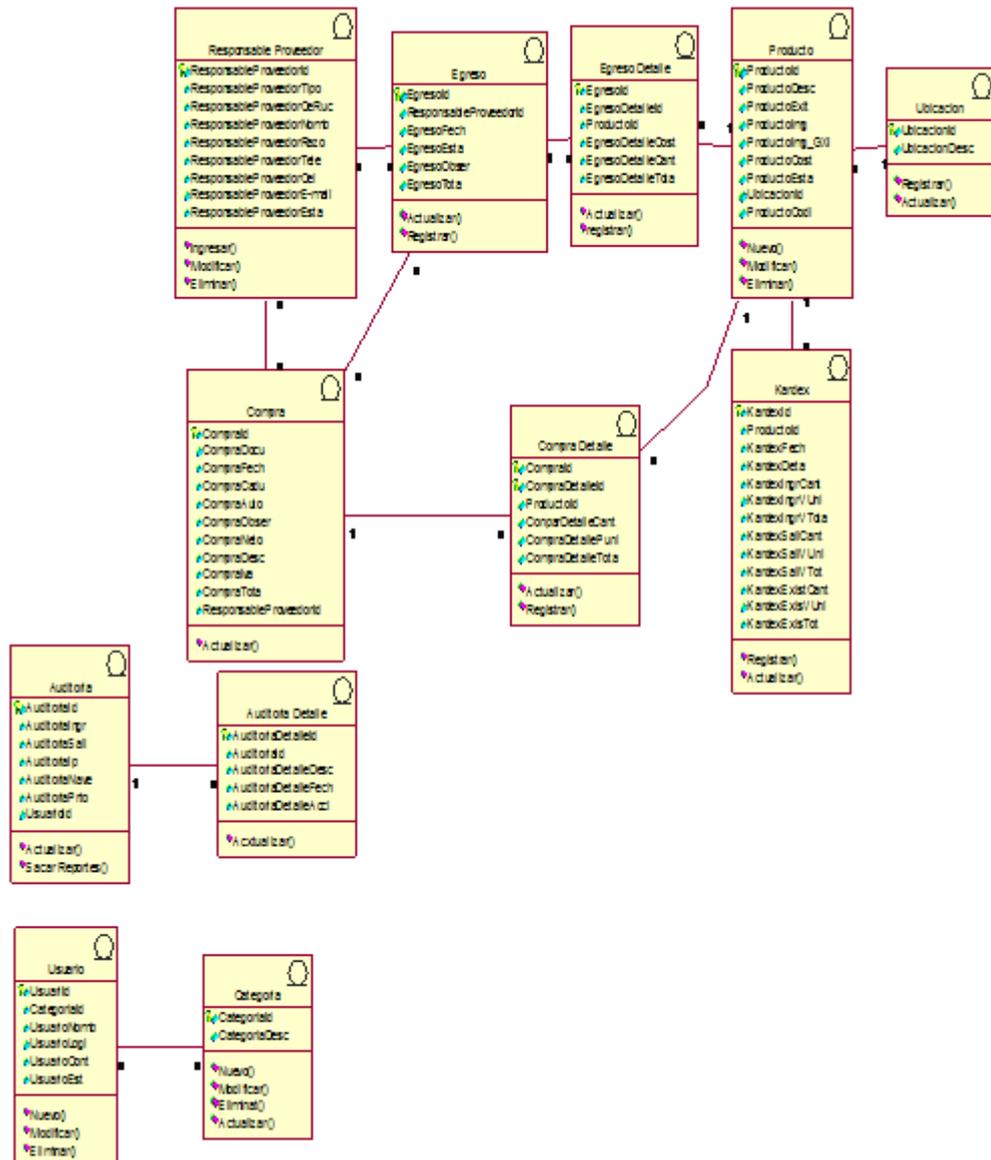


Figura 22 Diagrama de Clase – Describe las tablas del diseño de la Base de Datos.

#### 4.05 Modelo Lógico - Físico.

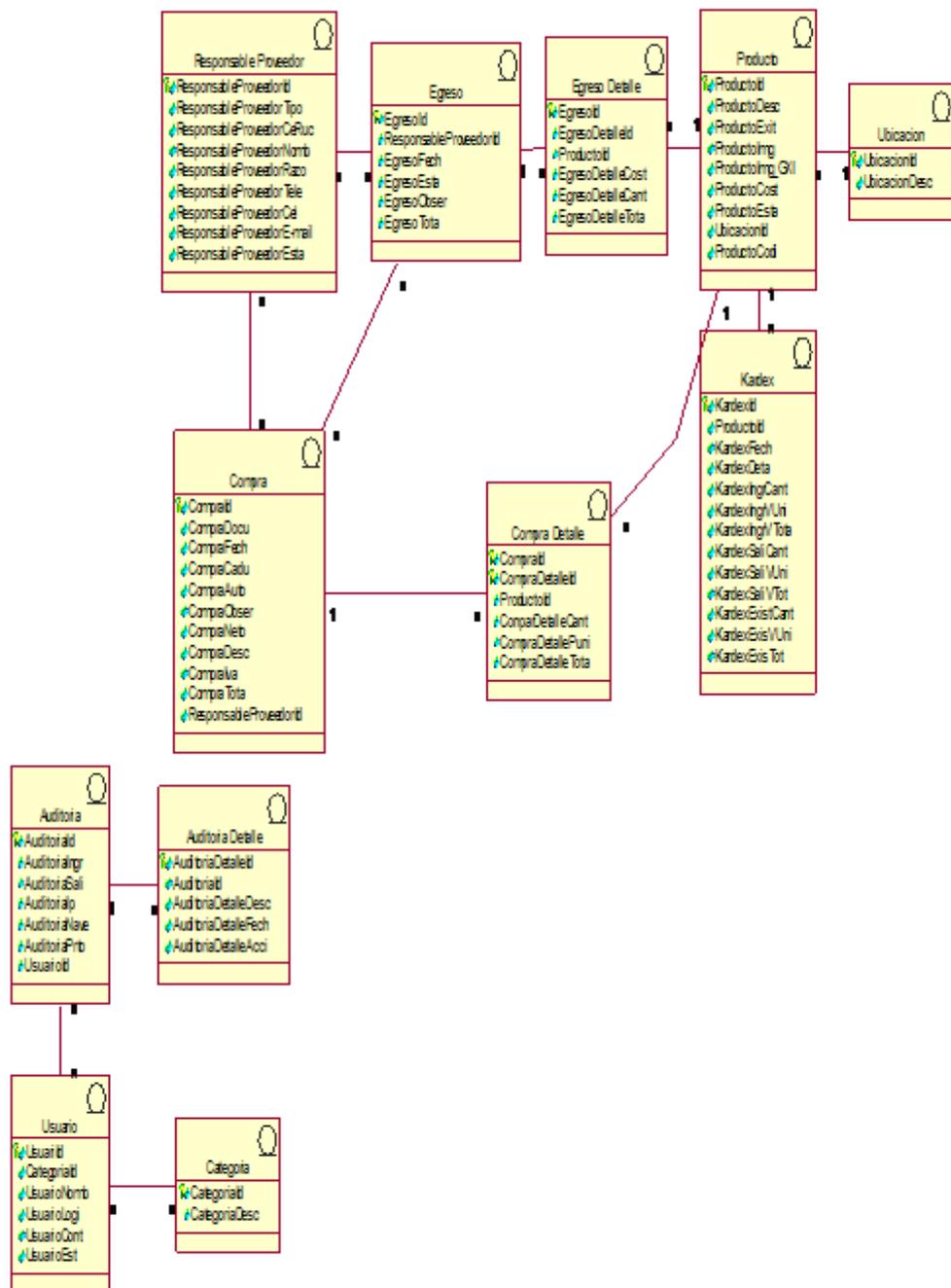


Figura 23 Diagrama de Clases Modelo Lógico – Físico - Describe las tablas del diseño de la Base de Datos en modelo lógico – físico.

#### 4.06 Diagrama de Componentes.

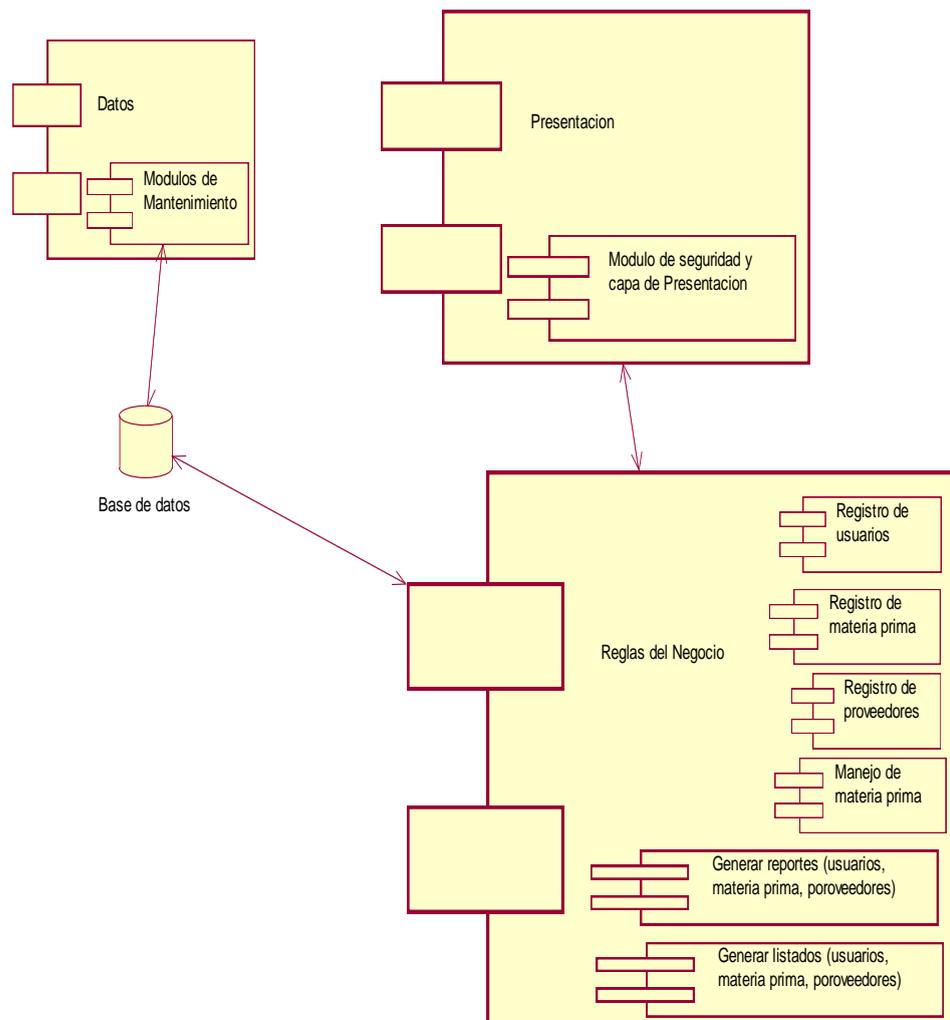


Figura 24 *Diagrama de Componentes* – Describe el diseño de la aplicación.

#### 4.07 Diagramas de Estrategias.

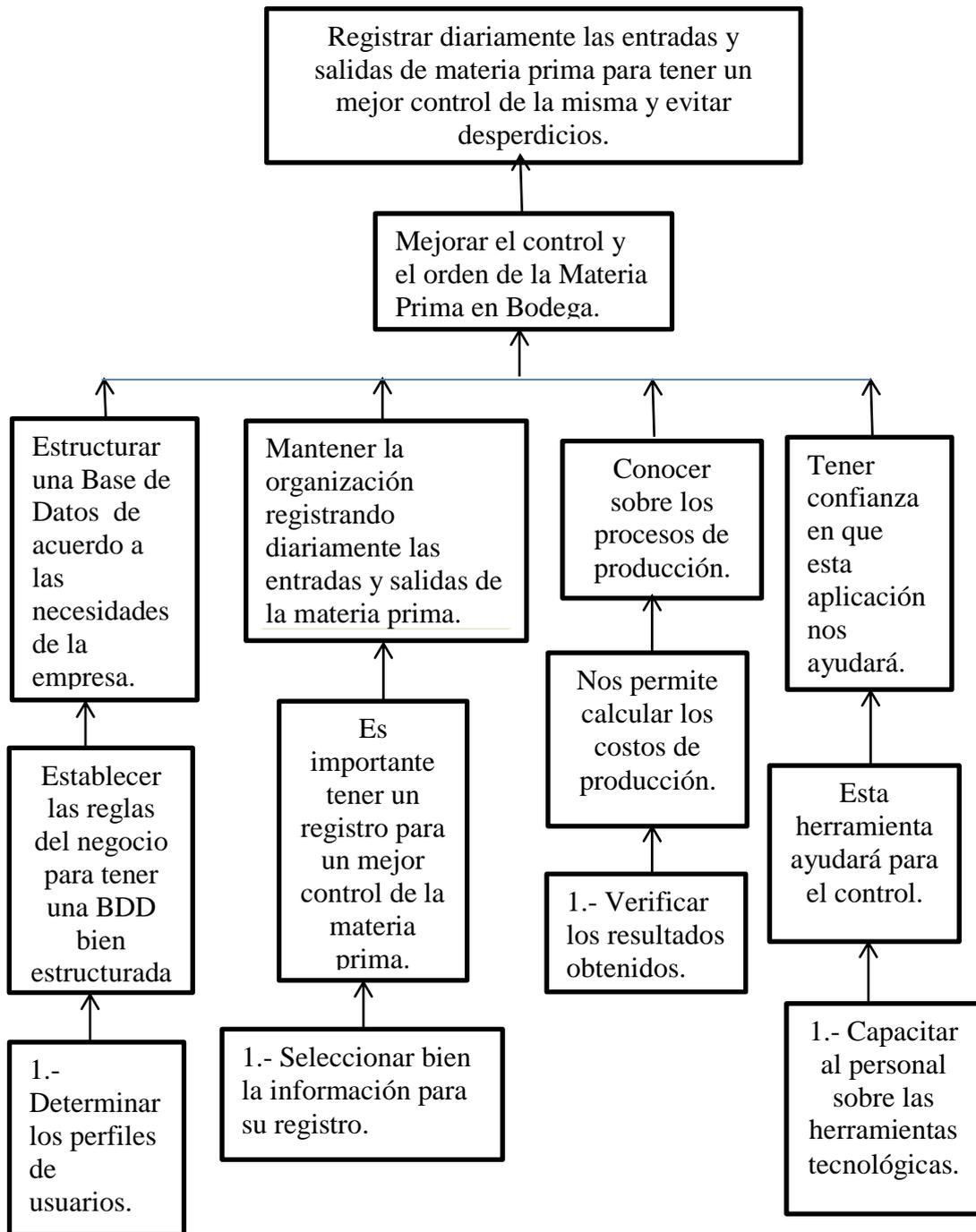


Figura 25 *Diagrama de Estrategias* – Describe las estrategias que se utilizan para el desarrollo de la aplicación.

#### 4.08 Matriz de Marco Lógico.

Tabla 28 Matriz de Marco Lógico.

Resumen narrativo de objetos	Indicadores	Medios de verificación	Supuestos
<b>Fin:</b> Mejorar el control de la materia prima agilitando procesos.	Proceso de obtención de la materia prima más rápida y eficiente.	Fácil obtención de datos de productos de materia prima, usuarios del sistema y proveedores.	Inducción a empleados de la empresa a que utilicen herramientas tecnológicas.
<b>Propósito:</b> Satisfacer las necesidades del cliente	Reducir desperdicios y aumentar la producción.	Optimizar tiempos en los procesos.	Mayores ingresos para la empresa al reducir los costos de producción.
<b>Componentes:</b> 1. Software fácil de utilizar y agradable a la vista.	1.- Realizar pruebas a la aplicación con usuarios múltiples comprobando la complejidad de su uso.	Realizar modelos que muestre las mejoras que se está logrando	No contemplar las restricciones de la aplicación.
<b>Actividad:</b> 1. Analizar cómo va a funcionar la aplicación. 2. Diseñar Base de datos. 3. Programar la aplicación.	1. Tener claras las reglas del negocio para realizar el aplicativo.	Documentar todo aquello que se esté realizando según los avances del proyecto.	Falta de tiempo para desarrollar la aplicación.

Notas: La Matriz de Marco lógico describe las actividades, componentes, propósito y el fin del proyecto.

## 4.09 Vistas Arquitectónicas.

### 4.01.01 Vista Lógica.

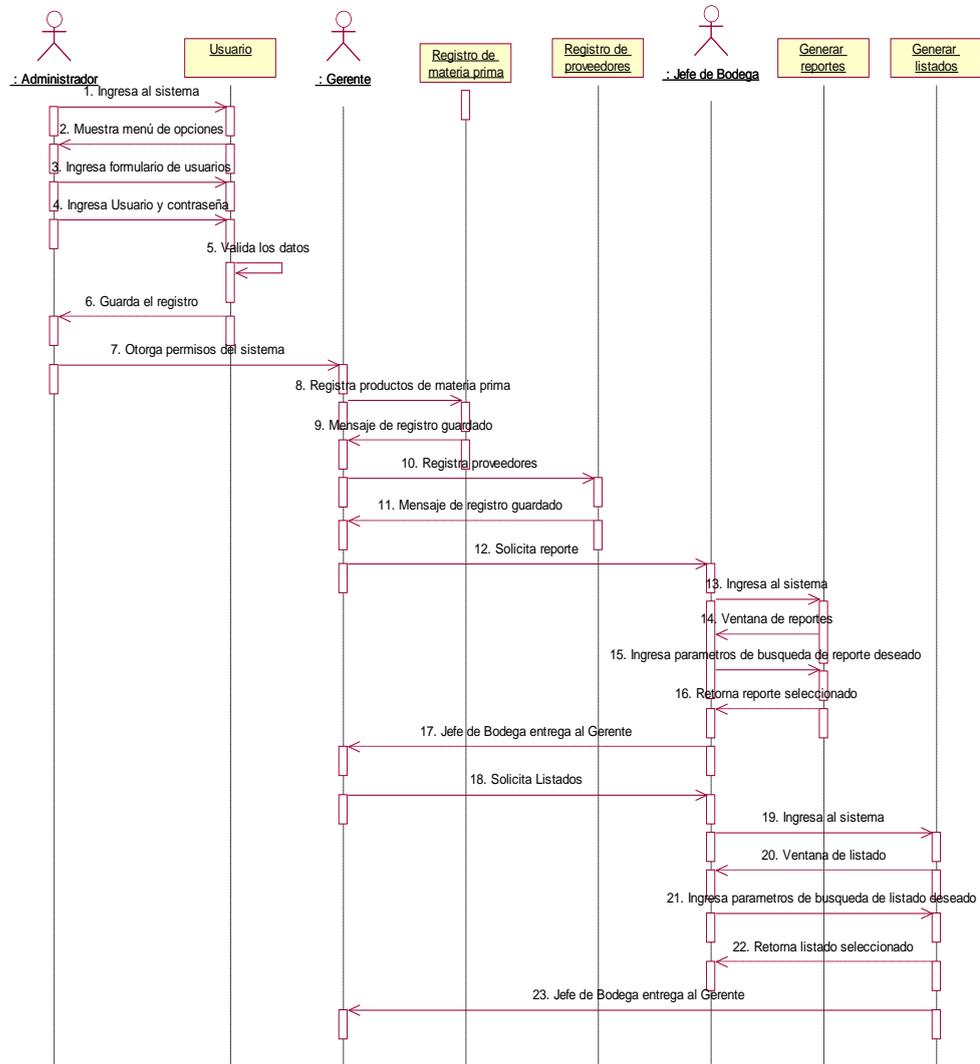


Figura 26 *Vistas Arquitectónicas* – Describe de forma lógica los procesos de la aplicación.

#### 4.01.02 Vista Física.



Figura 27 *Vistas Arquitectónicas* – Describe de forma física el proceso que realizan los usuarios para el manejo de la aplicación.

#### 4.01.03 Vista de Desarrollo.

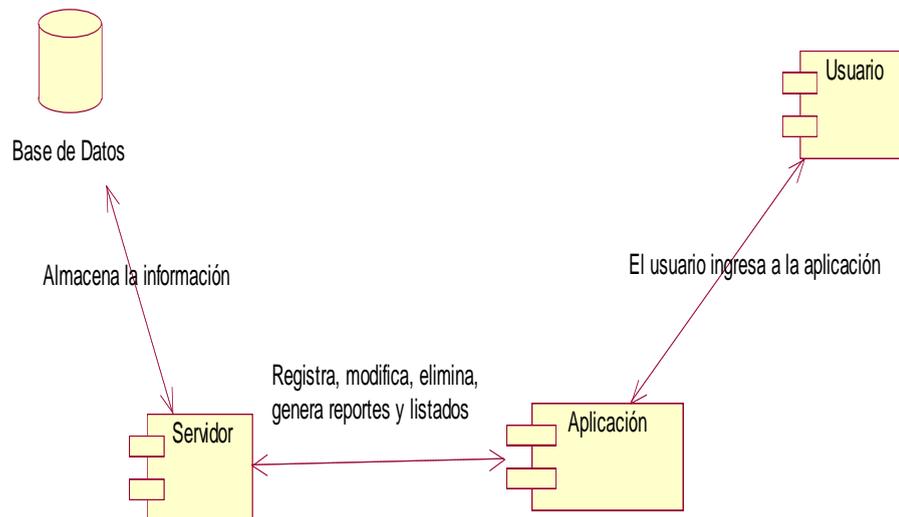


Figura 28 *Vistas Arquitectónicas* – Describe la vista de desarrollo de los procesos de la aplicación.

#### 4.01.04 Vista de Procesos.

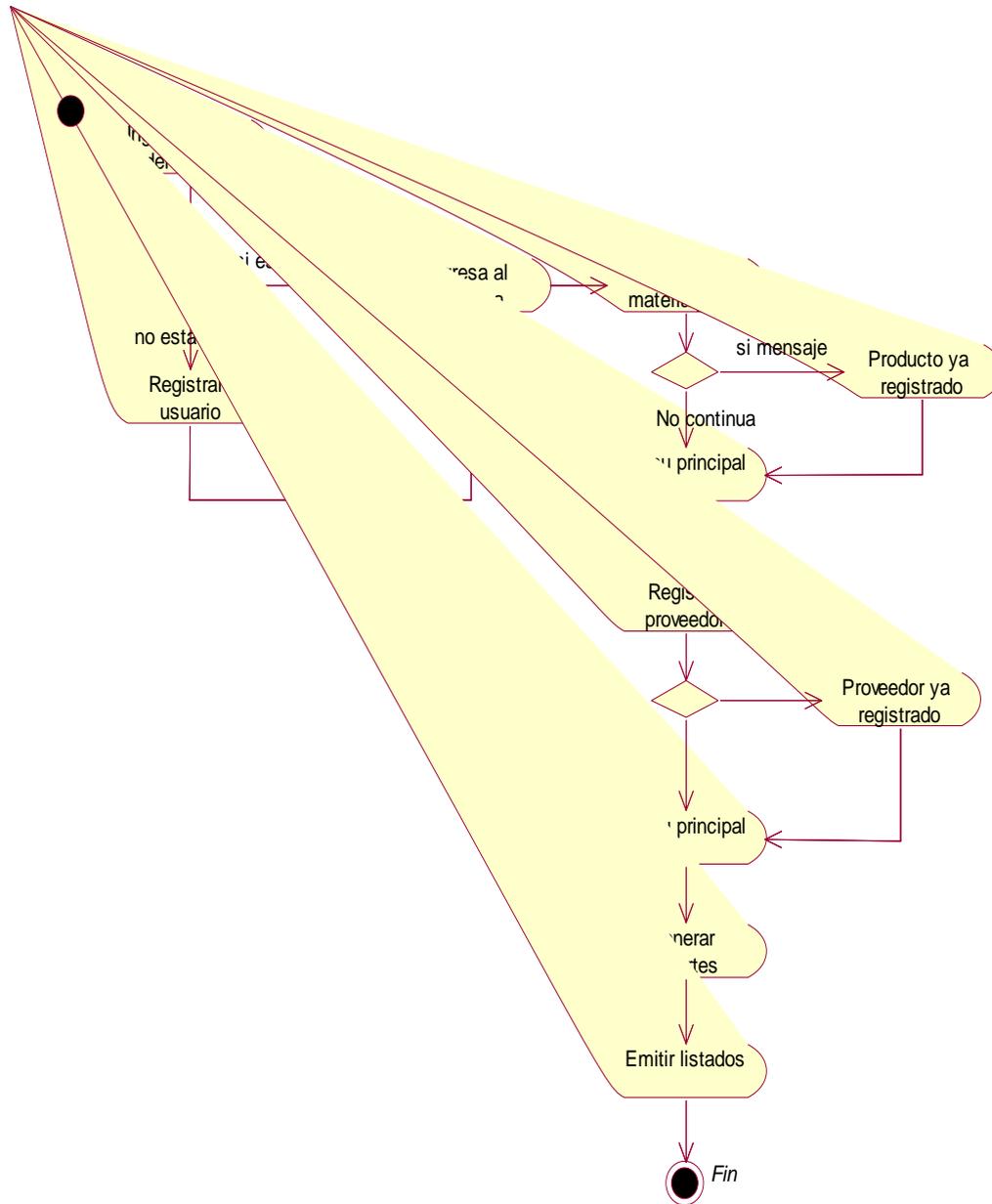


Figura 29 *Vistas Arquitectónicas* – Describe la vista de procesos de la aplicación mediante un algoritmo.

## **CAPITULO V: PROPUESTA**

### **5.01 Especificación de estándares de programación.**

“El objetivo de los estándares de programación es reglamentar la forma en que se implementará el código fuente del proyecto, pasando, por las variables, controles, ficheros, archivos y todo aquello que esté implicado en el código”. (oocities.org)

Tipos de variables

Declaración de clases como estoy utilizando las clases.

### **5.02 Diseño de Interfaces de Usuario.**

“La Interfaz de Usuario, de un programa es un conjunto de elementos hardware y software de una computadora que presentan información al usuario y le permiten interactuar con la información y con el computador. También se puede considerar parte de la Interfaz de Usuario la documentación (manuales, ayuda, referencia, tutoriales) que acompaña al hardware y al software.

Si la Interfaz de Usuario está bien diseñada, el usuario encontrará la respuesta que espera a su acción. Si no es así puede ser frustrante su operación, ya que el usuario habitualmente tiende a culparse a sí mismo por no saber usar el objeto”.

( monografías.com)

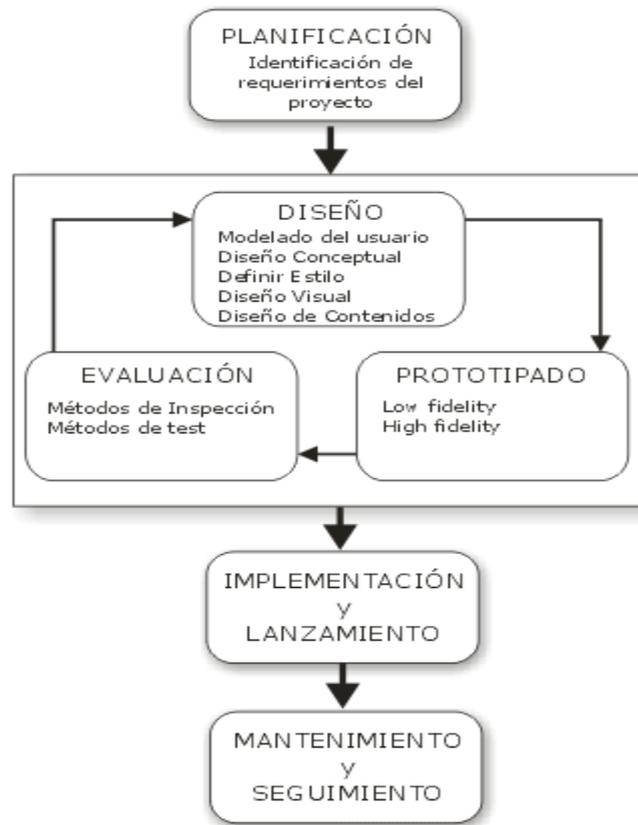


Figura 30 *Fases para el Diseño de Interfaz de Usuario* – Esta figura explica las fases para realizar la interfaz de usuario.

### Interfaz de logeo.

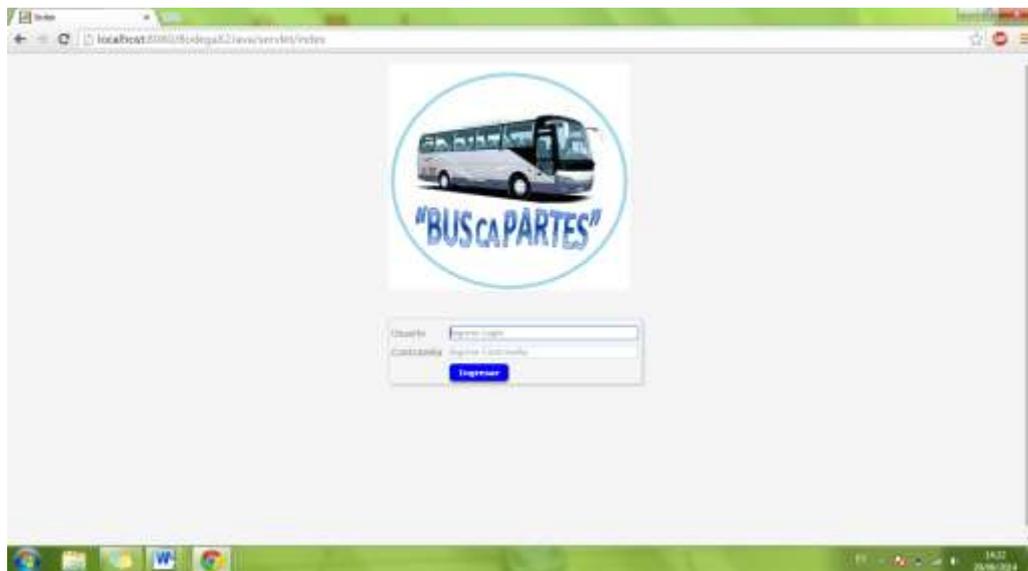


Figura 31 *Interfaz de Logeo de Usuario* – Interfaz principal para ingreso de datos de usuario para entrar a la aplicación.

## Interfaz aviso de stock bajo en bodega

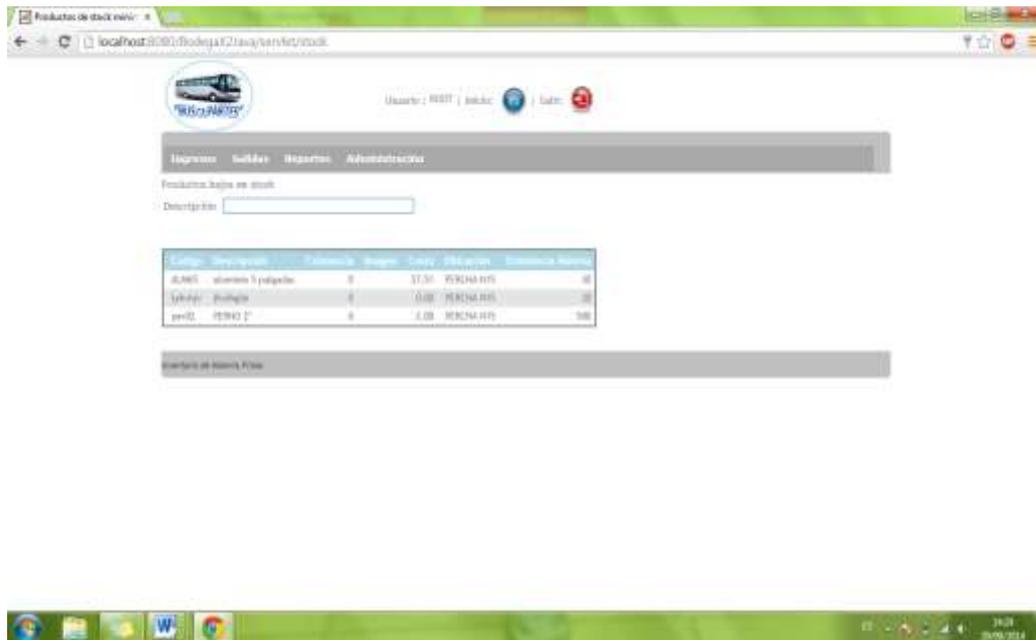


Figura 32 *Interfaz de aviso de stock bajo en bodega* – En esta interfaz informa sobre los productos bajos en stock.

## Interfaz ventana principal de la aplicación.



Figura 33 *Interfaz de Inicio* – Mediante esta interfaz podemos ingresar a los diferentes menús que tiene la aplicación.

### 5.03 Especificación de pruebas de unidad.

“La construcción de un sistema software tiene como objetivo satisfacer una necesidad planteada por el usuario. Para asegurar que se han alcanzado los niveles de calidad acordados es necesario evaluar el producto software a medida que se va construyendo. Por lo tanto se hace necesario llevar a cabo, en paralelo al proceso de desarrollo, un proceso de evaluación o comprobación de los distintos productos o modelos que se van generando”. (informatica-juridica.com)

**Tabla 29 Especificación de pruebas de unidad**

Identificador de la prueba:	EPU001
Método a probar:	Ingreso de usuarios al sistema mediante el login.
Objetivo de la prueba:	Verificar que la aplicación funcione de acuerdo a los requerimientos del usuario
Datos de entrada.	
<ol style="list-style-type: none"> <li>1) Ingresar nombre de usuario.</li> <li>2) Ingresar password de usuario.</li> </ol>	
Resultado esperado	
<ol style="list-style-type: none"> <li>1) Que el sistema valide los datos antes de ingresar a la aplicación.</li> </ol>	
Comentarios	

*Notas: Esta tabla describe los procesos que se realiza y los resultados esperados para el ingreso de usuarios al sistema.*

**Tabla 30 Especificación de pruebas de unidad**

Identificador de la prueba:	EPU002
Método a probar:	Registro de materia prima
Objetivo de la prueba:	Verificar que los datos de la materia prima sean correctos.
Datos de entrada.	
<ol style="list-style-type: none"> <li>1) Ingresar nombre de la materia prima.</li> <li>2) Ingresar la imagen de la misma si lo desea.</li> <li>3) Ingresar número de existencia mínima recomendada.</li> <li>4) Ingresar tipo de material.</li> <li>5) Ingresar ubicación de materia prima.</li> </ol>	
Resultado esperado	
Obtener información verdadera y actualizada de la materia prima.	
Comentarios	

*Notas: Esta tabla describe los procesos que se realiza y los resultados esperados para el registro de materia prima.*

**Tabla 31 Especificación de pruebas de unidad**

Identificador de la prueba:	EPU003
Método a probar:	Registro de proveedores.
Objetivo de la prueba:	Verificar el registro de proveedores y todos sus datos.
Datos de entrada.	
<ol style="list-style-type: none"> <li>1) Ingresar tipo de persona (proveedor).</li> <li>2) Ingresar ruc de proveedor.</li> <li>3) Ingresar nombre de proveedor.</li> <li>4) Ingresar teléfono de proveedor.</li> <li>5) Ingresar celular de proveedor.</li> <li>6) Ingresar e-mail de proveedor.</li> </ol>	
Resultado esperado	
Obtener información verdadera y actualizada de los proveedores.	
Comentarios	

*Notas: Esta tabla describe los procesos que se realiza y los resultados esperados para el registro de proveedores.*

**Tabla 32 Especificación de pruebas de unidad**

Identificador de la prueba:	EPU004
Método a probar:	Emitir reportes de entradas y salidas de materia prima.
Objetivo de la prueba:	Verificar mediante reportes la información.
Datos de entrada.	
<ol style="list-style-type: none"> <li>1) Ingresar parámetros de búsqueda del reporte deseado.</li> </ol>	
Resultado esperado	
Obtener reporte requerido con información actualizada.	
Comentarios	

*Notas: Esta tabla describe los procesos que se realiza y los resultados esperados para la obtención de reportes.*

**Tabla 33 Especificación de pruebas de unidad**

Identificador de la prueba:	EPU005
Método a probar:	Obtener listados de materia prima y proveedores.
Objetivo de la prueba:	Verificar datos de los mismos.
Datos de entrada.	
<ol style="list-style-type: none"> <li>1) Ingresar parámetros de búsqueda del listado deseado</li> </ol>	
Resultado esperado	
Obtener listados requeridos con información actualizada.	
Comentarios	

*Notas: Esta tabla describe los procesos y los resultados esperados para la obtención de listados.*

#### 5.04 Especificación de pruebas de aceptación.

“Las pruebas de aceptación aseguran el comportamiento del sistema, son escritas por [[nombre del cliente]] o por el usuario y especifican los aspectos a probar cuando una historia de usuario ha sido correctamente implementada.

En esta herramienta se definen las actividades que deben ser realizadas por

[[nombre del cliente]]”. (inf.utfsm.cl/~visconti/xp/Pruebas\_Aceptacion\_2.doc)

**Tabla 34 Especificación de pruebas de aceptación**

Identificador de la prueba:	EPA001
Caso de uso.	CU001
Tipo de usuario	Administrador (Gerente).
Objetivo de la prueba	Permitir ingresar a la aplicación al usuario
Secuencia de evento	
<ol style="list-style-type: none"> <li>1) Ingresamos a la aplicación.</li> <li>2) El usuario ingresa su nombre de perfil.</li> <li>3) Ingresa su password o contraseña.</li> <li>4) El sistema valida sus datos.</li> </ol>	
Resultados esperados	
El usuario ingresa a la aplicación.	
Comentarios	
<b>Estado:</b> Aceptado	

*Notas: Esta tabla describe el proceso que realiza el Administrador y el Gerente para permitir el ingreso a los usuarios.*

**Tabla 35 Especificación de pruebas de aceptación**

Identificador de la prueba:	EPA002
Caso de uso.	CU002
Tipo de usuario	Administrador, Usuario.
Objetivo de la prueba	Permitir al Gerente y Jefe de Bodega hacer el registro de la materia prima.
<b>Secuencia de evento</b>	
<ol style="list-style-type: none"> <li>1) El Gerente o el Jefe de Bodega deben ingresar al sistema mediante logeo.</li> <li>2) Se selecciona el formulario de productos.</li> <li>3) Se selecciona el botón nuevo.</li> <li>4) Se ingresa los datos de la materia prima.</li> <li>5) Se confirma el ingreso.</li> </ol>	
<b>Resultados esperados</b>	
Mostrará los datos ingresados del nuevo producto.	
<b>Comentarios</b>	
<b>Estado:</b> Aceptado.	

*Notas: Esta tabla describe el proceso que realiza el Gerente y Jefe de Bodega para registrar la materia prima.*

**Tabla 36 Especificación de pruebas de aceptación**

Identificador de la prueba:	EPA003
Caso de uso.	CU003
Tipo de usuario	Administrador (Gerente)
Objetivo de la prueba	Permitir al Gerente realizar el registro de proveedores.
<b>Secuencia de evento</b>	
<ol style="list-style-type: none"> <li>1) El Gerente debe ingresar al sistema mediante logeo.</li> <li>2) Se selecciona el formulario de proveedores.</li> <li>3) Se selecciona el botón nuevo.</li> <li>4) Se ingresa los datos del proveedor.</li> <li>5) Se confirma el ingreso.</li> </ol>	
<b>Resultados esperados</b>	
Mostrará los datos ingresados del nuevo proveedor.	
<b>Comentarios</b>	
<b>Estado:</b> Aceptado	

*Notas: Esta tabla describe el proceso que realiza el Gerente para registrar proveedores.*

**Tabla 37 Especificación de pruebas de aceptación**

Identificador de la prueba:	EPA004
Caso de uso.	CU004
Tipo de usuario	Usuario.
Objetivo de la prueba	Permitir al Jefe de Bodega generar reportes de entradas y salidas de materia prima.
<b>Secuencia de evento</b>	
<ol style="list-style-type: none"> <li>1) El Jefe de Bodega ingresa al sistema mediante logeo.</li> <li>2) Se selecciona el formulario de reportes.</li> <li>3) Se elige tipo de reporte.</li> <li>4) Ingresa parámetros de búsqueda del reporte deseado.</li> <li>5) Guarda el reporte o lo imprime.</li> </ol>	
<b>Resultados esperados</b>	
Se visualiza el reporte emitido.	
<b>Comentarios</b>	
El tipo de reporte puede ser de salida o entrada de materia prima.	
<b>Estado:</b> Aceptado	

*Notas: Esta tabla describe el proceso que realiza el Jefe de Bodega para generar reportes.*

**Tabla 38 Especificación de pruebas de aceptación**

Identificador de la prueba:	EPA005
Caso de uso.	CU005
Tipo de usuario	Usuario.
Objetivo de la prueba	Permitir al Jefe de Bodega emitir listados.
<b>Secuencia de evento</b>	
<ol style="list-style-type: none"> <li>1) El Jefe de Bodega ingresa al sistema mediante logeo.</li> <li>2) Se selecciona el formulario de listados.</li> <li>3) Se elige el listado requerido.</li> <li>4) Ingresa parámetros de búsqueda del listado deseado.</li> <li>5) Guarda el listado o lo imprime.</li> </ol>	
<b>Resultados esperados</b>	
Se visualiza el listado emitido.	
<b>Comentarios</b>	
El listado puede ser de usuarios, materia prima y proveedores.	
<b>Estado:</b> Aceptado	

*Notas: Esta tabla describe el proceso que realiza el Jefe de Bodega para emitir listados.*

### **5.05 Especificación de pruebas de carga.**

“Este es el tipo más sencillo de pruebas de rendimiento. Una prueba de carga se realiza generalmente para observar el comportamiento de una aplicación bajo una cantidad de peticiones esperada. Esta carga puede ser el número esperado de usuarios concurrentes utilizando la aplicación y que realizan un número específico de transacciones durante el tiempo que dura la carga. Esta prueba puede mostrar los tiempos de respuesta de todas las transacciones importantes de la aplicación.

Si la base de datos el servidor de aplicaciones, etc. Se realizan las funciones especificadas, Pruebas relacionadas con el rendimiento del sistema:

- Rendimiento (tiempos de respuesta adecuados)
- Volumen (funcionamiento con grandes volúmenes de datos)
  - Sobrecarga (funcionamiento en la Disponibilidad de datos (cuando se produce una recuperación ante fallos)
- Facilidad de uso (usabilidad) de desarranque, actualización de Operación e instalación (operaciones software)
- Entorno (interacciones con otros sistemas) y
- comunicaciones Seguridad (control de acceso e intrusiones)”.

(es.slideshare.net)

**Tabla 39 Especificación de pruebas de carga**

<b>Identificador de la prueba</b>	EPC001
<b>Tipo de prueba</b>	Desempeño de la aplicación realizando la prueba de logeo de usuarios.
<b>Objetivo de la prueba</b>	Verificar la validación de Nick name y password del usuario para ingresar a la aplicación.
<b>Descripción</b>	
<ul style="list-style-type: none"> <li>✓ Para la simulación se utilizó "JMeter Test".</li> <li>✓ Se crea un grupo de hilos para lanzar las peticiones.</li> <li>✓ Se configura para que tenga el grupo de hilos deseados.</li> <li>✓ Se configura para lanzar las peticiones. <ul style="list-style-type: none"> <li>- En este caso es una petición web.</li> </ul> </li> <li>✓ Configuramos esta petición asignándole <ul style="list-style-type: none"> <li>-Nombre, servidor, puerto y path al que tiene que acceder</li> </ul> </li> <li>✓ Se corre la aplicación para verificar la validación.</li> <li>✓ Se verifica los resultados en el Listener de resumen.</li> <li>✓ Se utilizó los siguientes navegadores: <ul style="list-style-type: none"> <li>- Google Chrome</li> <li>- Firefox</li> <li>- Internet Explorer</li> </ul> </li> </ul> <p>Duración de simulación 10 min.</p>	
<b>Resultados esperados</b>	
<ol style="list-style-type: none"> <li>1) Con esta prueba se verifico que al ingresar los datos erróneos del usuario la aplicación no nos permite acceder a la misma y que la validación funciona correctamente.</li> <li>2) Reportes creados satisfactoriamente.</li> </ol>	
<b>Comentarios</b>	
<p><i>Notas: Se detalla los procesos de cómo se realizó la prueba y los resultados que obtuvimos.</i></p>	

**Tabla 40 Especificación de pruebas de carga**

<b>Identificador de la prueba</b>	EPC002
<b>Tipo de prueba</b>	Verificar el desempeño de la aplicación cuando se registran varios usuarios y así poder tener acceso al sistema.
<b>Objetivo de la prueba</b>	Verificar cómo funciona el sistema al momento de registrar y cargar varios usuarios a la base de datos.
<b>Descripción</b>	
<ul style="list-style-type: none"> <li>✓ Para la simulación se utilizó "JMeter Test".</li> <li>✓ Se crea un grupo de hilos para lanzar las peticiones.</li> <li>✓ Se configura para que tenga el grupo de hilos deseados.</li> <li>✓ Se configura para lanzar las peticiones. <ul style="list-style-type: none"> <li>- En este caso es una petición web.</li> </ul> </li> <li>✓ Configuramos esta petición asignándole <ul style="list-style-type: none"> <li>- Nombre</li> <li>- Número de cédula.</li> <li>- Dirección.</li> <li>- Teléfono</li> <li>- Email.</li> <li>- Perfil de usuario.</li> <li>- Se guarda los registros.</li> </ul> </li> <li>✓ Se establece un número de usuarios ficticios.</li> <li>✓ Se utilizó los siguientes navegadores: <ul style="list-style-type: none"> <li>- Google Chrome</li> <li>- Firefox</li> <li>- Internet Explore</li> </ul> </li> <li>✓ Después de la simulación se procede a cargar registros de usuarios a la base de datos para verificar los resultados.</li> <li>✓ Se ejecuta el servidor de base de datos "SQL Server 2008 R2".</li> <li>✓ Los registros son cargados satisfactoriamente.</li> </ul> <p>Duración de simulación 10 min.</p>	
<b>Resultados esperados</b>	
<ol style="list-style-type: none"> <li>1) Con la simulación realizada y la carga de registros de usuarios ingresados a la base de datos, se puede apreciar que el sistema web actúa con mucha normalidad.</li> <li>2) Reportes creados satisfactoriamente.</li> </ol>	
<b>Comentarios</b>	
Esta prueba se realizó con 10 usuarios.	

*Notas: Se detalla los resultados después de realizar la prueba de ingreso de varios usuarios a la vez.*

### **5.06 Configuración del Ambiente mínima/ideal.**

Sistema Operativo Windows 7 o superior. Tipo de procesador Pentium 4 o superior.

Tamaño en disco duro de 4GB (espacio libre)

Memoria RAM de 2GB o más.

Monitor SVGA con resolución mínima de 1024 x768

Interfaz de la aplicación amigable con el usuario

Bases de datos en buen estado.

## **CAPITULO VI: ASPECTOS ADMINISTRATIVOS**

### **6.01 Recursos.**

#### **RECURSOS HUMANOS**

Autor del proyecto (Karina Morocho)

Tutor (Ing. Carlos Romero)

Lector (Ing. Jhonny Coronel)

Personal administrativo de la empresa (BUS CA PARTES)

#### **RECURSOS MATERIALES**

Computador portátil

Impresora

Hojas de papel bond

Carpetas

Lápices

Esferos

Medios de transporte.

Alimentación.

Internet

Energía

Eléctrica

Calendario de actividades.

Software para desarrollar.

## 6.02 Presupuesto.

**Tabla 41 Presupuesto.**

CANTIDAD	DESCRIPCIÓN	PRECIO UNITARIO	TOTAL
1	Computador portátil	950,00	950,00
5 meses	Servicios de electricidad	15,00	75,00
5 meses	Servicio de internet	21,00	105,00
1000	Impresiones	0,05	50,00
2	Anillados	13,00	26,00
1	Empastado	45,00	45,00
50	Medios de transporte	1,05	52,50
50	Alimentación	2,00	100,00
1	Taller de Profesionalización	720,00	720,00
		Total	2123,50

*Notas: Esta tabla describe el presupuesto económico que fue utilizado para el desarrollo de la aplicación.*

### 6.03 Cronograma.

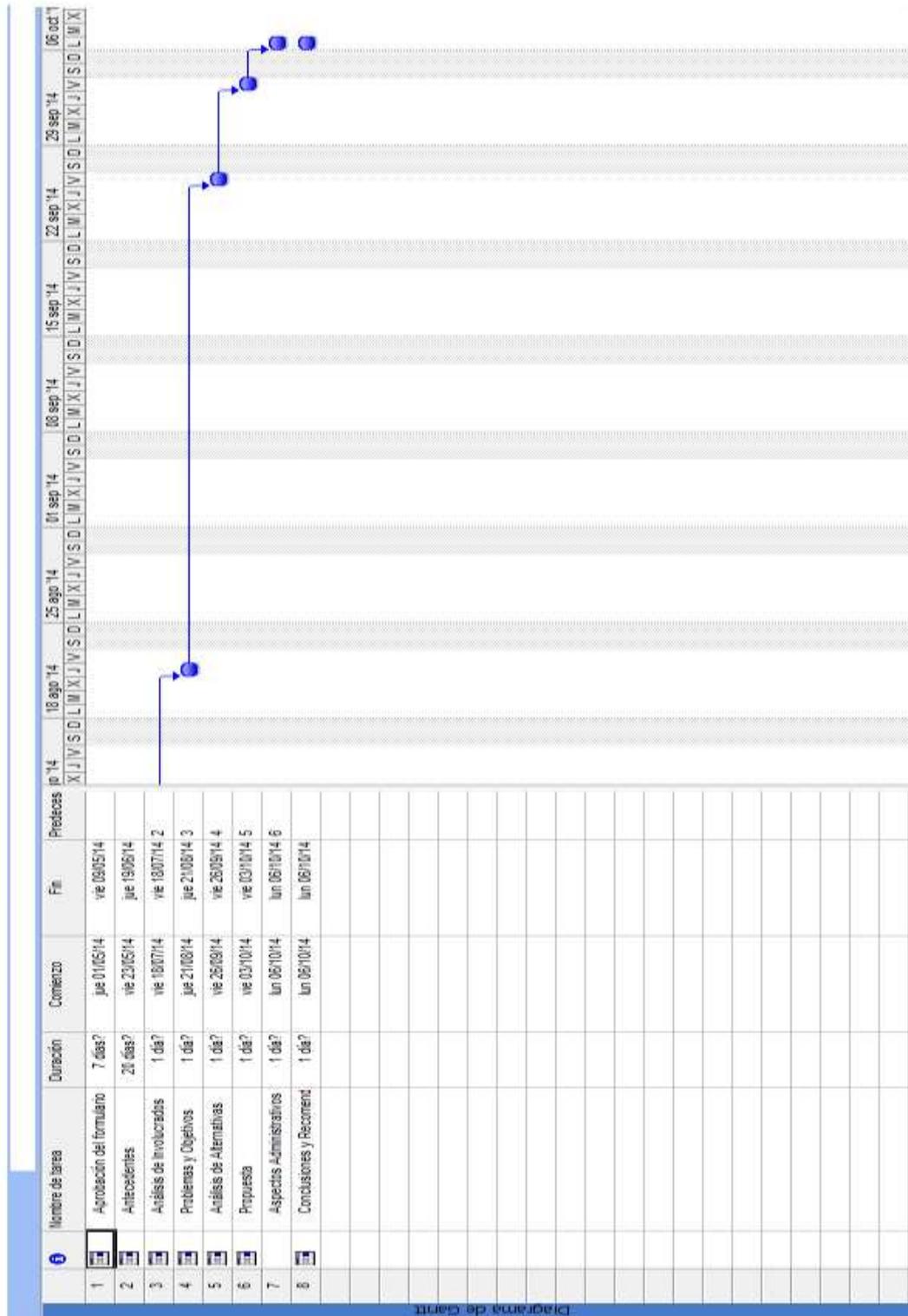


Figura 34 Cronograma de Actividades – Describe todas las actividades realizadas para el desarrollo del proyecto.

## CAPITULO VII: CONCLUSIONES Y RECOMENDACIONES

### 7.01 Conclusiones.

- ✓ Con esta aplicación se podrá mejorar los procesos para el control y manejo de la materia prima.
- ✓ Se pudo certificar la aplicabilidad de la propuesta y se constataron los beneficios en tiempo de desarrollo, mitigación de errores en etapas futuras y satisfacción de los requerimientos del usuario.
- ✓ Es importante adquirir herramientas tecnológicas ya que ayudan a optimizar y mejorar los procesos.

### 7.02 Recomendaciones.

- ✓ Es recomendable escoger adecuadamente las herramientas de diseño de software que se van a utilizar para realizar los diferentes diagramas, de tal manera que se agiliten y se realicen de forma correcta los modelos.
- ✓ Se recomienda que después de los casos de uso se pongan las especificaciones de casos de uso para que no existan contrariedades.
- ✓ Se debe realizar un mantenimiento constante a los equipos informáticos para que la aplicación siga funcionando sin contra tiempos.

## ANEXOS.

## A.01.01 Matriz de Requerimientos.

Matriz de requerimientos						
Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
<b>Requerimientos funcionales</b>						
RF001	La aplicación debe emitir un reporte de existencias de materia prima actualizado	Gerente	Alta	Sistema	En revisión	Gerente Encargado de bodega
RF002	La aplicación debe emitir un informe de la salida y entrada de materia prima	Gerente	Alta	Sistema	En revisión	Gerente Encargado de bodega
RF003	La aplicación debe contener un módulo para realizar el mantenimiento de la información.	Gerente	Alta	Sistema	En revisión	Gerente Encargado de bodega
RF004	El control del sistema deberá ser mediante una cuenta con usuario y password	Gerente Administrador	Alta	Sistema	En revisión	Gerente Encargado de bodega Administrador del sistema
<b>Requerimientos no funcionales</b>						
NRF001	Acceso desde internet.	Gerente	Media	Estudiante investigador	En revisión	Gerente

*Nota: Matriz de Requerimientos. Esta matriz detalla los requerimientos funcionales y no funcionales que obtuvimos de las entrevistas*

### A.01.02 Mapeo de Involucrados.



Figura 1. Mapeo de Involucrados.

### A.01.03 Matriz de Involucrados.

ACTOR INVOLUCRADO	INTERES SOBRE EL PROBLEMA CENTRAL	PROBLEMA PERCIBIDO	RECURSO MANDATO Y CAPACIDAD	INTERES SOBRE EL PROYECTO	CONFLICTO POTENCIAL
GERENTE	Facilidad para tener un inventario de bodega actualizado	Demora en la producción	Infraestructura de la empresa	Mejorar los procesos productivos	Disminución del interés en cuanto al proyecto.
ESTUDIANTE INVESTIGADOR	Ayudar automatizando el inventario de materia prima en bodega.	Conocimientos muy básicos de las reglas del negocio.	Herramientas de trabajo y conocimientos.	Apoyo a la empresa y obtención del título.	Tiempo y recursos.
JEFE DE BODEGA	Mejor organización en bodega	Dificultad en el desempeño laboral.	Herramientas de trabajo y horas laborables.	Menor carga laboral.	Exceso de personal.
CONTABLE	Organización y mejor desempeño en sus labores.	Dificultad en el desempeño laboral.	Económico.	Reducir costos de producción	Cuentas sin cuadrar.

*Nota: Matriz de involucrados. Con esta matriz podemos reconocer la intervención de los involucrados, los conflictos, problemas y recursos que los envuelven.*

### A.01.04 Matriz de análisis de alternativas.

Variables Estrategias	Necesidad de recursos			Factibilidad política	Duración del proyecto	Puntaje	Prioridad
	Humanos	Técnicos	Financieros				
Adquirir software de paquete.	3	4	3	4	4	18	1era.
Reducir la cantidad de desperdicios de materia prima aumentando su producción.	3	4	3	4	4	18	2da.
Incrementar la productividad.	3	4	4	4	4	19	3era.

*Nota: Matriz de Análisis de Alternativas. En esta matriz analizamos cada alternativa obtenida del árbol de objetivos para saber la prioridad de cada una.*

- 1= Bajo
- 2=Medio Bajo
- 3=Medio
- 4=Medio Alto
- 5=Alto

### A.01.05 Matriz de impacto de Objetivos.

ANÁLISIS DEL IMPACTO DE LOS OBJETIVOS					
Objetivos	Factibilidad a lograse	Impacto de género	Impacto Ambiental	Relevancia	Sostenibilidad
Adquirir un software de paquete para registrar y controlar los procesos de bodega.	La información se almacenara de mejor forma y con facilidad.	Este software podrá ser utilizado por cualquier persona que tenga autorización.	Se optimizara el tiempo, de modo que se utilizara por un corto lapso el computador.	Servirá para organizar y tener un mejor control de la Información.	Este software será la base para el buen manejo de la empresa.
Reducir la cantidad de desperdicios de materia prima aumentando su producción.	Determinar ganancias y pérdidas en un periodo de tiempo.	Todos los empleados podrán participar en esta actividad.	Minimizar la cantidad de desperdicios que se utiliza para realizar estos procesos.	La idea es ayudar a controlar la materia prima para aumentar la producción.	Este control ayudara a mejorar el rendimiento laboral.
Incrementar la productividad.	Aumentaran los ingresos y se sacara un producto de calidad.	Se cuenta con el apoyo de todos los empleados en general para realizar el cambio.	Mediante investigaciones se buscan herramientas menos contaminantes que den los mismos resultados.	Después de organizar la información se verificara si los cambios realmente están siendo de provecho.	Manteniendo ordenada la bodega reduce los tiempos de búsqueda y la producción se acelera.

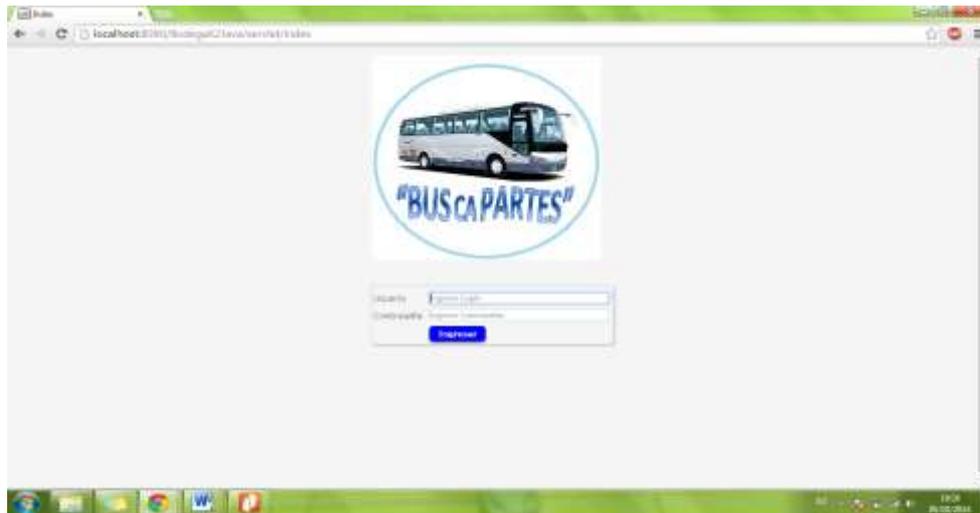
*Notas: Esta matriz describe el análisis de impacto de los objetivos de la aplicación.*

### **B.01.01 MANUAL DE USUARIO.**

Abrimos el navegador y ponemos la dirección de nuestra aplicación.

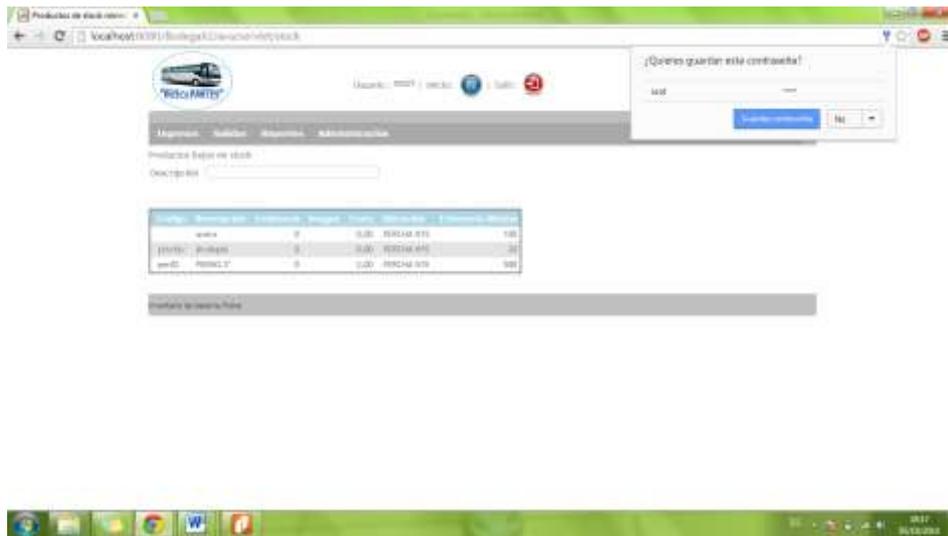
Nos muestra la pantalla principal de la aplicación.

Ingresamos el nombre de usuario y la contraseña para entrar a la aplicación.

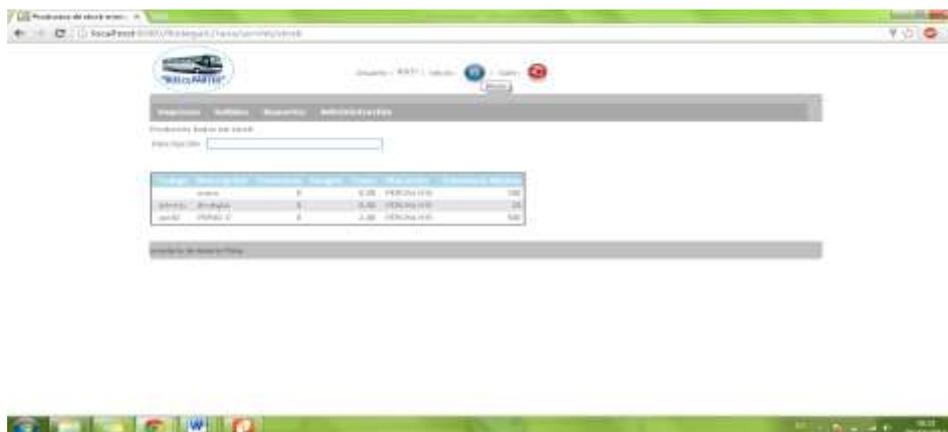


De inmediato nos mostrara otra ventana en la que aparecerá un mensaje si desea recordar la contraseña que es por defecto del navegador, debemos poner jamás para este sitio.

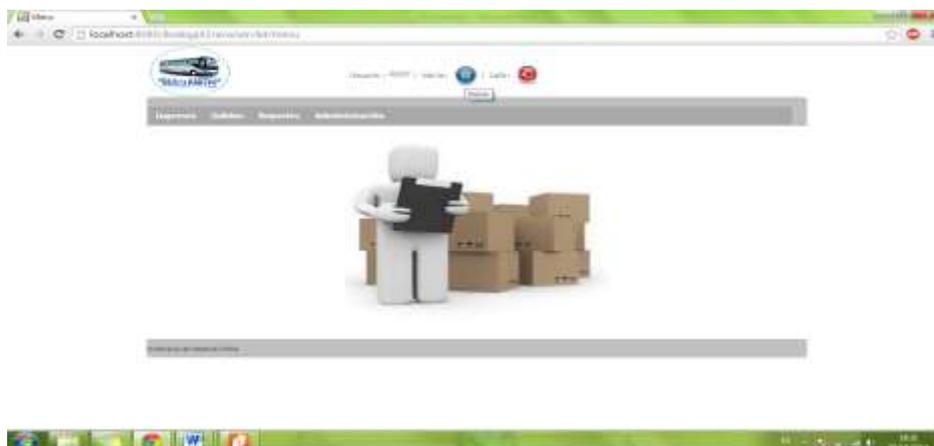
En la misma pantalla nos muestra una lista o un aviso para recordarnos los productos que están bajos en stock.



Después damos click en el botón de inicio para ir a la pantalla de inicio valga la redundancia.



Aparecerá la pantalla de inicio y ya podemos empezar a realizar los procesos del sistema.



Para empezar tenemos que ir al formulario de Tipo de Producto ya que es un dato que necesitamos para realizar el registro de la materia prima.

Dando click en el botón nuevo nos llevara al formulario de ingreso de datos de Tipo de Producto.



En esta pantalla ingresamos el tipo de producto o en otras palabras los tipos de materiales con que está hecha la materia prima.



Después vamos al formulario de Ubicación, este es otro dato que debemos ingresar antes de registrar la materia prima.

Dando click en el botón nuevo nos llevara al formulario de ingreso de datos de la ubicación.

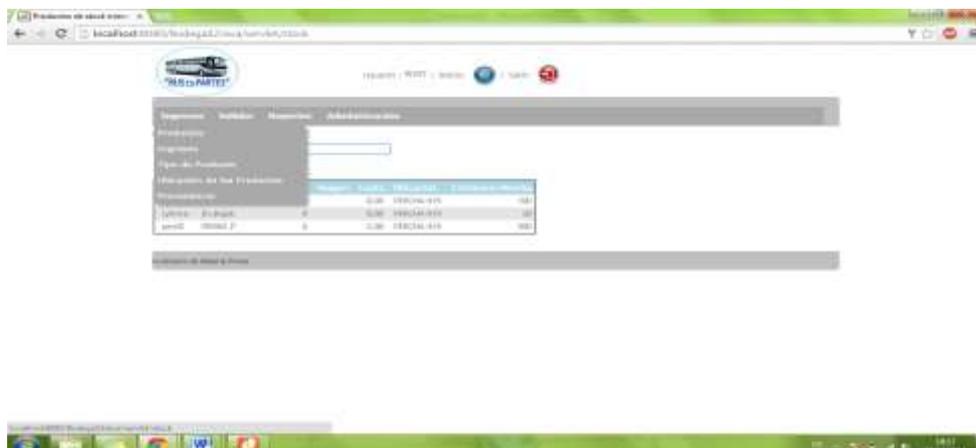


Aquí ingresamos el lugar donde está ubicada cada materia prima, es decir perchas, vitrinas, etc.



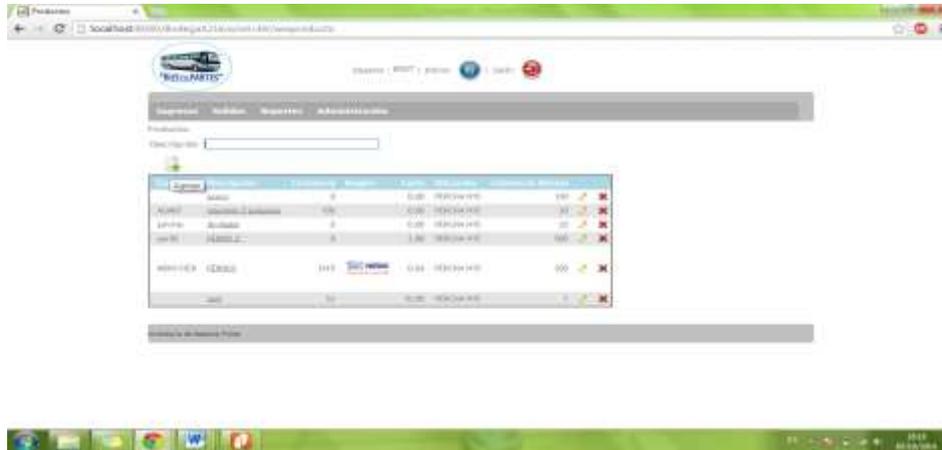
Luego de Ingresar estos datos ya podemos ir al formulario de registro de la materia prima.

Damos click en la pestaña de ingresos y luego en Productos.



Nos muestra el formulario de Productos.

Damos click en el agregar nuevo y nos lleva al formulario de ingreso de datos de la materia prima.



Esta ventana Muestra el formulario de ingreso de datos de la materia prima.

Se ingresa la descripción que se refiere al nombre del producto o materia prima.

La imagen de la materia prima si la tuviera.

La existencia mínima recomendada.

La referencia es opcional.

El tipo de producto que ya lo tenemos cargado.

Y la ubicación que también ya la tenemos.

Luego vamos a la ventana para registrar a proveedores.

Damos click en el botón agregar para registrar al proveedor.



Ingresamos el tipo de persona en este caso escogemos proveedor.

Escogemos ruc.

E ingresamos todos los datos solicitados.



En la pestaña de ingresos escogemos la opción Ingresos.

Click en agregar y se direcciona al formulario de ingreso de la compra.



En este formulario ingresamos la compra de la materia prima.

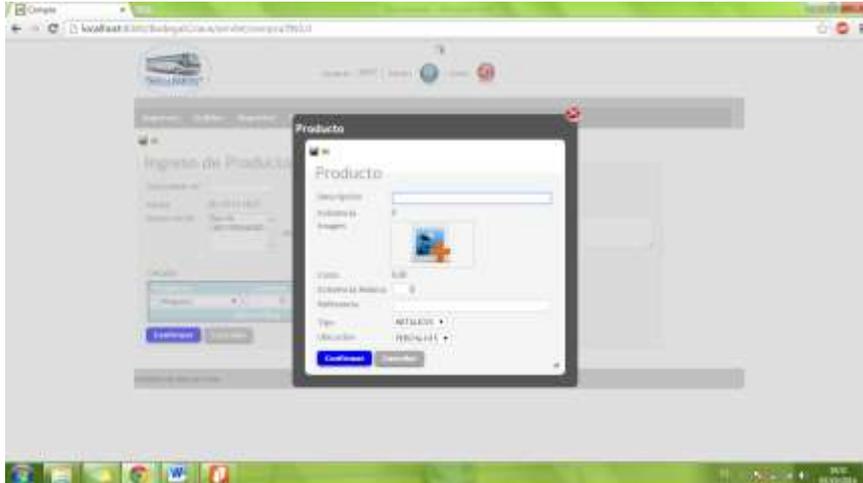


En caso de que el producto no este registrado damos click en el botón nuevo producto y nos direccionará a la ventana de registro de la materia prima.



Aquí nos aparece un pop – up o ventana emergente para registrar la materia prima sin abandonar el formulario anterior.

Una vez registrada la materia prima damos click en confirmar y continuamos registrando la compra.



En el menú escogemos salidas de productos y nos direccionará al formulario de egresos.

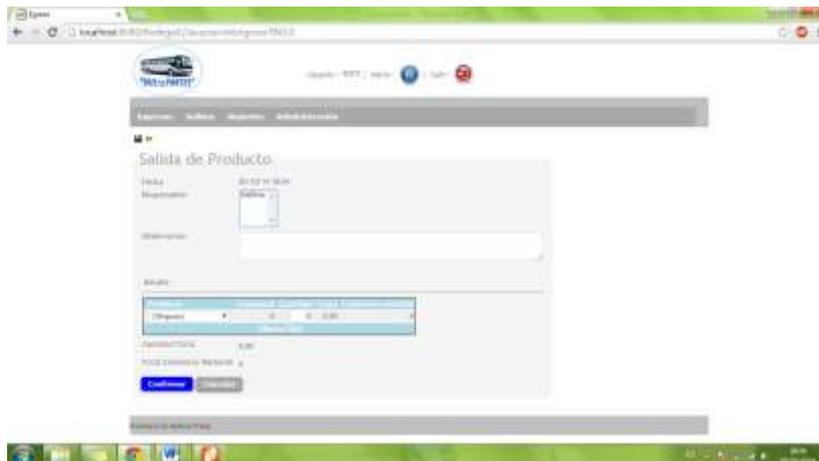


En este formulario damos click en el botón Agregar para realizar la salida de materia prima.



Nos muestra el formulario de registro de salida de productos.

Ingresamos todos los datos y confirmamos la salida de materia prima.



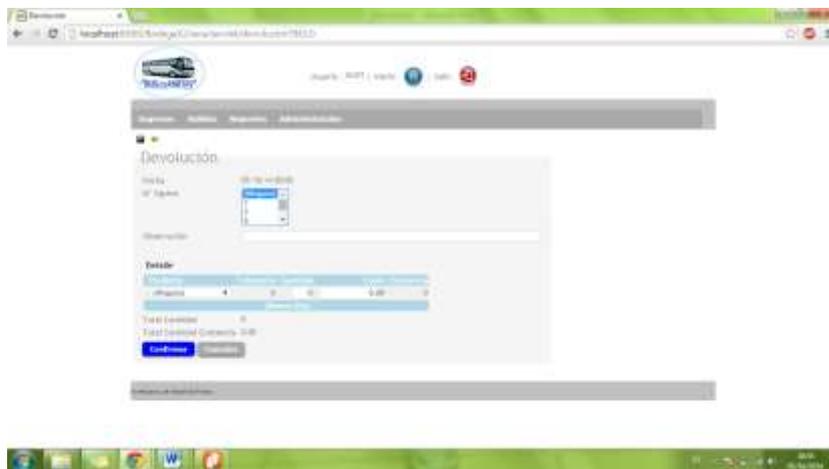
En caso de que no hayan utilizado el material de la salida antes realiza en este formulario hacemos la devolución del mismo.

En el menú escogemos la opción Devoluciones.

Después click en Agregar.



Ingresamos los datos solicitados y confirmamos la devolución.



Vamos al menú escogemos la opción Responsables, click en agregar.



Ingresamos los datos de la persona o el responsable de realizar la salida de materia prima.

En este caso escogemos cedula e ingresamos todos los datos solicitados.

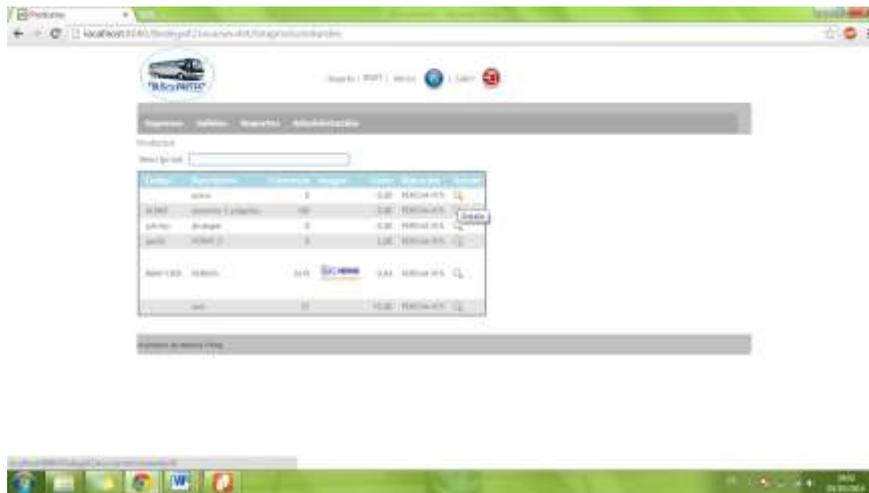


En el menú escogemos la opción reportes y luego Kardex.

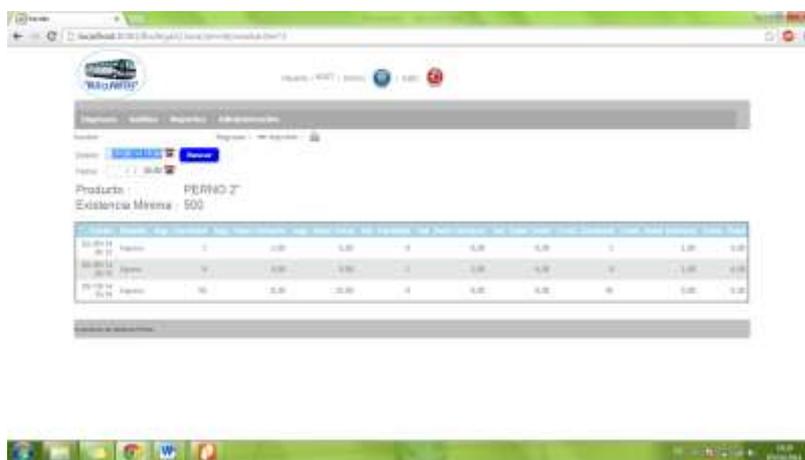
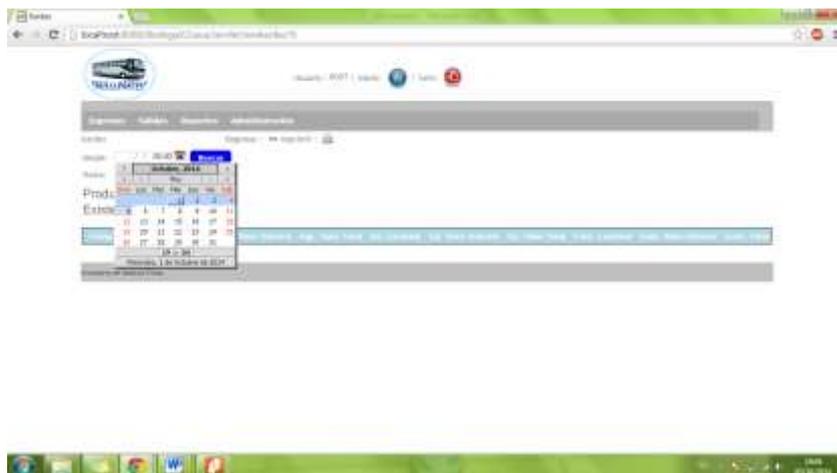


En esta ventana de kardex podemos observar a detalle cada una de las acciones realizadas.

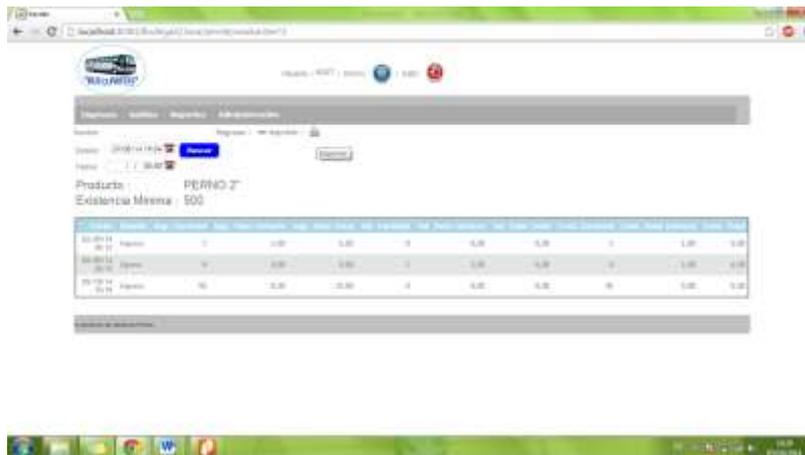
Damos click en el icono de búsqueda y nos mostrara el detalle del mismo.



Podemos realizar la búsqueda ingresando la fecha desde y hasta que fecha deseamos buscar.



Podemos imprimir el reporte si lo deseamos.



Producto: PERNO 2  
Existencia Mínima: 500

Fecha	Detalle	Ingreso	Egreso	Saldo	Costo	Valor	Existencia	Costo	Valor
2014/11/11	Ingreso	0	0.00	0.00	0	0.00	0.00	0	0.00
2014/11/11	Egreso	14	0.00	0.00	-14	0.00	0.00	0	0.00
2014/11/11	Ingreso	16	0.00	0.00	0	0.00	0.00	0	0.00

Esta ventana muestra el documento previo a la impresión.



Producto: PERNO 2  
Existencia Mínima: 500

Fecha	Detalle	Ingreso	Egreso	Existencia
2014/11/11	Ingreso	0	0.00	0.00
2014/11/11	Egreso	14	0.00	0.00
2014/11/11	Ingreso	16	0.00	0.00

En la pestaña de reportes podremos sacar un reporte de las existencias o el stock de productos.



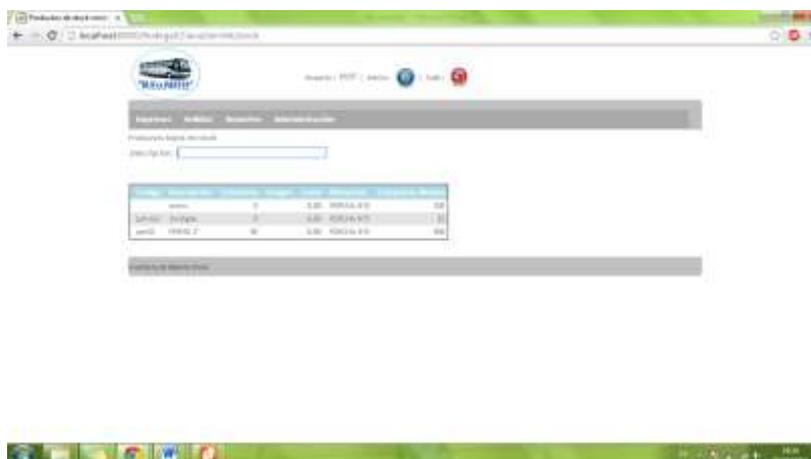
Fecha	Detalle	Ingreso	Egreso	Saldo	Costo	Valor	Existencia	Costo	Valor
2014/11/11	Ingreso	0	0.00	0.00	0	0.00	0.00	0	0.00
2014/11/11	Egreso	14	0.00	0.00	-14	0.00	0.00	0	0.00
2014/11/11	Ingreso	16	0.00	0.00	0	0.00	0.00	0	0.00

Esta ventana muestra el documento previo a la impresión.

De esta forma el usuario decide si desea guardar o imprimir el listado.



Dentro de la pestaña de reportes tenemos la opción de revisar una lista de los productos que están bajos en stock.



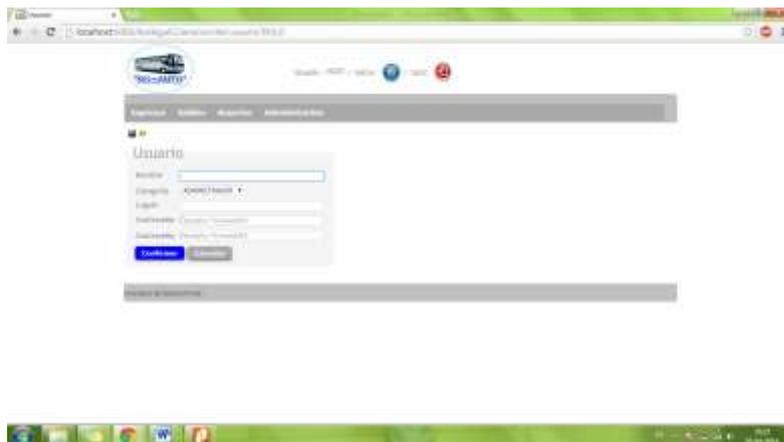
La parte de administración se habilita solo para la persona administradora de la aplicación.

Esta persona puede realizar la parte de creación de usuarios, verificar el listado o revisar la bitácora de auditoría.



Damos click en la pestaña de administración y luego en nuevo usuario.

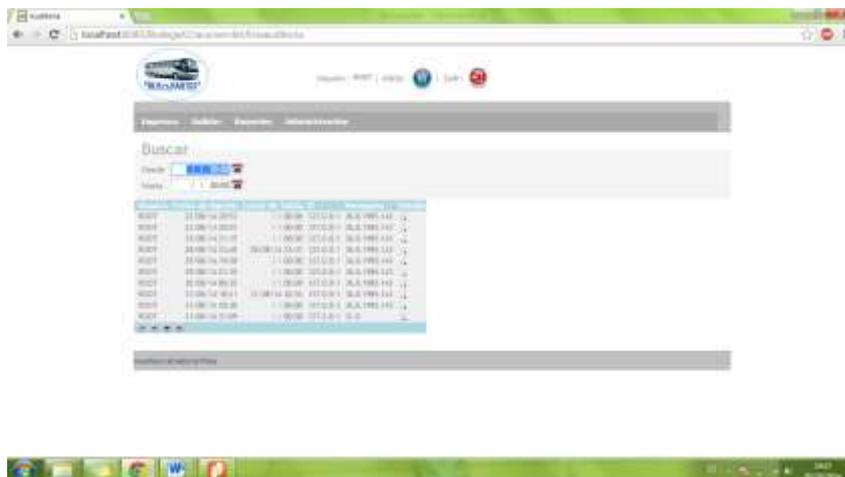
Nos muestra el formulario e ingresamos los datos solicitados.



Dando click en la pestaña lista de usuarios podremos obtener un listado de los usuarios registrados y autorizados para ingresar al sistema.



En la parte de Administración tenemos un formulario de Auditoria, el mismo que está diseñado para tener un control de quien ingresa o que acción realizó, a qué hora y la fecha, de esta manera llevaremos una bitácora de la empresa por asuntos de seguridad.



Esta ventana muestra el detalle de uno de los datos registrados en la Bitácora. Hacemos un click en el icono de la lupa de uno de los registros y podremos ver el detalle del mismo.



## B.01.02 MANUAL DE INSTALACIÓN.

### Guías de Instalación de SQL Server 2008 R2

Primero deben fijarse que en:

Panel de control

Todos los elementos de Panel de control

Configuración regional y de idiomas

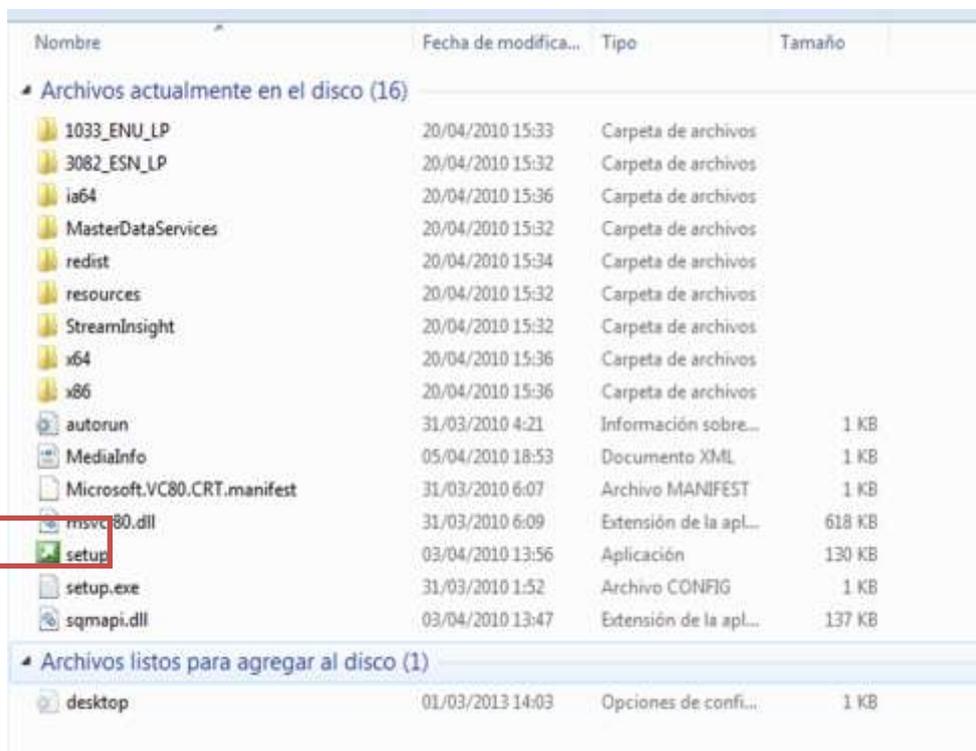


Para la instalación del SQL Server R2 debe estar configurado en ESPAÑOL (ESPAÑA)

1.- Se Abrirá automáticamente la instalación como en la imagen.



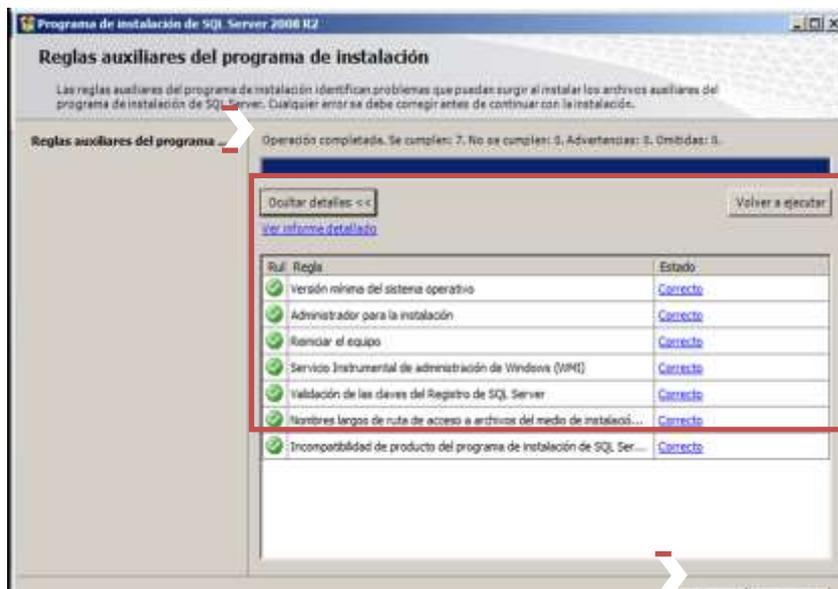
La otra forma sería hacer doble clic en el SETUP



2.- Se escogerá la opción **INSTALAR** y la Opción *Nueva instalación o agregar características a una instalación existente*



3.- En esta ventana **ACEPTAR** las *Reglas auxiliares del programa de Instalación.*

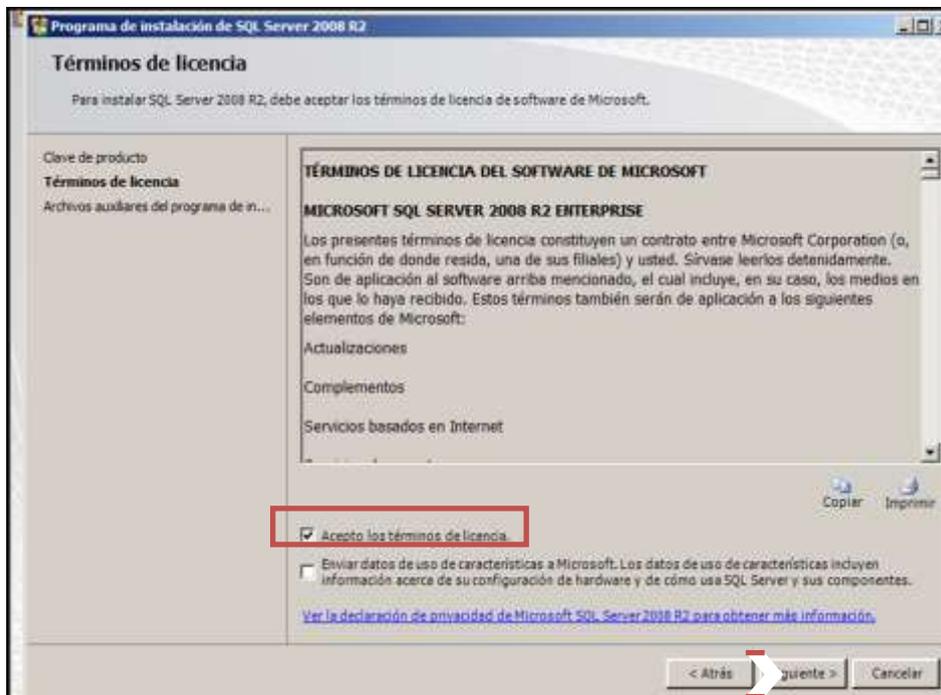


Donde también podrás ver los detalles de la instalación

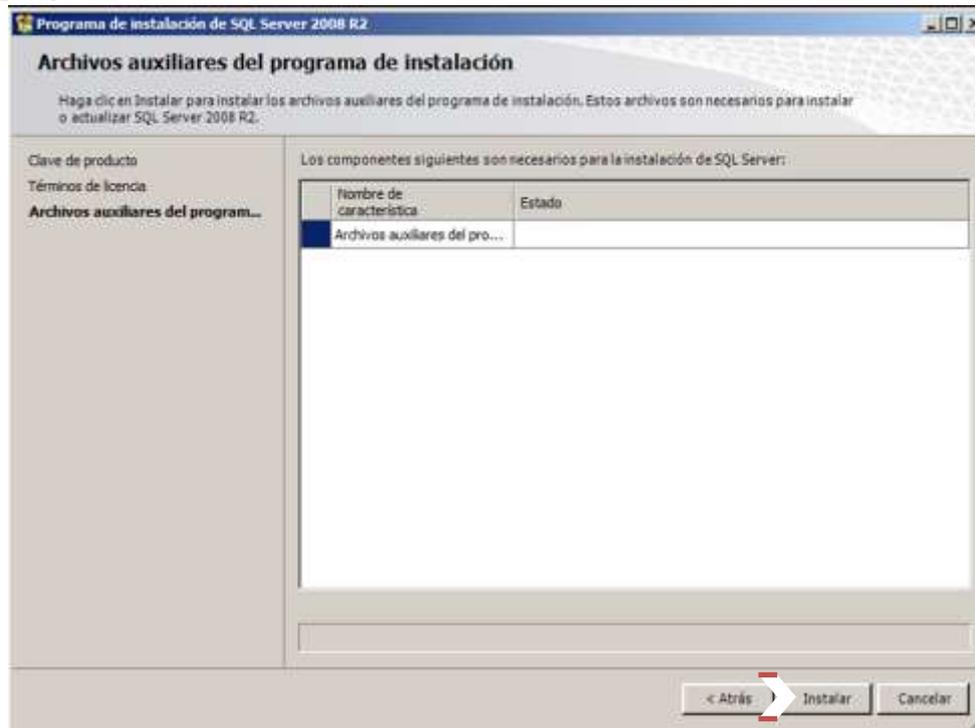
4.- En esta ventana damos **SIGUIENTE** la Clave predeterminada que te viene



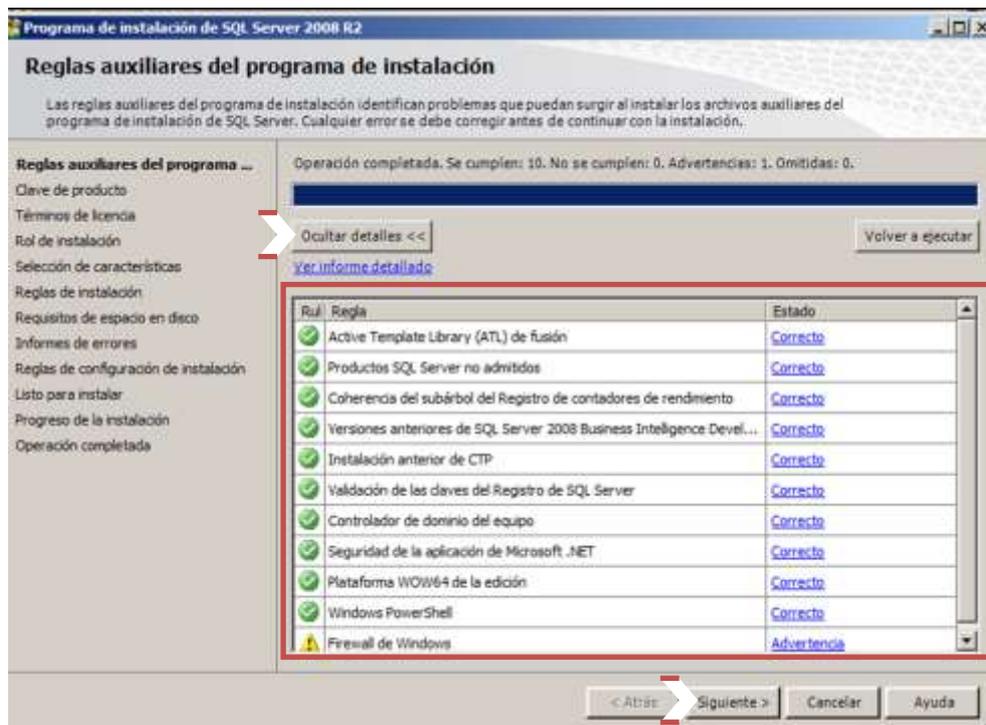
5.- **ACEPTAS** los términos de la licencia y damos **SIGUIENTE**



6.- En esta ventana solo tienes que **INSTALAR** los *Archivos auxiliares del programa de instalación*



7.- Al terminar la instalación se abrirá la ventana de *Reglas auxiliares del programa de instalación*

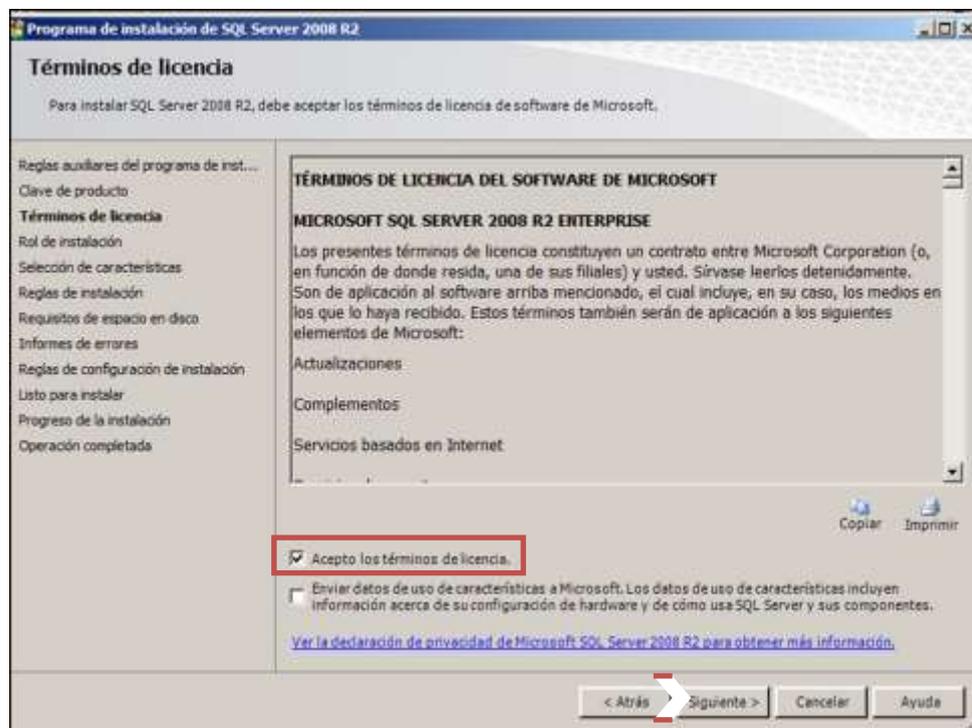


Donde también podrás ver los detalles de la instalación.

8.- Se abrirá *Clave de Producto* en donde damos **SIGUIENTE**

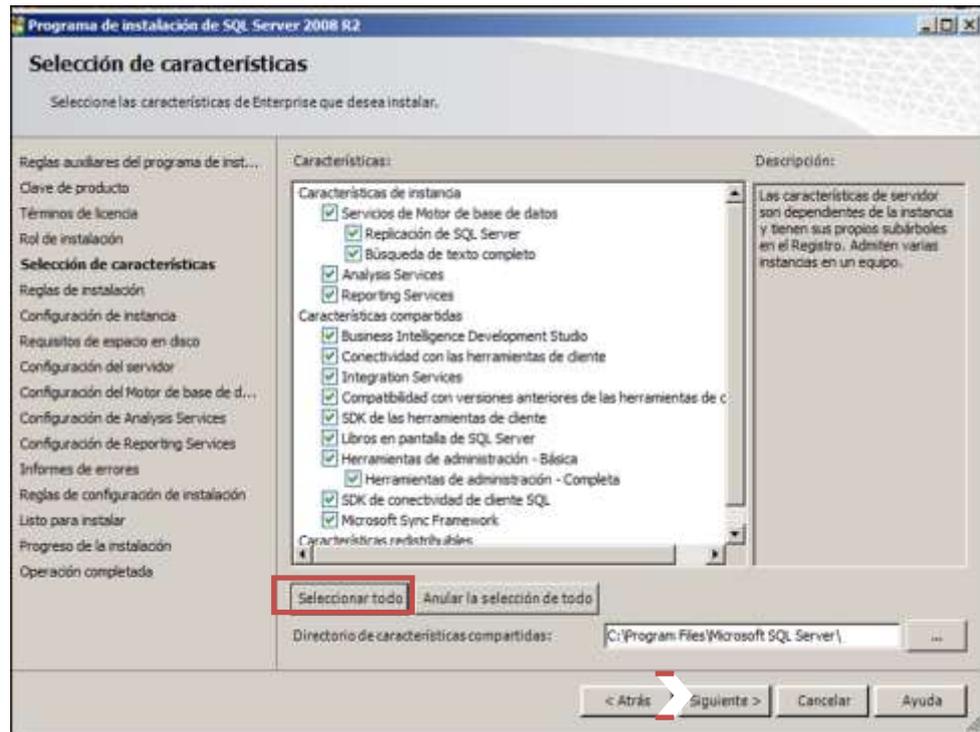


9.- *Términos de licencia* ahí Aceptamos y continuamos dando **SIGUIENTE**



10. En *Rol de Instalación* continuamos **SIGUIENTE**

11.- En esta ventana *Seleccionamos todo* y SIGUIENTE

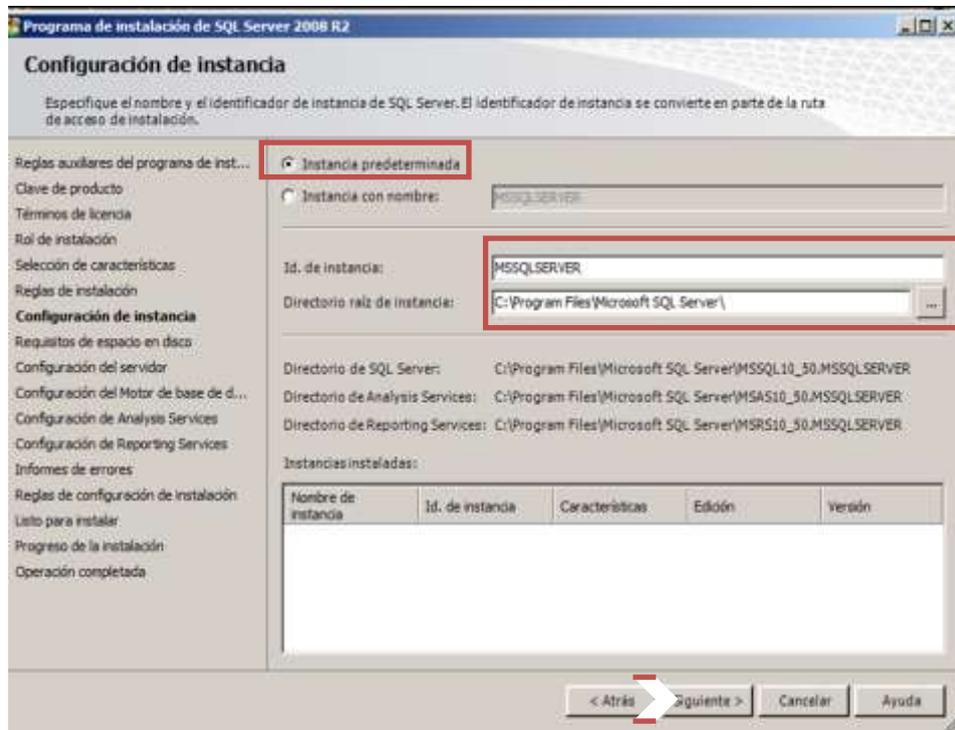


12.- En esta venta podemos ver las *Reglas de instalación* y continuamos SIGUIENTE

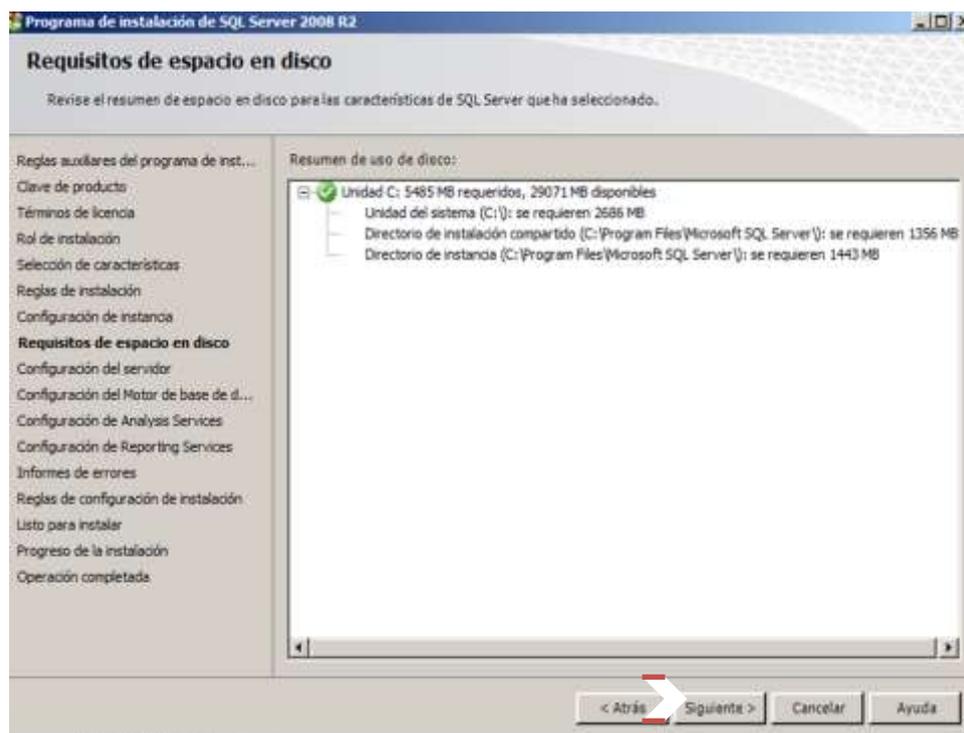


Nota: verifique que todo este  para que la instalación sea correcta

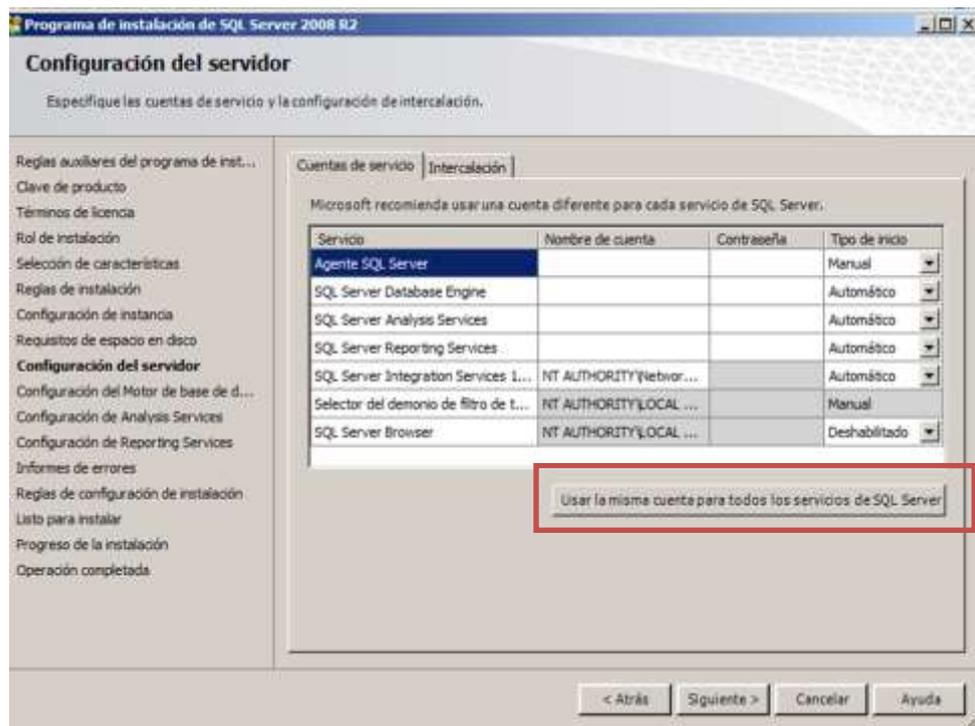
13.- En esta venta verifique que se encuentre como la imagen es lo más recomendable.



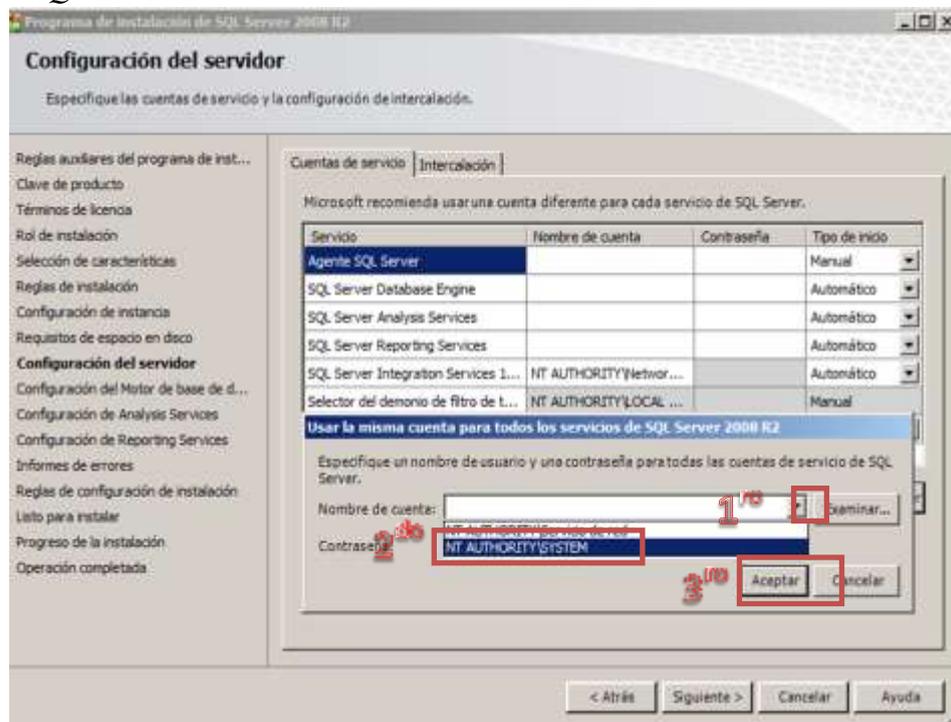
14.- Si los **Requisitos de espacio de disco** son aceptados continuamos **SIGUIENTE**



15.- En esta ventana vas a tener que *configuración del servidor*

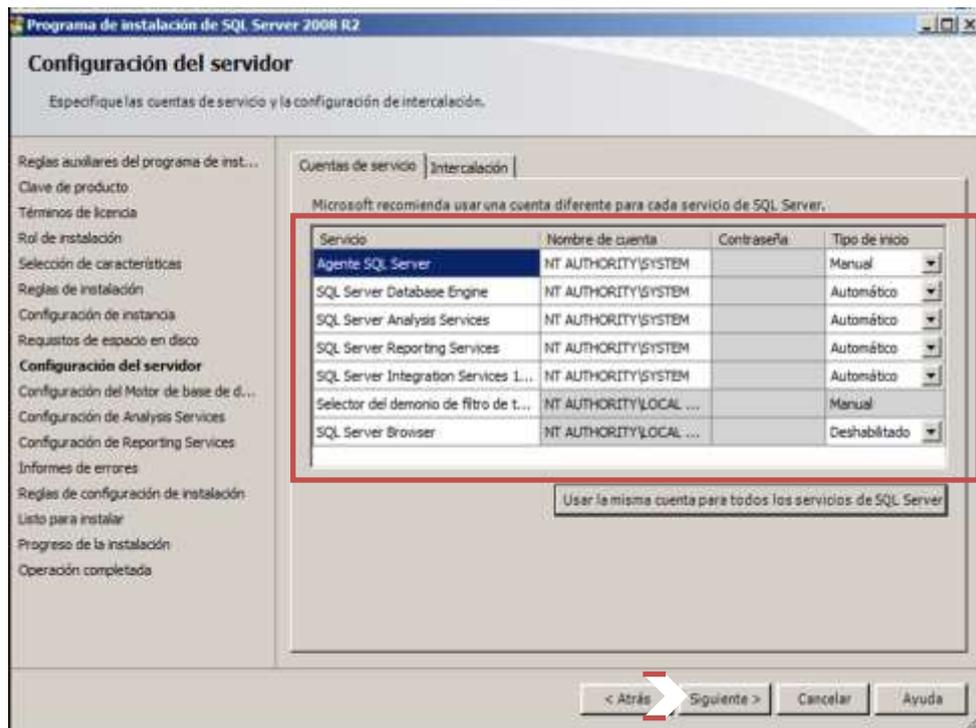


Lo más recomendable es hacer clic *Usar la misma cuenta para todos los servidores de SQL Servidor*

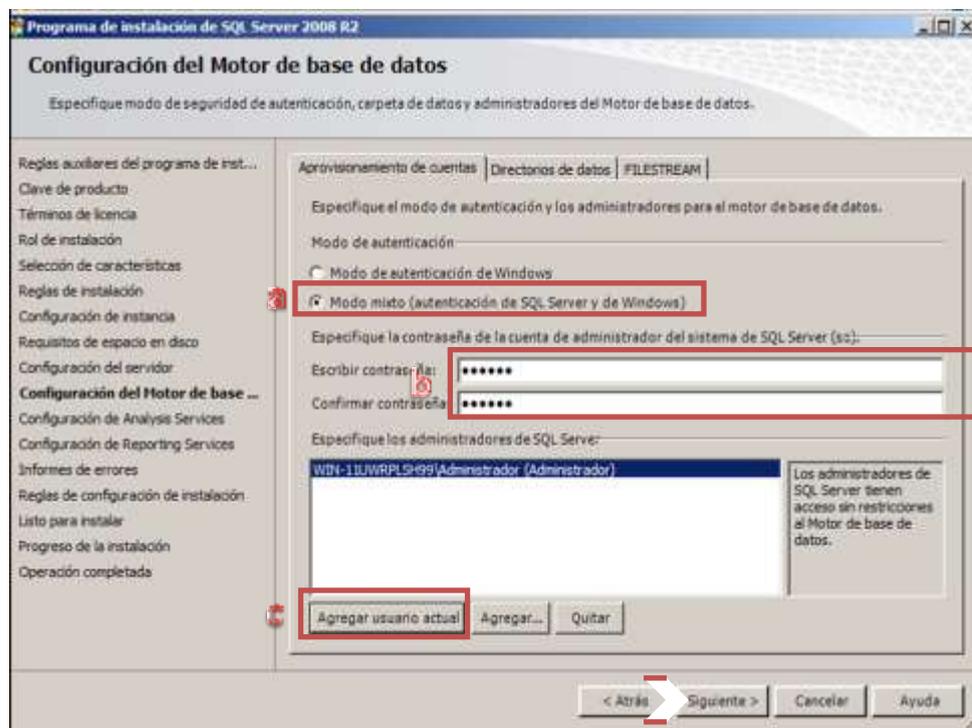


Donde seleccionaremos **AUTHORITY\SYSTEM** y Aceptaremos

Tendremos como resultado al finalizar esta ventana y damos **SIGUIENTE**

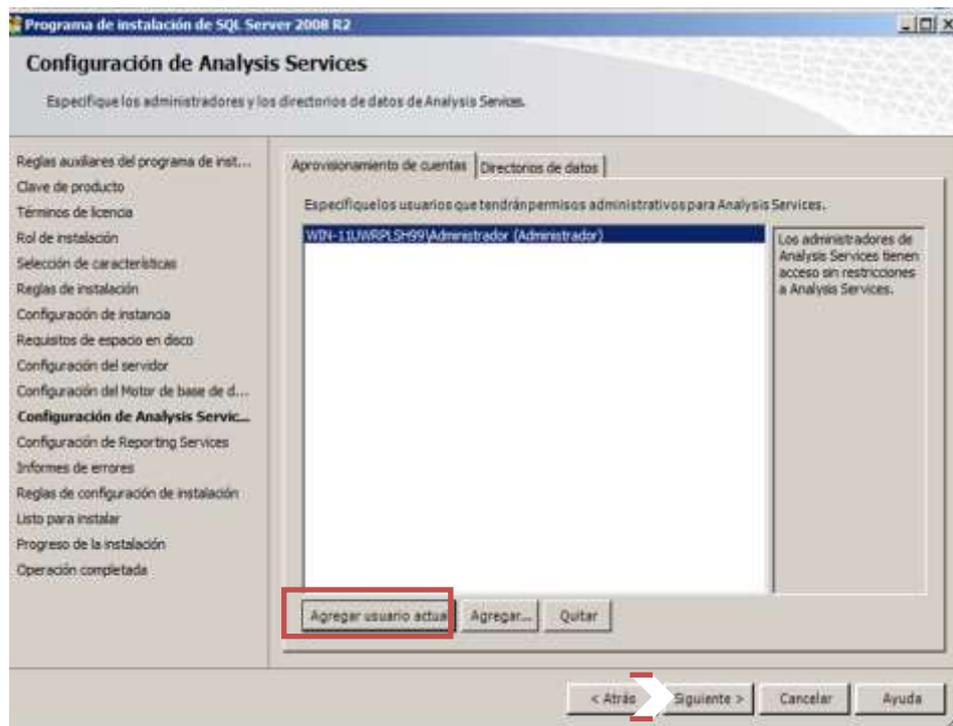


16.- En esta ventana seleccionamos modo de seguridad



- a) Seleccionamos modo autenticación de **Windows** o **Modo mixto**
- b) La contraseña es opcional dependiendo al Usuario c) Hacemos clic para **Agregar el usuario actual**.

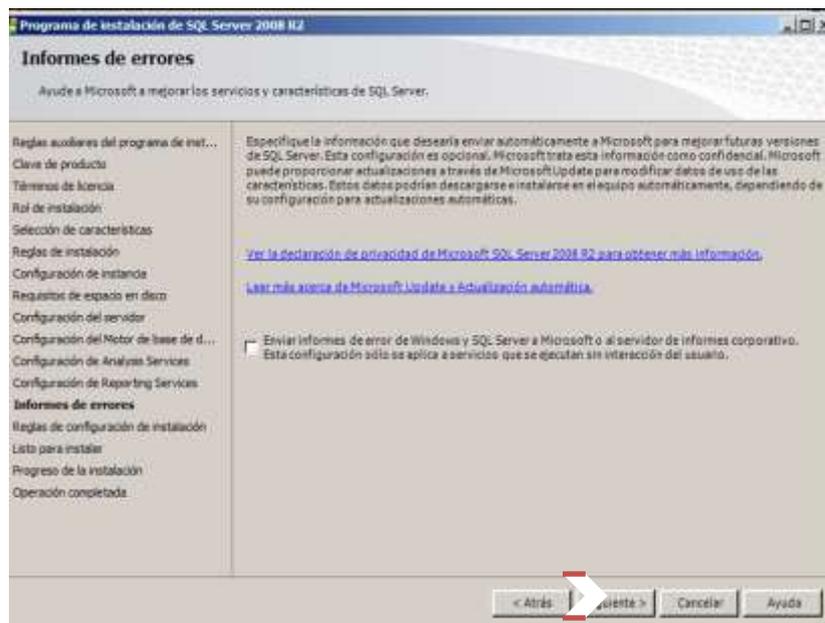
17.-En esta ventana también tendremos que hacemos clic para **Agregar el usuario actual**



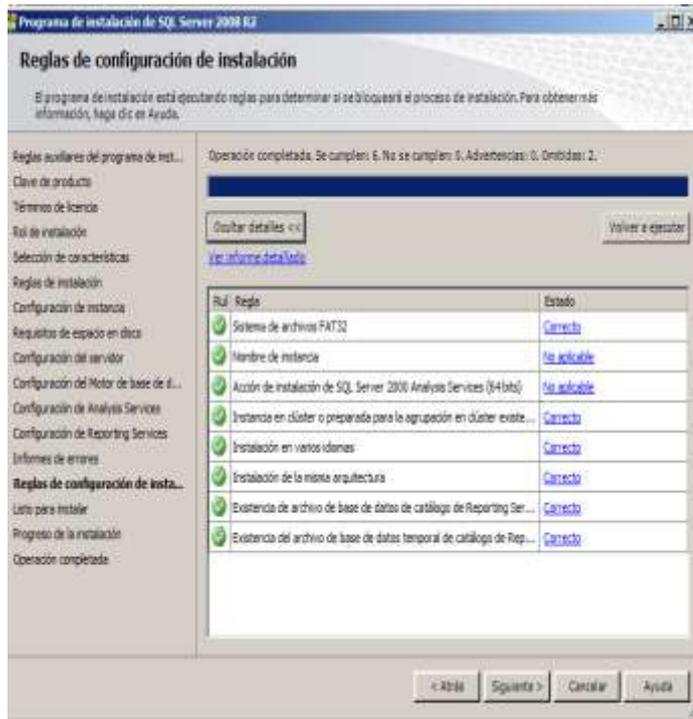
18.- En esta ventana dejamos en la opción marcada predeterminada Y **SIGUIENTE**



19.- En esta ventana dejamos sin marcar nada y continuamos **SIGUIENTE**

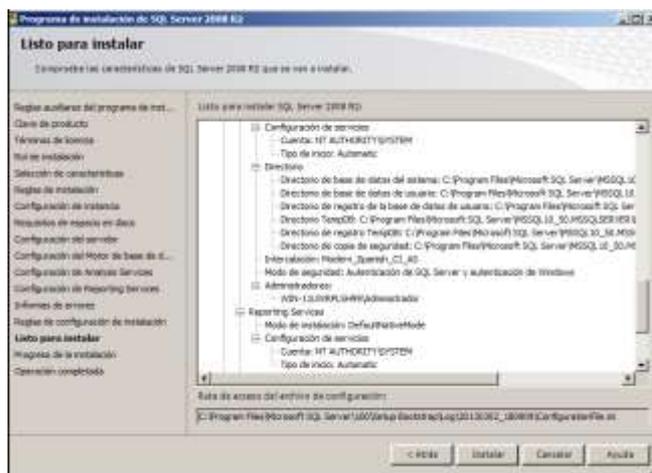


20.- en esta ventana comprobamos las **Reglas de configuración de Instalación**



21.- Aquí te mostrara la comprobación de las características del SQL Server R2 y comenzamos

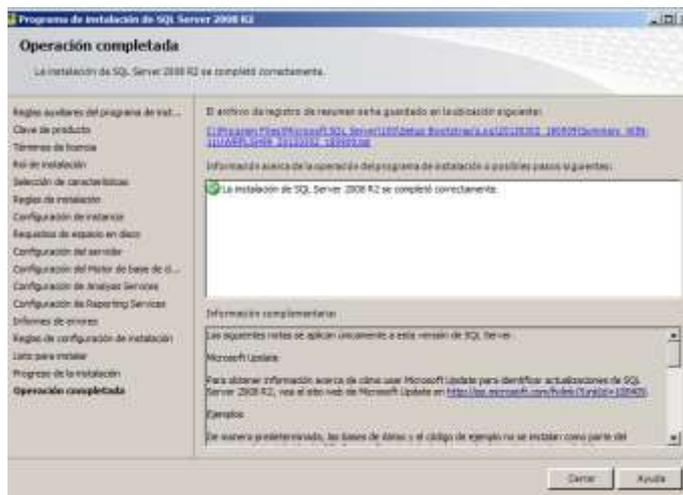
## INSTALAR



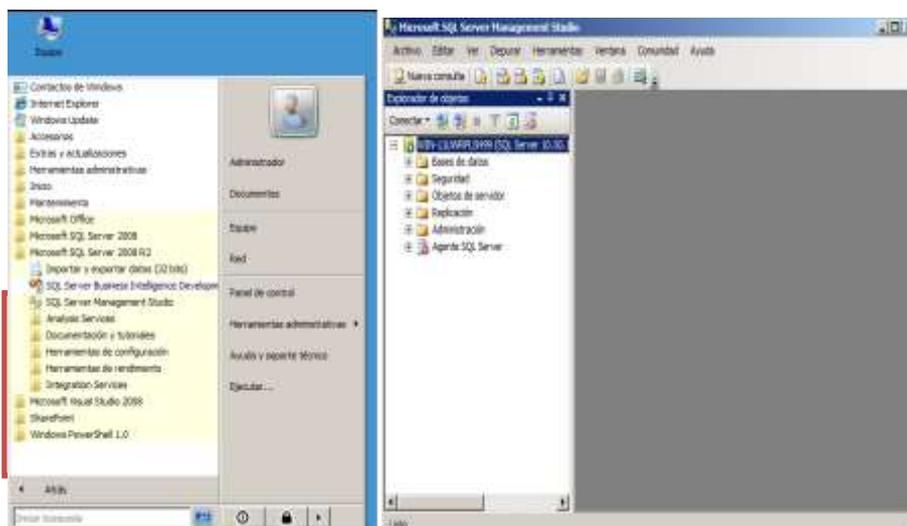
Tendrás que esperar al progreso de instalación varios minutos



Finalmente te mostrara esta pantalla con este mensaje

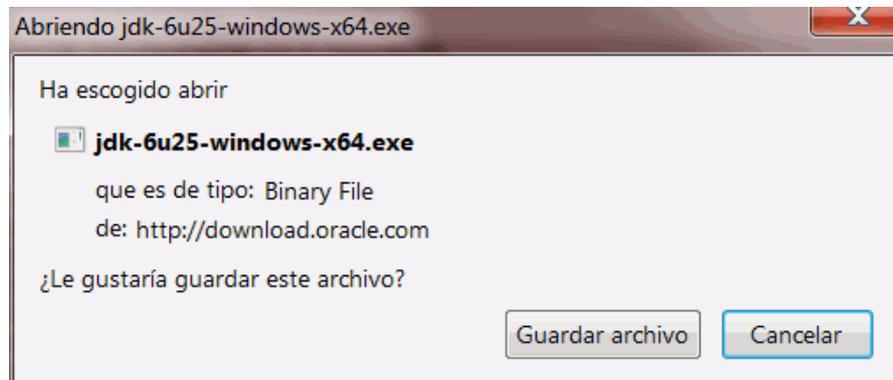


Comprobamos que nuestra instalación es correcta

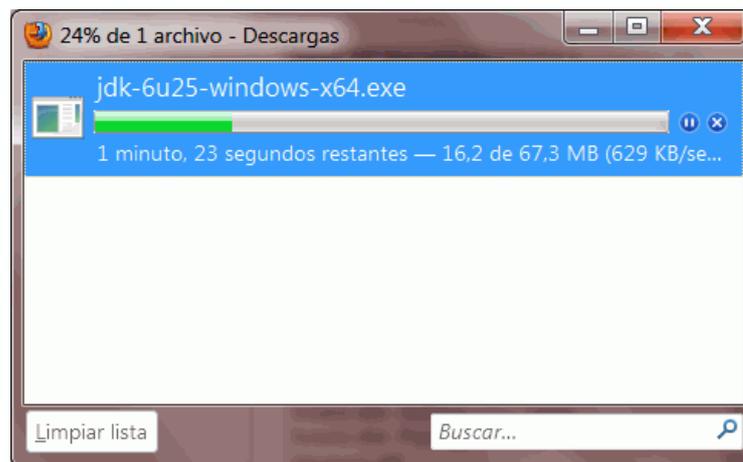


## Instalación de JDK y NETBEANTS 8.0

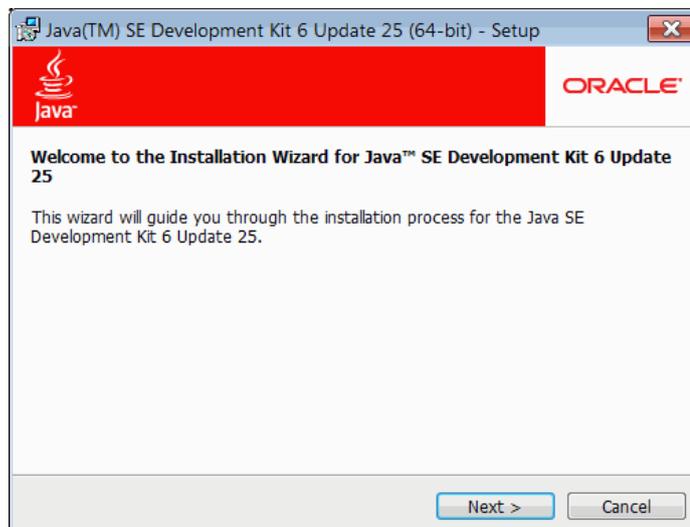
Entonces empezaremos a recibir un único fichero de gran tamaño (cerca de 70 Mb, según versiones):



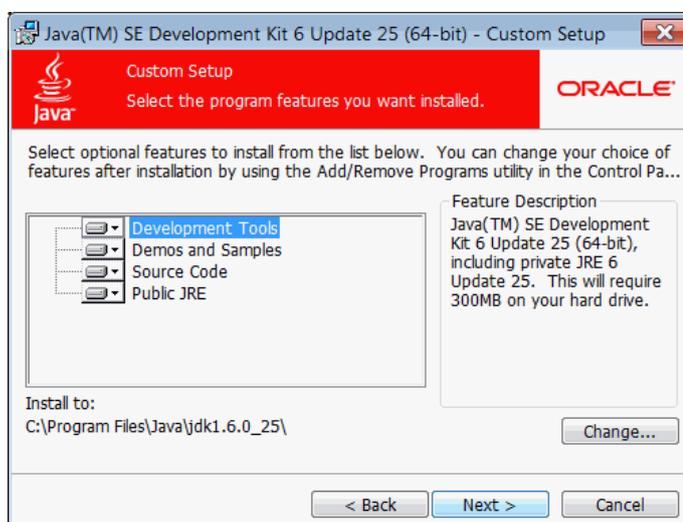
Al tener un tamaño tan grande, la descarga puede ser lenta, según la velocidad de nuestra conexión a Internet:



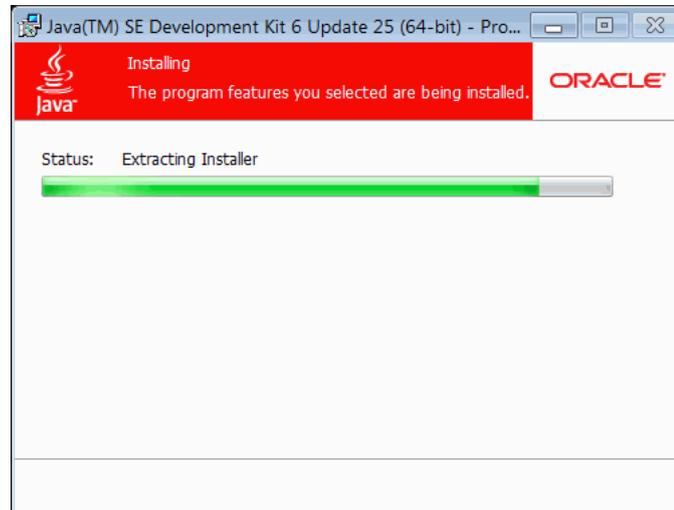
Cuando hayamos descargado, hacemos doble clic en el fichero, para comenzar la instalación propiamente dicha:



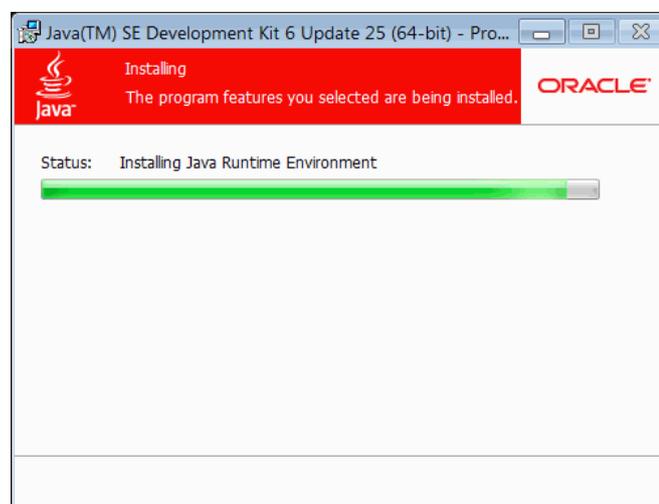
Podremos afinar detalles como la carpeta de instalación, o qué partes no queremos instalar (por ejemplo, podríamos optar por no instalar los ejemplos). Para no complicarnos, si tenemos suficiente espacio (posiblemente unos 400 Mb en total), podemos hacer una instalación típica, sin cambiar nada:



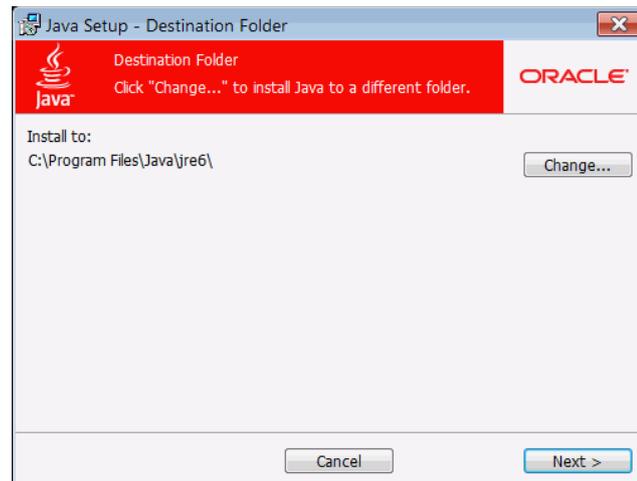
Ahora deberemos tener paciencia durante un rato, mientras se descomprime e instala todo:



En cierto punto se nos preguntará si queremos instalar la máquina virtual Java (Java Runtime Environment, JRE). Lo razonable será responder que sí:



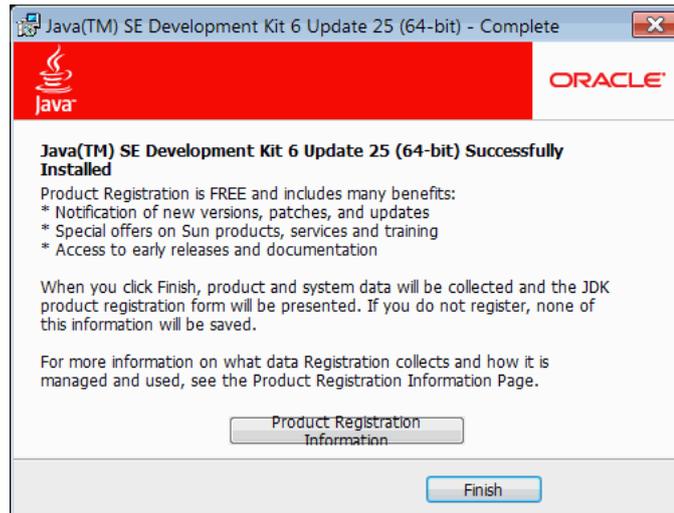
Igual que para el JDK, podríamos cambiar la carpeta de instalación:



Tendremos que esperar otro momento.



Si todo ha ido bien, deberíamos obtener un mensaje de confirmación:



Y se nos propondrá registrarnos nuestra copia en la página de Oracle (no es necesario):

Con eso ya tenemos instalada la herramienta básica.

Pero el kit de desarrollo (JDK) no incluye ningún editor con el que crear nuestros programas. Podríamos instalar un "editor genérico", porque tenemos muchos gratuitos y de calidad, como Notepad++. Aun así, si nuestro equipo es razonablemente moderno, puede ser preferible instalar un entorno integrado, como NetBeans, que encontraremos en

[www.netbeans.org](http://www.netbeans.org)

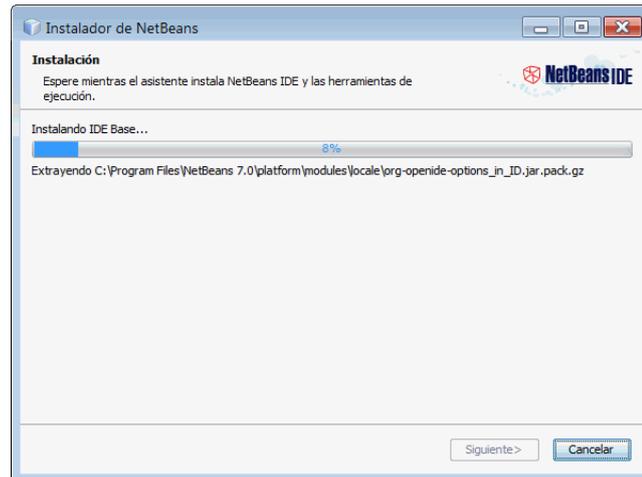


Si hacemos clic en "Download", se nos llevará a la página de descargas, en la que tenemos varias versiones para elegir. Lo razonable "para un novato" es descargar la versión para "Java SE" (las alternativas son otros lenguajes, como PHP o C++, versiones profesionales como Java EE -Enterprise Edition-, o una versión que engloba todas estas posibilidades).

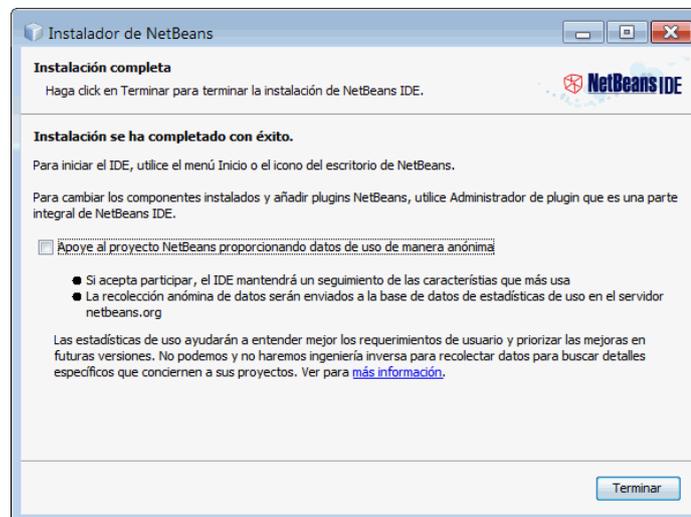
Es fácil que también podamos escoger el Español como idioma, en vez del inglés.



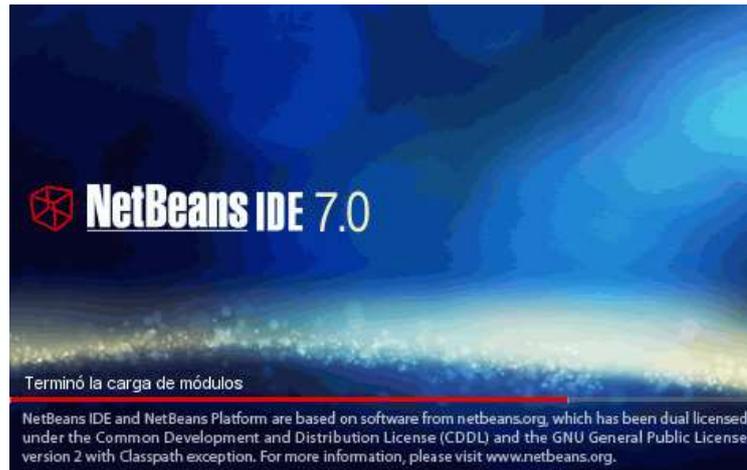
La instalación no será posible si no hemos instalado Java antes, pero si lo hemos hecho, debería ser simple y razonablemente rápida:



Y al final quizá se nos pregunte si queremos permitir que se recopile estadísticas sobre nuestro uso:



Todo listo. Tendremos un nuevo programa en nuestro menú de Inicio. Podemos hacer doble clic para comprobar que se ha instalado correctamente, y debería aparecer la pantalla de carga:



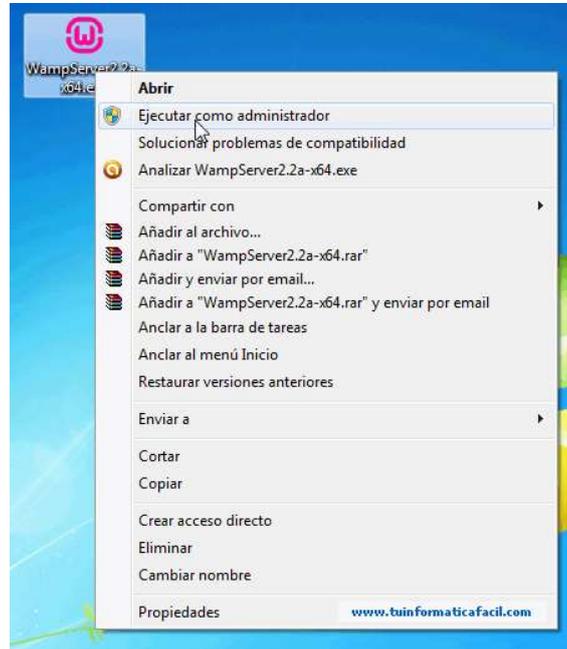
Y después de un instante, la pantalla "normal" de NetBeans:

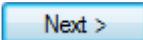


## INSTALACIÓN DE WAMP SERVER 2.0

Una vez tenemos descargado el software WampServer iniciamos la instalación ejecutando el archivo **WampServer2.1e-x32.exe**. Si va a instalar en Windows Vista, Windows 7 o Windows 2008 recordar que tienes que arrancar el ejecutable como administrador, sino lo hacemos así tendremos errores en la instalación y seguramente en la ejecución posterior de WampServer. Para instalar como administrador en el explorador de archivos seleccione el archivo **WampServer2.1e-**

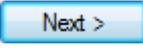
**x32.exe**, pulsas el botón derecho del ratón, aparecerá un menú, y seleccionas pulsando el botón izquierdo del ratón la opción 'Ejecutar como administrador', como podéis ver en la imagen.

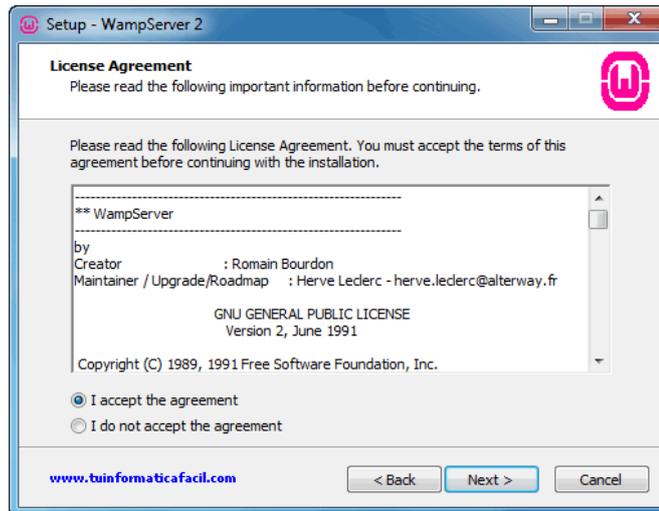


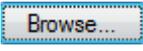
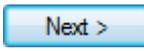
La primera ventana que obtenemos es la de bienvenida al Setup Wizard de la instalación, pulsamos el botón  para continuar.

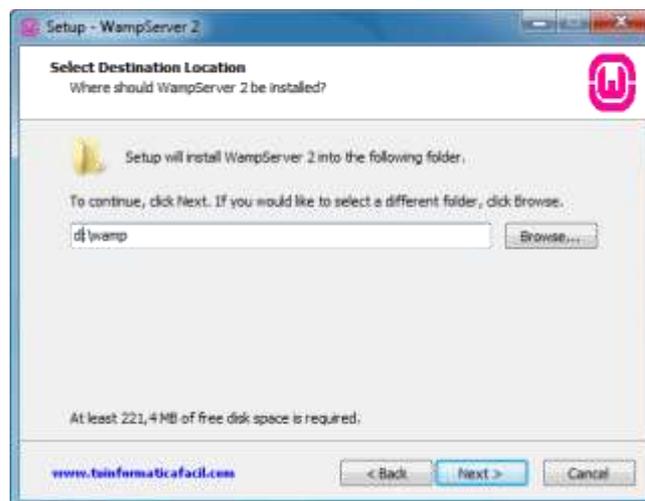


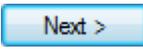
La siguiente ventana nos muestra el acuerdo de licencia, seleccionamos – **I**

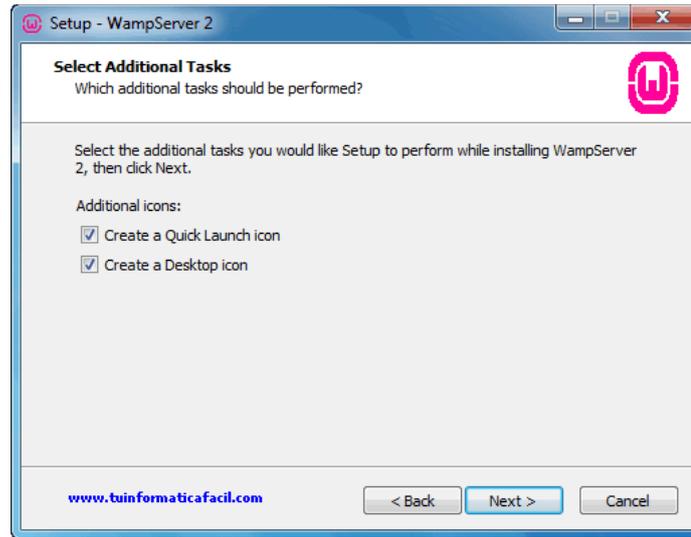
*accept the agreement* - y botón  para continuar.



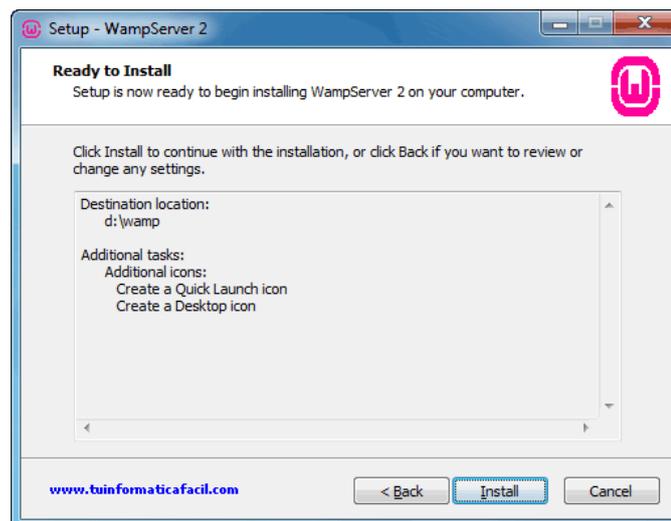
En esta ventana tenemos que determinar en qué directorio vamos realizar la instalación, por defecto nos muestra una ruta pero si deseamos instalar en otra localización pulsamos el botón , una vez tengamos el directorio destino decidido pulsamos el botón  para continuar.

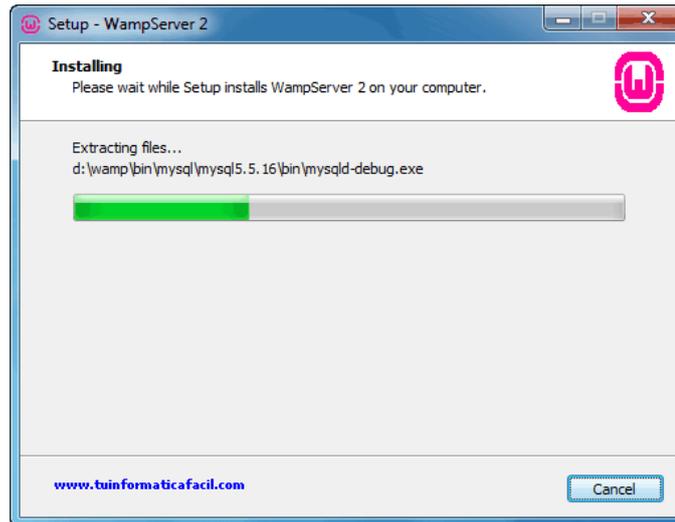


En esta ventana seleccionar los iconos que deseas que cree el instalador automáticamente, icono en el inicio rápido o/y icono en el escritorio, pulsamos el botón  para continuar.

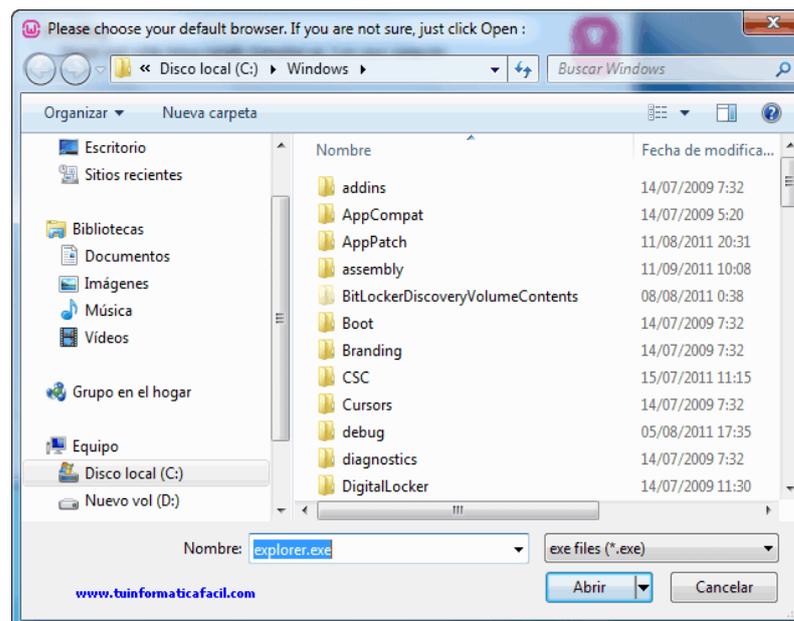


Finalmente obtenemos una ventana con el resumen de las tareas que va a realizar el Wizard de WampServer 2.1e, pulsamos el botón **Install** y la instalación comenzará.



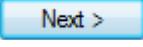


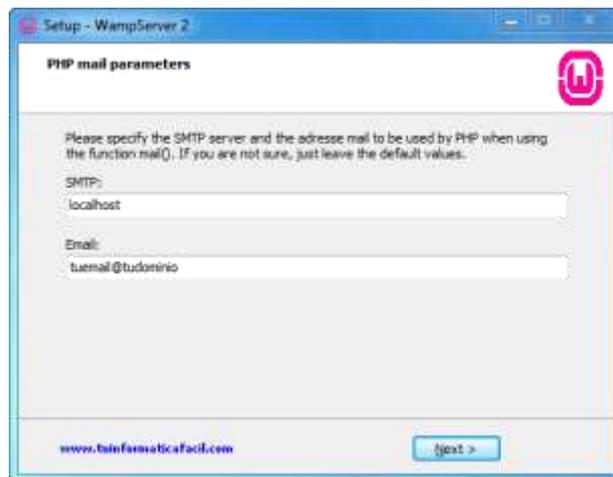
Durante el proceso de instalación aparece el explorador de archivos para que seleccionemos el navegador que vamos a utilizar, por defecto utilizará el Mozilla Firefox, si deseamos utilizar otro debemos buscar el ejecutable y pulsamos el botón **- Abrir -**.



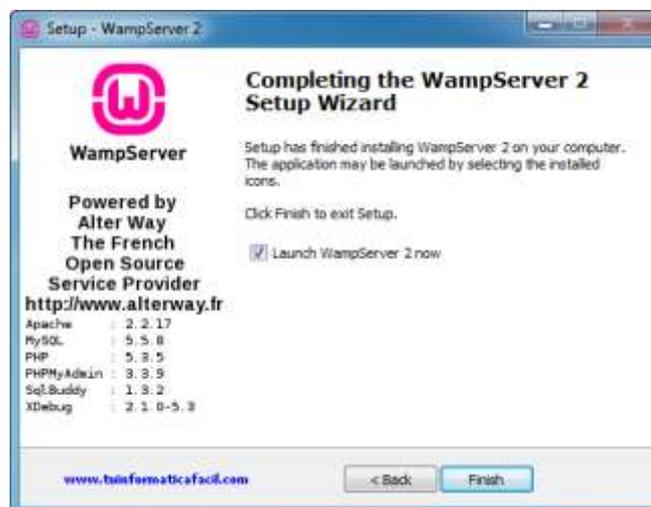
Si en el PC o servidor donde estamos instalando tenemos un servidor de correo (servidor SMTP) introduciremos el nombre del servidor de correo en campo SMTP

y una cuenta válida en el campo Email, esto será usado por la función mail() de PHP como remitente para correos de salida. Si no sabéis o no estáis seguros de tener un servidor de correo dejar los valores que aparecen por defecto.

Pulsamos el botón  para continuar.



Bien ya hemos terminado la instalación, si dejamos chequeado – **Launch WampServer 2 now** – WampServer 2.1e arrancará una vez pulsemos el botón – **Finish** –



### B.01.03 MANUAL TÉCNICO.

#### Programación registro de usuario.

```
gx.evt.autoSkip = !1;

function usuario() {

this.ServerClass = "Usuario";

    this.PackageName = "";

    this.setObjectType("trn");

    this.hasEnterEvent = !0;

    this.SetStandaloneVars = function () {

        this.AV12UsuarioId = gx.fn.getIntegerValue("vUSUARIOID", ".");

        this.A17UsuarioId = gx.fn.getIntegerValue("USUARIOID", ".");

        this.AV8Insert_CategoriaId =

gx.fn.getIntegerValue("vINSERT_CATEGORIAID", ".");

        this.AV16mensaje = gx.fn.getControlValue("vMENSAJE");

        this.AV15f = gx.fn.getControlValue("vF");

        this.A78UsuarioEst = gx.fn.getControlValue("USUARIOEST");

        this.AV19Pgmname =

            gx.fn.getControlValue("vPGMNAME");

        this.Gx_mode = gx.fn.getControlValue("vMODE")

    };

};
```

```
this.Valid_Usuarionomb = function () {  
  
    try {  
  
        var a = gx.util.balloon.getNew("USUARIONOMB");  
  
        this.AnyError = 0;  
  
        if (!gx.util.regExp.isMatch(this.A18UsuarioNomb, "^[\\s A-z ]*$"))  
  
            try {  
  
                a.setError("El campo debe contener solo letras"), this.AnyError = 1  
  
            } catch (b) {  
  
            }  
  
        }  
  
        if (" " == this.A18UsuarioNomb)  
  
            try {  
  
                a.setError("El Nombre es requerido"), this.AnyError = 1  
  
            } catch (c) {  
  
            }  
  
        } catch (d) {  
  
        }  
  
        try {  
  
            return null == a ? !0 : a.show()  
  
        } catch (e) {  
  
        }  
  
    }  
  
}
```

```
    }

    return!0

};

this.Valid_Categoriaid = function () {

    gx.ajax.validSrvEvt("dyncall",

        "valid_Categoriaid", ["gx.O.A74CategoriaId"], []);

    return!0

};

this.Valid_Usuariologi = function () {

    gx.ajax.validSrvEvt("dyncall", "valid_Usuariologi", ["gx.O.A17UsuarioId",

"gx.O.A76UsuarioLogi", "gx.O.AV16mensaje", "gx.O.AV15f"], ["AV16mensaje",

"AV15f"]);

    return!0

};

this.Valid_Usuariocont = function () {

    try {

        var a = gx.util.balloon.getNew("USUARIOCONT");

        this.AnyError = 0;

        if (!gx.util.regExp.isMatch(this.A77UsuarioCont,

"(?=^.{8,}$)((?=.*\\d)|(?=.*\\W+))(?![.\\n])(?=.*[A-Z])(?=.*[a-z]).*$"))
```

```
try {  
  
    a.setError("La contrase\u00f1a no tiene el formato requerido"),  
  
    this.AnyError = 1  
  
} catch (b) {  
  
}  
  
if (" " == this.A77UsuarioCont)  
  
    try {  
  
        a.setError("La Contrase\u00f1a es requerida"), this.AnyError = 1  
  
    } catch (c) {  
  
    }  
  
if (this.AV14UsuarioCont != this.A77UsuarioCont)  
  
    try {  
  
        a.setError("La Contrse\u00f1a no es igual"), this.AnyError = 1  
  
    } catch (d) {  
  
    }  
  
} catch (e) {  
  
}  
  
try {  
  
return null == a ? !0 : a.show()
```

```
    } catch (f) {  
  
    }  
  
    return!0  
  
};  
  
this.Validv_Usuariocont = function () {  
  
    try {  
  
        var a = gx.util.balloon.getNew("vUSUARIOCONT");  
  
        this.AnyError = 0  
  
    } catch (b) {  
  
    }  
  
    try {  
  
        return null == a ? !0 : a.show()  
  
    } catch (c) {  
  
    }  
  
    return!0  
  
};  
  
this.e120a2_client = function () {  
  
    this.executeServerEvent("AFTER TRN",  
  
        !0)
```

```
};

this.e130a12_client = function () {

    this.executeServerEvent("ENTER", !0)

};

this.e140a12_client = function () {

    this.executeServerEvent("CANCEL", !0)

};

this.GXValidFnc = [];

this.GXCtrlIds = [2, 5, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24,
27, 28, 34, 37, 39, 42, 44, 47, 49, 52, 54, 57, 59];

this.GXLastCtrlId = 59;

this.GXValidFnc[2] = {fld: "TABLEMAIN", grid: 0};

this.GXValidFnc[5] = {fld: "TABLETOOLBAR", grid: 0};

this.GXValidFnc[9] = {fld: "BTN_FIRST", grid: 0};

this.GXValidFnc[10] = {fld: "BTN_FIRST_SEPARATOR", grid: 0};

this.GXValidFnc[11] = {fld: "BTN_PREVIOUS",

    grid: 0};

this.GXValidFnc[12] = {fld: "BTN_PREVIOUS_SEPARATOR", grid: 0};

this.GXValidFnc[13] = {fld: "BTN_NEXT", grid: 0};

this.GXValidFnc[14] = {fld: "BTN_NEXT_SEPARATOR", grid: 0};
```

```
this.GXValidFnc[15] = {fld: "BTN_LAST", grid: 0};

this.GXValidFnc[16] = {fld: "BTN_LAST_SEPARATOR", grid: 0};

this.GXValidFnc[17] = {fld: "BTN_SELECT", grid: 0};

this.GXValidFnc[18] = {fld: "BTN_SELECT_SEPARATOR", grid: 0};

this.GXValidFnc[19] = {fld: "BTN_ENTER2", grid: 0};

this.GXValidFnc[20] = {fld: "BTN_ENTER2_SEPARATOR", grid: 0};

this.GXValidFnc[21] = {fld: "BTN_CANCEL2",

    grid: 0};

this.GXValidFnc[22] = {fld: "BTN_CANCEL2_SEPARATOR", grid: 0};

this.GXValidFnc[23] = {fld: "BTN_DELETE2", grid: 0};

this.GXValidFnc[24] = {fld: "BTN_DELETE2_SEPARATOR", grid: 0};

this.GXValidFnc[27] = {fld: "GROUPDATA", grid: 0};

this.GXValidFnc[28] = {fld: "TABLE1", grid: 0};

this.GXValidFnc[34] = {fld: "TABLE2", grid: 0};

this.GXValidFnc[37] = {fld: "TEXTBLOCKUSUARIONOMB", format: 0, grid:

0};

this.GXValidFnc[39] = {lvl: 0, type: "svchar", len: 40, dec: 0, sign: !1, ro: 0, grid:

0, gxgrid: null, fnc: this.Valid_Usuarionomb, isvalid: null, fld: "USUARIONOMB",
```

```
gxz: "Z18UsuarioNomb", gxold: "O18UsuarioNomb", gxvar:
"A18UsuarioNomb", ucs: [], op: [39], ip: [39], nacdep: [], ctrltype: "edit", v2v:
function (a) {

    gx.O.A18UsuarioNomb = a

    }, v2z: function (a) {

    gx.O.Z18UsuarioNomb = a

    }, v2c: function () {

    gx.fn.setControlValue("USUARIONOMB", gx.O.A18UsuarioNomb, 0)

    }, c2v: function () {

    gx.O.A18UsuarioNomb = this.val()

    }, val: function () {

    return gx.fn.getControlValue("USUARIONOMB")

    }, nac: function () {

    return !1

    }};

this.GXValidFnc[42] = {fld: "TEXTBLOCKCATEGORIAID", format: 0, grid:
0};

this.GXValidFnc[44] = {lvl: 0, type: "int",

len: 4, dec: 0, sign: !1, pic: "ZZZ9", ro: 0, grid: 0, gxgrid: null, fnc:

this.Valid_Categoriaid, isvalid: null, rgrid: [], fld: "CATEGORIAID", gxz:
```

```
"Z74Categoriald", gxold: "O74Categoriald", gxvar: "A74Categoriald", ucs: [], op:  
[], ip: [44], nacdep: [], ctrltype: "dyncombo", v2v: function (a) {
```

```
    gx.O.A74Categoriald = gx.num.intval(a)
```

```
  }, v2z: function (a) {
```

```
    gx.O.Z74Categoriald = gx.num.intval(a)
```

```
  }, v2c: function () {
```

```
    gx.fn.setComboBoxValue("CATEGORIAID", gx.O.A74Categoriald)
```

```
  }, c2v: function () {
```

```
    gx.O.A74Categoriald = this.val()
```

```
  }, val: function () {
```

```
    return gx.fn.getIntegerValue("CATEGORIAID",
```

```
      ".")
```

```
  }, nac: function () {
```

```
    return "INS" == this.Gx_mode && 0 != this.AV8Insert_Categoriald
```

```
  }  
});
```

```
this.GXValidFnc[47] = {fld: "TEXTBLOCKUSUARIOLOGI", format: 0, grid:  
0};
```

```
this.GXValidFnc[49] = {lvl: 0, type: "svchar", len: 40, dec: 0, sign: !1, ro: 0, grid:  
0, gxgrid: null, fnc: this.Valid_Usuariologi, isvalid: null, fld: "USUARIOLOGI",
```

gxz: "Z76UsuarioLogi", gxold: "O76UsuarioLogi", gxvar: "A76UsuarioLogi", ucs:

[], op: [49], ip: [49], nacdep: [], ctrltype: "edit", v2v: function (a) {

    gx.O.A76UsuarioLogi = a

}, v2z: function (a) {

    gx.O.Z76UsuarioLogi = a

}, v2c: function () {

gx.fn.setControlValue("USUARIOLOGI",

    gx.O.A76UsuarioLogi, 0)

}, c2v: function () {

    gx.O.A76UsuarioLogi = this.val()

}, val: function () {

    return gx.fn.getControlValue("USUARIOLOGI")

}, nac: function () {

    return !1

}};

this.GXValidFnc[52] = {fld: "TEXTBLOCKUSUARIOCONT", format: 0, grid:  
0};

this.GXValidFnc[54] = {lvl: 0, type: "svchar", len: 40, dec: 0, sign: !1, isPwd: !0,  
ro: 0, grid: 0, gxgrid: null, fnc: this.Valid\_Usuariocont, isvalid: null, fld:

"USUARIOCONT", gxz: "Z77UsuarioCont", gxold: "O77UsuarioCont", gxvar:

"A77UsuarioCont", ucs: [], op: [59, 54], ip: [59, 54], nacdep: [], ctrltype: "edit", v2v:

```
function (a) {
```

```
    gx.O.A77UsuarioCont =
```

```
        a
```

```
    }, v2z: function (a) {
```

```
        gx.O.Z77UsuarioCont = a
```

```
    }, v2c: function () {
```

```
        gx.fn.setControlValue("USUARIOCONT", gx.O.A77UsuarioCont, 0)
```

```
    }, c2v: function () {
```

```
        gx.O.A77UsuarioCont = this.val()
```

```
    }, val: function () {
```

```
        return gx.fn.getControlValue("USUARIOCONT")
```

```
    }, nac: function () {
```

```
return!1
```

```
    }};
```

```
    this.GXValidFnc[57] = {fld: "TEXTBLOCKUSUARIOCONT2", format: 0, grid:  
0};
```

```
    this.GXValidFnc[59] = {lvl: 0, type: "svchar", len: 40, dec: 0, sign: !1, isPwd: !0,  
ro: 1, grid: 0, gxgrid: null, fnc: this.Validv_Usuariocont, isvalid: null, fld:  
"vUSUARIOCONT", gxz: "ZV14UsuarioCont", gxold: "OV14UsuarioCont",
```

```
gxvar: "AV14UsuarioCont", ucs: [], op: [], ip: [], nacdep: [], ctrltype: "edit",
v2v: function (a) {

    gx.O.AV14UsuarioCont = a

    }, v2z: function (a) {

    gx.O.ZV14UsuarioCont = a

    }, v2c: function () {

    gx.fn.setControlValue("vUSUARIOCONT", gx.O.AV14UsuarioCont, 0)

    }, c2v: function () {

    gx.O.AV14UsuarioCont = this.val()

    }, val: function () {

    return gx.fn.getControlValue("vUSUARIOCONT")

    }, nac: function () {

    return!1

    }

    });

this.O18UsuarioNomb = this.Z18UsuarioNomb = this.A18UsuarioNomb = "";

this.O74CategoriaId = this.Z74CategoriaId = this.A74CategoriaId = 0;

this.OV14UsuarioCont =

    this.ZV14UsuarioCont = this.AV14UsuarioCont = this.O77UsuarioCont =

this.Z77UsuarioCont = this.A77UsuarioCont = this.O76UsuarioLogi =

this.Z76UsuarioLogi = this.A76UsuarioLogi = "";
```

```
this.AV18CategoriaId = 0;

    this.AV19Pgmname = "";

    this.AV9IsAuthorized = !1;

    this.AV10TrnContext = {};

    this.AV20GXV1 = this.AV8Insert_CategoriaId = 0;

    this.AV11TrnContextAtt = {};

    this.AV7Context = {};

    this.AV12UsuarioId = 0;

    this.AV17session = {};

    this.AV13WebSession = {};

    this.A17UsuarioId = 0;

    this.Gx_mode = this.A78UsuarioEst = this.AV15f = this.AV16mensaje = "";

    this.ServerEvents =

        ["e120a2_client", "e130a12_client", "e140a12_client"];

    this.EnterCtrl = ["BTN_ENTER2", "BTN_ENTER2_SEPARATOR",
"BTN_ENTER"];

    this.VarControlMap.AV12UsuarioId = {id: "vUSUARIOID", grid: 0, type: "int"};

    this.VarControlMap.A17UsuarioId = {id: "USUARIOID", grid: 0, type: "int"};

    this.VarControlMap.AV8Insert_CategoriaId = {id: "vINSERT_CATEGORIAID",
grid: 0, type: "int"};
```

```
this.VarControlMap.AV16mensaje = {id: "vMENSAJE", grid: 0, type: "svchar"};

this.VarControlMap.AV15f = {id: "vF", grid: 0, type: "svchar"};

this.VarControlMap.A78UsuarioEst = {id: "USUARIOEST",

    grid: 0, type: "char"};

this.VarControlMap.AV19Pgmname = {id: "vPGMNAME", grid: 0, type:

"char"};

this.VarControlMap.Gx_mode = {id: "vMODE", grid: 0, type: "char"};

    this.SetStandaloneVars()

}

usuario.prototype = new gx.GxObject;

gx.setParentObj(new usuario);
```

### **Scrip de la Base de Datos.**

```
SET FOREIGN_KEY_CHECKS=0;
```

```
-----
```

```
-- Table structure for auditoria
```

```
-----
```

```
DROP TABLE IF EXISTS `auditoria`;
```

```
CREATE TABLE `auditoria` (
```

```
    `AuditoriaId` smallint(6) NOT NULL AUTO_INCREMENT,
```

```
`AuditoriaIngr` datetime NOT NULL,  
  
`AuditoriaSali` datetime NOT NULL,  
  
`AuditoriaIp` varchar(20) NOT NULL,  
  
`AuditoriaNave` varchar(40) NOT NULL,  
  
`AuditoriaPrto` varchar(40) NOT NULL,  
  
`UsuarioId` smallint(6) NOT NULL,  
  
PRIMARY KEY (`AuditoriaId`),  
  
KEY `IAUDITORIA1` (`UsuarioId`),  
  
CONSTRAINT `IAUDITORIA1` FOREIGN KEY (`UsuarioId`) REFERENCES  
`usuario` (`UsuarioId`)  
) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=latin1;
```

-----

-- Records of auditoria

-----

```
INSERT INTO `auditoria` VALUES ('1', '2014-08-23 20:52:25', '1000-01-01  
00:00:00', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('2', '2014-08-23 20:53:35', '1000-01-01  
00:00:00', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('3', '2014-08-23 21:15:36', '1000-01-01  
00:00:00', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('4', '2014-08-28 23:40:43', '2014-08-28  
23:47:48', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('5', '2014-08-29 17:52:52', '2014-08-29  
17:53:16', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('6', '2014-08-29 17:53:27', '2014-08-29  
18:31:38', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('7', '2014-08-29 18:31:54', '1000-01-01  
00:00:00', '127.0.0.1', '36.0.1985.143', '', '2');
```

```
INSERT INTO `auditoria` VALUES ('8', '2014-08-31 11:41:01', '1000-01-01  
00:00:00', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('9', '2014-08-31 11:50:46', '2014-08-31  
12:28:00', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('10', '2014-08-31 12:28:13', '2014-08-31  
12:30:05', '127.0.0.1', '36.0.1985.143', '', '2');
```

```
INSERT INTO `auditoria` VALUES ('11', '2014-09-02 19:40:46', '1000-01-01  
00:00:00', '127.0.0.1', '36.0.1985.143', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('12', '2014-09-09 21:06:24', '1000-01-01  
00:00:00', '127.0.0.1', '37.0.2062.103', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('13', '2014-09-13 08:44:05', '1000-01-01  
00:00:00', '127.0.0.1', '37.0.2062.120', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('14', '2014-09-17 10:48:49', '1000-01-01  
00:00:00', '127.0.0.1', '37.0.2062.120', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('15', '2014-09-17 12:11:44', '1000-01-01  
00:00:00', '127.0.0.1', '37.0.2062.120', '', '1');
```

```
INSERT INTO `auditoria` VALUES ('16', '2014-09-25 17:40:10', '1000-01-01  
00:00:00', '127.0.0.1', '37.0.2062.120', '', '1');
```

```
-----  
  
-- Table structure for auditoriadetalle  
  
-----
```

```
DROP TABLE IF EXISTS `audoriadetalle`;
```

```
CREATE TABLE `audoriadetalle` (  
  
  `AudoriaDetalleId` smallint(6) NOT NULL AUTO_INCREMENT,  
  
  `AudoriaId` smallint(6) NOT NULL,  
  
  `AudoriaDetalleDesc` varchar(100) NOT NULL,  
  
  `AudoriaDetalleFech` datetime NOT NULL,  
  
  `AudoriaDetalleAcci` varchar(40) NOT NULL,  
  
  PRIMARY KEY (`AudoriaDetalleId`),  
  
  KEY `IAUDORIADETALLE1` (`AudoriaId`),
```

```
CONSTRAINT `IAUDITORIADETALLE1` FOREIGN KEY (`AuditoriaId`)
REFERENCES `auditoria` (`AuditoriaId`)
) ENGINE=InnoDB AUTO_INCREMENT=27 DEFAULT CHARSET=latin1;

-----

-- Records of auditoriadetalle

-----

INSERT INTO `auditoriadetalle` VALUES ('1', '6', 'Tipo de Producto', '2014-08-29
18:00:30', 'Insertar');

INSERT INTO `auditoriadetalle` VALUES ('2', '6', 'Ubicación', '2014-08-29
18:01:07', 'Insertar');

INSERT INTO `auditoriadetalle` VALUES ('3', '6', 'Producto', '2014-08-29
18:01:38', 'Insertar');

INSERT INTO `auditoriadetalle` VALUES ('4', '6', 'Persona', '2014-08-29 18:09:53',
'Insertar');

INSERT INTO `auditoriadetalle` VALUES ('5', '6', 'Ingresos', '2014-08-29 18:14:15',
'Insertar');

INSERT INTO `auditoriadetalle` VALUES ('6', '6', 'Ingresos', '2014-08-29 18:17:34',
'Insertar');

INSERT INTO `auditoriadetalle` VALUES ('7', '6', 'Persona', '2014-08-29 18:22:28',
'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('8', '6', 'Salida', '2014-08-29 18:25:08',  
'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('9', '6', 'Usuario', '2014-08-29 18:30:23',  
'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('10', '6', 'Usuario', '2014-08-29  
18:31:09', 'Actualizar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('11', '9', 'Producto', '2014-08-31  
11:54:48', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('12', '9', 'Tipo de Producto', '2014-08-31  
11:55:35', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('13', '9', 'Persona', '2014-08-31  
11:58:13', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('14', '9', 'Ingresos', '2014-08-31  
12:00:50', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('15', '9', 'Producto', '2014-08-31  
12:07:21', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('16', '9', 'Ingresos', '2014-08-31  
12:15:42', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('17', '9', 'Salida', '2014-08-31 12:18:06',  
'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('18', '9', 'Salida', '2014-08-31 12:18:16',  
'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('19', '11', 'Producto', '2014-09-02  
19:43:52', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('20', '11', 'Ingresos', '2014-09-02  
20:12:35', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('21', '11', 'Salida', '2014-09-02  
20:15:17', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('22', '11', 'Salida', '2014-09-02  
20:15:25', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('23', '11', 'Salida', '2014-09-02  
20:15:36', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('24', '11', 'Salida', '2014-09-02  
20:16:24', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('25', '11', 'Salida', '2014-09-02  
20:18:28', 'Insertar');
```

```
INSERT INTO `auditoriadetalle` VALUES ('26', '16', 'Producto', '2014-09-25  
17:51:35', 'Insertar');
```

```
-----  
  
-- Table structure for categoria
```

```
-----  
  
DROP TABLE IF EXISTS `categoria`;
```

```
CREATE TABLE `categoria` (  
  
  `CategoriaId` smallint(6) NOT NULL,  
  
  `CategoriaDesc` varchar(40) NOT NULL,  
  
  PRIMARY KEY (`CategoriaId`)  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
-----  
  
-- Records of categoria  
  
-----  
  
INSERT INTO `categoria` VALUES ('1', 'ADMINISTRADOR');  
  
INSERT INTO `categoria` VALUES ('2', 'USUARIO');  
  
-----  
  
-- Table structure for compra  
  
-----  
  
DROP TABLE IF EXISTS `compra`;  
  
CREATE TABLE `compra` (  
  
  `CompraId` smallint(6) NOT NULL AUTO_INCREMENT,  
  
  `CompraDocu` varchar(15) NOT NULL,
```

```
`CompraFech` datetime NOT NULL,  
  
`CompraCadu` date NOT NULL,  
  
`ResponsableProveedorId` smallint(6) NOT NULL,  
  
`CompraAuto` varchar(40) NOT NULL,  
  
`CompraObser` varchar(200) NOT NULL,  
  
`CompraEsta` char(1) NOT NULL,  
  
`CompraDesc` decimal(7,2) NOT NULL,  
  
PRIMARY KEY (`CompraId`),  
  
KEY `ICOMPRA1` (`ResponsableProveedorId`),  
  
CONSTRAINT `ICOMPRA1` FOREIGN KEY (`ResponsableProveedorId`)  
REFERENCES `responsableproveedor` (`ResponsableProveedorId`)  
  
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;  
  
-----  
  
-- Records of compra  
  
-----  
  
INSERT INTO `compra` VALUES ('1', '0000000000001', '2014-08-29 18:12:00',  
'2014-08-30', '1', '990989900', '', 'T', '0.00');  
  
INSERT INTO `compra` VALUES ('2', '0000000000002', '2014-08-29 18:17:00',  
'2014-08-30', '1', '88900', '', 'A', '0.00');
```

```
INSERT INTO `compra` VALUES ('3', '0000000000003', '2014-08-31 11:58:00',  
'2014-09-01', '3', '12', '', 'A', '0.05');
```

```
INSERT INTO `compra` VALUES ('4', '0010017899999', '2014-08-31 12:08:00',  
'2014-09-11', '1', '23333333333', 'prueba', 'A', '0.05');
```

```
INSERT INTO `compra` VALUES ('5', '0000000000004', '2014-09-02 20:03:00',  
'2014-09-03', '3', '0909090909', '', 'A', '5.00');
```

```
-----  
  
-- Table structure for compradetalle
```

```
-----  
  
DROP TABLE IF EXISTS `compradetalle`;
```

```
CREATE TABLE `compradetalle` (  
  
  `CompraId` smallint(6) NOT NULL,  
  
  `ProductoId` smallint(6) NOT NULL,  
  
  `CompraDetalleId` smallint(6) NOT NULL,  
  
  `CompraDetalleCant` int(11) NOT NULL,  
  
  `CompraDetallePuni` decimal(15,2) NOT NULL,  
  
  PRIMARY KEY (`CompraId`, `ProductoId`),  
  
  KEY `ICOMPRADETALLE1` (`ProductoId`),
```

```
CONSTRAINT `ICOMPRADETALLE1` FOREIGN KEY (`ProductId`)
REFERENCES `producto` (`ProductId`),

CONSTRAINT `ICOMPRADETALLE2` FOREIGN KEY (`CompraId`)
REFERENCES `compra` (`CompraId`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-- -----
```

```
-- Records of compradetalle
```

```
-- -----
```

```
INSERT INTO `compradetalle` VALUES ('1', '1', '0', '100', '0.25');
INSERT INTO `compradetalle` VALUES ('2', '1', '0', '100', '0.50');
INSERT INTO `compradetalle` VALUES ('3', '2', '0', '20', '10.00');
INSERT INTO `compradetalle` VALUES ('4', '1', '0', '3330', '0.04');
INSERT INTO `compradetalle` VALUES ('4', '2', '0', '54', '34.00');
INSERT INTO `compradetalle` VALUES ('5', '3', '0', '3', '2.00');
INSERT INTO `compradetalle` VALUES ('5', '4', '0', '56', '10.00');
```

```
-- -----
```

```
-- Table structure for egreso
```

```
-- -----
```

```
DROP TABLE IF EXISTS `egreso`;

CREATE TABLE `egreso` (

  `EgresoId` smallint(6) NOT NULL AUTO_INCREMENT,

  `ResponsableProveedorId` smallint(6) NOT NULL,

  `EgresoFech` datetime NOT NULL,

  `EgresoEsta` char(1) NOT NULL,

  `EgresoObser` varchar(200) NOT NULL,

  PRIMARY KEY (`EgresoId`),

  KEY `IEGRESO1` (`ResponsableProveedorId`),

  CONSTRAINT `IEGRESO1` FOREIGN KEY (`ResponsableProveedorId`)

REFERENCES `responsableproveedor` (`ResponsableProveedorId`)

) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=latin1;

-----

-- Records of egreso

-----

INSERT INTO `egreso` VALUES ('1', '2', '2014-08-29 18:22:00', 'T', '');

INSERT INTO `egreso` VALUES ('3', '2', '2014-08-31 12:17:00', 'A', '');

INSERT INTO `egreso` VALUES ('6', '2', '2014-09-02 20:14:00', 'A', 'jjjj');

INSERT INTO `egreso` VALUES ('7', '2', '2014-09-02 20:15:00', 'A', 'jjjj');
```

```
INSERT INTO `egreso` VALUES ('8', '2', '2014-09-02 20:16:00', 'A', 'jjjj');
```

```
-- -----
```

```
-- Table structure for egresodetalle
```

```
-- -----
```

```
DROP TABLE IF EXISTS `egresodetalle`;
```

```
CREATE TABLE `egresodetalle` (
```

```
  `EgresoId` smallint(6) NOT NULL,
```

```
  `ProductoId` smallint(6) NOT NULL,
```

```
  `EgresoDetalleId` smallint(6) NOT NULL,
```

```
  `EgresoDetalleExis` smallint(6) NOT NULL,
```

```
  `EgresoDetalleCost` decimal(15,2) NOT NULL,
```

```
  `EgresoDetalleCant` smallint(6) NOT NULL,
```

```
  PRIMARY KEY (`EgresoId`, `ProductoId`),
```

```
  KEY `IEGRESODETALLE1` (`ProductoId`),
```

```
  CONSTRAINT `IEGRESODETALLE1` FOREIGN KEY (`ProductoId`)
```

```
REFERENCES `producto` (`ProductoId`),
```

```
  CONSTRAINT `IEGRESODETALLE2` FOREIGN KEY (`EgresoId`)
```

```
REFERENCES `egreso` (`EgresoId`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-----  
  
-- Records of egresodetalle  
  
-----  
  
INSERT INTO `egresodetalle` VALUES ('1', '1', '0', '200', '0.37', '150');  
  
INSERT INTO `egresodetalle` VALUES ('3', '2', '0', '74', '27.51', '74');  
  
INSERT INTO `egresodetalle` VALUES ('6', '3', '0', '3', '2.00', '3');  
  
INSERT INTO `egresodetalle` VALUES ('7', '1', '0', '3430', '0.04', '4');  
  
INSERT INTO `egresodetalle` VALUES ('8', '1', '0', '3426', '0.04', '7');  
  
INSERT INTO `egresodetalle` VALUES ('8', '4', '0', '56', '10.00', '3');  
  
  
-----  
  
-- Table structure for kardex  
  
-----  
  
DROP TABLE IF EXISTS `kardex`;  
  
CREATE TABLE `kardex` (  
  
    `KardexId` smallint(6) NOT NULL AUTO_INCREMENT,  
  
    `ProductoId` smallint(6) NOT NULL,  
  
    `KardexFech` datetime NOT NULL,
```

```
`KardexDeta` varchar(40) NOT NULL,  
  
`KardexIngrCant` smallint(6) NOT NULL,  
  
`KardexIngrVUni` decimal(15,2) NOT NULL,  
  
`KardexIngrVTot` decimal(15,2) NOT NULL,  
  
`KardexSaliCant` smallint(6) NOT NULL,  
  
`KardexSaliVUni` decimal(15,2) NOT NULL,  
  
`KardexSaliVTot` decimal(15,2) NOT NULL,  
  
`KardexExisCant` smallint(6) NOT NULL,  
  
`KardexExisVUni` decimal(15,2) NOT NULL,  
  
`KardexExisTot` decimal(15,2) NOT NULL,  
  
`KardexComp` smallint(6) NOT NULL,  
  
`KardexEgre` smallint(6) NOT NULL,  
  
PRIMARY KEY (`KardexId`),  
  
KEY `IKARDEX1` (`ProductoId`),  
  
CONSTRAINT `IKARDEX1` FOREIGN KEY (`ProductoId`) REFERENCES  
`producto` (`ProductoId`)  
  
) ENGINE=InnoDB AUTO_INCREMENT=16 DEFAULT CHARSET=latin1;  
  
-----  
  
-- Records of kardex
```

-----  
INSERT INTO `kardex` VALUES ('1', '1', '2014-08-29 18:14:15', 'Ingreso', '100',  
'0.25', '25.00', '0', '0.00', '0.00', '100', '0.25', '25.00', '1', '0');

INSERT INTO `kardex` VALUES ('2', '1', '2014-08-29 18:17:34', 'Ingreso', '100',  
'0.50', '50.00', '0', '0.00', '0.00', '200', '0.37', '75.00', '2', '0');

INSERT INTO `kardex` VALUES ('3', '1', '2014-08-29 18:25:08', 'Egreso', '0', '0.00',  
'0.00', '150', '0.37', '55.50', '50', '0.37', '18.50', '0', '1');

INSERT INTO `kardex` VALUES ('4', '1', '2014-08-29 18:33:04', 'Anulación  
Egreso', '150', '0.37', '55.50', '0', '0.00', '0.00', '200', '0.37', '74.00', '0', '1');

INSERT INTO `kardex` VALUES ('5', '1', '2014-08-29 18:36:52', 'Anulacion', '0',  
'0.00', '0.00', '100', '0.25', '25.00', '100', '0.25', '25.00', '1', '0');

INSERT INTO `kardex` VALUES ('6', '2', '2014-08-31 12:00:50', 'Ingreso', '20',  
'10.00', '200.00', '0', '0.00', '0.00', '20', '10.00', '200.00', '3', '0');

INSERT INTO `kardex` VALUES ('7', '1', '2014-08-31 12:15:42', 'Ingreso', '3330',  
'0.04', '133.20', '0', '0.00', '0.00', '3430', '0.04', '170.19', '4', '0');

INSERT INTO `kardex` VALUES ('8', '2', '2014-08-31 12:15:42', 'Ingreso', '54',  
'34.00', '1836.00', '0', '0.00', '0.00', '74', '27.51', '2035.99', '4', '0');

INSERT INTO `kardex` VALUES ('9', '2', '2014-08-31 12:18:16', 'Egreso', '0', '0.00',  
'0.00', '74', '27.51', '2035.74', '0', '27.51', '0.00', '0', '3');

INSERT INTO `kardex` VALUES ('10', '3', '2014-09-02 20:12:35', 'Ingreso', '3',  
'2.00', '6.00', '0', '0.00', '0.00', '3', '2.00', '6.00', '5', '0');

```
INSERT INTO `kardex` VALUES ('11', '4', '2014-09-02 20:12:35', 'Ingreso', '56',  
'10.00', '560.00', '0', '0.00', '0.00', '56', '10.00', '560.00', '5', '0');
```

```
INSERT INTO `kardex` VALUES ('12', '3', '2014-09-02 20:15:36', 'Egreso', '0',  
'0.00', '0.00', '3', '2.00', '6.00', '0', '2.00', '0.00', '0', '6');
```

```
INSERT INTO `kardex` VALUES ('13', '1', '2014-09-02 20:16:24', 'Egreso', '0',  
'0.00', '0.00', '4', '0.04', '0.16', '3426', '0.04', '137.04', '0', '7');
```

```
INSERT INTO `kardex` VALUES ('14', '1', '2014-09-02 20:18:28', 'Egreso', '0',  
'0.00', '0.00', '7', '0.04', '0.28', '3419', '0.04', '136.76', '0', '8');
```

```
INSERT INTO `kardex` VALUES ('15', '4', '2014-09-02 20:18:28', 'Egreso', '0',  
'0.00', '0.00', '3', '10.00', '30.00', '53', '10.00', '530.00', '0', '8');
```

```
-----  
  
-- Table structure for producto  
  
-----
```

```
DROP TABLE IF EXISTS `producto`;
```

```
CREATE TABLE `producto` (
```

```
  `ProductoId` smallint(6) NOT NULL AUTO_INCREMENT,
```

```
  `ProductoDesc` varchar(40) NOT NULL,
```

```
  `ProductoExit` smallint(6) NOT NULL,
```

```
  `ProductoImg` longblob NOT NULL,
```

```
`ProductoImg_GXI` varchar(2048) DEFAULT NULL,  
  
`ProductoCost` decimal(15,2) NOT NULL,  
  
`ProductoEsta` char(1) NOT NULL,  
  
`ProductoCodi` varchar(40) NOT NULL,  
  
`TipoProductoId` smallint(6) NOT NULL,  
  
`UbacionId` smallint(6) NOT NULL,  
  
`ProductoMini` smallint(6) NOT NULL,  
  
PRIMARY KEY (`ProductoId`),  
  
KEY `IPRODUCTO1` (`UbacionId`),  
  
KEY `IPRODUCTO2` (`TipoProductoId`),  
  
CONSTRAINT `IPRODUCTO1` FOREIGN KEY (`UbacionId`) REFERENCES  
`ubicacion` (`UbacionId`),  
  
CONSTRAINT `IPRODUCTO2` FOREIGN KEY (`TipoProductoId`)  
REFERENCES `tipoproducto` (`TipoProductoId`)  
  
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;  
  
-----  
  
-- Records of producto  
  
-----  
  
INSERT INTO `producto` VALUES ('1', 'PERNOS', '3419',
```

```
INSERT INTO `producto` VALUES ('2', 'aluminio 5 pulgadas', '0', "", "", '27.51', "",  
'ALM65', '1', '1', '30');
```

```
INSERT INTO `producto` VALUES ('3', 'PERNO 2"', '0', "", "", '2.00', "", 'per02', '1',  
'1', '500');
```

```
INSERT INTO `producto` VALUES ('4', 'tool', '53', "", "", '10.00', "", "", '1', '1', '1');
```

```
INSERT INTO `producto` VALUES ('5', 'jhvjhgk', '0', "", "", '0.00', "", 'jyhvhjv', '1', '1',  
'20');
```

```
-----  
  
-- Table structure for responsableproveedor  
  
-----
```

```
DROP TABLE IF EXISTS `responsableproveedor`;
```

```
CREATE TABLE `responsableproveedor` (
```

```
  `ResponsableProveedorId` smallint(6) NOT NULL AUTO_INCREMENT,
```

```
  `ResponsableProveedorTipo` char(1) NOT NULL,
```

```
  `ResponsableProveedorCeRu` char(15) NOT NULL,
```

```
  `ResponsableProveedorNomb` varchar(40) NOT NULL,
```

```
  `ResponsableProveedorRazo` varchar(40) NOT NULL,
```

```
  `ResponsableProveedorTele` char(9) NOT NULL,
```

```
  `ResponsableProveedorCelu` char(10) NOT NULL,
```

```
`ResponsibleProveedorMail` varchar(100) NOT NULL,  
  
`ResponsibleProveedorEsta` char(1) NOT NULL,  
  
PRIMARY KEY (`ResponsibleProveedorId`)  
  
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;  
  
-----  
  
-- Records of responsableproveedor  
  
-----  
  
INSERT INTO `responsableproveedor` VALUES ('1', 'C', '1721064408', 'LUIS  
FERNANDO', 'LUIS FERNANDO', '022786051', '9999999999', 'M@M.COM', 'P');  
  
INSERT INTO `responsableproveedor` VALUES ('2', 'C', '1725700171', 'KARINA ',  
'KARINA ', '023450457', '0989407299', 'steff-piscis@hotmail.com', 'R');  
  
INSERT INTO `responsableproveedor` VALUES ('3', 'C', '0502725393', 'Darwin',  
'Darwin', '023748589', '0957834838', 'darwin@vega.com', 'P');  
  
-----  
  
-- Table structure for tipoproducto  
  
-----  
  
DROP TABLE IF EXISTS `tipoproducto`;  
  
CREATE TABLE `tipoproducto` (  
  
  `TipoProductoId` smallint(6) NOT NULL AUTO_INCREMENT,
```

```
`TipoProductoDesc` varchar(40) NOT NULL,  
  
PRIMARY KEY (`TipoProductoId`)  
  
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;  
  
-----  
  
-- Records of tipoproducto  
  
-----  
  
INSERT INTO `tipoproducto` VALUES ('1', 'METALICOS');  
  
INSERT INTO `tipoproducto` VALUES ('2', 'PLASTICO');  
  
  
  
-----  
  
-- Table structure for ubicacion  
  
-----  
  
DROP TABLE IF EXISTS `ubicacion`;  
  
CREATE TABLE `ubicacion` (  
  
  `UbicacionId` smallint(6) NOT NULL AUTO_INCREMENT,  
  
  `Ubicaciondesc` varchar(40) NOT NULL,  
  
PRIMARY KEY (`UbicacionId`)  
  
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;
```

```
-----  
  
-- Records of ubicacion  
  
-----  
  
INSERT INTO `ubicacion` VALUES ('1', 'PERCHA N15');
```

```
-----  
  
-- Table structure for usuario  
  
-----
```

```
DROP TABLE IF EXISTS `usuario`;  
  
CREATE TABLE `usuario` (  
  
  `UsuarioId` smallint(6) NOT NULL AUTO_INCREMENT,  
  
  `CategoriaId` smallint(6) NOT NULL,  
  
  `UsuarioNomb` varchar(40) NOT NULL,  
  
  `UsuarioLogi` varchar(40) NOT NULL,  
  
  `UsuarioCont` varchar(40) NOT NULL,  
  
  `UsuarioEst` char(1) NOT NULL,  
  
  PRIMARY KEY (`UsuarioId`),  
  
  KEY `IUSUARIOI1` (`CategoriaId`),
```

```
CONSTRAINT `IUSUARIO1` FOREIGN KEY (`CategoriaId`) REFERENCES  
`categoria` (`CategoriaId`)
```

```
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;
```

```
-----
```

```
-- Records of usuario
```

```
-----
```

```
INSERT INTO `usuario` VALUES ('1', '1', 'ROOT', 'ROOT', 'ROOT', 'A');
```

```
INSERT INTO `usuario` VALUES ('2', '2', 'Karina', 'Karina', 'Karina1234', '');
```

### **HERRAMIENTAS.**

- ✓ Windows 2007.
- ✓ NetBeans 8.0.
- ✓ SQL Server 2008 R2.
- ✓ Word 2010.
- ✓ Excel 2010
- ✓ Project 2007
- ✓ Rational Rose (untitled).
- ✓ Paint.
- ✓ Nitro Pro 8.
- ✓ Computador Portátil HP Core i5.
- ✓ Impresora Epson tinta continua.
- ✓ Flash

## BIBLIOGRAFÍA.

<http://paulacpascuas.wordpress.com/>

<http://www.monografias.com/trabajos6/mama/mama.shtml#riesgos>

## ESTÁNDARES DE PROGRAMACIÓN.

[https://www.google.com.ec/?gfe\\_rd=cr&ei=DngpVMPAEMfKgAS81oGoCw#q=objetivo+de+los+estandares+de+programacion](https://www.google.com.ec/?gfe_rd=cr&ei=DngpVMPAEMfKgAS81oGoCw#q=objetivo+de+los+estandares+de+programacion)

[https://www.google.com.ec/?gfe\\_rd=cr&ei=DngpVMPAEMfKgAS81oGoCw#q=estandares+de+programacion](https://www.google.com.ec/?gfe_rd=cr&ei=DngpVMPAEMfKgAS81oGoCw#q=estandares+de+programacion)

## INTERFAZ DE USUARIO

<http://www.monografias.com/trabajos10/diusuar/diusuar.shtml>

[http://es.wikipedia.org/wiki/Interfaz\\_de\\_usuario](http://es.wikipedia.org/wiki/Interfaz_de_usuario)

<https://www.google.com.ec/search?q=mapas+conceptuales+dise%C3%B1o+de+interfaz+de+usuario&biw=1366&bih=667&tbm=isch&tbo=u&source=univ&sa=X&ei=BsQhVMXPC4nIgwTsq4D4AQ&sqi=2&ved=0CBoQsAQ#imgdii=>

## PRUEBAS DE ACEPTACIÓN.

[https://www.google.com.ec/?gfe\\_rd=cr&ei=\\_B0wVKPONoOrhQTesoGoDA#q=especificacion+de+pruebas+de+aceptacion](https://www.google.com.ec/?gfe_rd=cr&ei=_B0wVKPONoOrhQTesoGoDA#q=especificacion+de+pruebas+de+aceptacion)

## PRUEBAS DE UNIDAD.

[https://www.google.com.ec/?gfe\\_rd=cr&ei=\\_B0wVKPONoOrhQTesoGoDA#q=especificacion+de+pruebas+de+unidad](https://www.google.com.ec/?gfe_rd=cr&ei=_B0wVKPONoOrhQTesoGoDA#q=especificacion+de+pruebas+de+unidad)

## **PRUEBAS DE CARGA.**

<http://es.slideshare.net/elgato801/pruebas-de-carga>

## **MANUAL DE INSTALACIÓN DE WAMP SERVER**

[http://www.ieonline.unan.edu.ni/av1/pluginfile.php/76730/mod\\_resource/content/1/manual\\_wampserver2.0.pdf](http://www.ieonline.unan.edu.ni/av1/pluginfile.php/76730/mod_resource/content/1/manual_wampserver2.0.pdf)

## **MANUAL DE INSTALACIÓN DE NETBEANS 8.0**

<http://www.aprendeaprogramar.com/mod/resource/view.php?id=205>

## **MANUAL DE INSTALACIÓN DE SQL SERVER R2 2008**

<http://puntocode.com.mx/2010/12/17/gua-de-instalacin-de-sql-server-2008-r2-paso-a-paso/>