



PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

Esta obra ha sido publicada bajo la licencia Creative Commons
Reconocimiento-No comercial-Compartir bajo la misma licencia 2.5 Perú.

Para ver una copia de dicha licencia, visite
<http://creativecommons.org/licenses/by-nc-sa/2.5/pe/>



PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE APOYO A LA GESTIÓN DE INFORMACIÓN EN EMPRESAS DE PRODUCCIÓN BASADO EN LOGÍSTICA INVERSA

Tesis para optar el Título de Ingeniero Informático, que presenta el
bachiller

Félix Melchor Santos López

Asesora: Ing. Olga Ramírez Pilares.

Lima, Noviembre de 2009

Resumen

El presente proyecto abarca el desarrollo de un sistema de apoyo a la gestión de información en empresas de producción basado en logística, siendo la aplicación de logística inversa en la etapa productiva y tomando en cuenta que las materias primas son no perecibles.

Por ello, en el capítulo 1 se muestra las generalidades del proyecto desarrollado así como sus conceptos básicos que permiten un entendimiento teórico del tema.

Asimismo, se presenta la definición del problema, es decir, la ausencia de un sistema de información para la aplicación de logística inversa y los beneficios de su aplicación y uso. Así también, se presenta el objetivo general del proyecto y los objetivos específicos necesarios para llevarlo a cabo. Seguidamente, se presenta el marco conceptual del tema y el problema a tratar en sus definiciones más importantes y detalladas, así como el entendimiento completo del proyecto a desarrollar.

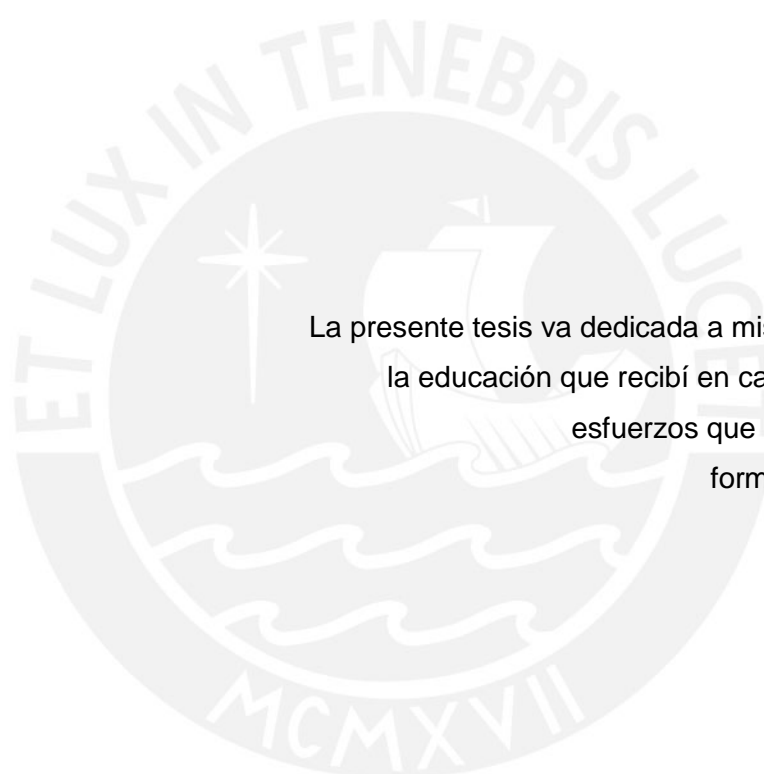
También se presenta el plan de proyecto, la justificación de la elección de la metodología elegida para la dirección de proyectos y el cronograma de trabajo. En la siguiente sección se presenta el estado del arte, que involucra a cuatro casos reales de la aplicación de logística inversa, así como los beneficios de estos sistemas. Finalmente, para concluir el capítulo 1 se presenta la solución planteada tanto para los planes de producción, la metodología de costos, los supuestos y restricciones, y la solución para el cálculo de la mejor decisión.

En el capítulo 2, se presenta la etapa de análisis mencionando la elección de la metodología de desarrollo, así como los requisitos funcionales y no funcionales, los diagramas de casos de uso por cada uno de los cuatro módulos de Maestros, Seguridad, Costos, Decisiones y Reportes. Después, se presenta el análisis de la solución en sí, mostrado mediante su viabilidad, análisis técnico económico, así como sus suposiciones, dependencias y restricciones.

En el capítulo 3 se presentan el diseño del sistema, la elección de la arquitectura en 3 capas, los paquetes de diseño, la vista de despliegue, los diagramas de interacción de componentes entre las clases de la Interfaz de usuario, lógica del negocio y el acceso a datos; por último el modelo de datos del sistema.

En el último capítulo se presentan la construcción y pruebas, donde se presenta el IDE NetBeans como el utilizado para el desarrollo, el motor de base de datos Oracle XE, el flujo básico del sistema en pantallas, una muestra de la clases de equivalencia y los casos de prueba. Finalmente, se desarrollan las conclusiones, recomendaciones y trabajos futuros.





La presente tesis va dedicada a mis padres, por toda la educación que recibí en casa y por todos los esfuerzos que realizaron para mi formación profesional.



Agradecimientos especiales a todos
los que fueron mis maestros en
estos años de estudios.

Tabla de Contenido

Introducción.....	1
1. Generalidades.....	2
1.1. Definición del Problema	2
1.1.1. Objetivo General	6
1.1.2. Objetivos Específicos.....	6
1.1.3. Resultados Esperados	6
1.2. Marco Conceptual	7
1.3. Plan de Proyecto	11
1.3.1. Planificación	11
1.3.2. Cronograma de trabajo	13
1.4. Estado del arte	14
1.4.1. Caso Acer América	14
1.4.2. Caso Coca Cola	15
1.4.3. Caso de una empresa de fabricación de bombas de agua	15
1.4.4. Caso de una empresa de fabricación de transformadores.....	16
1.5. Descripción y sustentación de la solución	16
1.5.1. Solución del plan de producción	18
1.5.2. Solución de los costos	19
1.5.2.1. Costos directos.....	19
1.5.2.2. Mano de obra y maquinaria	20
1.5.2.3. Materia prima	20
1.5.2.4. Otros costos directos	21
1.5.2.5. Costos indirectos.....	22
1.5.3. Solución de la aplicación de logística inversa.....	24
1.5.4. Algoritmo del cálculo de la mejor decisión	25
1.5.5. Ventajas de uso del sistema	27
2. Análisis.....	29
2.1. Metodología aplicada para el desarrollo de la solución	29
2.2. Identificación de requerimientos	34
2.2.1. Requisitos funcionales	34
2.2.2. Requisitos no funcionales	37
2.3. Modelo de casos de uso	38
2.3.1. Casos de uso por paquete	38
2.3.2. Catálogo de actores	39
2.3.3. Módulo de maestros.....	40
2.3.4. Módulo de costos y decisiones	41
2.3.5. Módulo de reportes	42
2.3.6. Módulo de seguridad.....	43
2.4. Análisis de la solución	44
2.4.1. Viabilidad del sistema	44
2.4.2. Análisis técnico y económico	46
2.4.2.1. Suposiciones y dependencias.....	46
2.4.2.2. Costos	47
2.4.2.3. Restricciones.....	48
2.5. Diagrama de clases	48
3. Diseño	51
3.1. Arquitectura de la solución.....	51
3.1.1. Descripción de la vista lógica.....	52
3.1.2. Paquetes de diseño	54
3.1.2.1. Paquetes de diseño del sistema cliente / servidor	54
3.1.2.2. Paquetes de diseño de los reportes Web	55
3.1.3. Vista de despliegue	56
3.1.4. Terminal	56
3.1.5. Servidor	56
3.2. Diagramas de Diseño.....	56
3.2.1. Interacción de Clases GUI	56

3.2.2.	Interacción de Clases Gestoras y de Acceso a Datos.....	56
3.2.3.	Modelo de Datos	57
3.3.	Conclusión de la arquitectura.....	61
3.4.	Diseño de interfaz gráfica	61
4.	Construcción y pruebas	64
4.1.	Construcción	64
4.1.1.	Tecnologías de construcción	64
4.1.2.	JAR, paquetes y librerías	66
4.1.3.	WAR, paquetes y librerías del módulo Web de reportes.....	70
4.1.4.	Flujo principal del sistema.....	72
4.1.5.	Reportes estadísticos de la aplicación de logística inversa.....	83
4.1.6.	Procedimientos almacenados.....	86
4.2.	Pruebas.....	88
4.2.1.	Clases de equivalencia	89
4.2.2.	Catálogo de pruebas.....	91
5.	Observaciones, conclusiones y recomendaciones.....	95
5.1.	Observaciones	95
5.2.	Conclusiones.....	96
5.3.	Recomendaciones y trabajos futuros.....	96
	Bibliografía	99



Índice de Figuras

Figura 1.1: Etapas de la logística inversa.....	3
Figura 1.2: Lazo de la cadena de producción y su relación con los retornos.....	4
Figura 1.3: Razones y decisiones de devoluciones.....	5
Figura 1.4: Orientación de la logística directa.....	7
Figura 1.5: Orientación de la logística inversa.....	8
Figura 1.6: Integración de ambas logísticas.....	9
Figura 1.7: Flujo de posibles devoluciones.....	10
Figura 1.8: Pirámide de nivel de reutilización.....	11
Figura 1.9: Estructura desagregada de trabajo.....	12
Figura 1.10: Cronograma general de trabajo.....	13
Figura 1.11: Diagrama de devoluciones Net Return.....	14
Figura 1.12: Bombas centrífugas.....	15
Figura 1.13: Gráfico del plan de producción.....	18
Figura 1.14: Costos directos.....	19
Figura 1.15: Almacén y lotes de materia prima.....	20
Figura 1.16: Plan de producción la falla producida en una actividad.....	24
Figura 1.17: Diagrama AHP.....	25
Figura 2.1: Fases de RUP.....	30
Figura 2.2: Diagrama de procesos del PMBOK.....	33
Figura 2.3: Paquetes del sistema.....	38
Figura 2.4: Diagrama de actores del sistema.....	39
Figura 2.5: Casos de uso del módulo maestros.....	40
Figura 2.6: Casos de uso del módulo costos y decisiones.....	41
Figura 2.7: Casos de uso del módulo de reportes.....	42
Figura 2.8: Casos de uso del módulo de seguridad.....	43
Figura 2.9: Diagrama de clases de análisis.....	50
Figura 3.1: Vista lógica del sistema.....	53
Figura 3.2: Modelo vista controlador.....	54
Figura 3.3: Vista de capas.....	55
Figura 3.4: Interacción modelo vista controlador.....	55
Figura 3.5: Vista de despliegue.....	56
Figura 3.6: Diagrama de clases de diseño, interacción GUI.....	58
Figura 3.7: Diagrama de clases de diseño, interacción BL-DA.....	59
Figura 3.8: Modelo de datos del sistema.....	60
Figura 3.9: Prototipo de ventana 1.....	63
Figura 3.10: Prototipo de ventana 2.....	63
Figura 4.1: Entorno de desarrollo NetBeans 6.1.....	65
Figura 4.2: Administrador del motor de base de datos Oracle XE 10.....	66
Figura 4.3: JAR del sistema y sus paquetes.....	69
Figura 4.4: WAR del módulo de reportes y sus paquetes.....	71
Figura 4.5: Ventana de validación de usuario.....	72
Figura 4.6: Prototipo de la ventana la ventana principal e ingreso a la función.....	72
Figura 4.7: Prototipo de la ventana de búsqueda de Producto en el Plan de Prod.....	73
Figura 4.8: Prototipo del resultado de búsqueda de Producto en el Plan de Prod.....	73
Figura 4.9: Prototipo del mantenimiento de plan de Producción.....	74
Figura 4.10: Prototipo del mantenimiento de actividades.....	74
Figura 4.11: Prototipo del mantenimiento de fases.....	75
Figura 4.12: Prototipo del registro satisfactorio de un plan de producción.....	75
Figura 4.13: Prototipo del acceso a la asignación de costos de un plan.....	76
Figura 4.14: Prototipo que muestra la asignación de costos de costos directos.....	76
Figura 4.15: Prototipo que muestra la asignación del valor monetario a los costos.....	77
Figura 4.16: Prototipo que muestra la asignación del valor monetario a los costos.....	77
Figura 4.17: Prototipo que muestra la asignación de productos y cantidades.....	78
Figura 4.18: Prototipo de asignación del costo indirecto e inductor.....	78
Figura 4.19: Prototipo del registro satisfactorio de una orden de producción.....	79
Figura 4.20: Prototipo que muestra los planes de producción en proceso.....	79

Figura 4.21: Prototipo que muestra los productos de la orden de producción.....	80
Figura 4.22: Prototipo que muestra el registro de una falla en una actividad.....	80
Figura 4.23: Prototipo que muestra el registro de una decisión ante una falla.....	81
Figura 4.24: Prototipo que muestra la actividades de las fallas.....	81
Figura 4.25: Prototipo que muestra el registro de la cantidad y precio de MP.....	82
Figura 4.26: Prototipo que muestra los lotes generados producto de la recuperación.....	82
Figura 4.27: Prototipo de la generación del reporte estadístico.....	83
Figura 4.28: Prototipo del reporte de frecuencia de decisiones tomadas.....	84
Figura 4.29: Prototipo de la frecuencia de las actividades falladas.....	84
Figura 4.30: Prototipo del reporte de materias primas recuperadas.....	85
Figura 4.31: Resultado de la búsqueda de órdenes vía Web.....	85
Figura 4.32: Pantalla de la generación de reportes estadísticos vía Web.....	86
Figura 4.33: Flujo de las pruebas de caja negra.....	88
Figura 4.34: Flujo de las pruebas de caja blanca.....	88
Figura 4.35: Ejemplo de clases equivalentes.....	89



Índice de Tablas

Tabla 1.1: Matriz para el cálculo del proceso de jerarquía analítica.....	26
Tabla 2.1: Lista de requisitos funcionales.....	34
Tabla 2.2: Lista de requisitos no funcionales.....	37
Tabla 2.3: Tabla descriptiva del tipo de requisito.....	37
Tabla 2.4: Tabla descriptiva de la prioridad del requisito.....	38
Tabla 2.5: Tabla descriptiva del catálogo de actores.....	39
Tabla 2.6: Tabla descriptiva del caso de uso mantener producto.....	40
Tabla 2.7: Tabla descriptiva del caso de uso mantener plan de producción.....	40
Tabla 2.8: Tabla descriptiva del caso de uso registrar falla.....	41
Tabla 2.9: Tabla descriptiva del caso de uso asignar decisión a producto.....	41
Tabla 2.10: Tabla descriptiva del caso de uso emitir reporte estadísticos.....	42
Tabla 2.11: Tabla descriptiva del caso de uso emitir reporte plan de producción.....	42
Tabla 2.12: Tabla descriptiva del caso de uso validar usuario.....	43
Tabla 2.13: Tabla descriptiva del caso de uso mantener usuario.....	43
Tabla 2.14: Cuadro comparativo de motores de base de datos.....	45
Tabla 2.15: Cuadro comparativo de lenguajes de programación.....	46
Tabla 2.16: Cuadro de costos del proyecto.....	47
Tabla 2.17: Clase BEParametroCosto.....	48
Tabla 2.18: Clase BEActividad.....	48
Tabla 2.19: Clase BEActividad.....	49
Tabla 2.20: Clase BEFalla.....	49
Tabla 2.21: Clase BEDecisión.....	49
Tabla 2.22: Clase BEProducto.....	49
Tabla 2.23: Clase BEOrdenProduccion.....	49
Tabla 2.24: Clase BEPlanProduccion.....	49
Tabla 3.1: Cuadro de ejemplo de botones e iconos.....	62
Tabla 4.1: Cuadro de tecnologías.....	64
Tabla 4.2: Cuadro de clases de equivalencia 1.....	89
Tabla 4.3: Cuadro de clases de equivalencia 2.....	90
Tabla 4.4: Cuadro de clases de equivalencia 3.....	90
Tabla 4.5: Cuadro de plan de prueba 1.....	91
Tabla 4.6: Cuadro de plan de prueba 2.....	93

Introducción

La logística inversa nace como una necesidad de reutilización de materiales colocados en nuestro entorno por algún proceso productivo. Con el discurso actual de conservación ambiental y reducción del calentamiento global, este concepto y su aplicación vienen tomando cada vez mayor fuerza en especial, se incrementa la preocupación en los países desarrollados, responsables de la mayor producción mundial, así como en países en vía de desarrollo que buscan una mejor calidad de vida.

Existe un alto grado de responsabilidad en las empresas productoras, de gran y mediana escala, para la reutilización de aquellos productos que fueron puestos en el mercado tales como las mermas o fallas de producción, incorporándolas como materias primas de sus procesos productivos. Asimismo, estas empresas se encuentran ante una oportunidad de adquisición y mejora en la productividad, a un menor costo que recalca en la eficiencia de la empresa y en la imagen institucional que proyecta a la sociedad.

A lo largo de los capítulos se irán mostrando las soluciones planteadas a la aplicación de logística inversa en la etapa productiva, apoyado por el sistema de información que permite su gestión desde detalles muy específicos como el registro de planes de producción hasta más generales como los reportes gerenciales.

Este proyecto presenta las tres etapas de la aplicación de logística inversa y el análisis, diseño e implementación del sistema de información. Cabe resaltar que, en la actualidad, es casi nula la presencia en el mercado de un software o herramientas informáticas comerciales y genéricas de logística inversa, motivo por el cual su desarrollo se da en forma específica a una organización en particular y a sus propios procesos. El trabajo busca, de alguna manera, el desarrollo de un sistema de información más genérico para empresas de producción de mediano y gran tamaño, en cuyos procesos no intervienen materias primas perecibles.

Finalmente, es importante mencionar que el proyecto abarca como solución conceptos de costos, producción, métodos de costeo, un proceso para la toma de decisiones extraído de la teoría de investigación de operaciones y la seguridad que todo sistema de información precisa. Así también, se menciona el desarrollo mediante el lenguaje de modelado UML y los Fundamentos de la Dirección de Proyectos (Guía del PMBOK) para la gestión del proyecto.

1. Generalidades

Este primer capítulo se enfoca en la explicación del problema a resolver, el marco conceptual, el objetivo principal y los objetivos específicos del proyecto. Asimismo, se presenta el plan de proyecto, los productos y formas actuales de la aplicación de logística inversa y se concluye con la solución del problema.

1.1. Definición del Problema

Hoy en día, el mundo presenta un acelerado crecimiento poblacional, y con ello, el aumento del consumo a gran escala, lo cual obliga a las empresas el incremento de su capacidad de producción, y como consecuencia de esto, la extracción de más cantidades de materias primas poniendo en peligro la conservación del ambiente.

Paralelamente, los productos terminados y puestos a la venta al consumidor final poseen un tiempo de vida limitado, que al finalizar, pasan a formar parte de la larga lista de desechos que se encuentran dispersos en todo el planeta. Éstos pueden ser del tipo electrónico (computadoras, televisores, radios, etc.) [24], mecánico (aviones, automóviles, barcos, etc.), consumo masivo (bolsas plásticas, cajas, empaques, etc.), manufactura y muchos más; lo cual lleva a los gobiernos, principalmente los países desarrollados, a adoptar normas y leyes en el control de

estos desechos delegando gran parte de la responsabilidad hacia las empresas de producción principalmente [27].

Ante esta problemática surge la logística inversa, para la cual se tienen las siguientes definiciones, en base a la de logística directa:

- **“Logística:** proceso de proyectar, implementar y controlar un flujo de materia prima, inventario en proceso, productos terminados e información relacionada desde el punto de origen hasta el punto de consumo de una forma eficiente y lo más económica posible con el propósito de cumplir con los requerimientos del cliente final.
- **Logística Inversa:** es el proceso de proyectar, implementar y controlar un flujo de materia prima, inventario en proceso, productos terminados e información relacionada desde el punto de consumo hasta el punto de origen de una forma eficiente y lo más económica posible con el propósito de recuperar su valor ó el de la propia devolución” [7].

El concepto de logística inversa se remonta a la década de los 70's, básicamente a una función de reciclaje. En la actualidad presenta tres partes, tal como se aprecia en la figura 1.1:



Figura 1.1: Etapas de la logística inversa [37]



Figura 1.2: Lazo de la cadena de producción y su relación con los retornos [22]

- **Logística inversa de devoluciones o retornos:** Se da por las devoluciones de productos terminados realizadas por los usuarios o compradores finales.
- **Logística inversa en la etapa de producción:** Se da desde el ingreso de materias primas hasta el producto terminado, es decir se aplica logística inversa cuando se produce alguna falla en el proceso de producción.
- **Logística inversa de reciclaje y recuperación:** Se da cuando la organización realiza procesos de reciclaje en afán de recolectar o adquirir productos en desuso, para clasificarlos y seleccionar los más convenientes al proceso de producción. [12]

Tal como se observa en la figura 1.2, las tres formas de ingreso de materias primas a la cadena de producción se dan por retornos de manufactura, retornos comerciales y reuso de productos en desecho.

Un problema identificado en la investigación realizada es que, en la actualidad, no existe en el mercado un software comercial basado en logística inversa, para lo cual la solución inmediata es la realización de uno a medida o la modificación de un sistema logístico existente. Sin este tipo de herramienta, las empresas no pueden manejar un adecuado control de fallas ni del costo de éstas, tampoco del registro de las decisiones tomadas al respecto, y por ende, se pierde el valor de los verdaderos costos incurridos en la producción.

Además, resulta vital para cualquier empresa de producción la aplicación de una política de logística inversa, tanto para la mejora de atención al cliente en los procesos de devoluciones, así como en la etapa de producción con respecto a las

fallas que se puedan presentar en cualquiera de las actividades que componen dicho proceso, teniendo en consideración un conjunto de decisiones para la reutilización del producto defectuoso, lo cual lleva a una reducción considerable de los costos y por último, la imagen de responsabilidad ambiental que la empresa proyecta ante la sociedad.

Nuestro país no es ajeno a esta realidad, ya que la competencia a enfrentar pasa a ser de un contexto local a uno global, para lo cual la reducción de costos juega un papel importante, así como la adopción de medidas por parte del gobierno central en favor de la conservación del ambiente¹, tal como se viene aplicando en el parlamento europeo [26].

A continuación, se muestra en la figura 1.3 las razones de las devoluciones y las decisiones que se suelen tomar:

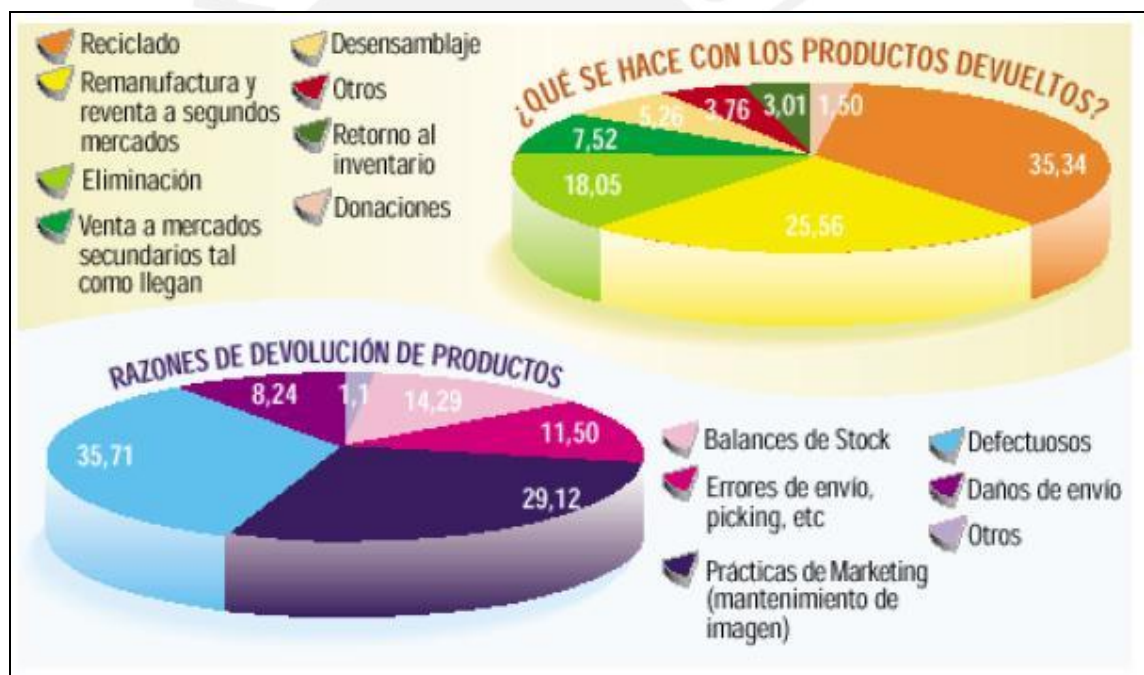


Figura 1.3: Razones y decisiones de devoluciones [9]

Como se ve en la figura 1.3, el mayor porcentaje de devoluciones se da en la falla de productos en proceso y/o productos terminados (defectuosos) y es ahí a donde apunta el presente proyecto, a la elaboración de una herramienta informática que permita el manejo y gestión en las decisiones a adoptar con respecto a las fallas que se puedan presentar en las actividades que componen el proceso de

¹ En el Perú existe la ley N° 27314, Ley general de Residuos Sólidos con Decreto Supremo N° 057-2004-PCM, donde se establece en la sección de Manejo de Residuos Sólidos, que los productos abandonados, adulterados o caducos son considerados residuos y por ende, los fabricantes y distribuidores emplearán mecanismos de recuperación.

producción, así como un adecuado control de costos en cada una de estas etapas y en la decisiones tomadas.

Por las razones expuestas en párrafos anteriores, se pretende brindar a las empresas el ingreso y configuración del proceso de producción, el cálculo automatizado del costo en cada actividad del proceso (desde ingreso de materias primas hasta el producto terminado). Asimismo, la configuración de las posibles tomas de decisiones en cada etapa, para finalmente proveer a la gerencia de reportes del manejo de la logística inversa en la organización, tanto con indicadores monetarios y de unidades.

1.1.1. Objetivo General

Realizar el análisis, diseño e implementación de un sistema de apoyo a la gestión de la información en empresas de producción basado en logística inversa de la etapa de producción, permitiendo un adecuado control de costos y toma de decisiones.

1.1.2. Objetivos Específicos

- Realizar el análisis, diseño e implementación de un módulo de producción, módulo de toma de decisiones y costos.
- Modelar una base de datos que permita almacenar la información del sistema.
- Diseñar la arquitectura para el soporte del sistema.
- Implementar un algoritmo para hallar la mejor decisión a tomar ante el registro de fallas en el proceso productivo.
- Realizar las pruebas del proceso y la simulación del sistema a implementar.
- Elaborar reportes que sirvan de apoyo al jefe de producción y a la gerencia.

1.1.3. Resultados Esperados

- Módulo de producción, toma de decisiones y costos.
- Modelo relacional de la base de datos del sistema.

- Arquitectura orientada a componentes que permitan las transacciones en un ambiente cliente/servidor.
- Algoritmo implementado del Proceso de Jerarquía Analítica para el apoyo de toma de decisiones.
- Catálogo de pruebas del sistema de información de logística inversa.
- Reportes estadísticos de la aplicación de logística inversa de producción.

1.2. Marco Conceptual

“Desde que el hombre ha necesitado mover cosas de un lugar a otro, casi desde el inicio de la Civilización, siempre se ha topado con un problema: ¿cómo hacerlo de la mejor manera posible? Este problema aun sigue vigente y solamente durante la segunda mitad del siglo veinte, es decir sobre un rango de tiempo de 50 años, los gerentes se han preocupado de hacerlo de una manera confiable y a bajo costo”.[10]

El objetivo de la logística directa radica en una adecuada planificación de movimiento o traslado de productos terminados, mercaderías, insumos, materias primas, etc. desde un determinado punto (inicial) hacia otro (final), buscando siempre el cumplimiento de tiempos establecidos con un controlado y aceptable costo.

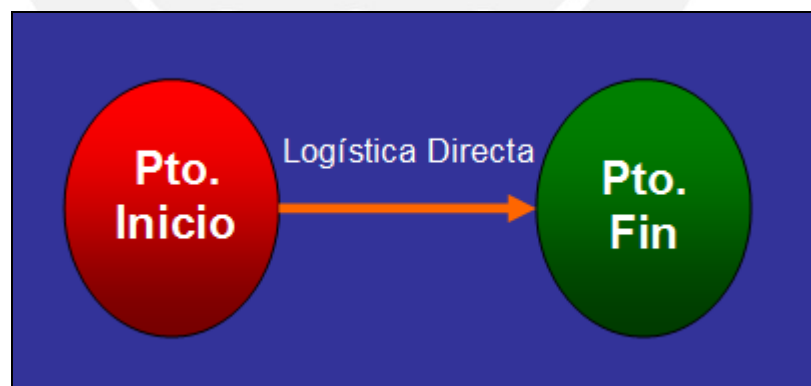


Figura 1.4: Orientación de la logística directa [propia autoría]

Sin embargo, las organizaciones y empresas siempre han visto a la logística en una sola dirección (hacia el mayorista, minorista, consumidor final, planta de producción, almacén MP, almacén PT, etc. Ver figura 1.4), presentando poco interés al flujo inverso que siempre se da por uno u otro motivo (definición del problema). Es ahí donde se puede señalar que se encuentra un “costo oculto” [21], ya que las empresas suelen creer que su responsabilidad acaba una vez que los productos

son entregados, poniendo todo su interés y esfuerzo en una sola dirección, dejando de lado una adecuada planificación de devoluciones en las diferentes etapas (compras, producción, ventas, etc.).

Este enfoque unidireccional prácticamente “ciega” a la empresa y aumenta los costos del flujo inverso, debido a que la empresa reacciona lentamente ante esta situación y mantiene estos bienes (materias primas, mercancía, productos terminados, etc.) por largos períodos [25], acarreando un costo innecesario.

Ante esta problemática surge la logística inversa, que se integra al sistema de logística directa, mejorando la eficiencia en la utilización de recursos de la organización que lo aplica (Ver figura 1.5).

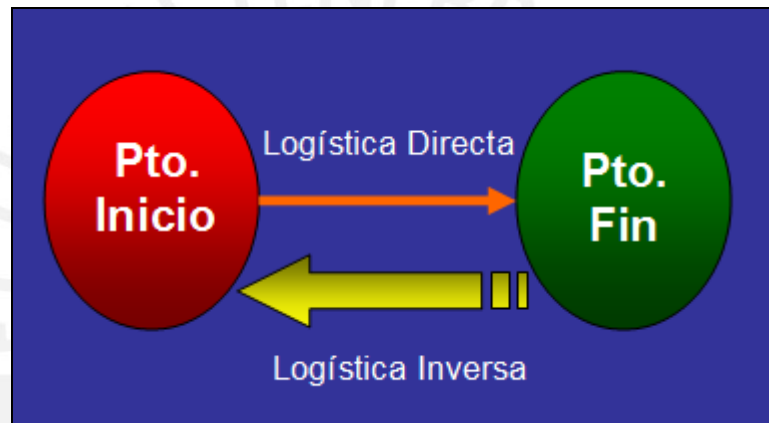


Figura 1.5: Orientación de la logística inversa [propia autoría]

En las últimas décadas, el concepto de logística inversa ha venido cambiando, específicamente se profundiza más en su definición, así como en su alcance. A continuación, se muestran algunas definiciones:

- “... El término comúnmente usado para referirse al rol de la logística en el reciclaje, disposición de desperdicios y el manejo de materiales peligrosos; una perspectiva más amplia incluye todo lo relacionado con las actividades logísticas llevadas en la reducción de entrada, reciclaje, sustitución y re-uso de materiales y su disposición final”. [11]
- “... El proceso de planear, implementar y controlar eficientemente y el costo eficaz de los flujos de materias primas, inventario en proceso, bienes terminados e información relacionada desde el punto de consumo al punto de origen con el propósito de recuperar el valor primario o disponer adecuadamente de ellos”. [8]

- “... El proceso de planeación, implementación y control de flujo de materias primas, inventario en proceso y bienes terminados, desde un punto de uso, manufactura o distribución a un punto de recuperación o disposición adecuada”. [13]

La primera definición data del año 1992, la segunda de 1998 y la tercera también de 1998 dada por el Grupo Europeo de Logística Inversa [23], siendo esta última la aceptada en la actualidad debido a que refleja una definición de los bienes en proceso, así como la opción de recuperación o toma de otra decisión.

A continuación se muestra en la figura 1.6 la integración de la logística directa y logística inversa:

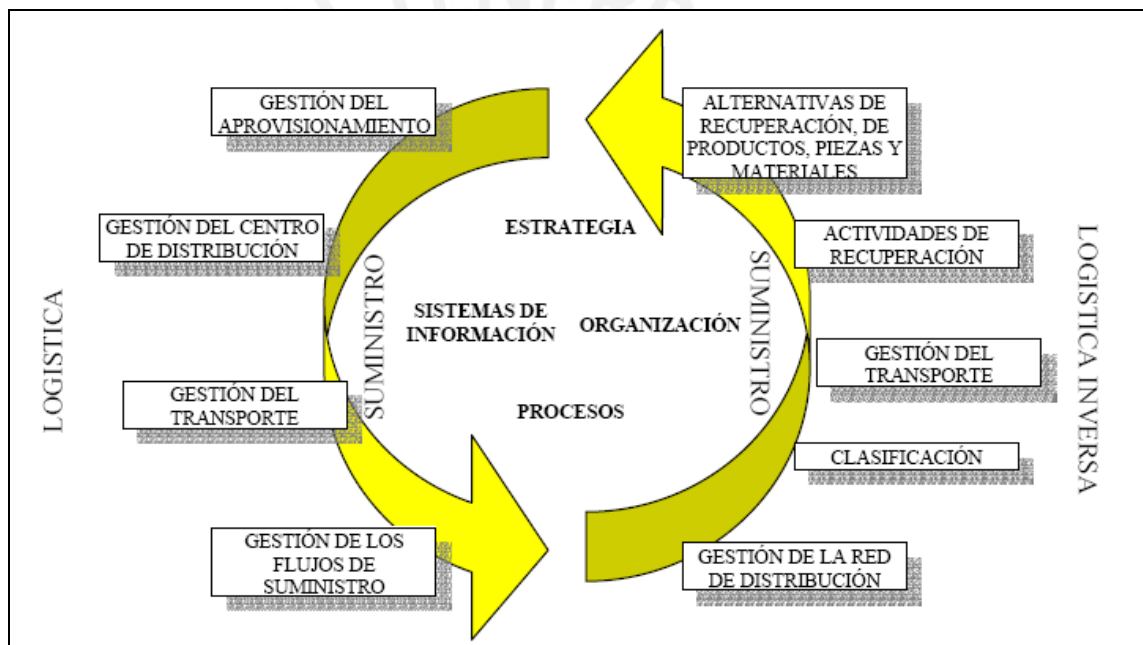


Figura 1.6: Integración de ambas logísticas [6]

Como se observa en la figura 1.6, logística inversa abarca actividades diferentes a la directa, pero ambas necesitan de procesos, estrategias, organización y sistemas de información.

Asimismo, las empresas deben preguntarse sobre el origen las fuentes de los retornos basado en la siguiente pregunta: *¿de dónde provienen los flujos inversos en una cadena de suministro?* [27]

Estos flujos no son tan visibles, sin embargo hay autores que han escrito al respecto y graficado los flujos como se aprecia en la figura 1.7.

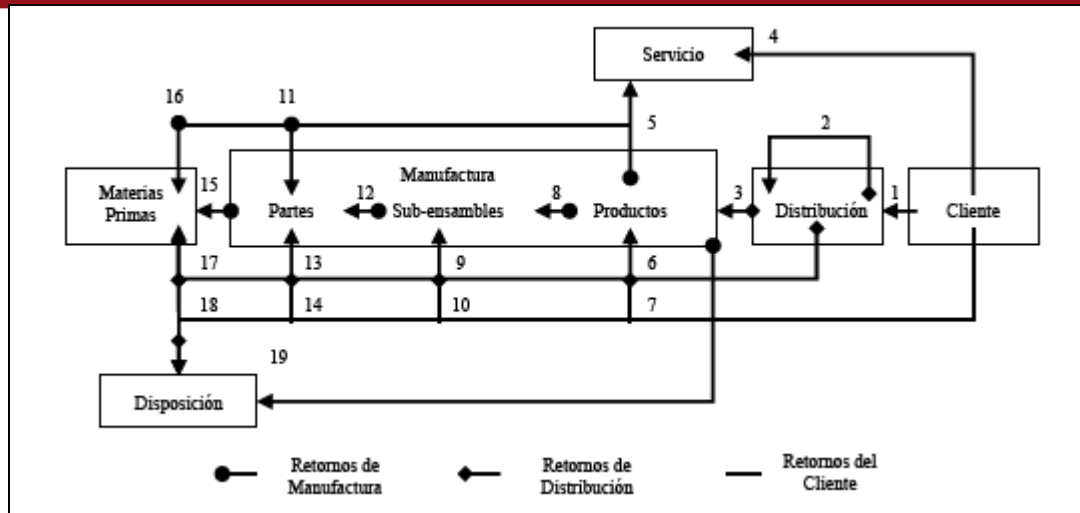


Figura 1.7: Flujo de posibles devoluciones [14]

Leyenda del gráfico:

- 1) Retornos producto equivocado (re-venta, re-uso)
- 2) Ajustes comerciales y de almacenamiento (re-distribución)
- 3) Recuperaciones (re-procesos)
- 4) Garantía, servicio (reparación)
- 5) Productos defectuosos (reparación)
- 6) Retornos comerciales (restauración)
- 7) Garantía de cliente (restauración)
- 8) Producto defectuoso (re-manufactura)
- 9) Retornos comerciales
- 10) Fin de vida (re-manufactura)
- 11) Productos defectuosos (recuperación)
- 12) Idem
- 13) Retornos comerciales (recuperación)
- 14) Fin de vida (recuperación)
- 15) Exceso de materias primas (re-uso, re-venta)
- 16) Productos defectuosos, sobras de producción (reciclaje)
- 17) Retornos comerciales (reciclaje)
- 18) Fin de vida (reciclaje)
- 19) Todos los tipos de flujo inverso (destrucción por incineración, disposición final en relleno sanitario)

Estos flujos fueron identificados por Marisa de Brito y Rommert Dekkert [14], donde se observa una lista de posibilidades de retorno y su posible aprovechamiento entre paréntesis. Además, estos autores señalan que el nivel del aprovechamiento está

sujeto a una pirámide (ver figura 1.8), donde los niveles más bajos muestran un aprovechamiento menor comparado con los niveles altos.

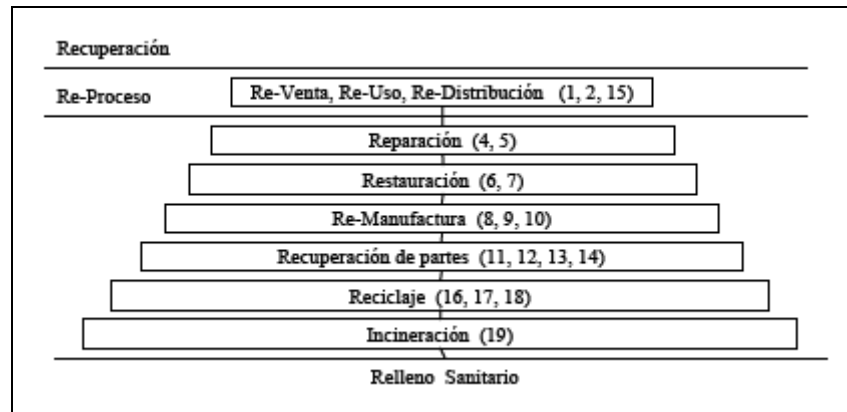


Figura 1.8: Pirámide de nivel de reutilización [14]

1.3. Plan de Proyecto

La presente tesis es un proyecto y como tal, su desarrollo está ubicado en un espacio finito de tiempo. Es decir, posee un inicio y un final que tendrá como resultado un conjunto de entregables como documentación, diseños y el producto software.

Por lo tanto, para la gestión del proyecto y sus procesos de inicio, planificación, ejecución, control y cierre se aplicará la “Guía de los Fundamentos de la Dirección de Proyectos”, es decir el PMBOK² en su tercera versión, propuesto por la organización PMI³.

Cabe mencionar, que el proyecto será dividido en un conjunto de fases (ver figura 1.9) que a su vez se divide en un conjunto de “componentes” que presentan en la mayoría de los casos los entregables [19].

1.3.1. Planificación

“El plan de gestión del alcance del proyecto proporciona orientación sobre cómo el equipo de dirección del proyecto definirá, documentará, verificará, gestionará, y controlará el alcance del proyecto” [36].

En la planificación se especifica el trabajo que debe realizarse para entregar el producto final con las funciones y características especificadas.

² Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK, “Project Management Body of Knowledge”).

³ “PMI”, el logotipo de PMI, “PMP”, el logotipo de “PMP”, “PMBOK” son marcas registradas de Project Management Institute, Inc.

“La EDT es una descomposición jerárquica, orientada al producto entregable, del trabajo que será ejecutado por el equipo del proyecto, para lograr los objetivos del proyecto y crear los productos entregables requeridos. La EDT organiza y define el alcance total del proyecto, subdivide el trabajo del proyecto en porciones de trabajo más pequeñas y fáciles de manejar” [5].

La EDT del proyecto refleja las etapas que habrá que realizar y completar para lo obtención del sistema de información, es decir cada uno de los pasos a seguir en forma general para alcanzar con éxito cada objetivo específico y el objetivo general del proyecto.

A continuación se presenta en la figura 1.9 las etapas del proyecto:



Figura 1.9: Estructura desagregada de trabajo

1.3.2. Cronograma de trabajo

“El desarrollo de cronograma del proyecto, un proceso iterativo, determina las fechas de inicio y finalización planificadas para las actividades del proyecto. El desarrollo del cronograma exige que se revisen y corrijan las estimaciones de duración y las estimaciones de los recursos para crear un cronograma del proyecto aprobado que pueda servir como línea base con respecto a la cual poder medir el avance” [5].

A diferencia de la EDT, el cronograma de trabajo indica cuando y por quienes debe de realizarse cada entregable.

A continuación se presenta en la figura 1.10 el cronograma general de trabajo del proyecto (el cronograma detallado se muestra en los anexos, ver sección 11 Cronograma de Trabajo).











		Nombre de tarea	Duración	Comienzo	Fin
1		 Plan de Proyecto de Fin de Carrera	282 días	lun 17/03/08	mar 14/04/09
2		 Inicio del Proyecto	11 días	lun 17/03/08	lun 31/03/08
6		 Entregable1	4 días	mar 01/04/08	vie 04/04/08
11		 Entregable2	13 días	lun 21/04/08	mié 07/05/08
18		 Entregable3	8 días	lun 12/05/08	mié 21/05/08
22		 Planificación del Proyecto	2 días	lun 21/07/08	mar 22/07/08
24		 Análisis	19 días	lun 04/08/08	jue 28/08/08
31		 Diseño	27 días	lun 01/09/08	mar 07/10/08
39		 Construcción	112 días	lun 13/10/08	mar 17/03/09
46		 Pruebas	90 días	mié 26/11/08	mar 31/03/09
50		Cierre	26 días	mar 10/03/09	mar 14/04/09

Figura 1.10: Cronograma general de trabajo

El cronograma de trabajo fue elaborado tomando como base la estructura desagregada del proyecto y el alcance del proyecto.

Asimismo, en el cronograma de trabajo detallado que se encuentra en los anexos se puede observar la lista de actividades que lo componen, así como el tiempo de ejecución y el recurso humano que lo llevará a cabo. Para el proyecto en particular, solo se considera a un recurso humano, ya que es un proyecto de desarrollo individual.

1.4. Estado del arte

El concepto actual de logística inversa desde un enfoque general, tomando en cuenta devoluciones, fallas en producción y reciclaje viene siendo aplicado por diferentes empresas alrededor del mundo. A continuación se analizarán diferentes casos:

1.4.1. Caso Acer América

Acer es considerado uno de los mayores fabricantes de computadoras en el mundo, sin embargo experimentó problemas ligados a los cambios fuertes y rápidos que se producen en esta industria, debido a los ciclos cortos de vida (90 a 120 días) que poseen las computadoras por aparición de nuevas tecnologías.

Acer América tenía serios problemas con los retornos de los productos, que significan 15 días y una seria depreciación. Además, no tenía medios para determinar cuando llegaba un envío de retorno, ni del tipo de paquete que tenía, por lo tanto solo era contabilizado cuando se abría.

La solución a este problema fue el *outsourcing*, el cual fue designado a NetReturn de Fedex[15] que posee su propio sistema de logística inversa, basado en una plataforma de e-Commerce, donde las devoluciones son registras y asignadas a una orden, la cual es recibida por Acer América, permitiendo una adecuada planificación para la atención de la mercadería, disponibilidad de almacén, reparación, remanufactura o reciclaje (ver figura 1.11).

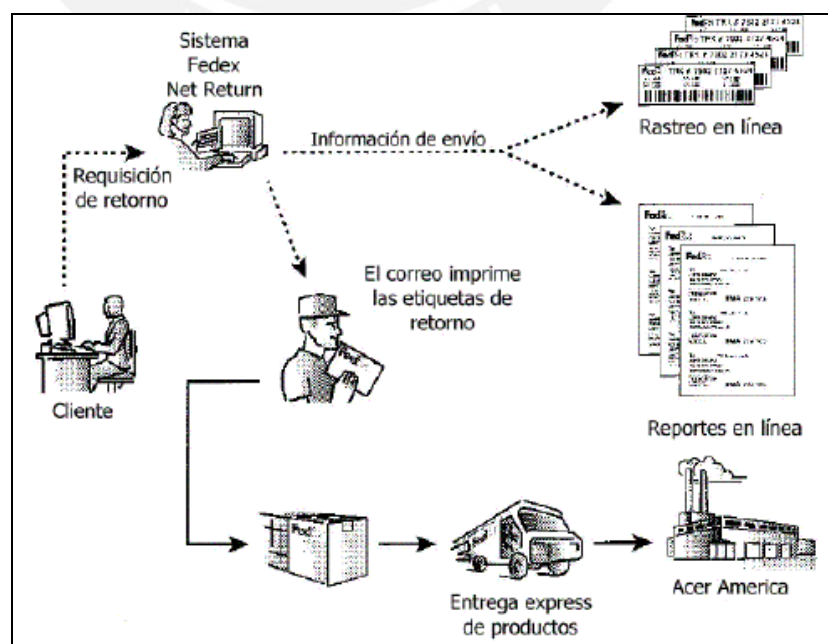


Figura 1.11: Diagrama de devoluciones Net Return [15]

1.4.2. Caso Coca Cola

FEMSA Coca Cola, localizada en la Ciudad de México, es el cuarto productor más grande de plásticos a nivel mundial. FEMSA Coca Cola implementó un software especializado en logística inversa: *Numetrix/3* [6].

Es una herramienta táctica de planeación, que permite controlar la fabricación de botellas en respuesta a los picos de demandas producidas por promociones, logrando además el ahorro en almacenamiento, debido a que la disponibilidad de botellas pasó a ser conocida por la empresa.

1.4.3. Caso de una empresa de fabricación de bombas de agua

En el Perú existe una empresa líder en la fabricación de bombas centrífugas, helicoidales y piezas fundidas (ver figura 1.12). Dicha organización aplica logística inversa en su proceso de producción y sus compras de materias primas principalmente.

Las materias primas son compradas en condición de chatarra para después pasar por un proceso de fundición. Es en este proceso de reciclado donde se aplica uno de los puntos de logística inversa, ya que la empresa recurre a materiales existentes y usados, sin comprarlo a un proveedor que extrae constantemente de la naturaleza.

Por otro lado, en sus procesos de producción aplica actividades como el de fundición, limado, torneado, taladrado, etc. que generan una gran cantidad de merma acumulada, la cual es aprovechada nuevamente regresando a la etapa de fundición.



Figura 1.12: Bombas centrífugas [16]

Este proceso de logística inversa se repite para cada una de las piezas que forman parte del producto terminado. Por ende, logran un uso máximo de sus materiales y una gran reducción de costos.

Sin embargo, la empresa registra la cantidad de chatarra que adquirió y no hace una diferenciación de cuánto de esa cantidad fue usada por medio de logística inversa. El sistema de información de la empresa registra muy bien todos los parámetros referentes solo de la logística directa.

1.4.4. Caso de una empresa de fabricación de transformadores

Esta empresa peruana se dedica a la fabricación de transformadores industriales, empleando principalmente planchas de flejes magnéticos, acero y hierro.

El peso del transformador más pequeño que fabrica es de aproximadamente una tonelada. Sus procesos de producción incluyen actividades como los de corte, taladrado, limado, granallado, soldado, etc. Esta empresa no aplica logística inversa y toda la merma generada es desechada, lo cual lleva a un aumento en los costos (compra de más materia prima), así como el aprovechamiento económico de algunos empleados de la empresa con la venta de estas mermas.

Esta empresa posee un sistema de información de logística directa, mas no de la inversa ya que no lo aplica. Sin embargo, un marco legal más estricto en un futuro, así como ciertas certificaciones ambientales como el caso del ISO 14 000; cambiará la forma de trabajar de esta empresa. [21]

Finalmente, los dos primeros casos expuestos se refieren a la aplicación de logística inversa en el caso de devoluciones, los cuales son tratados adecuadamente debido a la existencia de soluciones informáticas. Por otro lado, los dos últimos casos reflejan la falta de un sistema de información de apoyo a la gestión de logística inversa, el cual es precisamente el objetivo de la presente tesis.

1.5. Descripción y sustentación de la solución

Según lo indicado en la sección anterior, logística inversa provee una solución en el máximo aprovechamiento de los insumos y materias primas adquiridas para el proceso de producción. Sin embargo, se observa que este mecanismo no es aplicado en las empresas del medio local y menos aún cuentan con una herramienta o sistema de información referente a este tema.

Por esta razón, el objetivo de la presente tesis es proporcionar una herramienta de apoyo a la gestión de logística inversa para aquellas empresas (de mediano y gran tamaño) que se encuentren aplicándolo sin un adecuado control y registros de toma de decisiones y cálculo automático de costos, así como para aquellas otras

empresas que se decidan a aplicarlo ya sea por una necesidad de mejora productiva, responsabilidad ambiental o aplicación de un marco legal.

Asimismo, el sistema a elaborar será novedoso ya que en el mercado existen soluciones informáticas para el punto de devoluciones y control de inventarios como los paquetes Numetrix/3 y NetRetun mencionados en la sección de estado del arte, mas no para el punto aplicado al proceso productivo. En realidad, se sabe que estos sistemas son implementados a medida.

El sistema a desarrollar permitirá el ingreso actual del proceso productivo para cada línea de producción, dividido en actividades y la vez permitirá la configuración de fases que agruparán a un subconjunto de las actividades ingresadas. Además, el sistema ofrecerá el ingreso de la actividad, por parte del usuario, donde ocurrió una falla y obtendrá como respuesta el costo calculado hasta ese momento así como un conjunto de decisiones posibles como solución a la falla, mostrándolos de acuerdo a un nivel de máximo aprovechamiento.

Adicionalmente, el sistema permitirá la emisión de reportes gerenciales donde se verá reflejado el nivel de fallas ocurridas en un cierto periodo, realizando comparaciones entre la aplicación de logística inversa y la no aplicación de esta, los cuales permitirán a la gerencia ver y/o comprobar los beneficios de la aplicación de la metodología (logística inversa) con el apoyo del sistema informático.

Otro punto importante a tomar en cuenta es el costo y el beneficio, siendo el primero representado por el precio que conlleve el análisis, diseño, implementación, implantación y mantenimiento del sistema informático que ayude a gestionar la aplicación de logística inversa y por otro lado, el beneficio se verá representado por toda la reducción de costos en la etapa productiva, aprovechamiento máximo de recursos y el reconocimiento del compromiso de protección ambiental que reflejará la empresa que aplique la metodología.

Finalmente, cabe destacar que el proceso productivo suele contener una gran o poca cantidad de actividades que la envuelven, así como un número finito de productos, los cuales multiplican la cantidad de actividades. Por esta razón, la aplicación de logística inversa de forma manual resultaría poco viable y tediosa. Para ello, la presente tesis presenta una herramienta de apoyo que permitirá la gestión de logística inversa de un modo más viable.

A continuación se explicarán las diferentes partes que componen la solución:

1.5.1. Solución del plan de producción

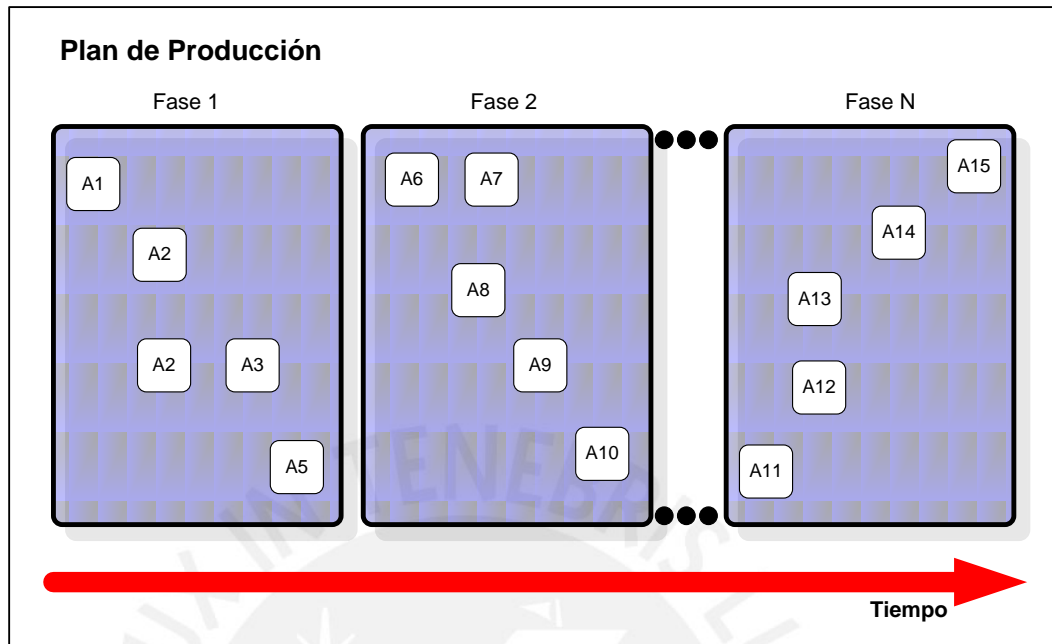


Figura 1.13: Gráfico del plan de producción [propia autoría]

Como se ve en la figura 1.13, un plan de producción está compuesto por un conjunto de fases.

Una fase a su vez agrupa a un conjunto de actividades, es decir a las tareas, labores o pasos previos que se tienen que cumplir para la obtención del producto final. Las fases permiten definir puntos que son importantes en la etapa productiva, es decir, establecer que al finalizar una fase se obtendrá un denominado producto intermedio. Los productos intermedios forman parte importante del proceso productivo, los cuales eventualmente pasarían a venta; por ello la importancia de identificarlos.

Cabe resaltar que un proceso productivo de complejidad baja probablemente no tenga productos intermedios, razón por la cual se definiría una única fase. Por otro lado, en el caso de productos cuya elaboración es de complejidad alta, por ejemplo la fabricación de un automóvil o un avión, las fases resultan vitales para abstraer el proceso productivo para su mejor comprensión.

1.5.2. Solución de los costos

“El costo o coste es el gasto económico que representa la fabricación de un producto o la prestación de un servicio. Al determinar el costo de producción, se puede establecer el precio de venta al público del bien en cuestión” [29].

1.5.2.1. Costos directos

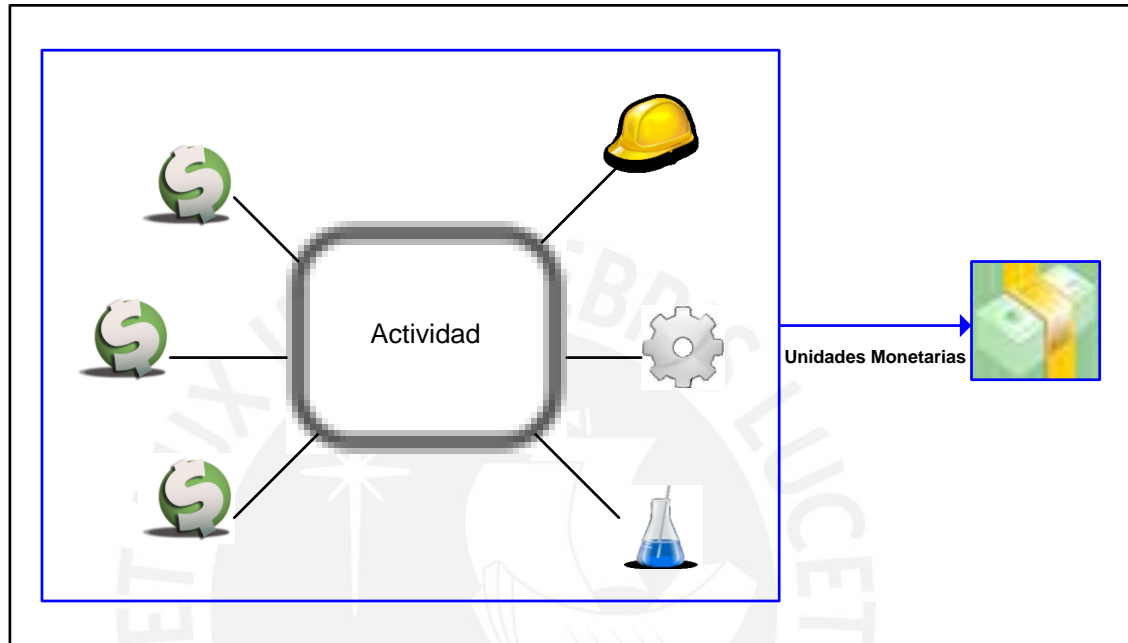


Figura 1.14: Costos directos [propia autoría]

Es necesario definir el concepto de costo directo, para lo cual se tiene lo siguiente: “Costos relacionados con un objeto del costo en particular que pueden rastrearse de manera económicamente factible (efectiva en cuanto a costos se refiere)” [2].

La definición presentada se refiere a aquellos costos que son identificables para un producto en particular y específico, por ejemplo: mano de obra directa, maquinaria, materiales directos, etc. Esta definición de costo directo es válida desde un enfoque gerencial y general. Sin embargo, para llevar a cabo la aplicación de logística inversa en la etapa productiva es necesario tener una asignación de costos más a detalle que la definición tradicional, es decir una vez identificados los costos directos de un producto, también se deben de identificar los costos directos de una actividad en particular como se muestra en la figura 1.14.

Se recomienda proceder del modo indicado, ya que las fallas ocurren en cualquier momento de la etapa productiva, con lo cual se necesita identificar los costos incurridos hasta ese punto de error.

1.5.2.2. Mano de obra y maquinaria

La asignación de los costos directos concernientes a la mano de obra y maquinaria directa se realizará en base una lista de precios de cada una de ellas. Es decir, se podrá abarcar el caso de que una mano de obra cueste 'A', otra 'B' y otra 'C' unidades monetarias y de la misma manera para la maquinaria.

La unidad de medida para la mano de obra será 'hora/hombre' y para la maquinaria 'hora/maquina'.

Por lo tanto, el costo de una maquinaria o mano de obra estará dado por el producto del precio y la cantidad a utilizar en cada actividad.

$$\text{Costo}_{\text{Maquinaria}_{\text{Actividad}_y}} \equiv \text{Precio}_{\text{Maquinaria}} \times \text{Cantidad}_{\text{Maquinaria}_{\text{Actividad}_y}}$$

$$\text{Costo}_{\text{ManoObra}_{\text{Actividad}_y}} \equiv \text{Precio}_{\text{ManoObra}} \times \text{Cantidad}_{\text{ManoObra}_{\text{Actividad}_y}}$$

1.5.2.3. Materia prima

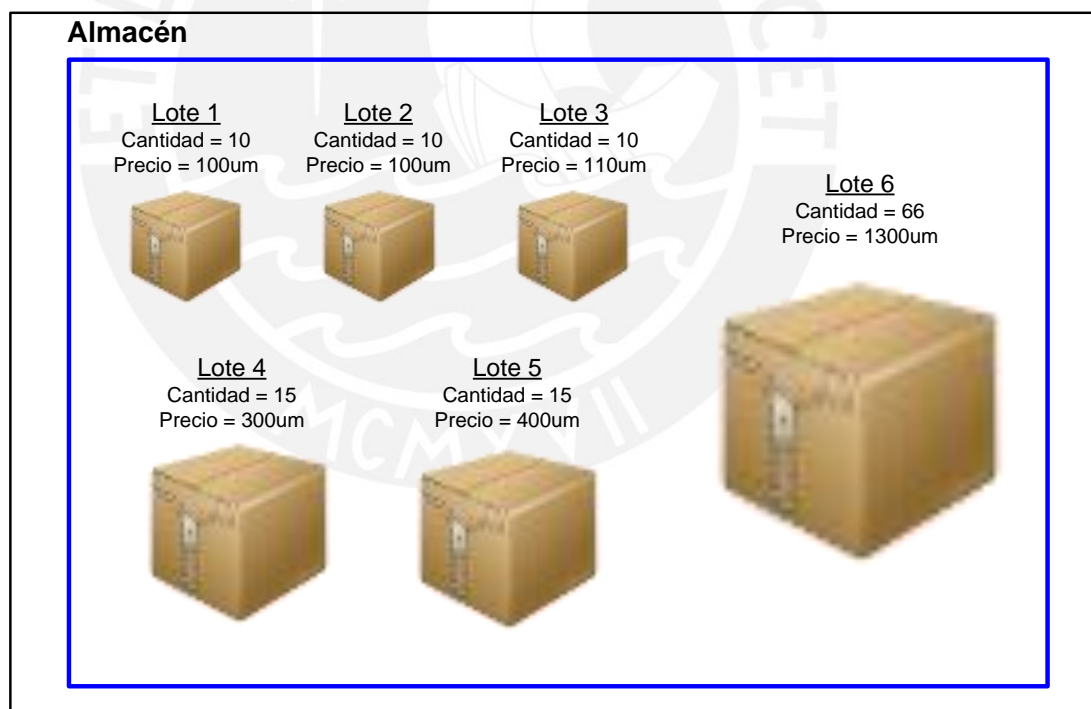


Figura 1.15: Almacén y lotes de materia prima [propia autoría]

El sistema de información permitirá realizar los registros y dada de baja de los lotes de materias primas a utilizar en los diversos planes de producción de las órdenes de producción ingresadas.

Se manejará un solo almacén, donde se registrarán los lotes por producto con su respectivo precio y la cantidad de materia prima en cada una de ellas, como se aprecia en la figura 1.15.

Para el cálculo de los costos de materias prima en cada actividad se realizará lo siguiente:

- Se verificará que, para una orden de producción, exista stock suficiente para cada materia prima necesaria.
- Con la cantidad necesaria se procederá a obtener cada lote necesario, mediante el método PEPS (Primero en Entrar Primero en Salir).
- Con la lista de lotes, se procederá a obtener un “precio promedio” del valor de la materia prima, ya que los lotes pueden no ser uniformes, es decir ser de diferentes cantidades y precios.

$$\text{Precio}_{\text{Promedio}} \equiv \frac{\sum \left(\frac{\text{Precio}_{\text{Total}_{\text{del}_{\text{Lote}}}}}{\text{Cantidad}_{\text{Total}_{\text{en}_{\text{el}_{\text{Lote}}}}} \times \text{Cantidad}_{\text{a}_{\text{Usar}_{\text{del}_{\text{Lote}}}}} \right)}{\sum \text{Cantidad}_{\text{de}_{\text{cada}_{\text{Lote}}}}}$$

- Con el Precio Promedio calculado, se procede a asignar el costo de la materia prima en cada actividad, el cual estará dado por el producto de la cantidad necesaria y el precio promedio.

$$\text{CostoMateriaPrima}_{y_{\text{Actividad}}_x} \equiv \text{Precio}_{\text{Promedio}}_y \times \text{Cantidad}_{\text{MateriaPrima}}_{y_{\text{Actividad}}_x}$$

- Finalmente, una vez asignado el costo de materia prima en cada actividad de cada plan de producción necesario, se procederá a actualizar la cantidad disponible en cada lote y a dar de baja a aquellos en los que se usó su totalidad de materia prima.

1.5.2.4. Otros costos directos

La asignación de otros costos directos que no sean materia prima, mano de obra o maquinaria se realizará paramétricamente, es decir se asignará el costo directo en cada actividad (costo directo registrado en el sistema por el usuario) y se ingresará directamente el valor en unidades monetarias.

1.5.2.5. Costos indirectos

“Costos relacionados con un objeto de costo particular que no pueden rastrearse a ese objeto de manera económicamente factible (efectiva en cuanto a costes se refiere). Los costos indirectos se asignan al objeto del costo a través de un método de asignación de costos”. [2]

Los costos indirectos son aquellos costos que no son identificables con un producto en particular, es decir son costos que pueden o no estar en el proceso productivo, sin embargo son necesarios para el funcionamiento y operatividad de la empresa. Por ejemplo: Los costos de la fuerza de ventas, la gerencia general, amortizaciones de intangibles, combustible, etc.

Para la distribución y cálculo adecuado de estos costos indirectos es necesario optar por un sistema de costeo específico y acorde con el tipo de producción a realizar. El sistema de costeo indirecto a utilizar será el de “Costeo por órdenes de producción”.

“Las empresas que comúnmente utilizan este sistema son:

- *de impresión*
- *astilleros*
- *aeronáutica*
- *de construcción*
- *de ingeniería” [28]*

Asimismo, este sistema es el más adecuado en aquellas industrias en las cuales la producción es unitaria (por ejemplo: una bomba de agua, un transformador, un auto, un avión, etc.) siendo el más adecuado en este caso en particular, ya que el alcance del sistema de información está sustentado en la aplicación de logística inversa en empresas de producción manufactureras, donde se emplean materias primas y materiales no perecibles (no se utilizan alimentos u otros con fecha de vencimiento).

Una orden de producción estará conformada por líneas de órdenes, donde cada una de ellas reflejará un producto y la cantidad a producir, tal como se ve en el siguiente ejemplo:

Orden N°: 0000001

<u>Nombre del producto</u>	<u>Cantidad</u>
Producto 'A'	12
Producto 'B'	10
Producto 'C'	8
.....
.....
Producto 'X'	200

Para la distribución o prorrateo de los costos indirectos será necesario utilizar una base de asignación o inductor, los cuales serán los siguientes:

- Unidades de producción
- Horas máquina
- Costo de mano de obra directa

Por ejemplo:

	<u>Producto 'A'</u>	<u>Producto 'B'</u>	<u>Producto 'X'</u>
Unidades de producción	12	10		200
Horas máquina	100	120		10
Mano de obra directa	3000	2500		30 000

Los inductores se calcularán con la división de lo que le corresponde a cada producto entre el total.

$$ind\phi_x \equiv \frac{Valor_del_Producto_x}{\sum_{i=Producto_A'} Valor_del_Producto_i}$$

Además, después de calcular el inductor correspondiente a cada producto se procederá a distribuir el costo indirecto para cada producto.

$$\text{Costo_Indirecto_Producto}_x \equiv \text{ind}\varphi_x \times \text{CostoIndirectoTotal}$$

Para costear los costos indirectos solo se utilizará un inductor por cada orden de producción.

1.5.3. Solución de la aplicación de logística inversa

La solución para la aplicación de logística inversa en la etapa productiva consiste en el registro de la actividad en la cual se produjo la falla, luego el registro de la decisión tomada con respecto a la falla y finalmente, si se da el caso, la recuperación de la materia prima involucrada hasta el momento, indicando la cantidad usada y valor de recuperación, generando un nuevo lote automáticamente y registrándolo en el almacén para su posterior uso.

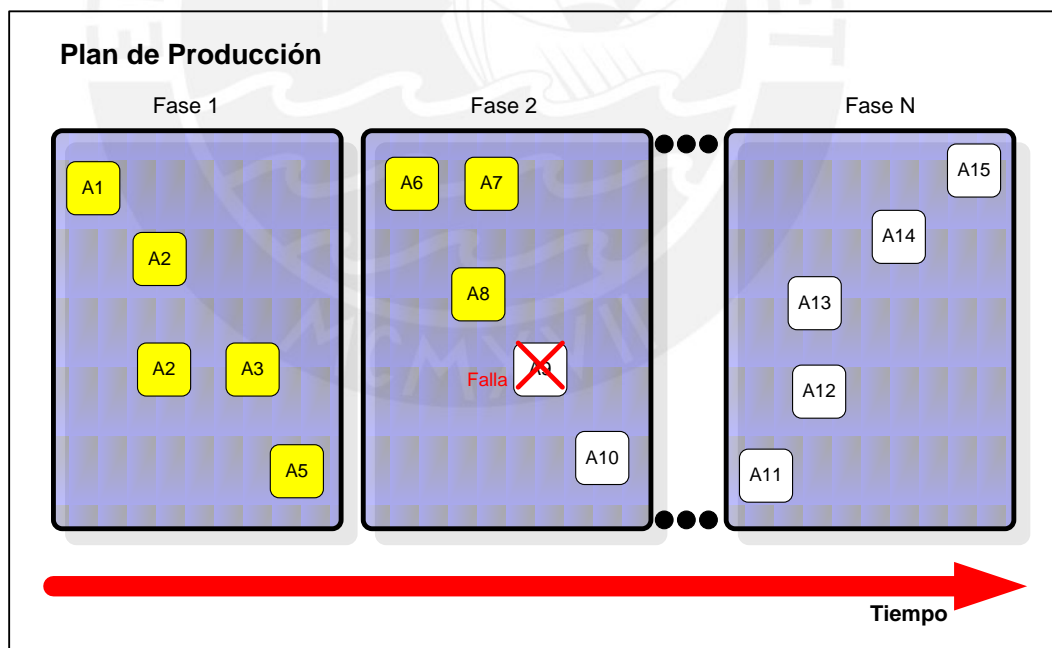


Figura 1.16: Plan de producción mostrando la falla producida en una actividad [propia autoría]

Asimismo, se calculará los costos incurridos hasta el momento de la actividad en que se produjo, como se ve en la figura 1.16.

1.5.4. Algoritmo del cálculo de la mejor decisión

El sistema tendrá la funcionalidad de calcular la mejor decisión ante una determinada falla que se produzca en el proceso productivo, de una determinada orden de producción en base a la actividad que falló. Por ello, para el cálculo de la mejor decisión se tomará como fuente teórica un proceso extraído de la Investigación de Operaciones.

“El proceso de jerarquía analítica (AHP, por sus siglas en inglés) está diseñado para casos en que las ideas, sentimientos y emociones se cuantifican con base en juicios subjetivos para obtener una escala numérica para dar prioridades a las alternativas de decisión”. [3]

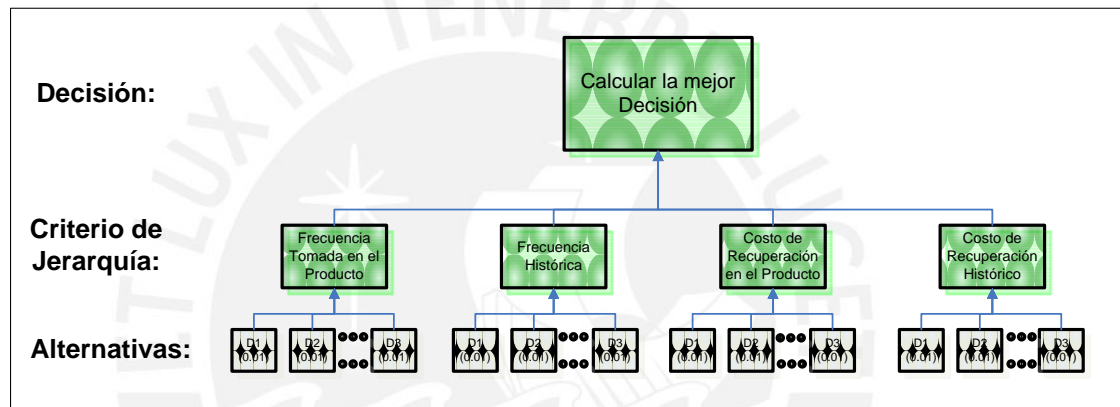


Figura 1.17: Diagrama AHP [autoría propia]

Como se puede apreciar en la figura 1.17, el proceso de jerarquía analítica está compuesto por tres partes, las cuales se aplicarán de la siguiente manera para el sistema de logística inversa:

- **Decisión:** es el objetivo del proceso, que en este caso es el cálculo de la mejor decisión a tomar.
- **Criterio de Jerarquía:** es aquel juicio subjetivo o criterios para el cálculo de la decisión. Para este sistema en particular se escogieron cuatro criterios de jerarquía, que son los siguientes:
 - Frecuencia de la decisión tomada en el producto para una actividad en particular.
 - Frecuencia histórica en que la decisión fue tomada.
 - Costo de recuperación logrado con la toma de la decisión para el producto y la actividad específica.
 - Costo de recuperación histórico logrado con la toma de la decisión.

- **Alternativas:** son todas aquellas posibles decisiones a tomar, es decir es la lista de las cuales se escogerá a la mejor según el cálculo.

Para llevar a cabo este proceso se establecerá una matriz de dos dimensiones que será la siguiente:

	Frecuencia en el Producto	Frecuencia Histórica	Costo de Recuperación en el Producto	Costo de Recuperación Histórica
Decisión 1	xx.xx	mm.mm	kk.kk	mm.mm
Decisión 2	yy.yy	xx.xx	zz.zz	zz.zz
Decisión 3	zz.zz	yy.yy	xx.xx	yy.yy
Decisión 4	mm.mm	mm.mm	pp.pp	xx.xx
Decisión 5	kk.kk	pp.pp	kk.kk	mm.mm
Decisión 6	xx.xx	yy.yy	aa.aa	kk.kk
Decisión 7	pp.pp	xx.xx	mm.mm	pp.pp
Decisión 8	oo.oo	zz.zz	xx.xx	mm.mm
Decisión 9	tt.tt	kk.kk	yy.yy	xx.xx

Tabla 1.1: Matriz para el Cálculo del proceso de Jerarquía Analítica

De la matriz, se asignará un peso porcentual a cada “criterio de jerarquía”, siendo esta de 25% para cada una ellas por defecto. Este porcentaje es modificable y estará disponible en el módulo de seguridad como la configuración de criterios de jerarquía AHP.

Asimismo, los valores de cada “jerarquía de decisión” para cada decisión serán extraídas de la data acumulada en la base de datos del sistema, por ello a mayor cantidad de datos que se vayan acumulando en el tiempo permitirán un cálculo de mejor calidad.

Finalmente, la fórmula para el cálculo de la mejor decisión será como sigue:

$$\beta_{Decision1} \equiv 0.25 \times xx.xx + 0.25 \times mm.mm + 0.25 \times kk.kk + 0.25 \times mm.mm$$

$$\beta_{Decision2} \equiv 0.25 \times yy.yy + 0.25 \times xx.xx + 0.25 \times zz.zz + 0.25 \times zz.zz$$

$$\beta_{Decision3} \equiv 0.25 \times zz.zz + 0.25 \times yy.yy + 0.25 \times xx.xx + 0.25 \times mm.mm$$

.....

.....

$$\beta_{Decision9} \equiv 0.25 \times tt.tt + 0.25 \times kk.kk + 0.25 \times yy.yy + 0.25 \times xx.xx$$

Se calculará un valor β eta para cada una de las decisiones, siendo la de mayor valor la mejor decisión a tomar, es decir la respuesta.

1.5.5. Ventajas de uso del sistema

Según lo explicado y sustentado en la descripción de la solución, se puede inferir que el uso del sistema de información de logística inversa en la etapa productiva trae consigo varias ventajas como las siguientes:

- Se permite registrar, de forma detallada, un plan de producción indicando tanto las fases que la componen, así como las actividades de cada fase. Además, se permite el registro del tiempo de duración de la actividad y del tiempo de inicio con respecto tiempo del plan de producción en su totalidad.
- Se permite registrar varios planes de producción para un producto y establecer uno como activo.
- Se permite el registro de los costos directos a cada actividad.
- Se permite el prorroteo de los costos indirectos en base a un inductor y una orden de producción.
- Se registra una orden de producción donde se valida lo siguiente:
 - Que todos los productos ingresados tengan un plan de producción activo, con ello se valida generar ordenes de producción coherentes.
 - Que exista stock de materia prima disponible para todas las materias primas de las diferentes órdenes de producción, es decir se garantiza un registro coherente con lo disponible en el almacén.
 - Que todos los empleados y maquinarias necesarios, tengan un precio para su respectiva unidad de medida. Es decir, se garantiza que estos costos directos asignados tengan un valor monetario en el momento de registrar la orden de producción.
- Una vez validada la orden de producción se procede al registro de la orden de producción de la siguiente forma:
 - Se reduce el stock de la materia prima utilizada en la orden de producción, es decir hay una reducción automática de los lotes y con ello se permite mantener constantemente actualizado la cantidad de bienes disponibles en el almacén.

- Se registra los costos incurridos de cada actividad, tanto la mano de obra, maquinaria y otros ingresados como parámetros.
- Se registra la orden y los productos que la componen.

- Se permite el registro de las fallas y la decisión de para cada una de ellas.

- Se dispone de una funcionalidad de consulta de la mejor decisión a tomar en caso del registro de una falla, todo ello con los datos históricos almacenados en la base de datos, permitiendo que el cálculo se logre de manera automática y rápida.

- Finalmente, se permite el registro de la cantidad de materia prima recuperada y su precio (en caso se dé) en base a las fallas que se registraron previamente.

Se concluye que la principal ventaja del sistema de información de logística inversa es la integración de los planes de producción, sus costos, el registro de las fallas, decisiones y la posibilidad de registrar materia prima en base a la recuperación generada. Por ello, se recalca que esta es una herramienta novedosa con respecto a los tradicionales sistemas de información de producción.

2. Análisis

“El Análisis de sistemas trata básicamente de determinar los objetivos y límites del sistema objeto de análisis, caracterizar su estructura y funcionamiento, marcar las directrices que permitan alcanzar los objetivos propuestos y evaluar sus consecuencias” [30].

El presente capítulo presenta la metodología aplicada en el desarrollo del sistema, así como su justificación. Además, se muestran la lista de los artefactos del proyecto, los requerimientos funcionales y no funcionales, los casos de uso del sistema, el diagrama de clases de análisis, la viabilidad del sistema y el análisis técnico económico.

2.1. Metodología aplicada para el desarrollo de la solución

Para el proceso de desarrollo de software del presente proyecto se empleará el Lenguaje Unificado de Modelado UML en su versión 2.0. La elección de este lenguaje de modelado está basada principalmente por ser la más acorde y utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Asimismo, se aplicará la metodología RUP (*Rational Unified Process*)[17] para la obtención del conjunto de entregables que son expresados como artefactos.

La metodología tiene tres características principales:

- Dirigido por Casos de Uso.
- Centrado en la Arquitectura.
- Iterativo e Incremental.

Además, divide el proceso de desarrollo general en ciclos, teniendo un conjunto de productos al culminar cada uno de ellos, tal como se puede muestra en la figura 2.1:

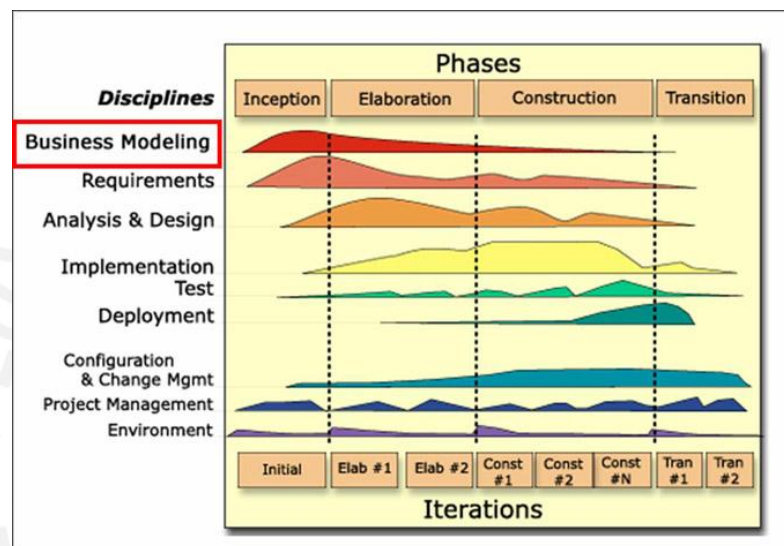


Figura 2.1: Fases de RUP [18]

- **Concepción:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos.
- **Elaboración:** se hace un plan de proyecto, se completan los casos de uso y se mitigan los riesgos.
- **Construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente.
- **Transición:** Se maneja el producto β eta con un usuario (no forma parte del alcance del proyecto). [1]

UML, por medio del uso de los artefactos RUP, es un lenguaje adecuado para proyectos de software de gran envergadura porque permite la planificación de un equipo de desarrollo con una gran cantidad de integrantes. Sin embargo, para esta tesis en particular, el proyecto no es considerado de tamaño grande ya que el número de integrantes es bajo y el alcance corresponde a un proyecto de fin de carrera, sin embargo esto no impide su utilización debido a que, al ser éste un

proyecto con presentaciones y entregables constantes, RUP ofrece una serie de artefactos como por ejemplo: modelo de casos de uso, catálogos, código fuente, etc. siendo obtenidos de un conjunto de iteraciones e incrementándose en cada una de ellas. El alcance de este proyecto solo contempla los ciclos de concepción, elaboración y construcción.

Además, otro punto importante en cualquier desarrollo de software es el conocimiento adecuado y experiencia en la metodología elegida, siendo esta también una razón de la elección.

A continuación se presenta la lista de artefactos RUP que formarán parte del proyecto de desarrollo y que se serán necesarios para la obtención del producto final:

- **Catálogo de requerimientos:** Son los requisitos funcionales y no funcionales del sistema. Es de suma importancia para establecer el alcance y las funcionalidades del proyecto. Destacado en requisitos funcionales y no funcionales, ver sección 2.2.1 y 2.2.2.
- **Diagramas de casos de uso:** Son los casos de uso identificados a partir de los requisitos del sistema. Es de vital importancia para el proyecto ya que es la entrada principal para la especificación de requisitos, así como poder entender la lógica del negocio, sus procesos e identificar los actores del sistema. Destacado en los casos de uso por paquete, ver sección 2.3.1.
- **Especificación de requisitos de software - ERS:** El especificación de cada uno de los casos de uso, permitiendo establecer la interacción entre el usuario y el sistema, así como los flujos básicos, alternativos y excepcionales que debe tener el sistema. Ver sección 1 de anexos.
- **Diagrama de clases de análisis:** Presenta un primer diagrama general de las clases a utilizar en el sistema, es decir muestra un primer bosquejo para el diseño. Asimismo, se presenta el diccionario de clases. Ver figura 2.9 Diagrama de clases de análisis.
- **Diagrama de clases de diseño:** Presenta el diagrama de clase de diseño, donde se aprecia las clases que pertenecen a cada una de las capas del

sistema, su interrelación entre ellas y los métodos y atributos de cada una de ellas. Ver sección 3.2 de anexos.

- **Diagrama de secuencias:** Presenta las secuencias del sistema, es decir el conjunto de pasos y la relación entre las diferentes clases del sistema, todo ello mediante la llamada a métodos e instancias de las clases. Resulta importante para llevar a cabo una correcta construcción del sistema.

Todos los artefactos descritos se pueden revisar en los anexos del proyecto.

Es importante mencionar que para el control de calidad de del producto software será preciso utilizar las denominadas “clases de equivalencia” y el “catálogo de pruebas”, ver secciones 7 y 8 de anexos. En las clases de equivalencia se establecen los valores aceptados y no aceptados de cada uno de los campos del sistema. Asimismo, el catálogo de pruebas esta enfocado a las pruebas funcionales del sistemas en sí, es decir se basa en las pantallas construidas.

Asimismo, este proyecto necesitará de una metodología para su gestión en todo momento, es decir para su administración. La elegida es la del Instituto de Administración de Proyectos (PMI), tal cual se señaló en el sección 1.3 *Plan de Proyecto*, que ofrece el estándar PMBOK® reconocido por la IEEE⁴ ANSI que se adapta perfectamente a este proyecto debido a que está orientado a procesos.

Este estándar define 9 áreas del conocimiento con un conjunto de procesos básicos que se enumeran a continuación [5]:

- **Integración**, destacado en el plan de proyecto, ver sección 1.3.
- **Alcance**, destacado en el objetivo general y los objetivos específicos, ver sección 1.1.1 y 1.1.2.
- **Tiempo**, destacado en el cronograma de trabajo, ver sección 1.4.
- **Costo**, destacado en la sección de Costos del capítulo 3.
- **Calidad**, destacado en pruebas de software, ver capítulo 4
- **Recursos Humanos**, no aplica en el proyecto.
- **Comunicaciones**, no aplica en el proyecto.
- **Riesgos**, destacado en la sección de riesgos (ver sección 12 de anexos).

⁴ IEEE corresponde a las siglas de *The Institute of Electrical and Electronics Engineers*, que es una asociación técnico-profesional mundial dedicada a la estandarización e investigación de desarrollo de tecnologías.

- **Adquisiciones**, no aplica en el proyecto.

También, PMBOK® utiliza una variación del Ciclo de Deming para el mejoramiento continuo de las 5 etapas del ciclo de vida, como se muestra en la figura 2.2, la cual se aplicará en cada una de los fases de RUP.

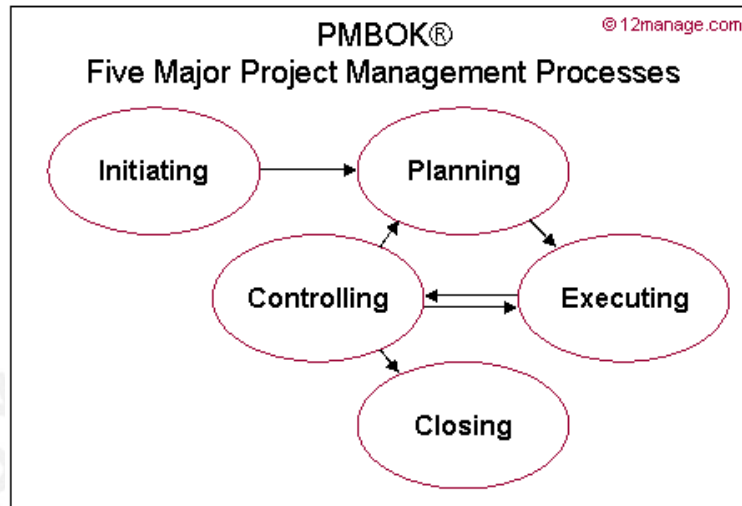


Figura 2.2: Diagrama de procesos del PMBOK [19]

“Un proceso es un conjunto de acciones y actividades interrelacionadas que se llevan a cabo para alcanzar un conjunto previamente especificado de productos, resultados o servicios.

Los procesos de de la dirección de proyectos comunes a la mayoría de los proyectos por lo general están relacionados entre sí por el hecho de que se llevan a cabo para un propósito integrado. El propósito es integrar, planificar ejecutar, supervisar y controlar, y cerrar un proyecto” [5].

Para cada una de las fases del proyecto se tendrá primero que realizar el proceso de iniciación, luego el planeamiento de su ejecución, seguido la ejecución y su control continuo para finalmente llegar al proceso de cierre.

Los procesos del PMBOK se encargan de lo siguiente:

- **Iniciación:** define y autoriza el proyecto / fase.
- **Planificación:** refina los objetivos, planifica el curso de acción para lograr objetivos del proyecto.

- **Ejecución:** integra los recursos para llevar adelante el plan de gestión del proyecto.
- **Control:** mide y supervisa regularmente a fin de identificar variaciones y tomar medidas correctivas para el logro de objetivos.
- **Cierre:** formaliza aceptación de producto y termina ordenadamente el proyecto/fase.

2.2. Identificación de requerimientos

Los requerimientos funcionales fueron obtenidos de acuerdo a un levantamiento de información que se realizó a un empleado de una empresa industrial del medio local. Asimismo, otros de estos requerimientos fueron obtenidos en base a lo señalado en el capítulo 1, en torno al marco conceptual y la justificación de la solución.

Estos requerimientos solucionan el problema ya que se contempla desde lo básico como son los mantenimientos (productos, materias primas, empleados, maquinarias), así como el del proceso productivo y las variables para el cálculo de los costos y un mantenimiento de toma decisiones. Finalmente también abarca los reportes gerenciales, útiles para poder apreciar los beneficios de la aplicación de logística inversa, además de los requisitos para la implementación del módulo de seguridad.

2.2.1. Requisitos funcionales

A continuación se muestra la lista de requisitos funcionales.

No	Descripción	Prioridad
Módulo Maestros		
1.	El sistema permitirá el mantenimiento de los productos que se van a producir.	1
2.	El sistema permitirá el mantenimiento de los planes de producción de cada producto.	1
3.	El sistema permitirá el mantenimiento de las fases que componen el plan de producción.	1
4.	El sistema permitirá el mantenimiento de las actividades que conforman cada fase.	1

5.	El sistema permitirá la asignación de actividades en cada fase.	2
6.	El sistema permitirá realizar el mantenimiento de materias primas.	1
7.	El sistema permitirá el mantenimiento de la cantidad de materia prima a usar en cada actividad del plan de producción.	2
8.	El sistema permitirá el mantenimiento de los costos de la materia prima en cada actividad del plan de producción.	1
9.	El sistema permitirá realizar el mantenimiento de empleados.	1
10.	El sistema permitirá la asignación de los empleados para cada proceso productivo	1
11.	El sistema permitirá la asignación del(os) empleado(s) a cada actividad.	1
12.	El sistema permitirá el ingreso de la cantidad de horas-hombre a utilizar en cada actividad del plan de producción.	1
13.	El sistema permitirá realizar el mantenimiento de maquinarias.	1
14.	El sistema permitirá la asignación de la(s) máquina(s) a cada actividad del plan de producción.	1
15.	El sistema permitirá realizar el tendrá un mantenimiento de variables para el cálculo de costo unitario de producción (por defecto y obligatorias son: horas-hombre, maquinaria y materia prima).	2
16.	El sistema permitirá el registro, finalización y anulación de órdenes de producción.	1
17.	El sistema permitirá el registro de los precios para cada materia prima y maquinaria.	1
18.	El sistema permitirá el registro, confirmación y anulación de una nota de ingreso de materias primas.	1
19.	El sistema permitirá registrar las líneas de notas de ingreso (Materia prima, cantidad de lotes, cantidad por lote, precio por lote).	1
20.	El sistema permitirá visualizar y dar de baja a lotes activos.	3
21.	El sistema permitirá el registro del código externo para cada lote.	3
Módulo de costos y decisiones		
22.	El sistema permitirá el registro de los costos de la materia prima por fechas en cada actividad del proceso productivo.	1
23.	El sistema calculará el costo de las horas-hombre en cada actividad del proceso productivo.	2
24.	El sistema permitirá el registro de los costos de maquinaria por fechas en cada actividad del proceso productivo.	1
25.	El sistema registrará los costos de cada variable registrada para cada actividad del proceso productivo.	1
26.	El sistema permitirá la selección de un inductor para el cálculo de costos indirectos en una orden de producción.	1

27.	El sistema registrará y calculará los costos indirectos de cada producto, en unidades monetarias, para una orden de producción determinada.	1
28.	El sistema calculará automáticamente los costos acumulados hasta cada una de las fases del proceso productivo.	2
29.	El sistema calculará automáticamente el costo unitario de un producto específico.	3
30.	El sistema dará de baja automáticamente y reducirá la cantidad de materia prima, para los lotes utilizados en el cálculo de los costos para una orden de producción.	1
31.	El sistema permitirá el registro de las fallas producidas en el proceso de producción, para una determinada orden de producción.	1
32.	El sistema permitirá realizar el mantenimiento de decisiones para las fallas producidas en el proceso de producción.	1
33.	El sistema permitirá la asignación de decisiones a cada actividad del proceso productivo.	1
34.	El sistema mostrará automáticamente en el registro de una falla el conjunto de decisiones posibles a tomar.	1
35.	El sistema permitirá registrar la cantidad de materia prima recuperada ante una falla registrada para una orden de producción.	1
36.	El sistema calculará y mostrará automáticamente la mejor decisión a tomar ante el registro de una falla.	1
37.	El sistema permitirá visualizar el proceso de producción en un entorno gráfico, mostrando el tiempo de duración de cada actividad.	3
Módulo de seguridad		
38.	El sistema deberá permitir el registro de los usuarios que emplearán las diferentes funcionalidades.	1
39.	El sistema deberá permitir el cambio de contraseñas a los diferentes usuarios.	2
40.	El sistema deberá permitir la validación de los usuarios y contraseñas al momento que se ingrese al mismo.	1
41.	El sistema deberá permitir el establecimiento de permisos para las diferentes funcionalidades que brinda. Los permisos deberán ser considerados desde ingreso a módulos principales.	1
42.	El sistema permitirá la configuración de las variables de seguridad (longitud mínima contraseña, longitud máxima, cantidad de caracteres numéricos, cantidad de caracteres no numéricos, validez de la contraseña)	1
Módulo de reportes		
43.	El sistema emitirá un reporte mostrando un plan de producción en un entorno gráfico.	3

44.	El sistema emitirá un reporte del detalle de los planes de producción utilizados en una orden de producción (mostrará las fases y actividades).	2
45.	El sistema emitirá un reporte orden de producción indicando los productos, cantidades y costos totales de materias primas, mano de obra, maquinaria, otros costos directos y costos indirectos para la orden de producción.	1
46.	El sistema emitirá un reporte del detalle de los costos indirectos de producción de una orden de producción determinada, agrupadas por producto.	2
47.	El sistema emitirá un reporte del detalle de los costos directos de producción de una orden de producción determinada, mostrando para cada fase y actividad, agrupadas por producto.	2
48.	El sistema emitirá un reporte estadístico de las frecuencias de tomas de decisiones, para un rango de fechas.	1
49.	El sistema emitirá un reporte estadístico de las fallas presentadas en las actividades, para un rango de fechas.	1
50.	El sistema emitirá un reporte estadístico mostrando la cantidad de materia prima recuperada, para un rango de fechas.	1

Tabla 2.1: Lista de requisitos funcionales.

2.2.2. Requisitos no funcionales

A continuación se muestra la lista de requisitos funcionales.

No	Descripción	Tipo
1.	El sistema será elaborado empleando el lenguaje de programación JAVA.	E
2.	El sistema tendrá como motor de base de datos Oracle Express.	E
3.	El sistema deberá funcionar en estaciones basadas en Windows.	E
4.	El sistema deberá poder ejecutarse en red.	E
5.	Todos los reportes podrán ser exportados a formato PDF.	E
6.	El sistema será trabajado en capas que permitan una rápida movilidad y modificación de interfaz gráfica.	E

Tabla 2.2: Lista de requisitos no funcionales.

Tipos:

Letra	Descripción
E	Exigible
D	Deseable

Tabla 2.3: Tabla descriptiva del tipo de requisito.

Prioridad:

Número	Descripción
1	Alto
2	Medio
3	Bajo

Tabla 2.4: Tabla descriptiva de la prioridad del requisito.

2.3. Modelo de casos de uso

“El diagrama de casos de uso representa la forma en cómo un Cliente (Actor) opera con el sistema en desarrollo, además de la forma, tipo y orden en como los elementos interactúan (operaciones o casos de uso)” [31].

A continuación se presenta los paquetes del sistema, los casos de uso por módulo y los actores que interactúan con ellos.

2.3.1. Casos de uso por paquete

En esta sección se muestra el paquete general (ver figura 2.3) del sistema Logística Inversa, el cual está constituido por los paquetes de: maestros, costos y toma de decisiones, reportes y seguridad.

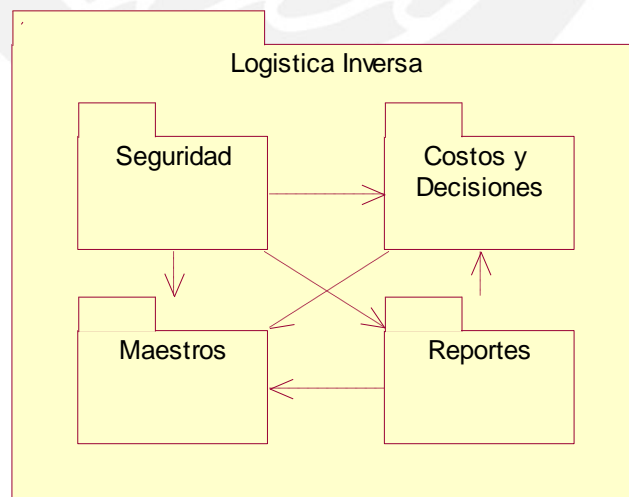


Figura 2.3: Paquetes del sistema

2.3.2. Catálogo de actores

A continuación en la figura 2.4 se identifican a los actores del sistema.

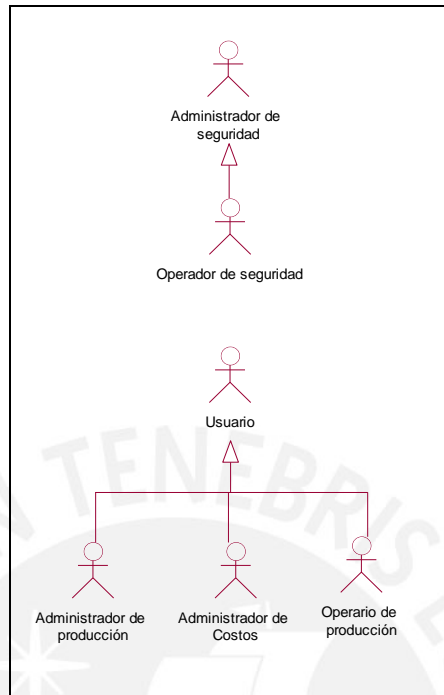


Figura 2.4: Diagrama de actores del sistema

Actor	Descripción
Administrador de seguridad	Usuario encargado del control de la seguridad en la organización. Posee acceso a todas las funcionalidades del módulo de seguridad.
Operador de seguridad	Usuario encargado de manejar funcionalidades básicas del módulo de seguridad como crear usuarios, administrar perfiles, etc.
Usuario	Usuario que sólo interactúa con el sistema validando su acceso al mismo.
Administrador de producción	Usuario encargado del manejo de los maestros del sistema. Puede realizar todos los mantenimientos permitidos para aplicación de logística inversa en el proceso productivo.

Administrador de costos	Usuario encargado de la asignación de costos a cada uno de los procesos productivos.
Operario de producción	Usuario encargado de realizar el proceso productivo y/o manejar el sistema, asignando las decisiones a tomar en caso de fallas.

Tabla 2.5: Tabla descriptiva del catálogo de actores

2.3.3. Módulo de maestros

A continuación en la figura 2.5 se aprecian los casos de uso correspondientes al módulo de maestros. Para consultar sus descripciones, ver en anexos la sección 1.

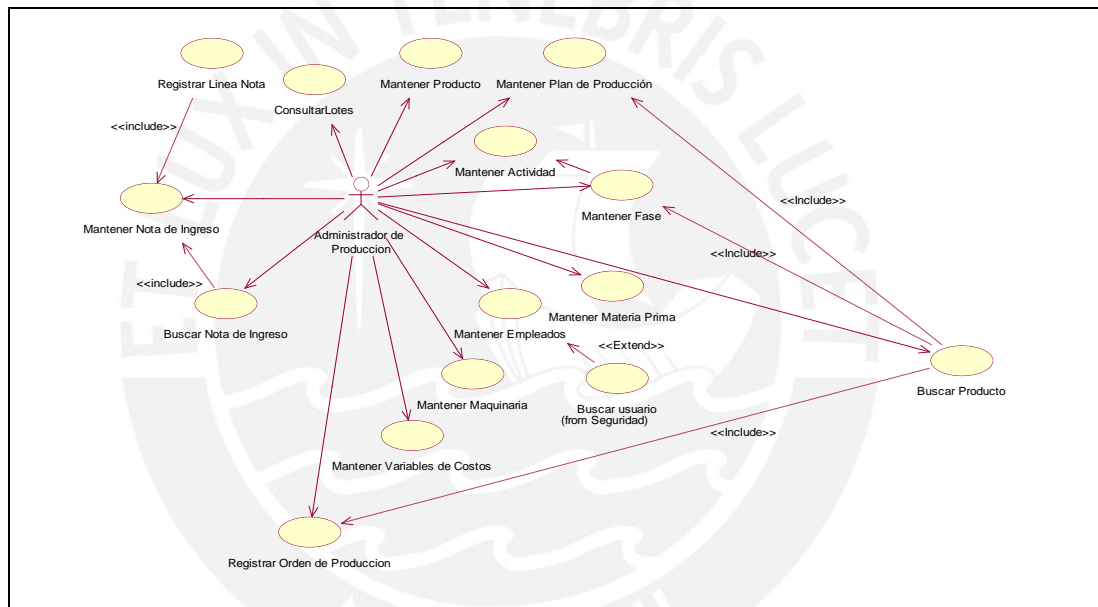


Figura 2.5: Casos de uso del módulo maestros

Mantener Producto:

Actor	Administrador de producción.
Descripción	Este caso de uso permite registrar, modificar y eliminar los productos del sistema.
Pre-condiciones	El usuario debe estar registrado al sistema y tener el permiso de acceso a este mantenimiento.
Post-condiciones	Se realiza el mantenimiento de un producto.

Tabla 2.6: Tabla descriptiva del caso de uso Mantener producto.

Mantener Plan de Producción:

Actor	Administrador de producción.
Descripción	Este caso de uso permite registrar, modificar y eliminar los planes de producción de un producto.
Pre-condiciones	El usuario debe estar registrado en el sistema y tener el permiso de acceso a este mantenimiento. Debe existir al menos un producto registrado en el sistema.
Post-condiciones	Se realiza el mantenimiento de un plan de producción.

Tabla 2.7: Tabla descriptiva del caso de uso Mantener plan de producción.

2.3.4. Módulo de costos y decisiones

A continuación en la figura 2.6 se aprecia los casos de uso correspondientes al módulo de costos y decisiones. Para consultar sus descripciones, ver en anexos la sección 1.

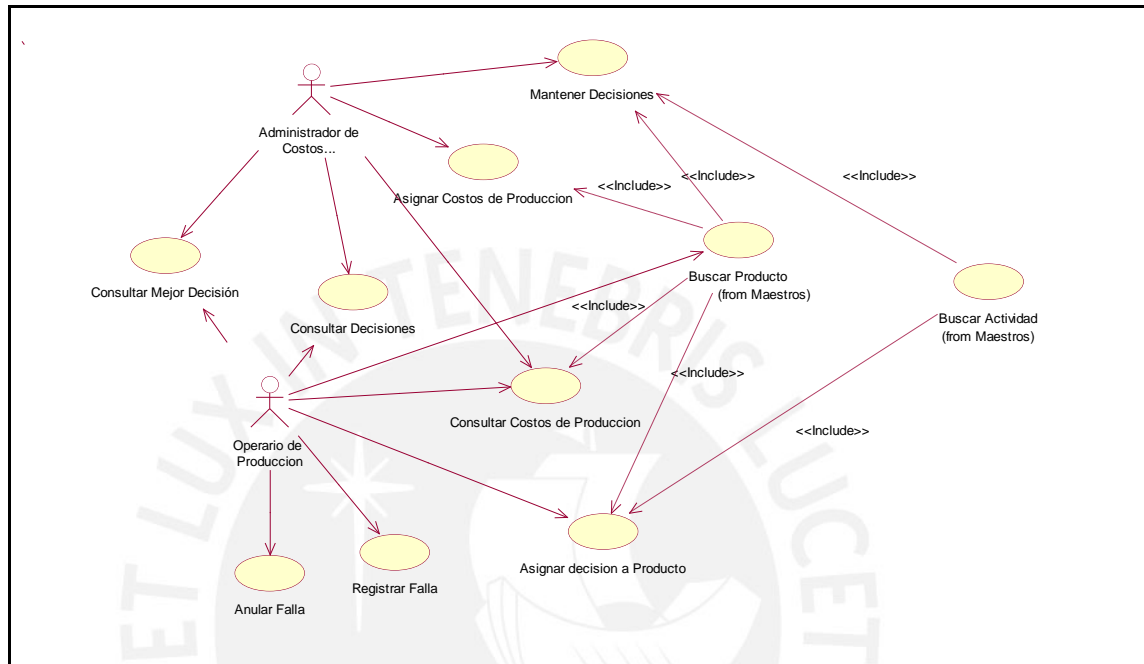


Figura 2.6: Casos de uso del módulo costos y decisiones

Registrar Falla:

Actor	Operador de producción.
Descripción	Este caso de uso permite registrar las fallas producidas en un proceso productivo.
Pre-condiciones	El usuario debe estar registrado en el sistema, tener el permiso de acceso a esta funcionalidad, haber registrado una orden de producción.
Post-condiciones	Se realiza el registro de una falla en el proceso productivo, para una orden de producción.

Tabla 2.8: Tabla descriptiva del caso de uso registrar falla.

Asignar decisión a producto:

Actor	Operador de producción.
Descripción	Este caso de uso permite asignar un conjunto de decisiones para un proceso de producción específico.
Pre-condiciones	El usuario debe estar registrado en el sistema, tener el permiso de acceso a esta funcionalidad, haber registrado al menos un proceso productivo y un conjunto de decisiones en el sistema.
Post-condiciones	Se realiza la asignación de decisiones para un proceso productivo.

Tabla 2.9: Tabla descriptiva del caso de uso asignar decisión a producto.

2.3.5. Módulo de reportes

A continuación en la figura 2.7 se aprecian los casos de uso correspondientes al módulo de reportes. Para consultar sus descripciones, ver en anexos la sección 1.

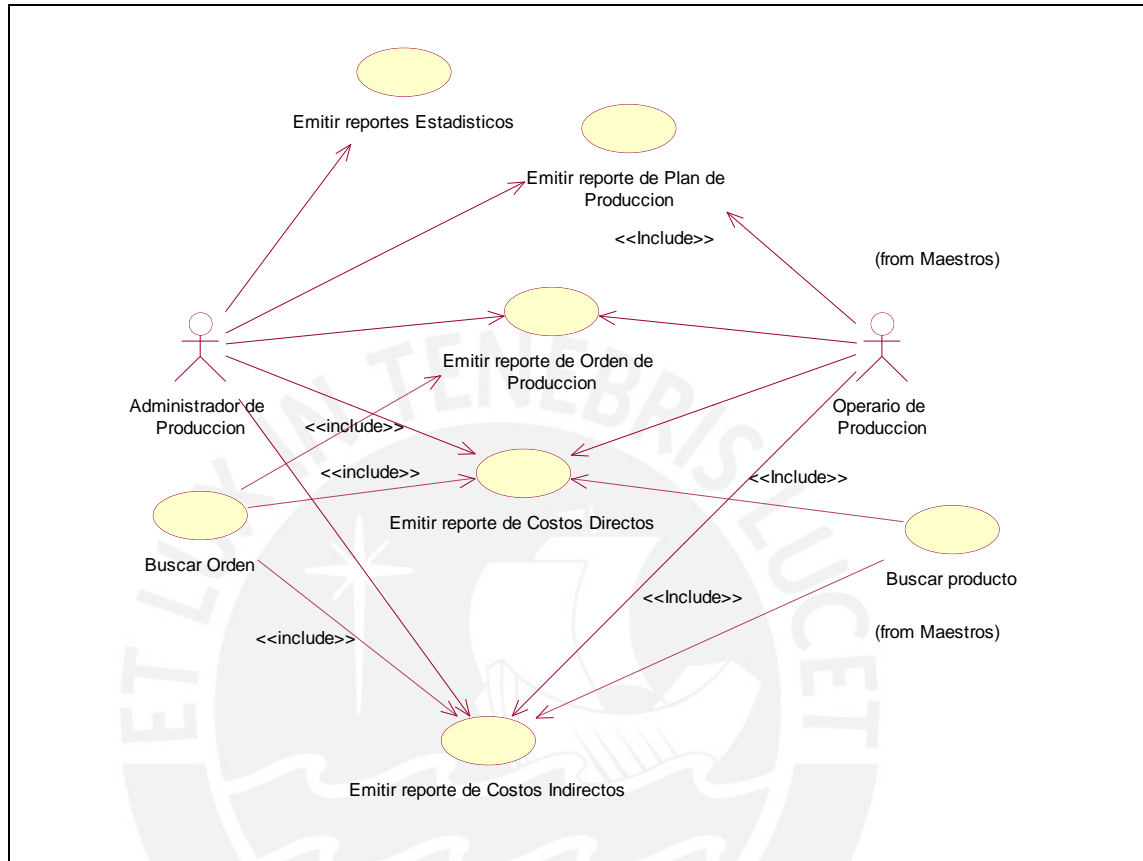


Figura 2.7: Casos de uso del módulo de reportes

Emitir reporte estadísticos:

Actor	Administrador de producción.
Descripción	Este caso de uso permite visualizar el reporte estadístico para un rango de fechas determinadas.
Pre-condiciones	El usuario debe estar registrado en el sistema, tener el permiso de acceso a esta funcionalidad.
Post-condiciones	Se visualiza correctamente el reporte.

Tabla 2.10: Tabla descriptiva del caso de uso emitir reporte estadísticos.

Emitir reporte de Plan de Producción

Actor	Administrador de producción.
Descripción	Este caso de uso permite visualizar en un reporte el plan de producción de un producto determinado.
Pre-condiciones	El usuario debe estar registrado en el sistema, tener el permiso de acceso a esta funcionalidad y haber registrado al menos un plan de producción.
Post-condiciones	Se visualiza correctamente el reporte.

Tabla 2.11: Tabla descriptiva del caso de uso Emitir reporte plan de producción.

2.3.6. Módulo de seguridad

A continuación en la figura 2.8 se aprecian los casos de uso correspondientes al módulo de seguridad. Para consultar sus descripciones, ver en anexos la sección 1.

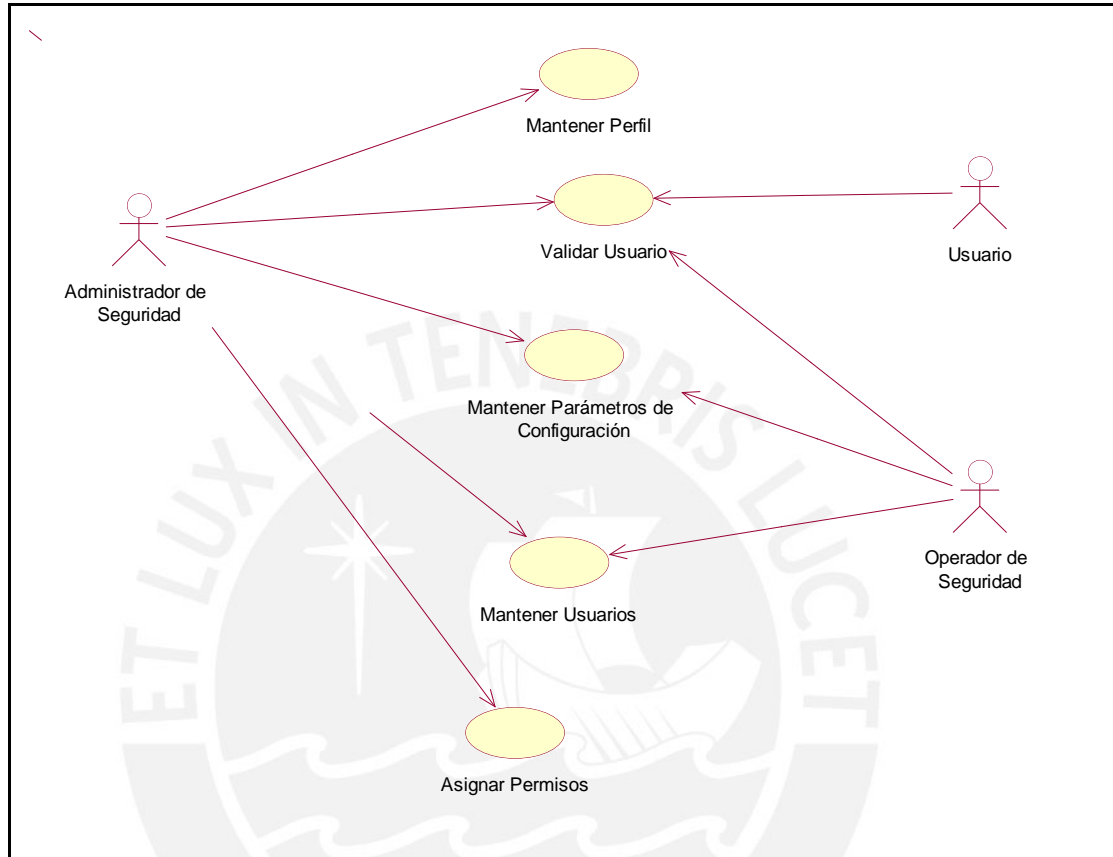


Figura 2.8: Casos de uso del módulo de seguridad

Validar Usuario:

Actor	Administrador de seguridad, usuario.
Descripción	Este caso de uso permite verificar los datos del usuario ingresados y el perfil al cual pertenece.
Pre-condiciones	El usuario debe ingresar en la pantalla principal, su usuario y contraseña.
Post-condiciones	El usuario ingresó al sistema correctamente.

Tabla 2.12: Tabla descriptiva del caso de uso validar usuario

Mantener Usuario:

Actor	Administrador de seguridad, operador de seguridad.
Descripción	Este caso de uso permite crear y modificar los datos del usuario, así como el perfil asignado al mismo.
Pre-condiciones	El usuario debe ser administrador de seguridad u operador del sistema.
Post-condiciones	El usuario fue registrado correctamente.

Tabla 2.13: Tabla descriptiva del caso de uso mantener usuario

2.4. Análisis de la solución

En esta sección se presenta las diversas características y conclusiones que conllevan al estudio de la viabilidad del desarrollo del sistema. Todo esto mediante la elección de las tecnologías a utilizar en el desarrollo, tanto como el lenguaje de programación elegido y el motor de base de datos, mediante una comparación cualitativa entre diferentes productos. Además, de la sustentación económica y las restricciones, suposiciones y dependencias del sistema de logística inversa.

2.4.1. Viabilidad del sistema

El proyecto a desarrollar contempla el análisis, diseño e implementación del sistema de información. Para llevarlo a cabo, se toma en cuenta los aspectos económicos, legales y de tiempo.

En lo económico el proyecto no presenta mayores gastos o inversión, ya que es un desarrollo individual y de investigación, lo cual no implica contratación de personal y recursos de terceros.

En lo legal, se opta por herramientas de desarrollo *Open Source*⁵ como es Java y la base de datos Oracle Express que también posee una licencia sin costo, siempre y cuando sea un fin académico, el cual es el caso. Sin embargo, para un uso real en producción por parte de una empresa se debe tener en cuenta el costo de la licencia del motor de base de datos Oracle Estándar Edition 10g, que se presenta en la sección 2.2.4 Costos y que funciona correctamente con el desarrollo realizado en Oracle Express. Esto debido a la compatibilidad de versiones y a la mayor capacidad y rendimiento por parte de la versión comercial.

Asimismo, se encuentra en el mercado ERP's que comprenden módulos como los de contabilidad y logística, es decir que existen desarrollos en estos temas, documentación al respecto y diseño de los mismos. Por ello, se refleja que su viabilidad es adecuada ya que se le añade el concepto de logística inversa (recuperación y toma de decisiones) a lo que sería un sistema de contabilidad de costos y logística directa tradicional.

Es viable tanto en su nivel de dificultad de desarrollo como el de su aplicación en contextos reales, ya que se perfila como una herramienta de uso continuo en las empresas de producción, sin tener que ser necesariamente de un uso complicado o de algún desarrollo adicional para la integración con un sistema existente.

⁵ *Open Source* o Código Abierto se refiere al software desarrollado y distribuido libremente, lo cual no implica gratuidad en un 100%, ya que si se pretende lucrar o comercializar un sistema desarrollado con estas herramientas, es decir no proporcionar el código fuente o el uso del software fuera de una alcance académico o de investigación; se tendrá que pagar un determinado monto (mucho menor que el software propietario).

Además, se toma en cuenta que en el mercado existen diversos sistemas de información de producción desarrollados bajo el lenguaje de modelado UML, aplicando RUP, con la tecnología Java y el motor de base de datos Oracle. Es decir, se puede deducir que la viabilidad del proyecto está garantizada en base a la experiencia y éxito en otros desarrollos de este tipo de sistemas.

La metodología escogida es iterativa-incremental, basada en un conjunto de entregables o artefactos; por ende se adapta perfectamente al cronograma establecido de la presentación de los avances del proyecto. Asimismo, Java es un lenguaje de programación orientado a objetos, que es en lo que se basa el diseño del sistema. El motor de base de datos garantiza un modelo y funcionamiento, ya que Oracle es la marca líder a nivel mundial, con la mayor capacidad de almacenamiento y un mayor rendimiento en las transacciones.

Ítems de comparación	Bases de Datos			
	Oracle	MySQL	Microsoft SQL Server	PostGres SQL
Soporte Sistema Operativo				
Linux	✓	✓	✗	✓
Windows	✓	✓	✓	✓
Mac OS X	✓	✓	✗	✓
Características Fundamentales				
ACID	✓	✗	✓	✓
Integridad Referencial	✓	✗	✓	✓
Transacciones	✓	✓	✓	✓
UNICODE	✓	✓	✓	✓
Tablas y vistas				
Tabla Temporal	✓	✓	✓	✓
Vista Materializada	✓	✗	✗	✗
Índices				
Árbol R-/R+	✓	✓	✗	✓
Hash	✓	✗	✗	✓
Expresión	✓	✓	✗	✓
Reversa	✓	✗	✗	✗
Mapa de bits	✓	✗	✗	✗
Otros Objetos				
Dominio	✓	✗	✗	✓
Cursor	✓	✓	✓	✓
Trigger	✓	✓	✓	✓
Función	✓	✓	✓	✓
Procedimiento	✓	✓	✓	✓
Rutina Externa	✓	✓	✓	✓

Tabla 2.14: Cuadro comparativo de motores de base de datos [20]

Ítems de comparación	Lenguajes de programación		
	Java	.NET	Power Builder
Soporte Sistema Operativo			
Linux	✓	✗	✗
Windows	✓	✓	✓
Mac OS X	✓	✗	✗
IDE			
Precio bajo (Open Source)	✓	✗	✗
Utilitarios instalados del IDE	✗	✓	✗
Licencia			
Precio Bajo (Open Source)	✓	✗	✗

Tabla 2.15: Cuadro comparativo de lenguajes de programación

De los cuadros anteriores se puede apreciar que el motor de base de datos Oracle es el único que posee todas las funcionalidades puestas a comparación. Paralelamente, el lenguaje de programación Java cumple con todos los ítems de comparación a excepción de uno, siendo también el que posee la mejor evaluación. Por ello, se justifica la elección del motor de base de datos Oracle y el lenguaje de programación Java, siendo los que mejor calificación poseen.

2.4.2. Análisis técnico y económico

En la presente sección se analizará las dependencias y suposiciones a tomar en cuenta para el desarrollo del sistema de logística inversa. Además, se evaluará económicamente, mediante los precios de desarrollo y necesidades de hardware y licencias, el costo del producto software final.

2.4.2.1. Suposiciones y dependencias

Se van a tener las siguientes suposiciones y dependencias para el sistema:

- El sistema se muestra correctamente bajo una resolución de 800x600 pixeles o superior.
- El lenguaje de programación Java trabaja correctamente y tiene compatibilidad con el motor de base de datos Oracle XE.
- Las materias primas a ingresar son únicamente no perecibles y no poseen fecha de vencimiento.
- La unidad de medida de la maquinaria es únicamente hora/máquina.
- La unidad de medida de los empleados es únicamente hora/hombre.
- El terminal cliente tiene instalado un programa para la lectura correcta de archivos en formato pdf.

- El terminal cliente tiene instalado la maquina virtual de java (JVM).

Los requerimientos deseables para una estación de trabajo tipo cliente son:

- Procesador Intel P4 2.4Ghz ~ 3Ghz o Intel DualCore 1.6Ghz.
- 512MB de memoria RAM.
- Disco duro de 40GB.
- Tarjeta de red.

Estos requerimientos son los mínimos aceptables para un correcto funcionamiento del sistema, para características inferiores no se garantiza un adecuado rendimiento.

Los requerimientos deseables para una estación de trabajo tipo servidor son:

- Procesador Intel DualCore Xeon X3065 2.33Ghz.
- 4GB de memoria RAM.
- Disco duro SAS de 360GB a 7200RPM.
- Tarjeta de red.
- La base de datos a usar será Oracle Express.
- Se deberá implementar una red LAN para el funcionamiento del sistema, en el caso se desee usar un número mayor de estaciones.

2.4.2.2. Costos

El presente proyecto tiene por finalidad la aplicación del sistema información de logística inversa de producción, por ello el precio ofrecido a las empresas sería el que se detalla a continuación:

- Número de Horas = 564.3
- Precio por hora = S/.20

Concepto	Comercial
Precio de desarrollo	S/.11 286
Precio de licencia de desarrollo(Java)	S/.0
Precio de base de datos*	S/.52 500 (Oracle Estándar Edition 10g) [32]
Servidor de base de datos*	S/.2 728 (Servidor HP Proliant ML110) [33]

Sistema Operativo*	S/.540 (Microsoft XP Professional) [34]
Total	S/.67 054

Tabla 2.16: Cuadro de costos del proyecto

(*)Costo referencial y sujeto a variación.

2.4.2.3. Restricciones

- El sistema será desarrollado en Java, con base de datos Oracle Express bajo sistema operativo Windows.
- El sistema no abarca el desarrollo interfaces.
- El sistema será desarrollado bajo una arquitectura cliente/servidor.

2.5. Diagrama de clases

El diagrama de clases mostrado en la figura 2.9 será la base para el desarrollo del sistema. Cumple con la solución esperada, ya que sus clases están relacionadas entre sí y con las cardinalidades correctas. En especial, se tiene un mayor énfasis en las clases BEParametroCosto, BEActividad, BEFalla y BEDecision; siendo estas las primordiales para resolver el problema de aplicación de logística inversa, relacionadas con cada producto. Para observar el diccionario de clases, ver en anexos la sección 2. Diccionario de clases de análisis.

A continuación una muestra de la descripción de las clases:

Clase BEParametroCosto	
Clase que representa a los parámetros de costos utilizados en los planes de producción	
Atributo	Descripción
StrCodigo	Identificador del parámetro
strNombre	Nombre del parámetro
strDescripcion	Definición del parámetro
DbiCosto	Costo del parámetro

Tabla 2.17: Clase BEParametroCosto

Clase BEActividad	
Clase que representa a las actividades que componen el plan de producción	
Atributo	Descripción
strCodigo	Identificador de la actividad
strNombre	Nombre de la actividad
strDescripcion	Definición de la actividad
strEstado	Estado de la actividad

Tabla 2.18: Clase BEActividad

Clase BEFase	
Clase que representa las fases que componen el plan de producción	
Atributo	Descripción
strCodigo	Identificador de la fase
strNombre	Nombre de la fase
strDescripcion	Definición de la fase

Tabla 2.19: Clase BEActividad

Clase BEFalla	
Clase que representa a la falla que se presenta en una actividad	
Atributo	Descripción
StrCodigo	Identificador de la falla
strCodigoDecision	Código de la decisión tomada
strDescripcion	Definición de la falla

Tabla 2.20: Clase BEFalla

Clase BEDecision	
Clase que representa a la decisión que se puede tomar ante una falla de una actividad	
Atributo	Descripción
StrCodigo	Identificador de la decisión
strNombre	Nombre de la decisión
strDescripcion	Definición de la decisión
StrEstado	Estado de la decisión

Tabla 2.21: Clase BEDecision

Clase BEProducto	
Clase que representa al producto que fabrica la empresa	
Atributo	Descripción
StrCodigo	Identificador del producto
StrNombre	Nombre del producto
StrDescripcion	Definición del producto
StrEstado	Estado actual del producto
FecRegistro	Fecha de registro del producto

Tabla 2.22: Clase BEProducto

Clase BEOrdenproduccion	
Clase que representa a la orden de producción	
Atributo	Descripción
strCodigo	Identificador de la orden de producción
strDescripcion	Definición de la orden de producción
strEstado	Estado actual de la orden de producción
fecRegistro	Fecha de registro de la orden de producción

Tabla 2.23: Clase BEOrdenProduccion

Clase BEPlanProduccion	
Clase que representa al plan de producción del producto que fabrica la empresa	
Atributo	Descripción
strCodigo	Identificador del plan de producción
strNombre	Nombre del plan de producción
strDescripcion	Definición del plan de producción
strEstado	Estado actual del plan de producción
fecInicio	Fecha de registro del plan de producción
fecFin	Fecha de finalización del plan de producción

Tabla 2.24: Clase BEPlanProduccion

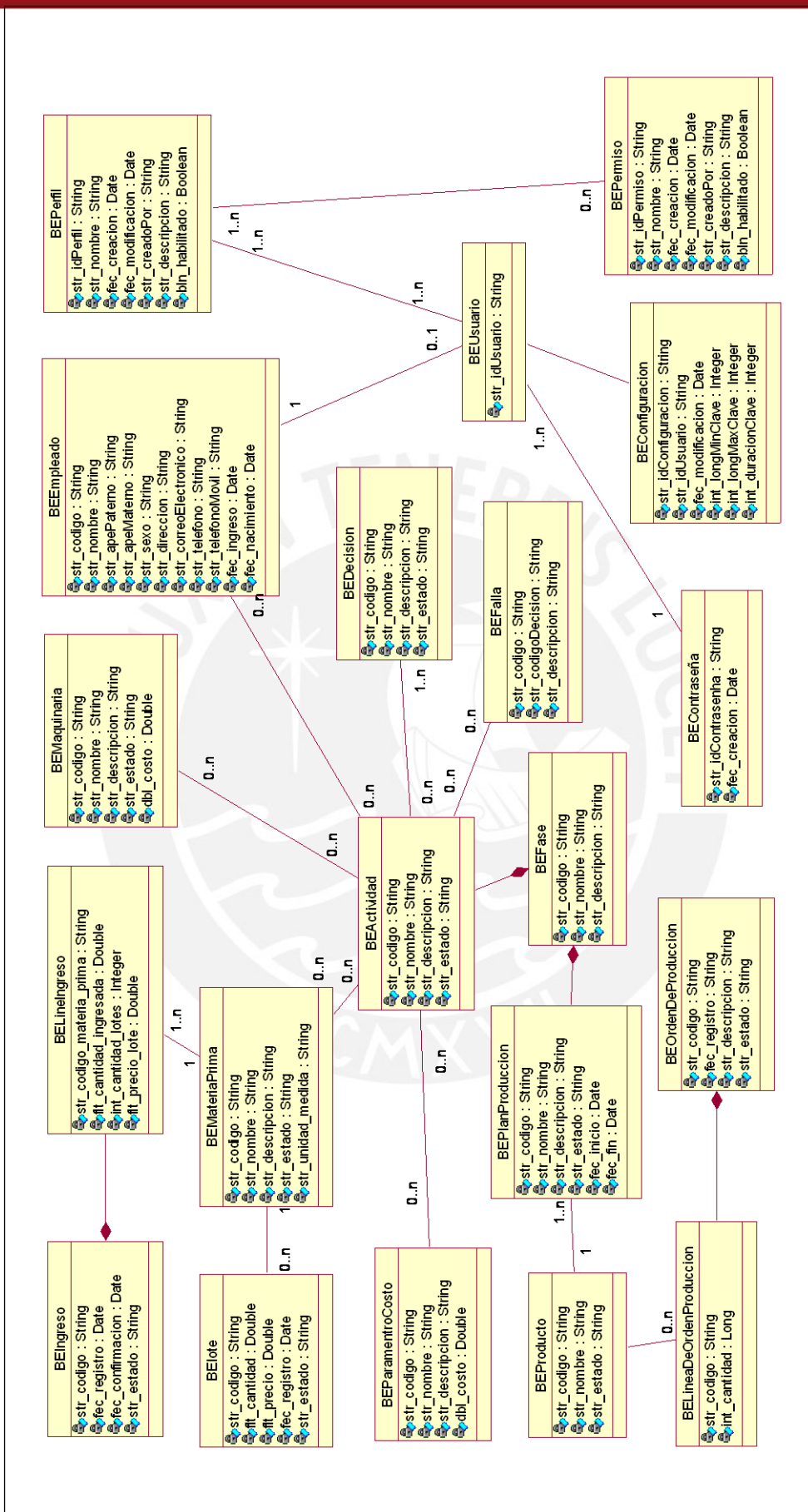


Figura 2.9: Diagrama de clases de análisis

3. Diseño

“El diseño es una representación significativa de ingeniería de algo que se va a construir. Se puede hacer el seguimiento basándose en los requisitos del cliente, y al mismo tiempo la calidad se puede evaluar y cotejar con el conjunto de criterios predefinidos para obtener un diseño <<bueno>>. En el contexto de la ingeniería del software, el diseño se centra en cuatro áreas importantes de interés: datos, arquitectura, interfaces y componentes” [4].

En las siguientes secciones se muestran las áreas de datos, arquitectura y componentes que forman parte del diseño del sistema de logística inversa.

3.1. Arquitectura de la solución

“El diseño arquitectónico representa la estructura de los datos y los componentes del programa que se requieren para construir un sistema basado en computadora. Constituye el estilo arquitectónico que tendrá el sistema, la estructura y las propiedades de los componentes que ese sistema comprende, y las interrelaciones que tienen lugar entre todos los componentes del sistema” [4].

A continuación, se detalla la arquitectura del sistema.

3.1.1. Descripción de la vista lógica

Los diagramas de componentes describen todos los elementos físicos del sistema y sus relaciones con otros elementos. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de la aplicación.

Un componente es una pieza del software reemplazable de un sistema que empaqueta su implementación y un conjunto de interfaces que proporciona su funcionalidad.

Se han definido tres capas dentro de las cuales se definirá todo el sistema cliente/servidor:

- Capa de presentación

Contiene las clases de ventanas y formularios que servirán para que el usuario pueda interactuar con el sistema. Su función es permitir al usuario la creación, modificación, consulta y eliminación de la información a la que éste puede acceder según su nivel de acceso.

- Capa de lógica del negocio

Esta capa incluye las clases que permitirán realizar las operaciones del negocio, así como los procesos y servicios que se manejan en el mismo.

Esta capa depende de la capa acceso a datos debido a que esta última se comunica directamente con la base de datos, confiriendo mayor seguridad a las modificaciones que desee hacer el usuario.

- Capa de acceso a datos

Esta capa incluye las clases para el acceso seguro a la base de datos del sistema, manteniendo una relación directa con ésta. Su función es agregar, modificar, consultar y eliminar – de manera lógica - la información de la base de datos misma, proporcionando así soporte a las capas superiores en cuanto a suministración de información a la cual se refiere.

A continuación se muestra en la figura 3.1 la vista lógica de sistema cliente/servidor.

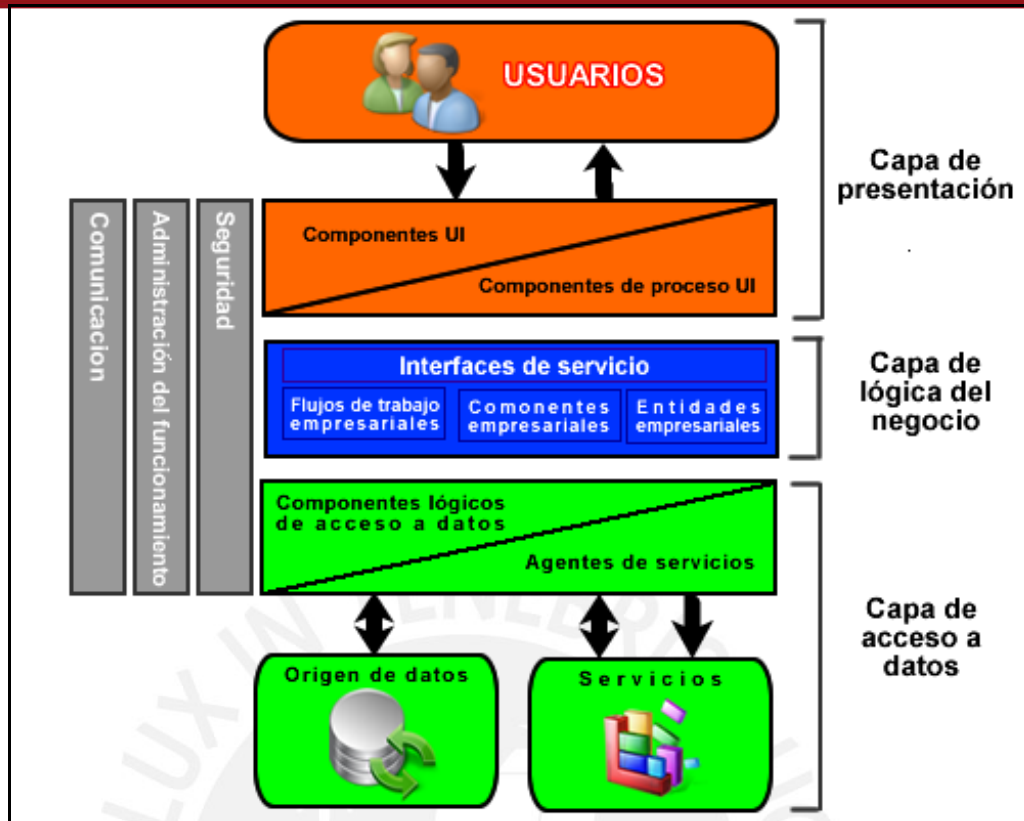


Figura 3.1: Vista lógica del sistema [propia autoría]

Además, para los reportes vía Web se empleará el patrón Modelo – Vista – Controlador (MVC) para lo cual se han definido los siguientes paquetes:

- Vista

En esta sección se desarrolla las páginas necesarias para la implementación de los reportes, es decir los JSP.

- Controlador

En esta sección se desarrolla los servlets que serán necesarios para administrar las peticiones del cliente, derivarlo al Modelo y retornarlo nuevamente a la Vista del sistema.

- Modelo

En esta sección se encuentran las clases *Business Logic* y *Data Access*, los cuales serán las mismas clases de la arquitectura cliente/servidor explicada anteriormente

Asimismo, las clases *Business Entity* de la arquitectura cliente/servidor serán utilizadas en el MVC donde se requieran, ya que forman parte del modelo y Bean de la Vista.

A continuación se muestra en la figura 3.2 un gráfico descriptivo del modelo.

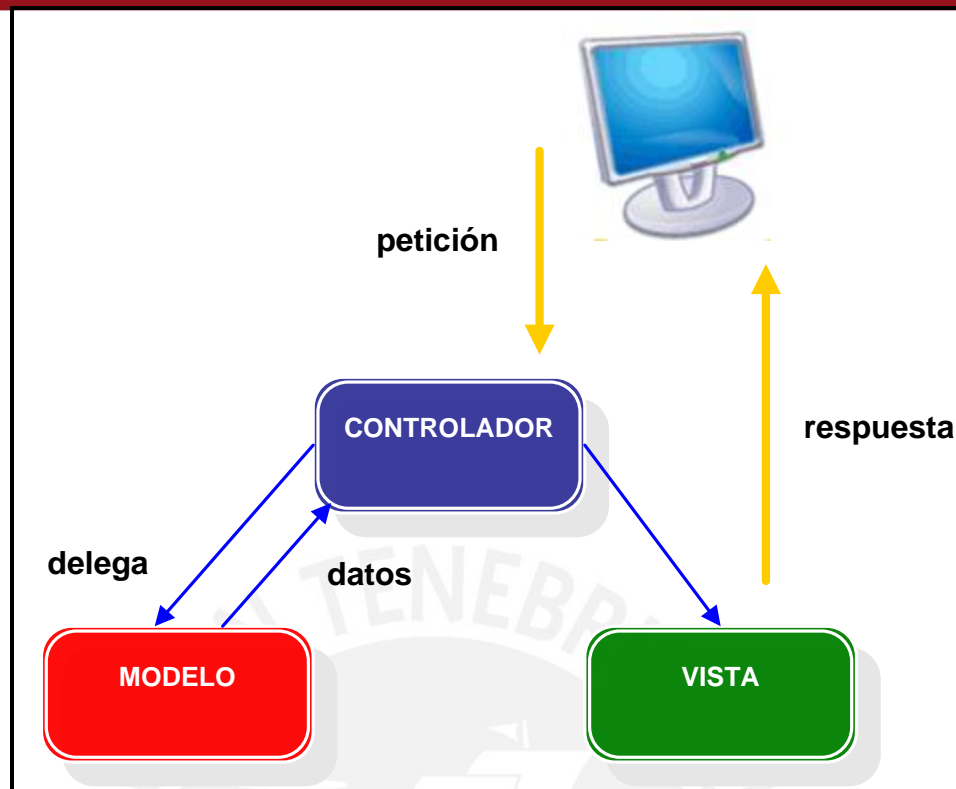


Figura 3.2 Modelo vista controlador [propia autoría]

3.1.2. Paquetes de diseño

3.1.2.1. Paquetes de diseño del sistema cliente / servidor

La realización de todos los casos de uso está comprendida por cuatro módulos principales (maestros, costos y decisiones, reportes y seguridad) y una base de datos.

Cada uno de estos módulos contiene en cuatro tipos de clases (Interfaces de usuario, gestoras, servicios del negocio y las de acceso a base de datos).

- Clases de interfaz de usuario (*Graphics Unified Interface*)

Contiene las clases para cada uno de los formularios que utilizan los actores para comunicarse con el sistema.

- Clases gestoras (*Bussiness Logic*)

Las clases gestoras se encargan del manejo de las clases del negocio. Son una especie de nexo entre la interfaz de usuario y los objetos de negocio. Ejemplos de estas clases son: BLProducto, BLActividad, BLFalla, etc.

- Clases del negocio (*Bussiness Entity*)

Las clases que representan las entidades del negocio. Contienen todos los datos del sistema. Ejemplos de estas clases son: BEProducto, BEPersona, BEUsuario, BEDecision, etc.

- Clases de acceso a base de datos (*Data Access*)

Las clases que proporcionan la comunicación con la base de datos del sistema. Ejemplos de estas clases serian las de DAProducto, DADecision, etc.

A continuación se muestra en la figura 3.3 la vista de capas del sistema.

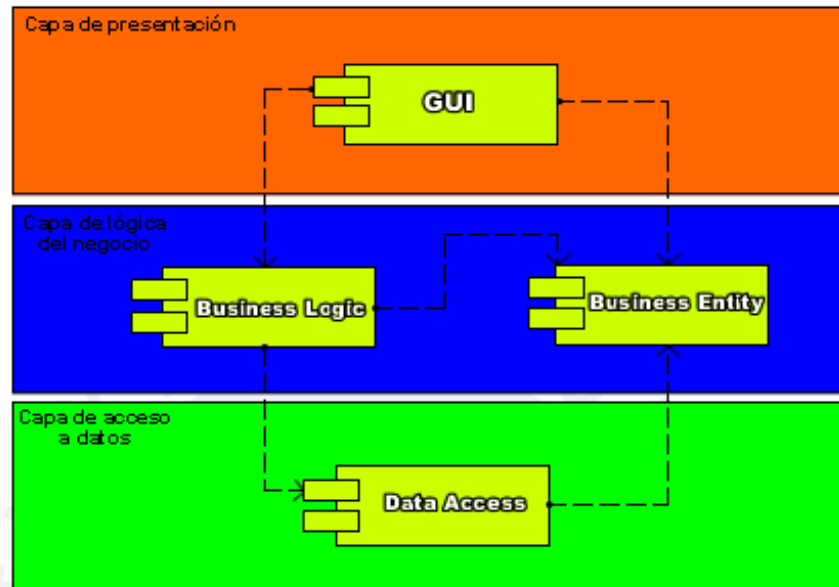


Figura 3.3: Vista de capas [propia autoría]

3.1.2.2. Paquetes de diseño de los reportes Web

A continuación en la figura 3.4 se presenta los paquetes de diseño a utilizar en el desarrollo del módulo de reportes en Web.

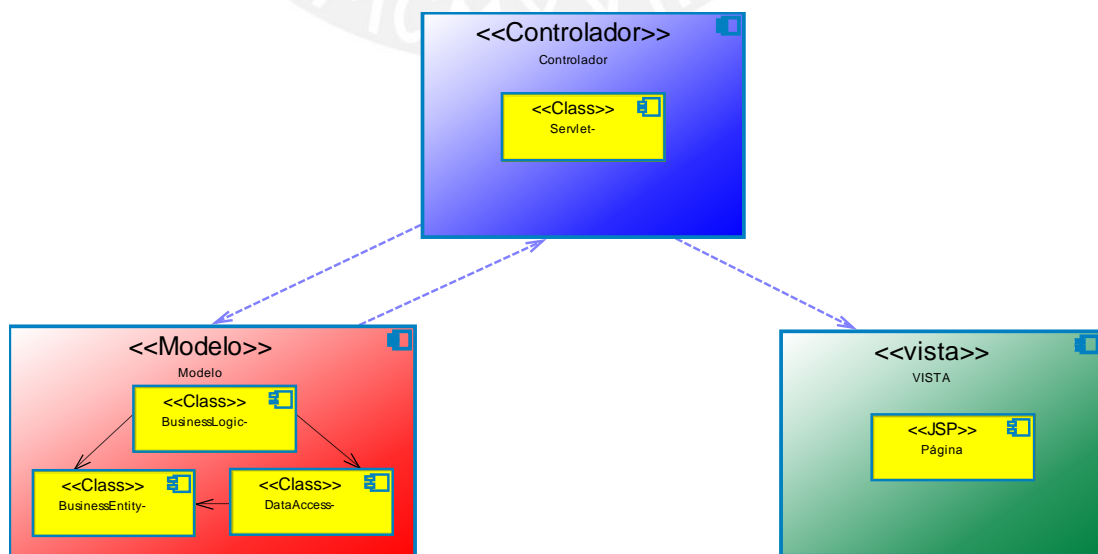


Figura 3.4: Interacción modelo vista controlador [propia autoría]

3.1.3. Vista de despliegue

A continuación se muestra en la figura 3.5 la vista de despliegue del sistema.

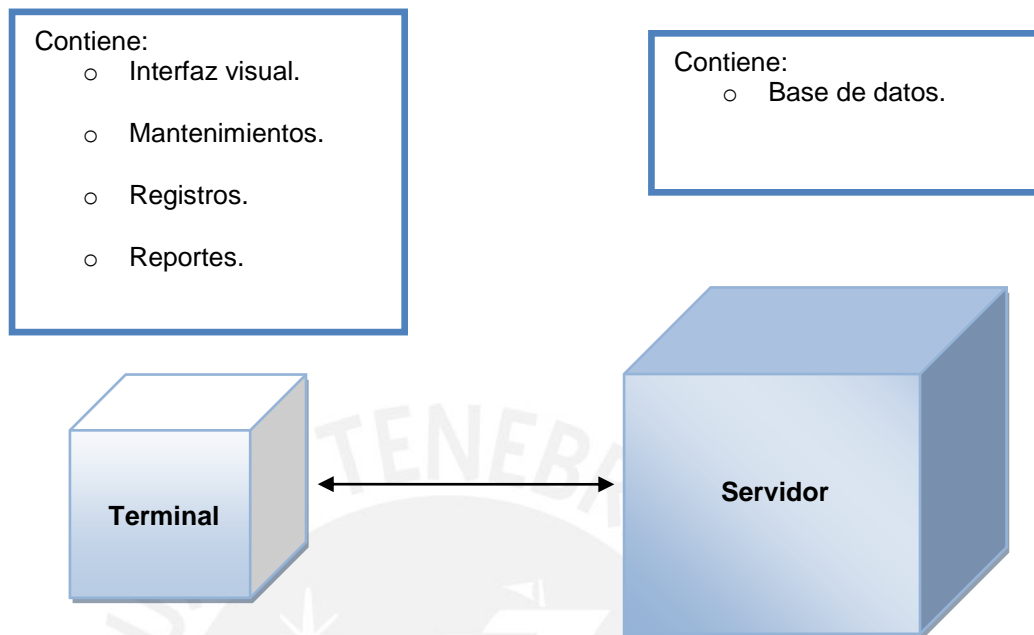


Figura 3.5: Vista de despliegue [propia autoría]

3.1.4. Terminal

Los terminales son las PC's y laptops de los usuarios mediante las cuales se accederá al sistema; es decir, se desplegarán las pantallas de interacción con el usuario, realizarán las consultas, ejecutarán procesos de registro y mantenimientos.

3.1.5. Servidor

El servidor es el lugar físico donde se encontrará la base de datos y al cual todos los terminales tendrán acceso a través de la red LAN.

3.2. Diagramas de Diseño

En esta sección se presentan los diversos diagramas de diseño y su interacción entre las diferentes clases, sin incluir atributos ni métodos.

3.2.1. Interacción de Clases GUI

En la figura 3.4 se aprecian las diversas clases utilizadas en la interfaz de usuario, así como la interacción entre ellas, sin incluir métodos ni atributos.

3.2.2. Interacción de Clases Gestoras y de Acceso a Datos

En la figura 3.5 se aprecian las diversas clases gestoras y de acceso a datos utilizados, así como la interacción entre ellas, sin incluir métodos ni atributos.

3.2.3. Modelo de Datos

En la figura 3.8 se aprecia el modelo de datos del sistema, cuyo diseño satisface los requerimientos de almacenamiento y es necesario para su uso en el servidor

A todo ello, se puede apreciar en la sección 3.2 Vistas de clases de diseño de los anexos, los atributos y métodos de las clases graficadas en los diagramas de interacción de las figuras 3.6 y 3.7.



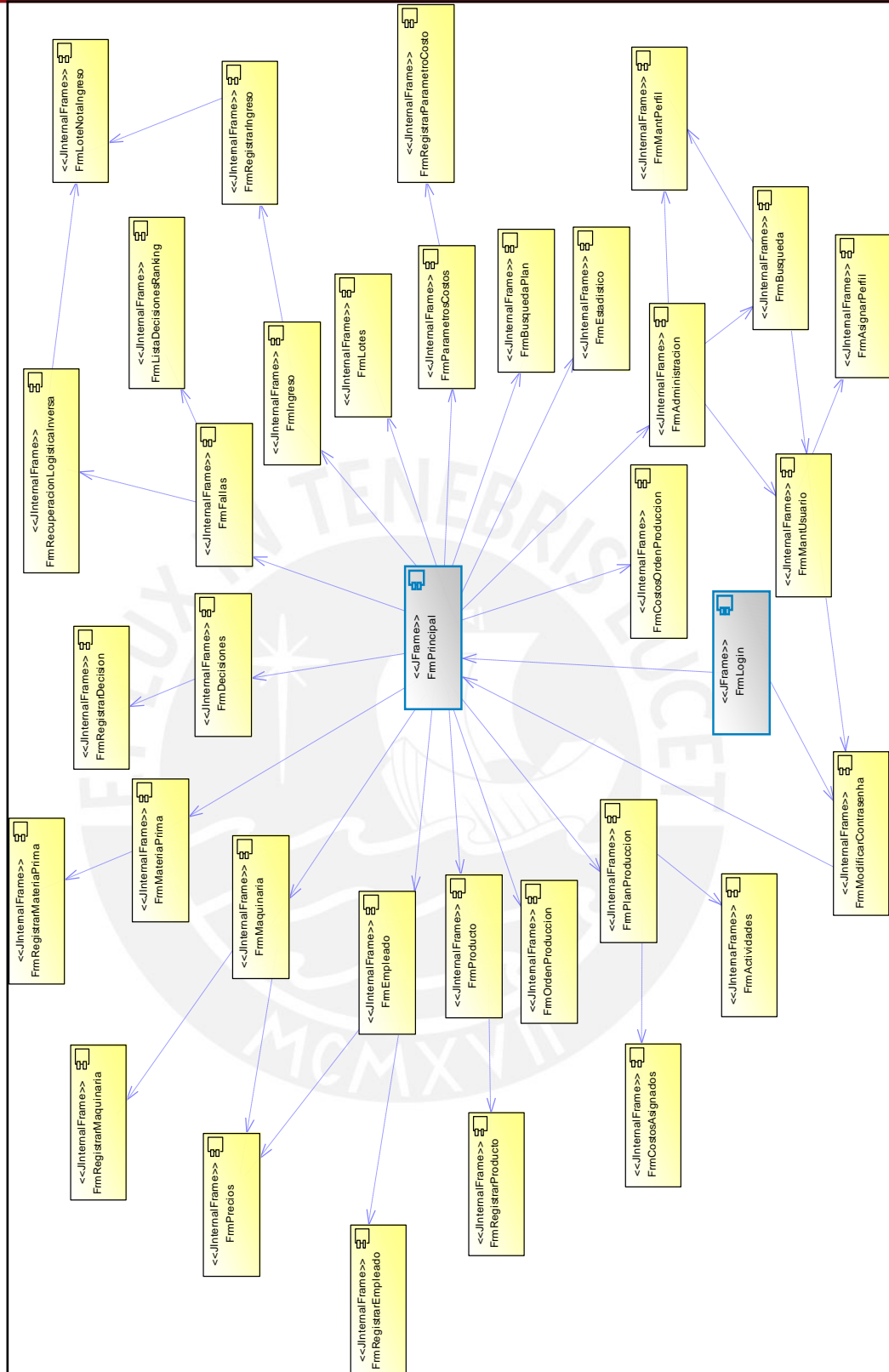


Figura 3.6: Diagrama de clases de diseño, interacción GUI

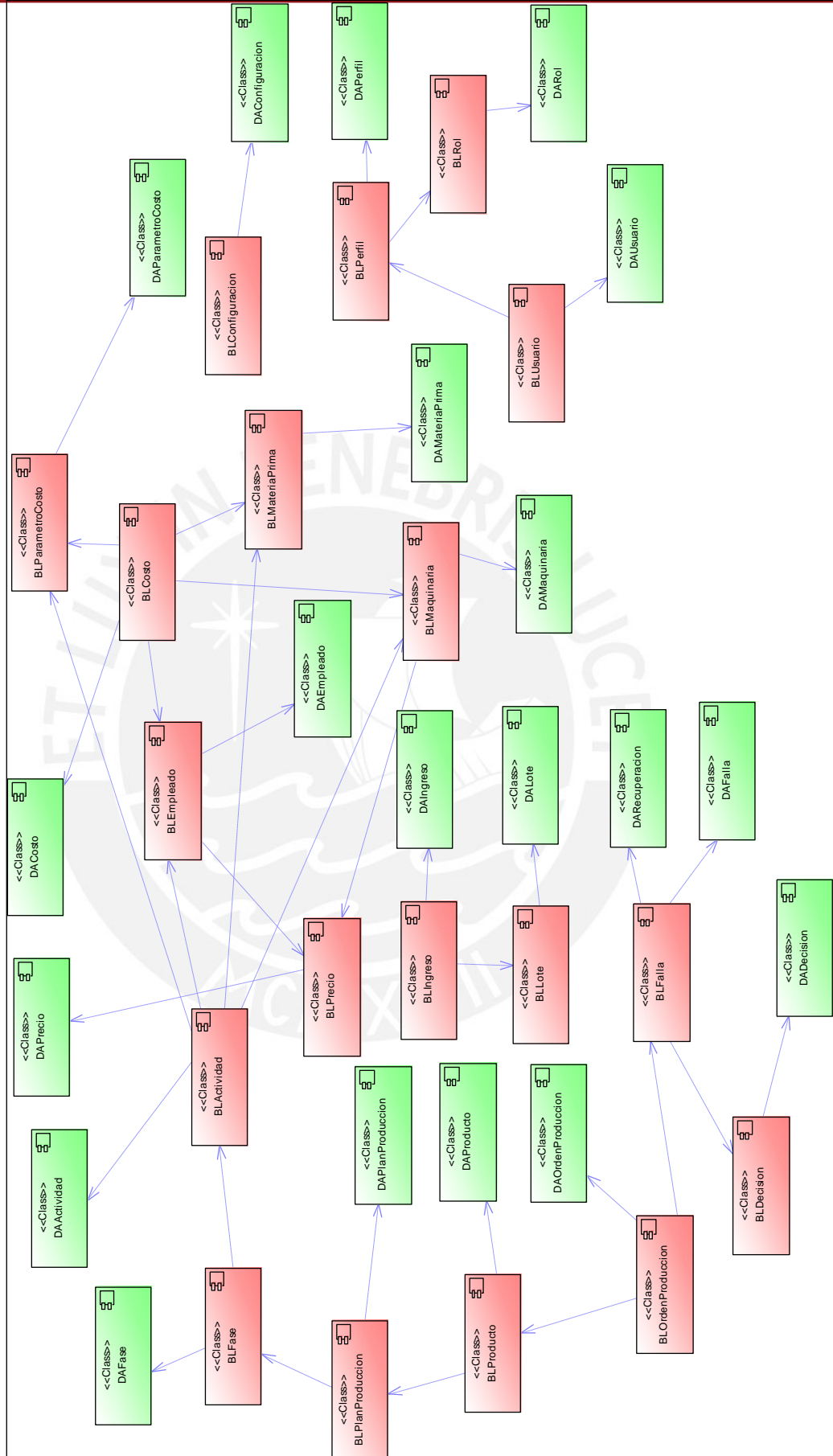


Figura 3.7: Diagrama de clases de diseño, interacción Business Logic – Data Access

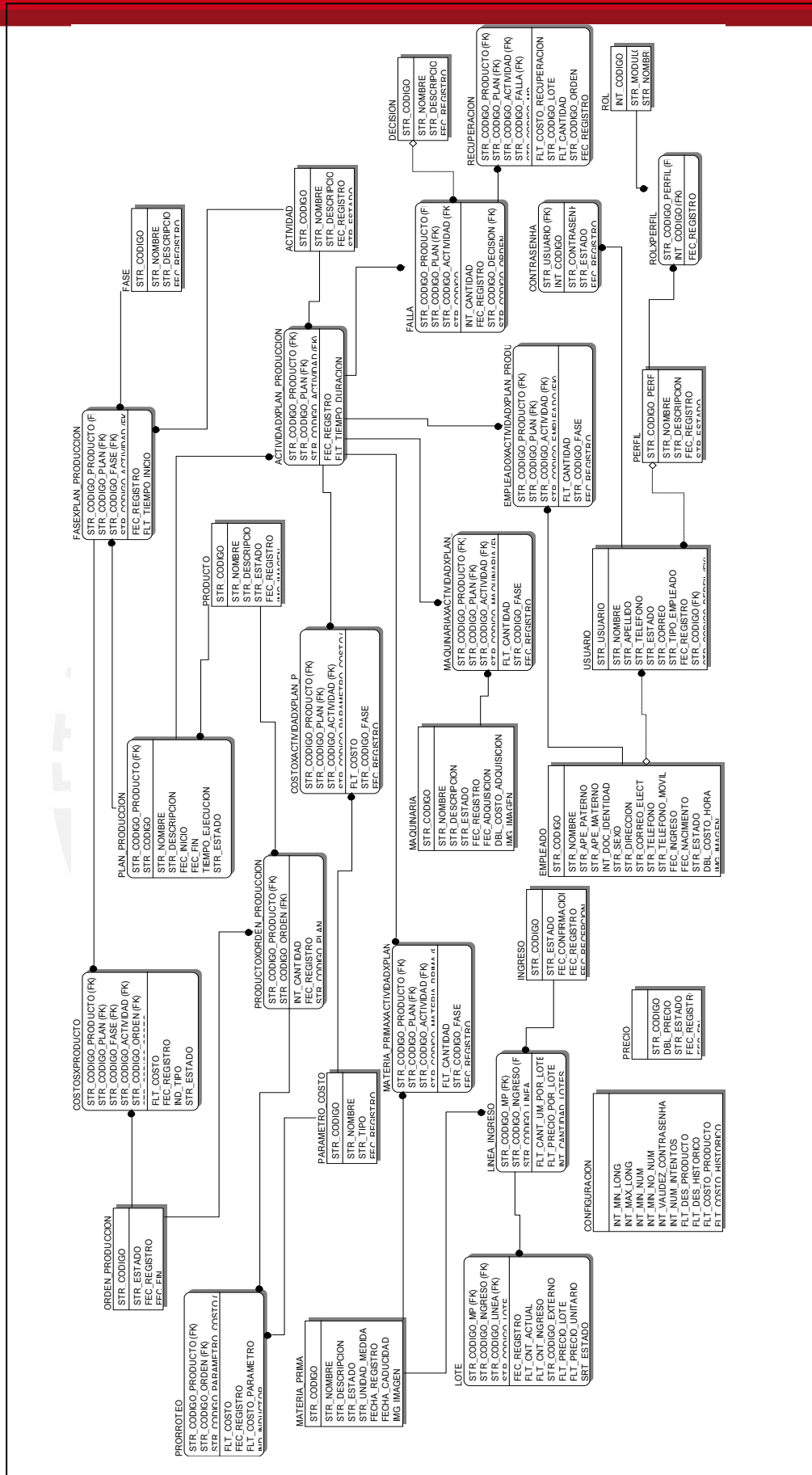


Figura 3.8: Modelo de datos del sistema, ver sección 9 Modelo de Datos.

3.3. Conclusión de la arquitectura

De lo explicado en los puntos anteriores, se deduce que una arquitectura de capas garantiza un adecuado orden en la programación e independencia de cada una de ellas, permitiendo su modificación sin generar grandes cambios y/o gastos en los demás.

Este planteamiento es adecuado para el desarrollo del sistema de logística inversa, ya que en un primer momento el usuario interactúa directamente con su interfaz gráfica, sin conocer detalles interiores del sistema, es decir se busca la simplicidad y fácil uso con esta arquitectura.

Además, permite separar la lógica del negocio en una capa independiente, basándose únicamente en los procesos y reglas propios del sistema de información de logística inversa, permitiendo, si se da el caso, inclusive utilizar otro motor de base de datos y otra interfaz de usuario (Web, PDA, etc.), sin que la lógica del negocio se vea afectada.

Finalmente, se concluye que la arquitectura cliente/servidor es la más adecuada para el desarrollo del sistema de logística inversa, debido a que la cantidad de usuarios del sistema es baja y la información que se maneja es de carácter confidencial de la empresa. Por ello, no resultaría conveniente por ejemplo una aplicación Web de los módulos de Maestros, Costos y Decisiones y Seguridad donde se pueda tener el acceso sin control al sistema, es decir ser más vulnerables a la sustracción de información. Sin embargo, se establece conveniente poder visualizar los reportes por medio de una aplicación Web y con ello el uso de las clases creadas en el sistema cliente/servidor, demostrando su correcto funcionamiento.

3.4. Diseño de interfaz gráfica

Para el diseño de la interfaz gráfica se tendrán en consideración los siguientes criterios:

- Se debe tener en cuenta solo las funcionalidades descritas en el catálogo de requerimientos (ver sección 2.2 Identificación de requerimientos).
- Se manejará una sola ventana principal, que será el contenedor de las ventanas internas, es decir solo se levantará una ventana para el uso del sistema.
- El sistema será desarrollado bajo una resolución de 800x600 píxeles, por ende, ésta será la resolución mínima.

- La ventana principal tendrá una barra de menú, con botones y texto de cada una de las funcionalidades principales del sistema.
- Cada funcionalidad principal a su vez contendrá uno o más accesos a las diferentes ventanas que conforman dichas funcionalidades.

Para el manejo de las variables de los componentes GUI y los estándares establecidos, ver en anexos el documento de estándares de interfaz gráfica.

Muestra de botones:














Funcionalidad de Iconos		
Icono	Botón	Descripción
	Iniciar sesión	Permite el ingreso del usuario y contraseña.
	Nuevo	Permite registrar o abrir la ventana de un nuevo registro.
	Perfil	Permite el mantenimiento de los perfiles.
	Configuración	Permite la configuración de una funcionalidad.
	Reportes	Permite ingresar a las funcionalidades de los reportes.
	Mejor Decisión	Permite realizar el cálculo de la mejor decisión ante una falla.
	Empleado	Permite el ingreso a las funcionalidades del mantenimiento de empleados.
	Materia Prima	Permite el ingreso a las funcionalidades del mantenimiento de materias primas.
	Maquinaria	Permite el ingreso a las funcionalidades del mantenimiento de maquinarias.
	Producto	Permite el ingreso a las funcionalidades que se realizan con los productos.
	Costos	Permite el ingreso a las funcionalidades que se realizan con los costos.
	Logística Inversa	Permite ingresar a las funcionalidades de la aplicación de Logística Inversa.
	Recuperación	Permite acceder a la ventana de recuperación de materia prima.

Tabla 3.1: Cuadro de ejemplo de botones e iconos.

A continuación en las figuras 3.7 y 3.8 se muestran ejemplos de la interfaz gráfica de usuario.



Figura 3.8: Prototipo de ventana 1

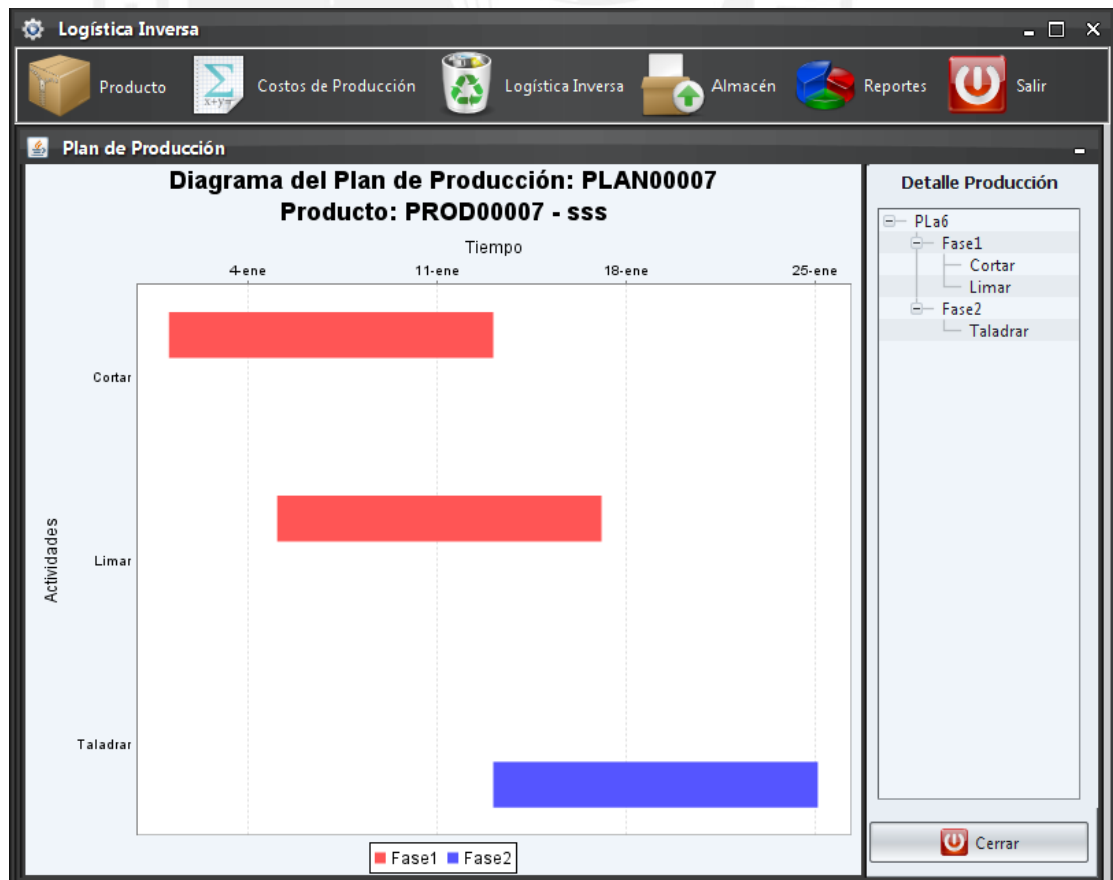


Figura 3.9: Prototipo de ventana 2

4. Construcción y pruebas

En el presente capítulo se presenta la etapa de construcción del sistema de información de logística inversa. Para ello, se muestra el lenguaje de programación elegido, así como el motor de base de datos, el IDE de desarrollo y el administrador de base de datos. Asimismo, se muestra el JAR principal que compone el sistema, los paquetes en que fue dividido y las librerías necesarias.

Finalmente, se muestra el flujo principal del sistema mediante pantallas, los reportes estadísticos, los procedimientos almacenados desarrollados y las pruebas del sistema de logística inversa.

4.1. Construcción

En la sección de construcción se muestra y detalla la construcción del sistema de información, tanto mostrando las utilizadas para la programación como son los paquetes y librerías.

4.1.1. Tecnologías de construcción

La construcción del sistema cliente/servidor se desarrollará bajo las siguientes tecnologías:

Tipo	Tecnología
Lenguaje de programación	Java
IDE	NetBeans 6.1 (ver figura 4.1)
Motor de base de datos	Oracle XE 10g
Administrador de base de datos	Application Express 2.1.0.00.39 (ver figura 4.2)

Tabla 4.1: Cuadro de tecnologías.

La elección del lenguaje de programación y del IDE se basó principalmente en una decisión de índole económica y legal, ya que Java y NetBeans 6.1 poseen licencias que permiten su libre uso, sin ningún desembolso de dinero y es el más adecuado en comparación con otras tecnologías (ver tabla 2.15).

Asimismo, el IDE seleccionado permite un rápido desarrollo del sistema por medio del uso del diseñador de ventanas que posee, contribuyendo de esta manera a reducir ampliamente el tiempo de construcción del sistema. Además, posee un compilador y generador de solución, siendo estos muy útiles tanto en la codificación como en la generación final del sistema.

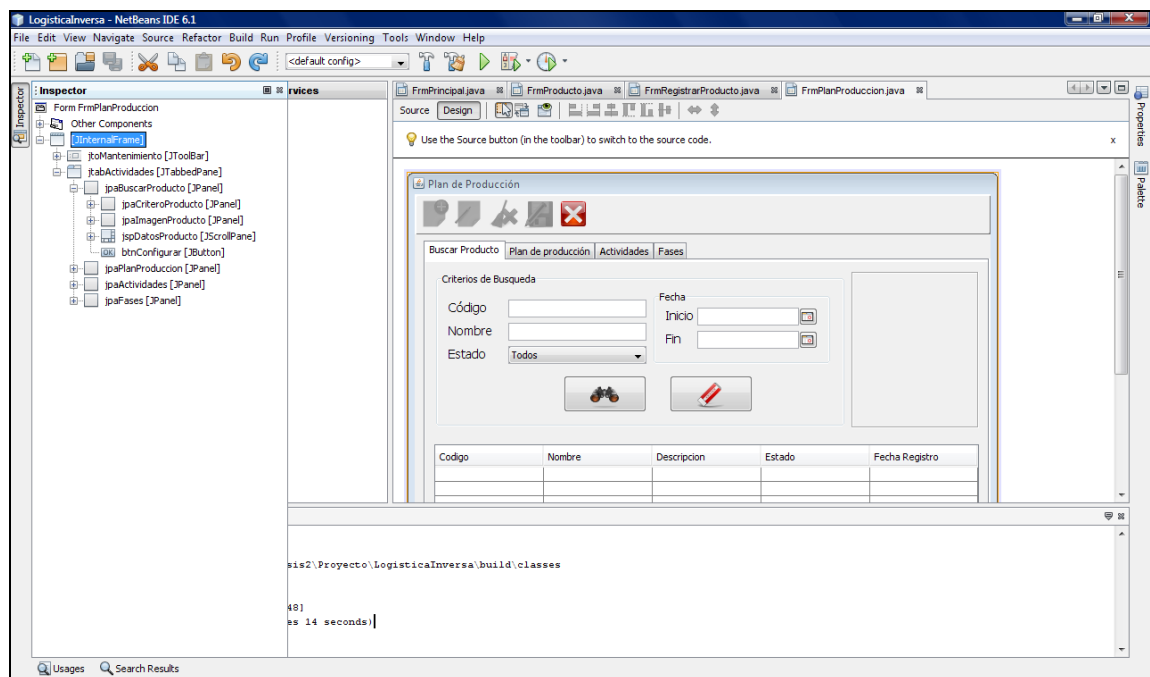


Figura 4.1: Entorno de desarrollo NetBeans 6.1

Adicionalmente, el motor de base de datos elegido es el adecuado para el tipo de sistema, donde los registros tienden a crecer exponencialmente (registro de fallas diarias) y por las funcionalidades que posee en comparación con otros motores de bases de datos (ver tabla 2.13) siendo Oracle el líder a nivel mundial. Específicamente se utilizará la versión académica Oracle XE 10g, con la cual no se desembolsará costo alguno por su uso, cuya licencia no genera costo de adquisición adicional.

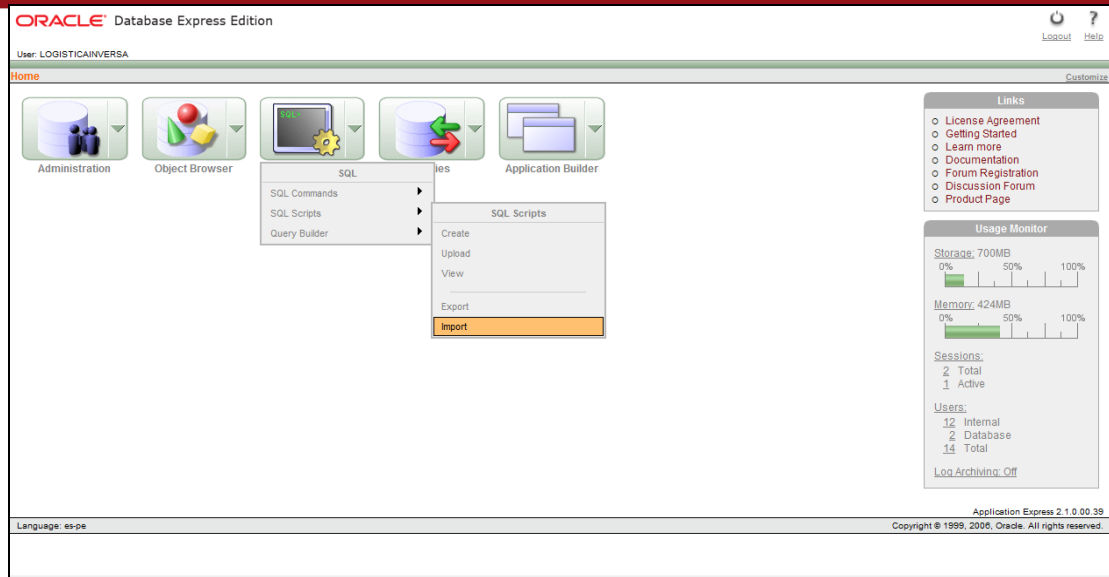


Figura 4.2: Administrador del motor de base de datos Oracle⁶ XE 10g

Finalmente, para la construcción del módulo Web de Reportes se utilizará el servidor Apache Tomcat⁷, ya que este es de libre distribución, para poder desplegar el WAR de la solución. Así mismo su elección es adecuada ya que solo es un módulo de consulta y generación de reporte que no incluyen modificación de datos.

4.1.2. JAR, paquetes y librerías

En la figura 4.3 se presenta el JAR del sistema de logística de inversa, así como sus diferentes paquetes divididos en las tres capas de la arquitectura del sistema (BusinessEntity, BusinessLogic, DataAccess, Frames), los jars utilizados como los del reporte, el driver de conexión del motor de base de datos, el de diagramas JFreeChart, etc. Asimismo, se observa los paquetes de iconos, la clase conexión, las plantillas JRXML de reportes, etc.

Cabe mencionar, que dentro de los paquetes que conforman las capas del sistema (BusinessEntity, BusinessLogic, DataAccess, Frames) a su vez, cada una de ellas posee otros paquetes nombrados por los módulos del sistema (Maestros, CostosDecisiones, Seguridad y Reportes) y dentro de cada uno de ellos se encuentran las clases, archivos con la extensión java, que conforman el sistema de logística inversa.

Para el desarrollo de los reportes y su funcionamiento es necesario de los siguientes Jars: iReport.jar, itext-1.3.1.jar, jxl-2.6.jar, jdt-compiler-3.1.1.jar, xml-apis.jar, log4j-1.2.9.jar, xml-apisext.jar, jcommon-1.0.13.jar, commons-beanutils-

⁶ Oracle es una marca registrada de Oracle Corporation y/o de sus afiliados. Otros nombres pueden ser marcas registradas de sus respectivos dueños.

⁷ Apache y el logo de Apache Tomcat son marcas registradas de Apache Software Foundation.

1.7.jar, commons-collections-2.1.jar. Asimismo, para el diagrama de plan de producción es necesario contar con el jfreechart-1.0.10.jar.

Además, para la apariencia del sistema se utiliza el look and feel Substance, proporcionado mediante el Jar substance.jar y para los campos fechas se utilizará una funcionalidad de calendario mediante el jcalendar-1.3.2.jar y el driver del motor de base de datos mediante el ojdbc14.jar.

A continuación se presenta las diferentes clases que componen cada paquete:

BusinessEntity.Maestros: BEActividad.java, BEEmpleado.java, BEFase.java, BEMaquinaria.java, BEMateriaPrima.java, BEOrdenProduccion.java, BEPlanProduccion.java, BEProducto.java.

BusinessEntity.CostosDecisiones: BEDecision.java, BEIngreso.java, BELinealIngreso.java, BELote.java, BEParametroCosto.java, BEPrecio.java, BERecuperacion.java.

BusinessEntity.Seguridad: BEConfiguracion.java, BEPerfil.java, BERol.java, BEUsuario.java.

BusinessEntity.Reportes: BECostoDirecto.java, BECostoIndirecto.java, BEDetalleOrden.java, BEFrecuencia.java, BERecupMP.java.

BusinessLogic.Maestros: BLActividad.java, BLEmpleadojava, BLFase.java, BLMaquinaria.java, BLMateriaPrima.java, BLOrdenProduccion.java, BLPlanProduccion.java, BLProducto.java.

BusinessLogic.CostosDecisiones: BLCosto.java, BLDecision.java, BLFalla.java, BLIngreso.java, BLLote.java, BLParametroCosto.java, BLPrecio.java.

BusinessLogic.Seguridad: BLConfiguracion.java, BLPerfil.java, BLRol.java, BLUsuario.java.

DataAccess.Maestros: DAActividad.java, DAEmpleado.java, DAFase.java, DAMaquinaria.java, DAMateriaPrima.java, DAOrdenProduccion.java, DAPlanProduccion.java, DAProducto.java.

DataAccess.CostosDecisiones: DACosto.java, DADecision.java, DAFalla.java, DAIngreso.java, DALote.java, DAParametroCosto.java, DAPrecio.java, DARecuperacion.java.

DataAccess.Seguridad: DAConfiguracion.java, DAPerfil.java, DARol.java, DAUsuario.java.

Frames.Maestros: FrmActividades.java, FrmEmpleado.java, FrmMaquinaria.java, FrmMateriaPrima.java, FrmOrdenProduccion.java, FrmPlanProduccion.java, FrmProducto.java, FrmProductosActivos.java, FrmRegistrarEmpleado.java, FrmRegistrarMaquinaria.java, FrmRegistrarMateriaPrima.java, FrmRegistrarProducto.java.

Frames.CostosDecisiones: FrmCostosAsignados.java, FrmDecisiones.java, FrmFallas.java, FrmIngreso.java, FrmListaDecisionesRanking.java, FrmLoteNotaIngreso.java, FrmLotes.java, FrmParametrosCostos.java, FrmPrecios.java, FrmRecuperacionLogisticaInversa.java, FrmRegistrarDecision.java, FrmRegistrarIngreso.java, FrmRegistrarParametroCosto.java.

Frames.Seguridad: FrmAdministracion.java, FrmAsignarPerfil.java, FrmBusqueda.java, FrmLogin.java, FrmMantPerfil.java, FrmMantUsuario.java, FrmModificarContraseña.java.

Frames.Reportes: FrmBusquedaPlan.java, FrmCostosOrdenProduccion.java, FrmEstadistico.java.

Para poder visualizar los atributos y métodos de cada una de las clases que componen el sistema de logística inversa, ver en anexos la sección 3.2. Vista de clases de diseño.

Adicionalmente, se muestran el paquete que contiene las plantillas de los reportes utilizados en el sistema de logística inversa.

Util.JRXML: Costo_Decisiones.jrxml, Costos_Directos.jrxml, Costos_Indirectos.jrxml, Costos_MP.jrxml, Detalle_Orden.jrxml, Frecuencia_Decisiones.jrxml, Frecuencia_Fallas.jrxml.

A continuación se presenta la clase conexión del sistema.

Util.Conexion: Conexión.java.

Otras clases útiles en el sistema fueron las clases necesarias para la utilización de las estructuras de árboles y nodos GUI.

Util.Tree: CheckBoxNode.java, CheckBoxNodeEditor.java, NamedVector.java, CheckBoxNodeRenderer.java.

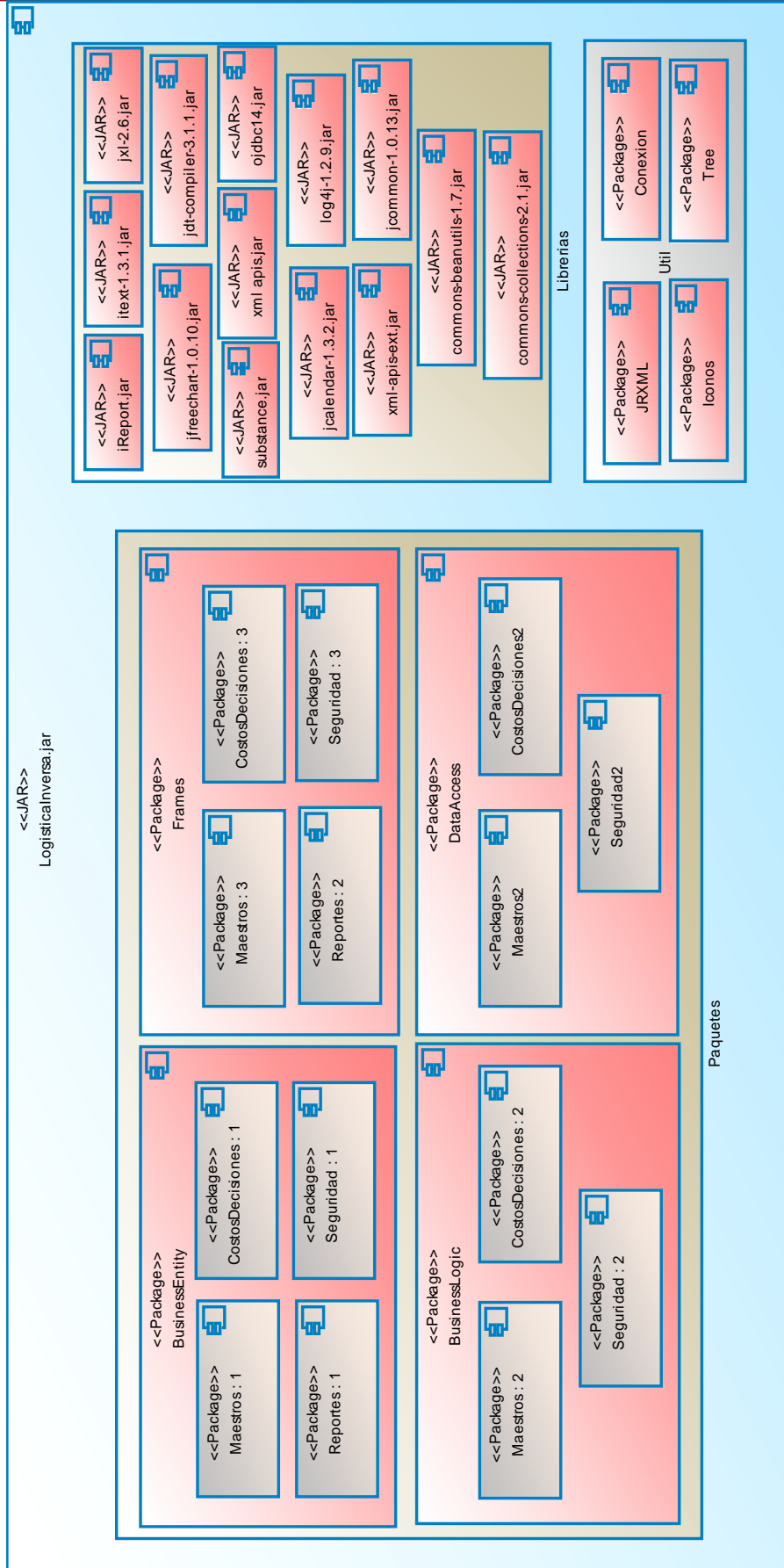


Figura 4.3: JAR del sistema y sus paquetes.

4.1.3. WAR, paquetes y librerías del módulo Web de reportes

En la figura 4.4 se presenta el WAR del módulo de Reportes del sistema de logística de inversa, como sus diferentes paquetes que implementan el patrón Modelo – Vista – Controlador.

Los paquetes BusinessEntity, BusinessLogic y Data Access son los mismos de los explicados en la sección anterior 4.1.2, utilizando solo aquellas clases que son necesarias para la implementación del módulo de Reportes y las clases del módulo de Seguridad que permiten la validación del usuario en el sistema. Además, se implementan las clases de los servlets y los JSP necesarios para la aplicación Web.

A continuación se detalla las clases utilizadas:

BusinessEntity.Maestros: BEOrdenProduccion.java, BEProducto.java.

BusinessEntity.Reportes: BECostoDirecto.java, BECostoIndirecto.java, BEDetalleOrden.java, BEFrecuencia.java, BERecupMP.java.

BusinessLogic.CostosDecisiones: BLDecision.java, BLFalla.java.

BusinessLogic.Maestros: BLOrdenProduccion.java, BLProducto.java.

BusinessLogic.Seguridad: BLUsuario.java.

Servlet.Reportes: SerReportes.java

Servlet.Seguridad: SerSeguridad.java

WebContent: jspLogin.jsp, jspReportes.jsp, jspEstadistico.jsp, jspOrdenProduccion.jsp.

Adicionalmente se detalla las plantillas de los reportes y los archivos javascript utilizados:

WebContent.JRXML: Costo_Decisiones.jrxml, Costos_Directos.jrxml, Costos_Indirectos.jrxml, Costos_MP.jrxml, Detalle_Orden.jrxml, Frecuencia_Decisiones.jrxml, Frecuencia_Fallas.jrxml.

WebContent.JS: jsValidación.js, jsCal2.js.

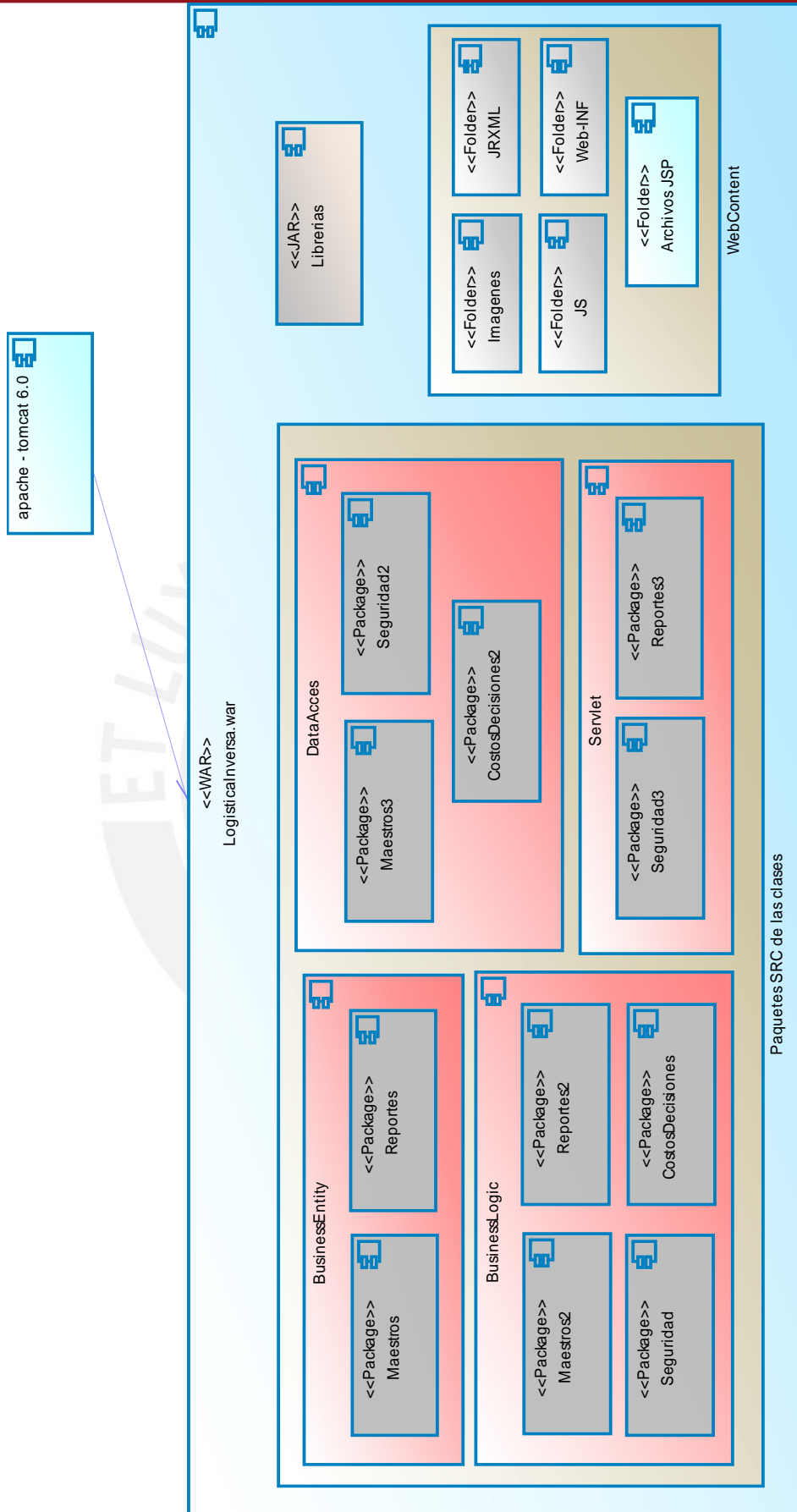


Figura 4.4: WAR del módulo de reportes y sus paquetes

4.1.4. Flujo principal del sistema

Se presentará, mediante pantallas extraídas de la etapa de construcción, el flujo básico del sistema. Se empieza con el registro de un plan de producción, luego se procede a la asignación de costos para el plan ingresado, después se registra una orden de producción y finalmente se registra una falla y la decisión tomada para esta última.

A continuación el flujo de principal del sistema:



Figura 4.5: Ventana de validación de usuario.

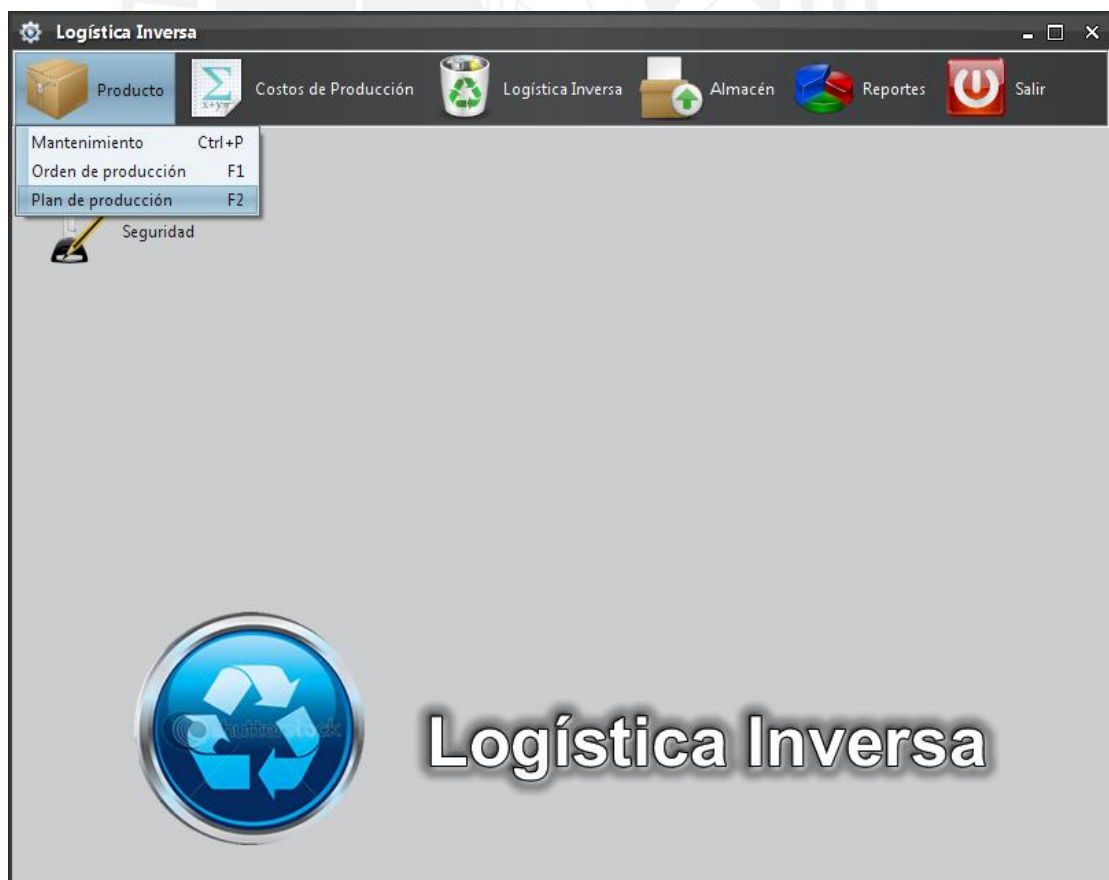


Figura 4.6: Prototipo de la ventana principal e ingreso a la funcionalidad del plan de producción.

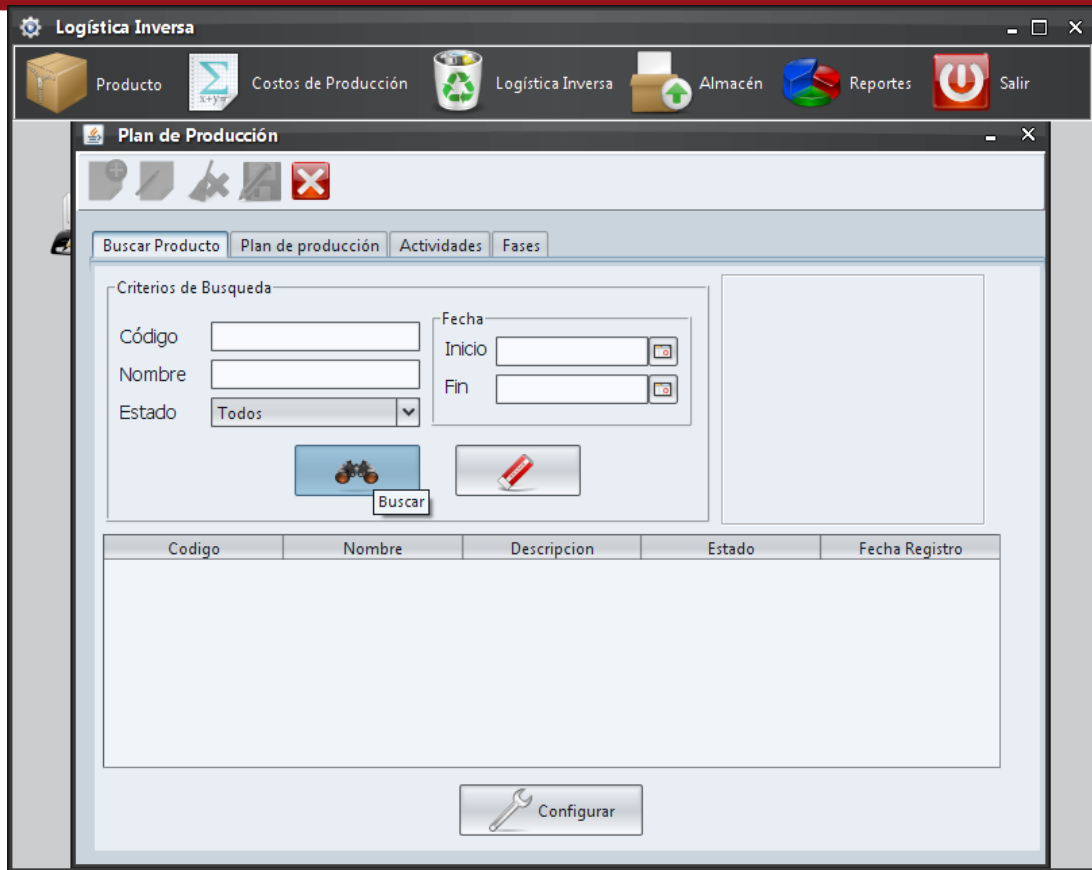


Figura 4.7: Prototipo de la ventana de búsqueda de producto en el plan de producción.

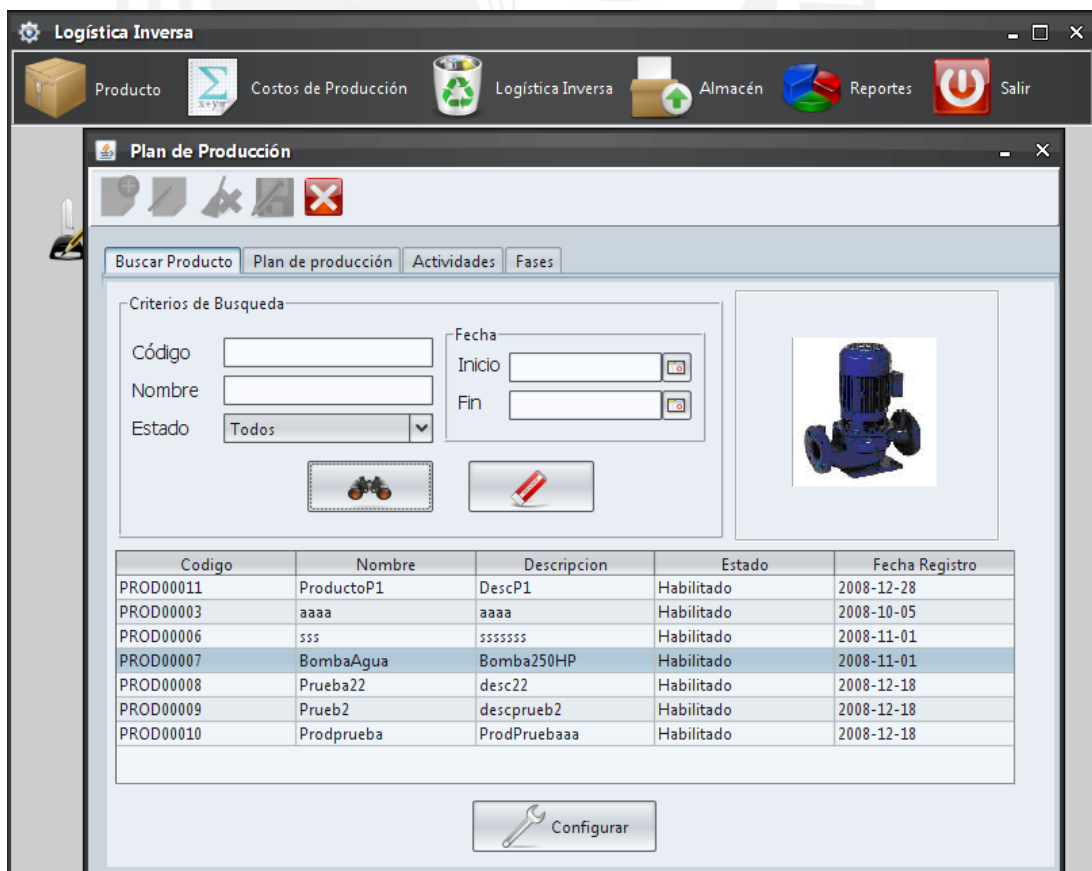


Figura 4.8: Prototipo del resultado de búsqueda de producto en el plan de producción.

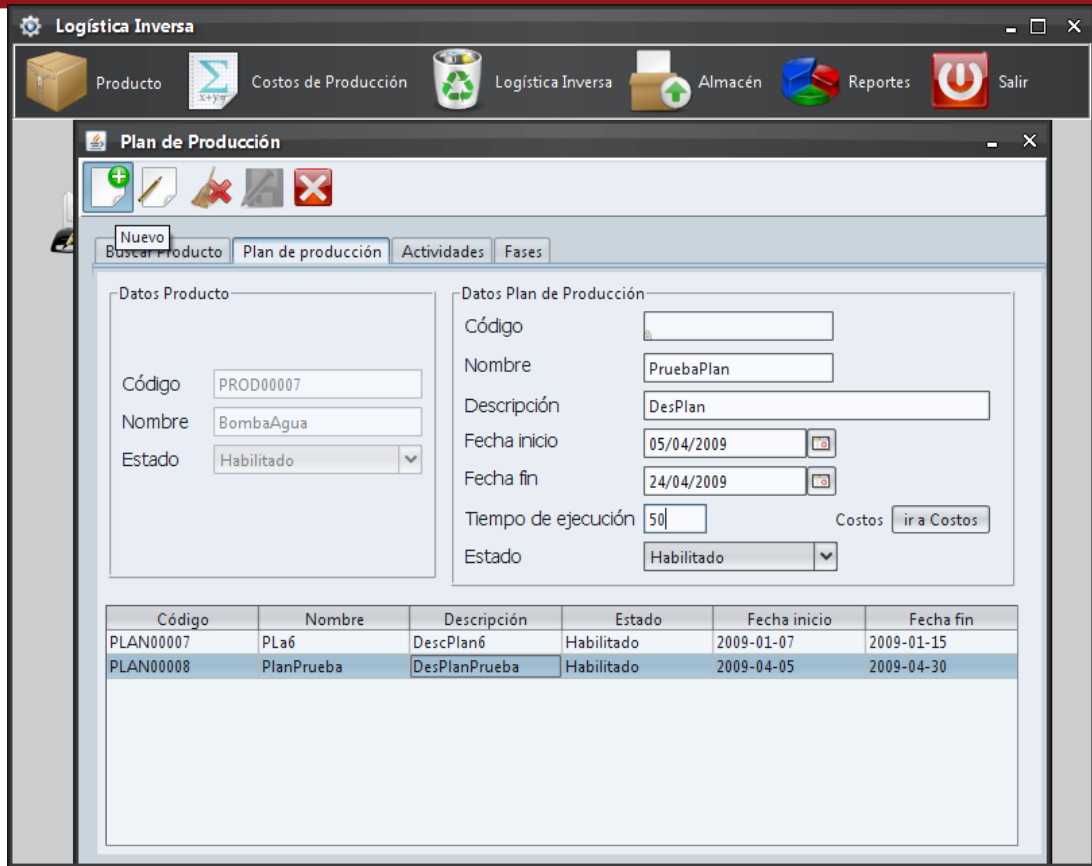


Figura 4.9: Prototipo del mantenimiento de plan de producción.

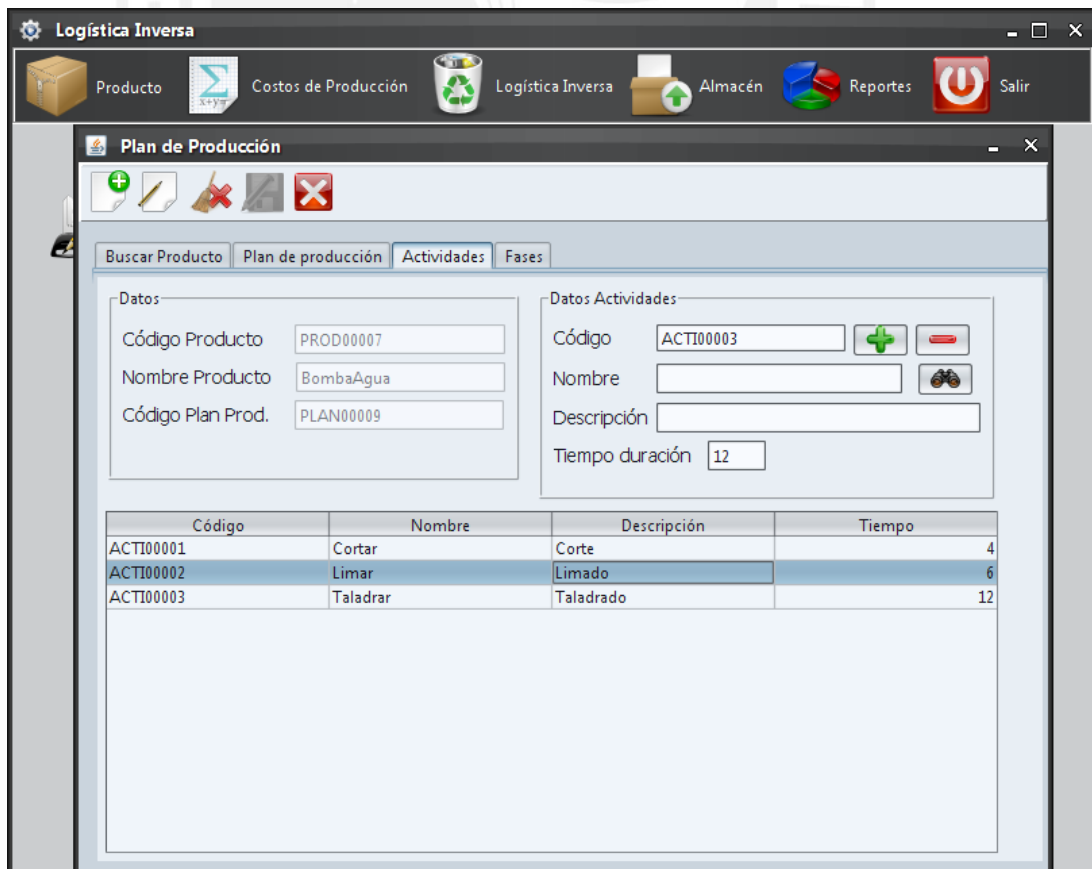


Figura 4.10: Prototipo del mantenimiento de actividades

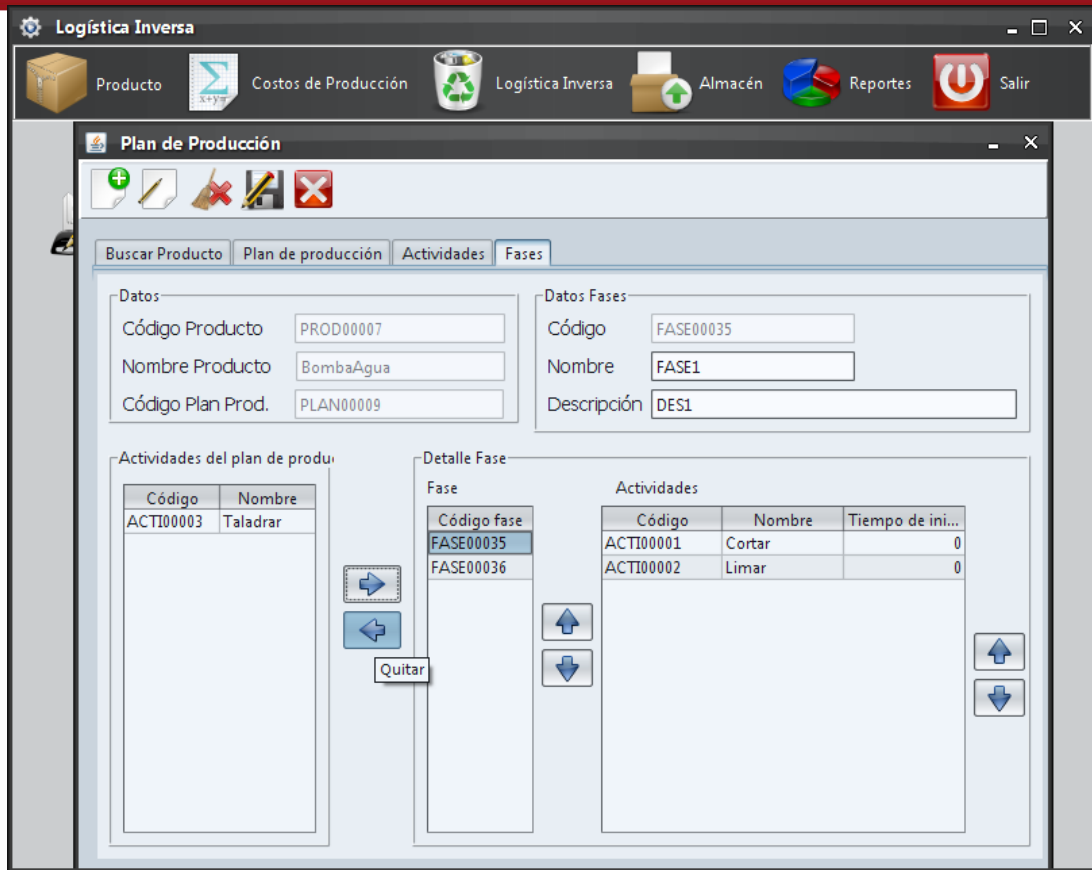


Figura 4.11: Prototipo del mantenimiento de fases.

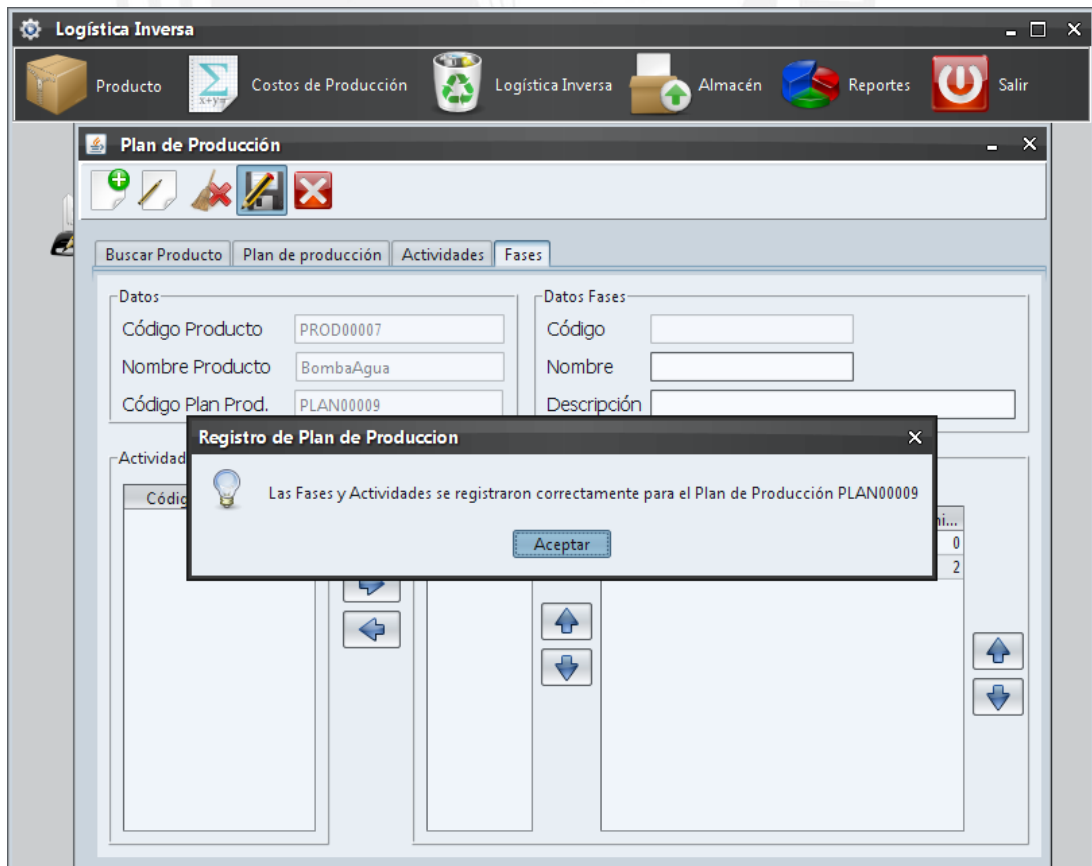


Figura 4.12: Prototipo del registro satisfactorio de un plan de producción.

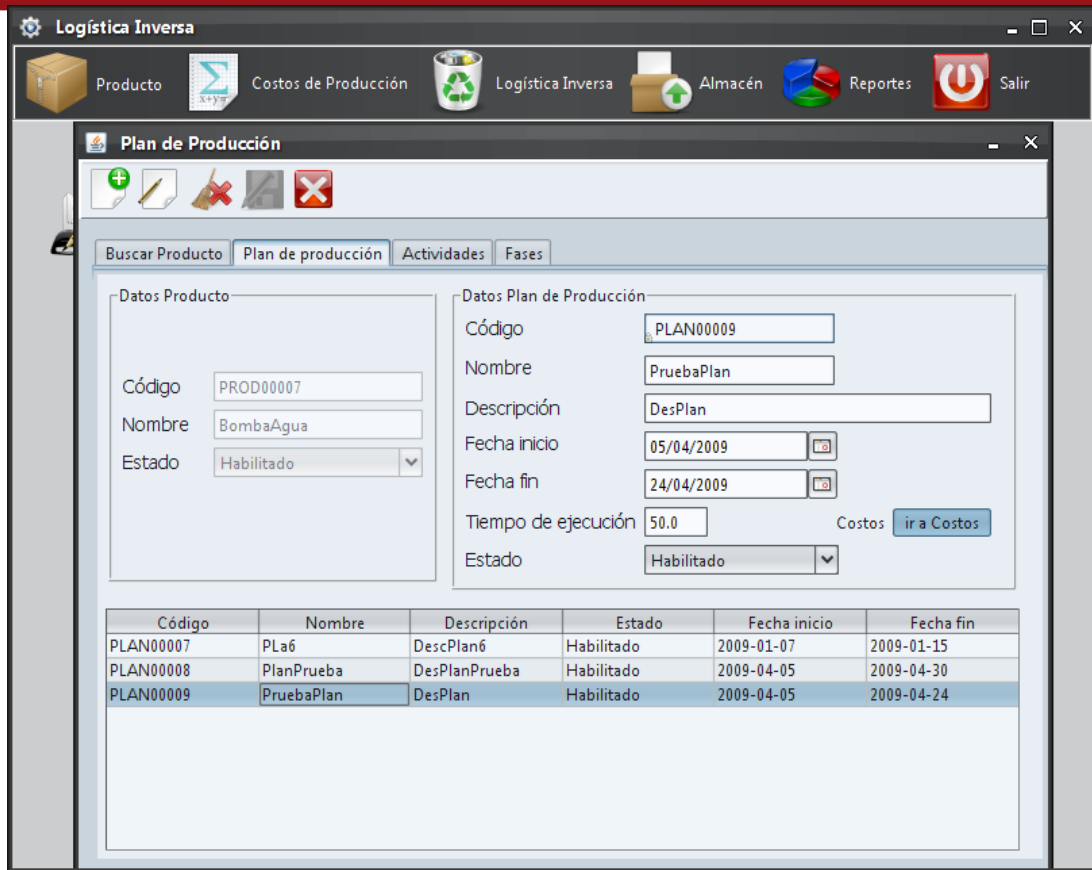


Figura 4.13: Prototipo del acceso a la asignación de costos de un plan de producción.

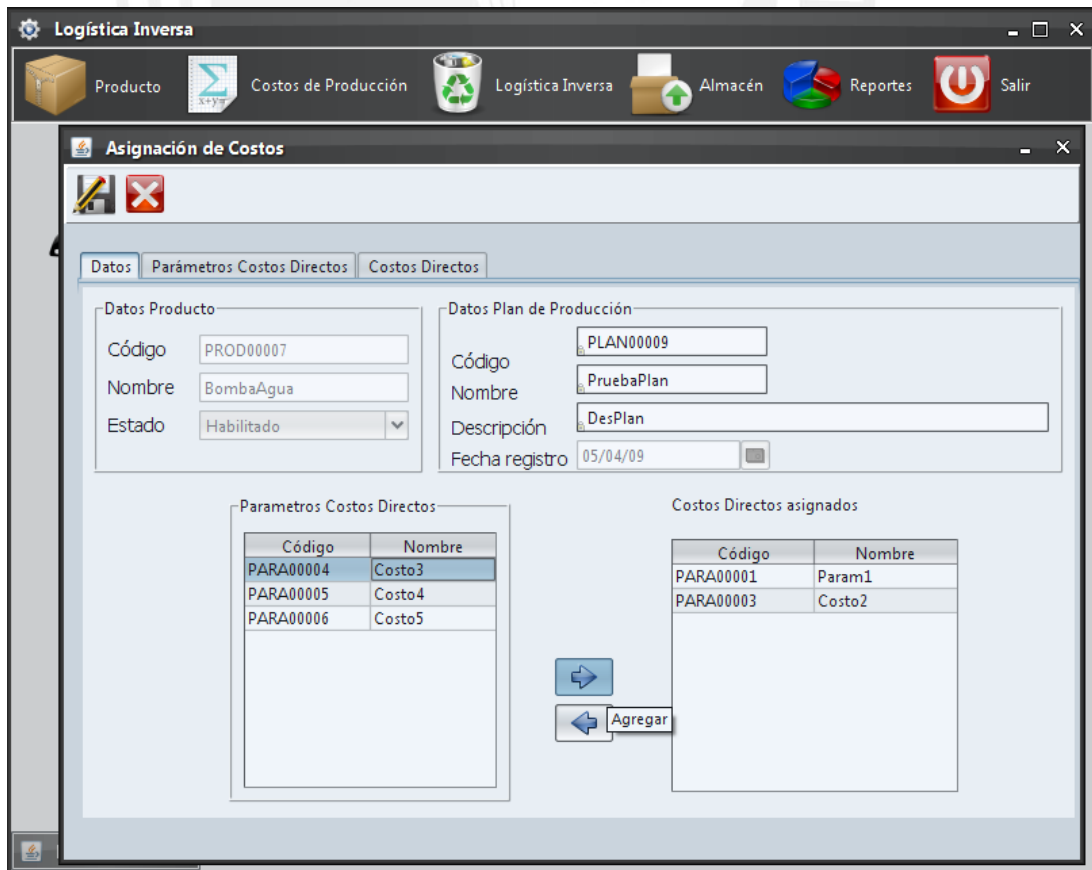


Figura 4.14: Prototipo que muestra la asignación de costos de costos directos.

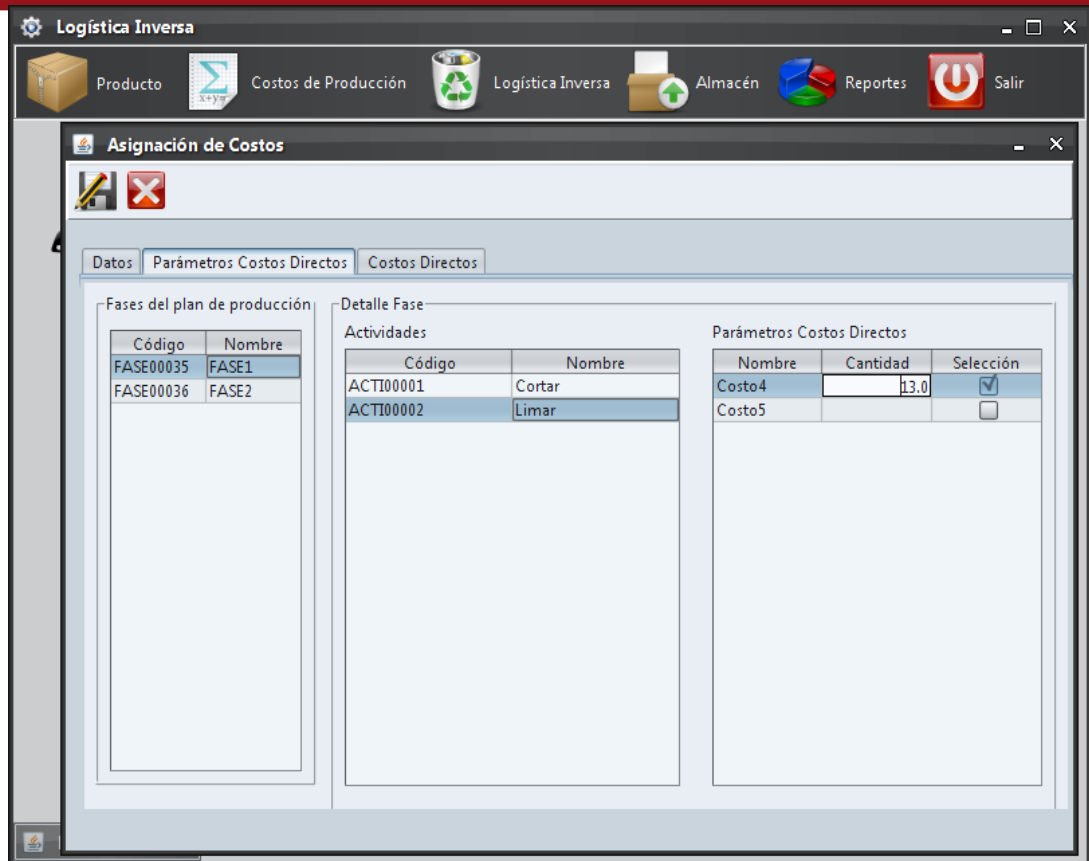


Figura 4.15: Prototipo que muestra la asignación del valor monetario a los costos asignados.

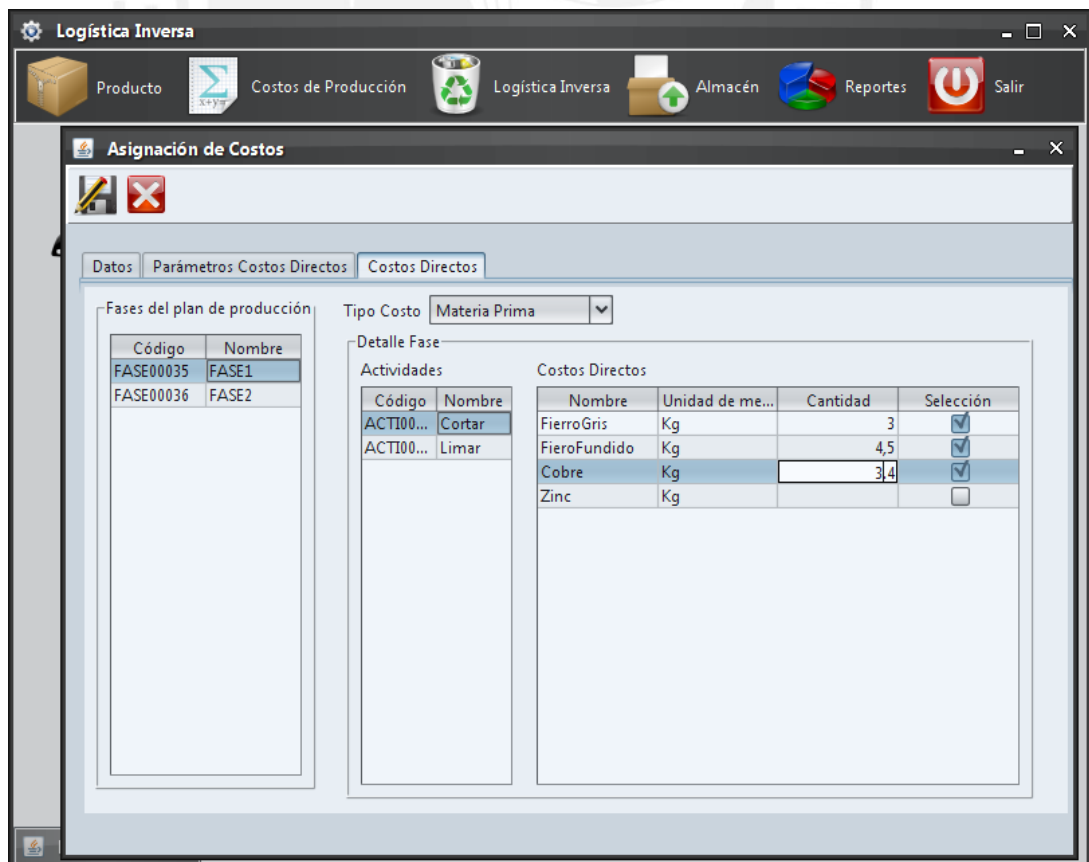


Figura 4.16: Prototipo que muestra la asignación del valor monetario a los costos obligatorios (materia prima, mano de obra, maquinaria).

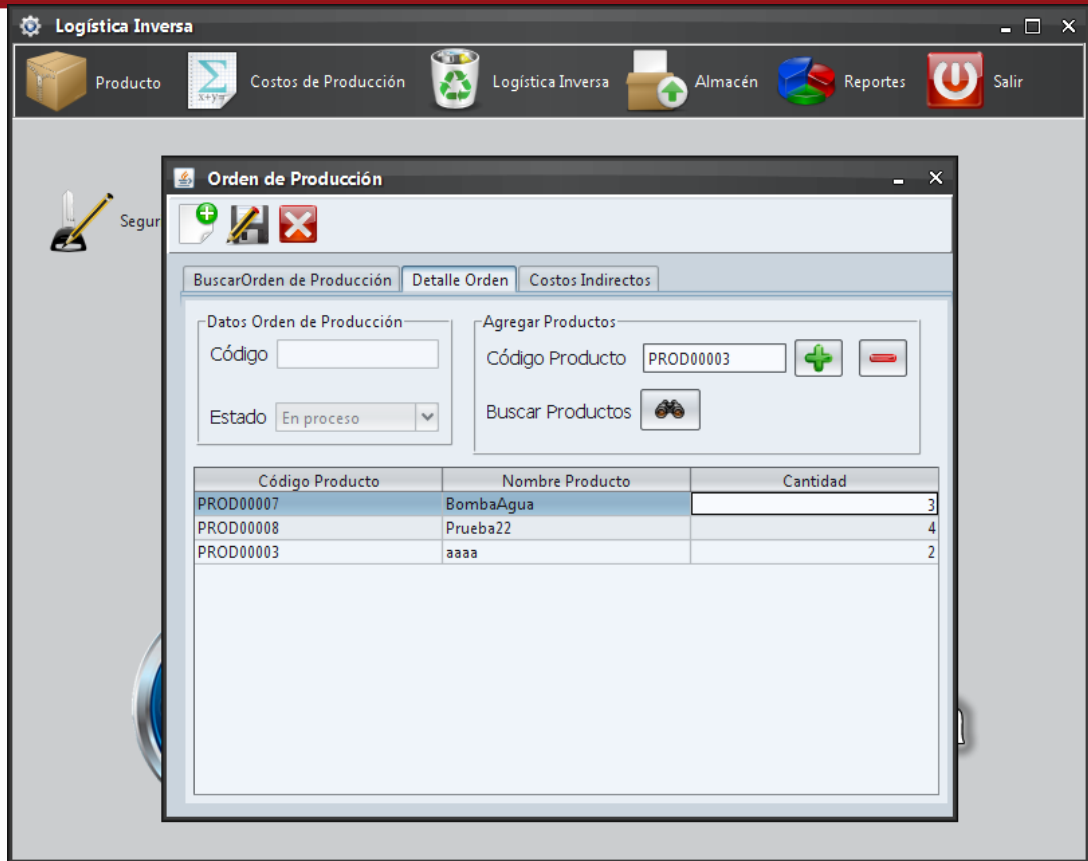


Figura 4.17: Prototipo que muestra la asignación de productos y cantidades de una orden de producción.

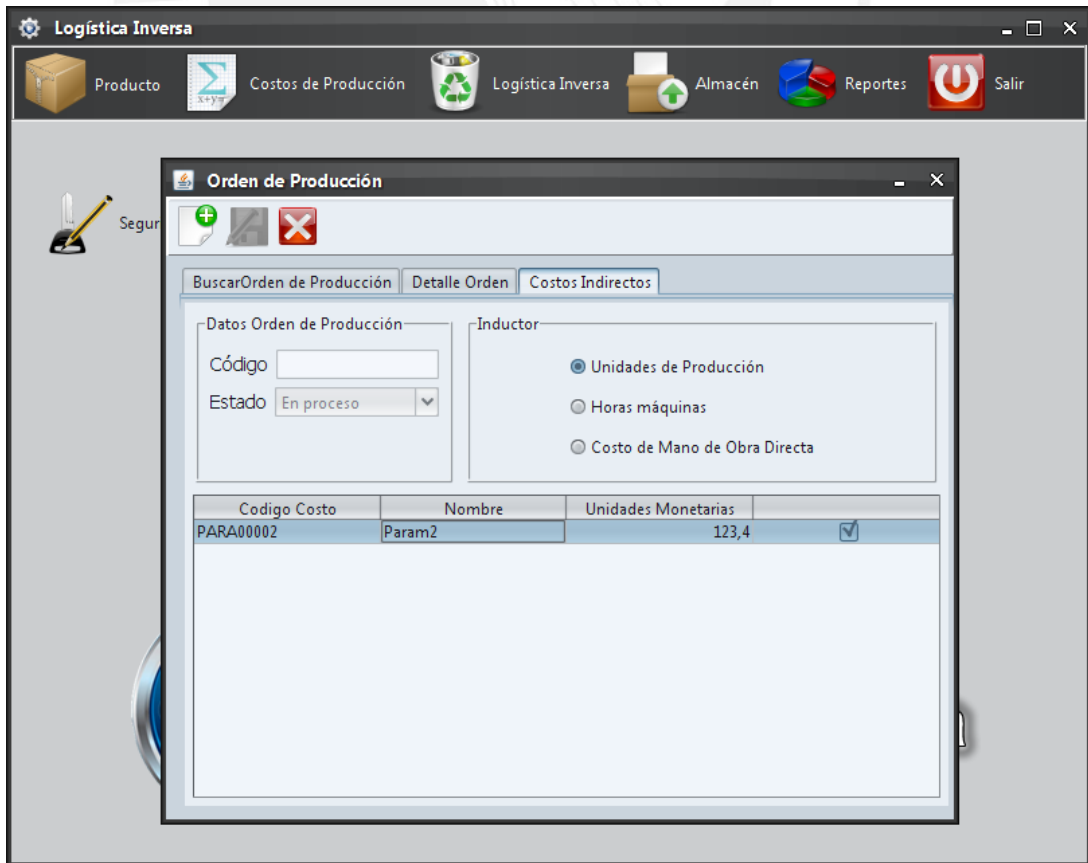


Figura 4.18: Prototipo de asignación del costo indirecto e inductor.

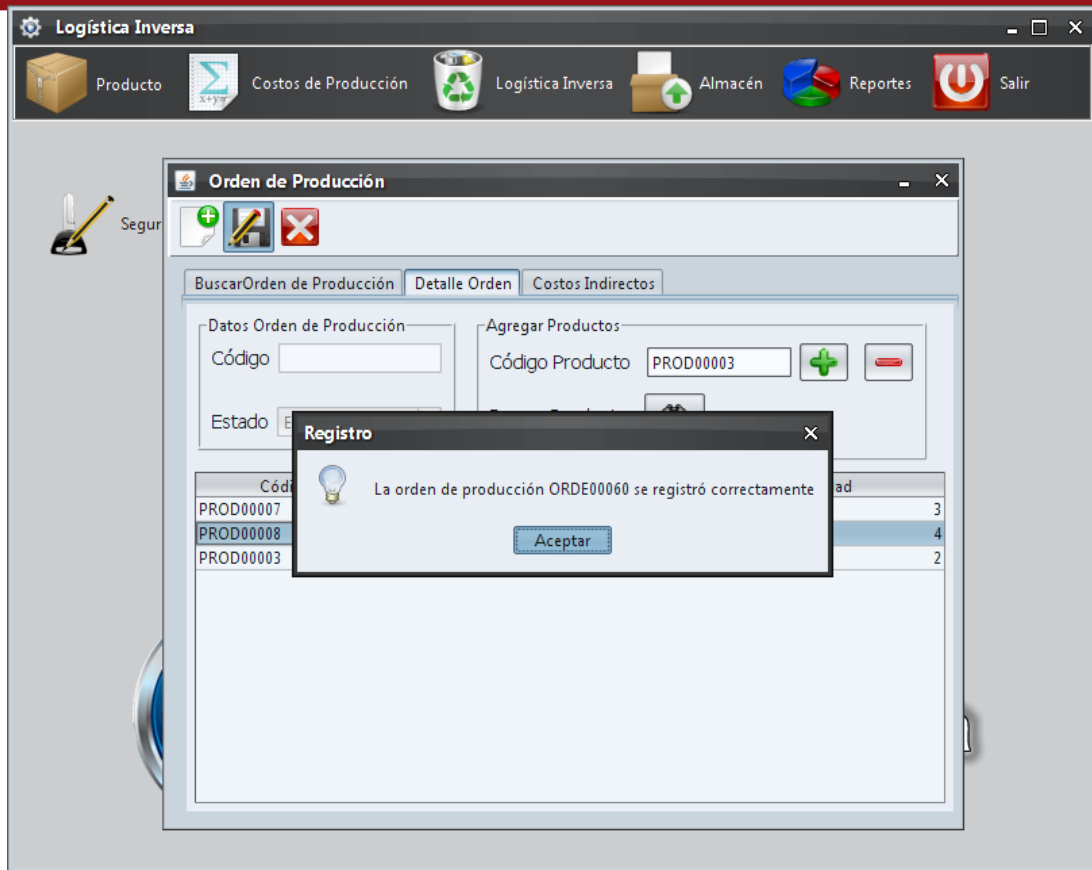


Figura 4.19: Prototipo del registro satisfactorio de una orden de producción

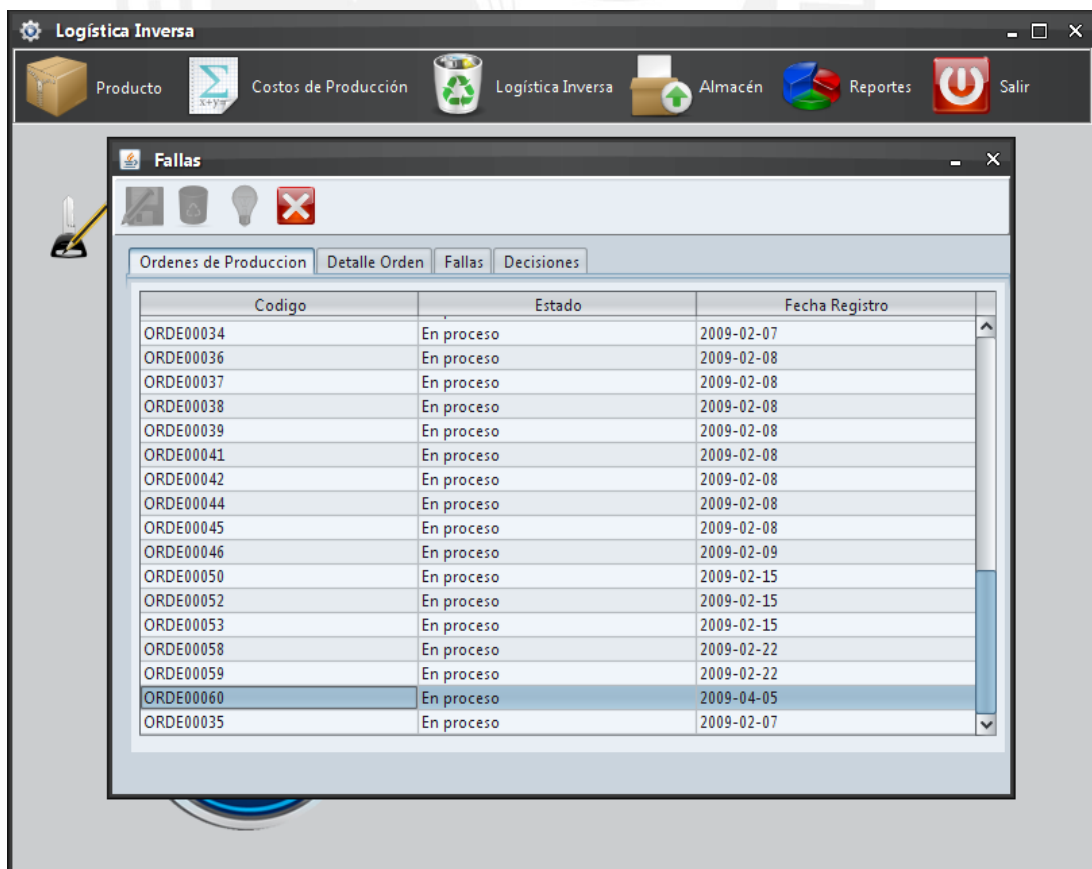


Figura 4.20: Prototipo que muestra los planes de producción en proceso

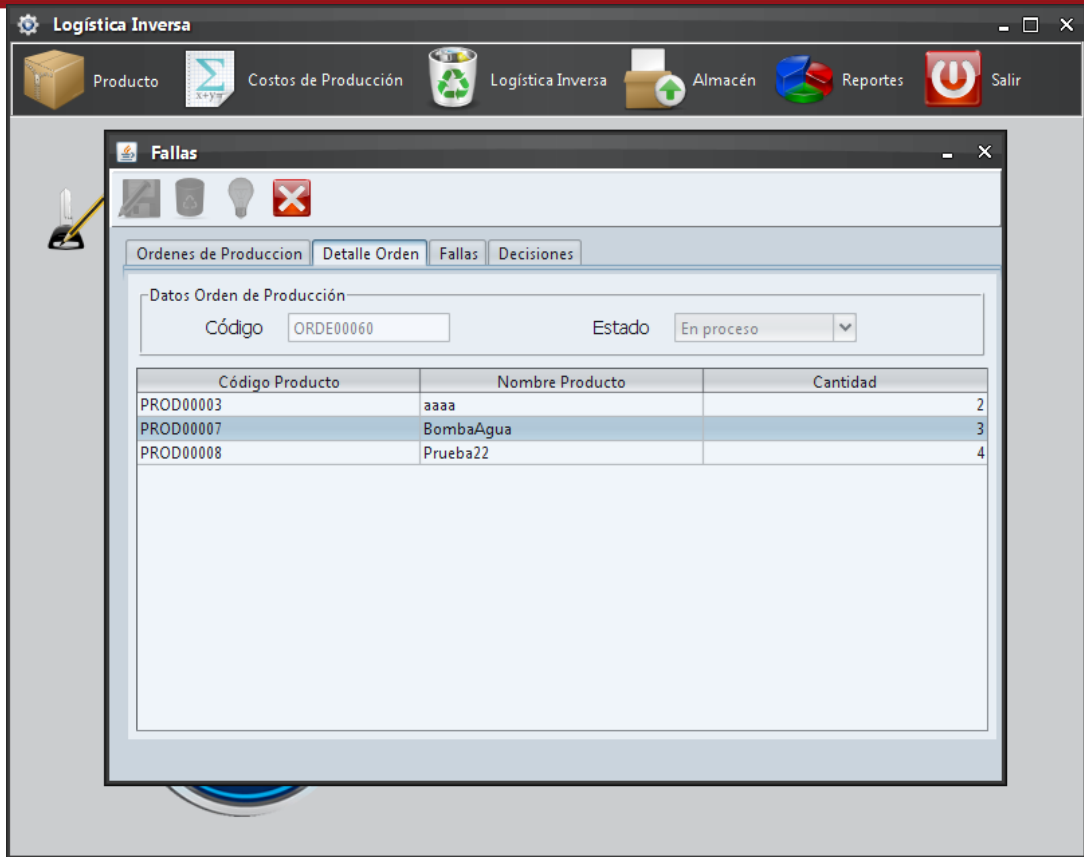


Figura 4.21: Prototipo que muestra los productos de la orden de producción.

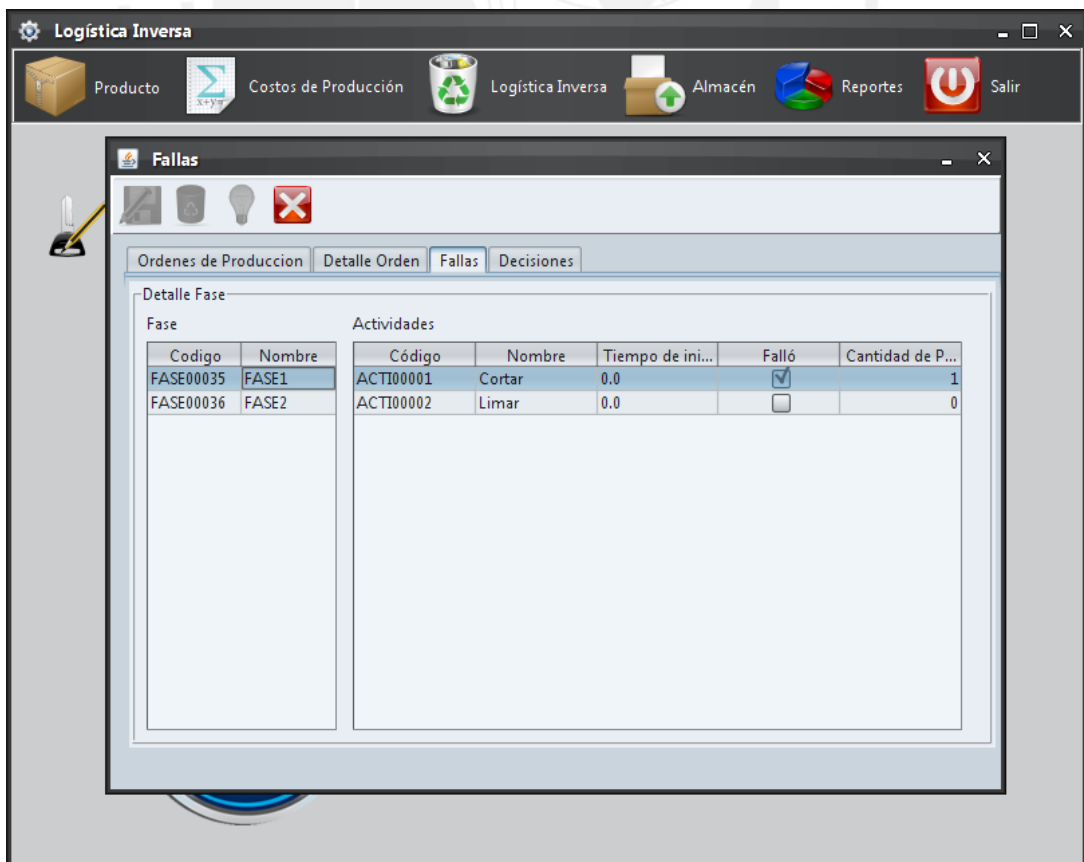


Figura 4.22: Prototipo que muestra el registro de una falla en una actividad.

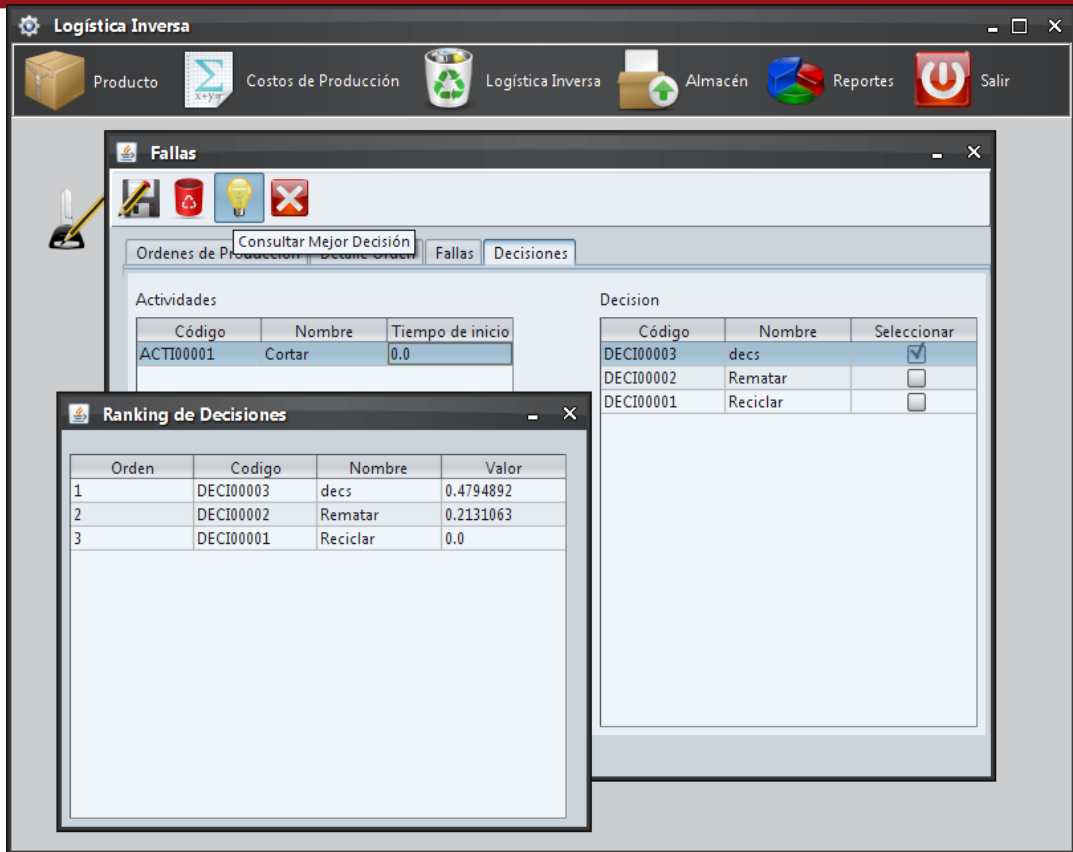


Figura 4.23: Prototipo que muestra el registro de una decisión ante una falla y la consulta de la mejor decisión.

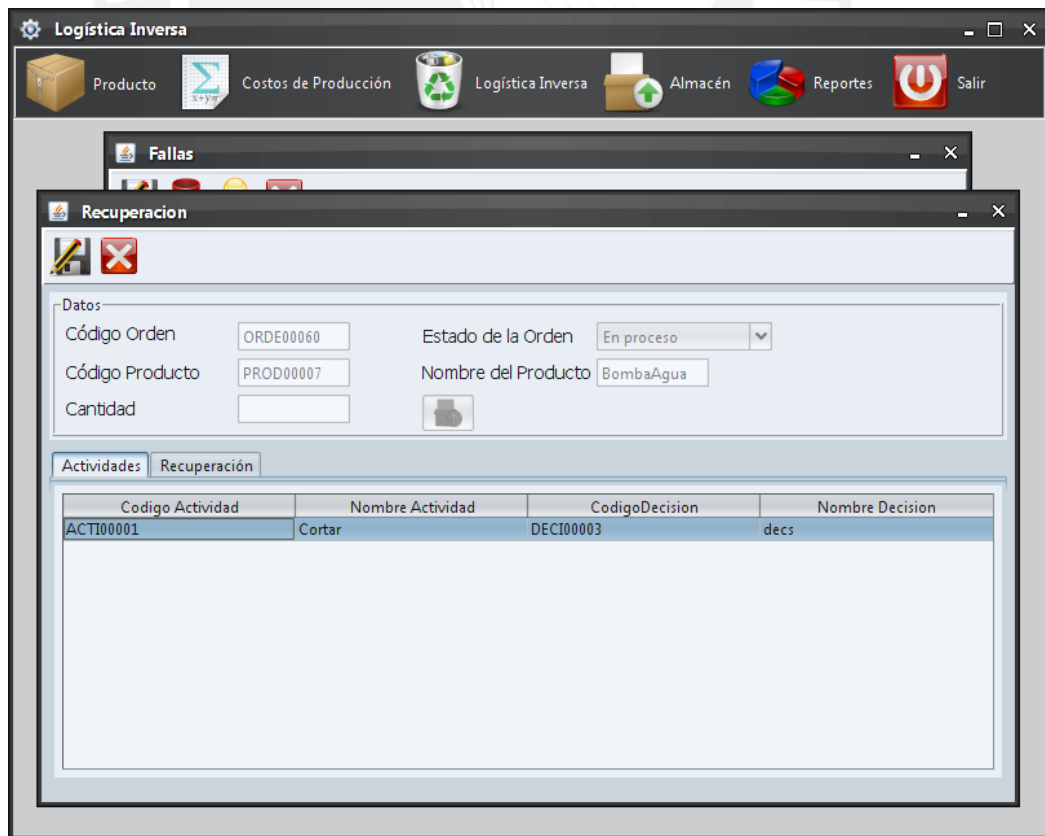


Figura 4.24: Prototipo que muestra la actividades de las fallas en el proceso de recuperación.

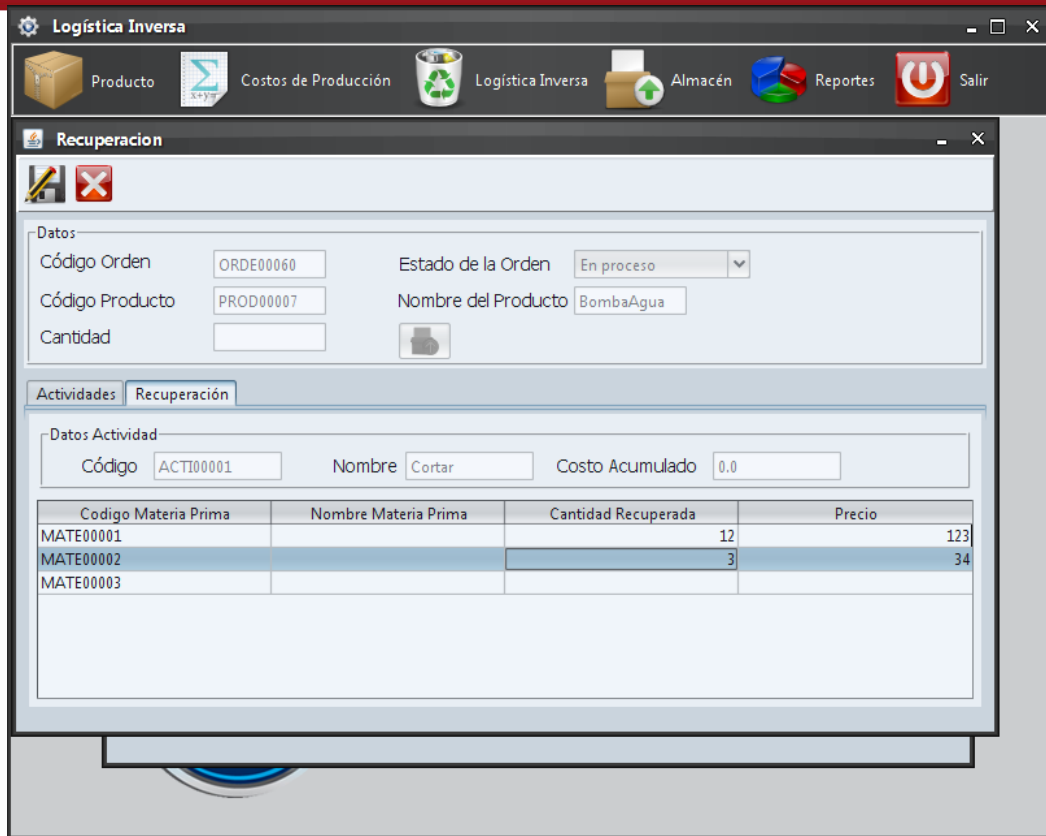


Figura 4.25: Prototipo que muestra el registro de la cantidad y precio de materia prima recuperada.

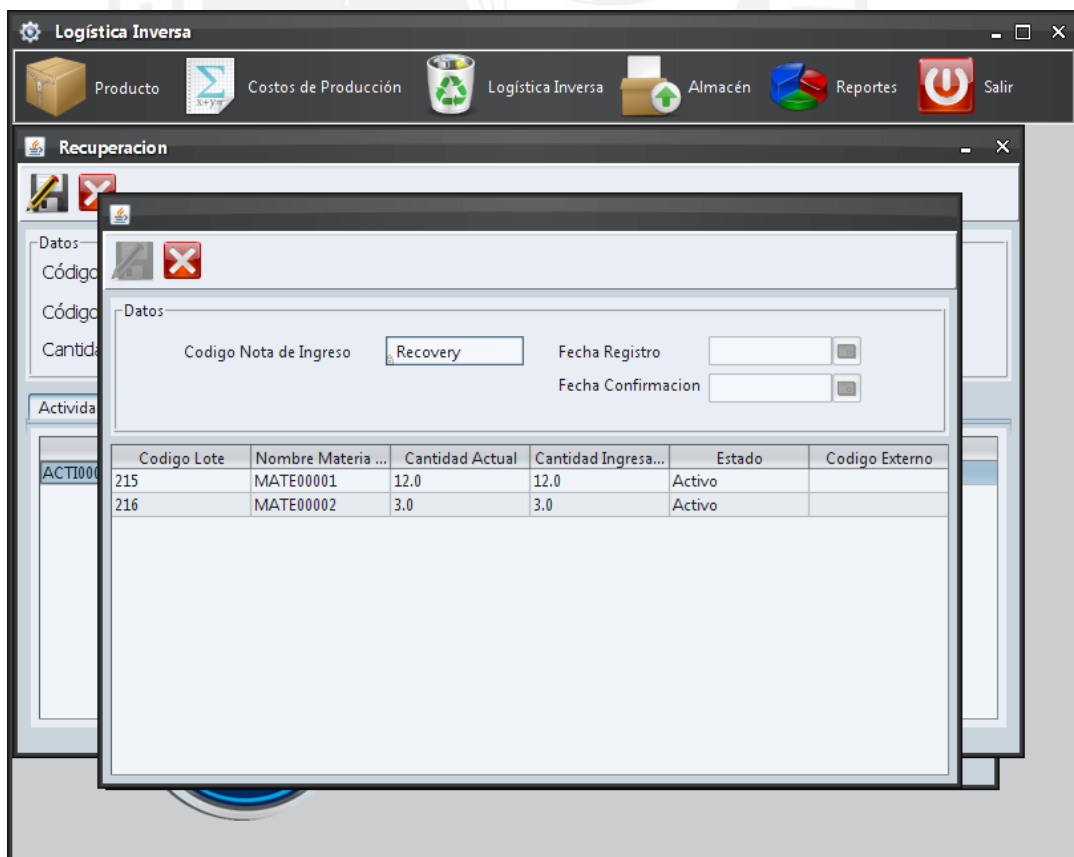


Figura 4.26: Prototipo que muestra los lotes generados producto de la recuperación.

4.1.5. Reportes estadísticos de la aplicación de logística inversa

Además, como el objetivo de la aplicación de logística inversa es la mejora en el uso y reutilización de la materia prima que se necesita en el proceso productivo, es necesario poder demostrar y apreciar, mediante valores numéricos, que su aplicación conlleva efectivamente a una reducción de costos, así como poder identificar las actividades con mayor frecuencia de falla y las decisiones que fueron tomadas con mayor regularidad.

Por ello, a continuación se muestra el flujo de los reportes gerenciales del sistema. El primer paso es seleccionar la opción “Reportes\Reportes Estadísticos” del menú principal, luego el sistema mostrará la ventana para el ingreso del rango de fechas del reporte. Seguido se deberá de seleccionar un tipo de reporte y este se generará en formato pdf.

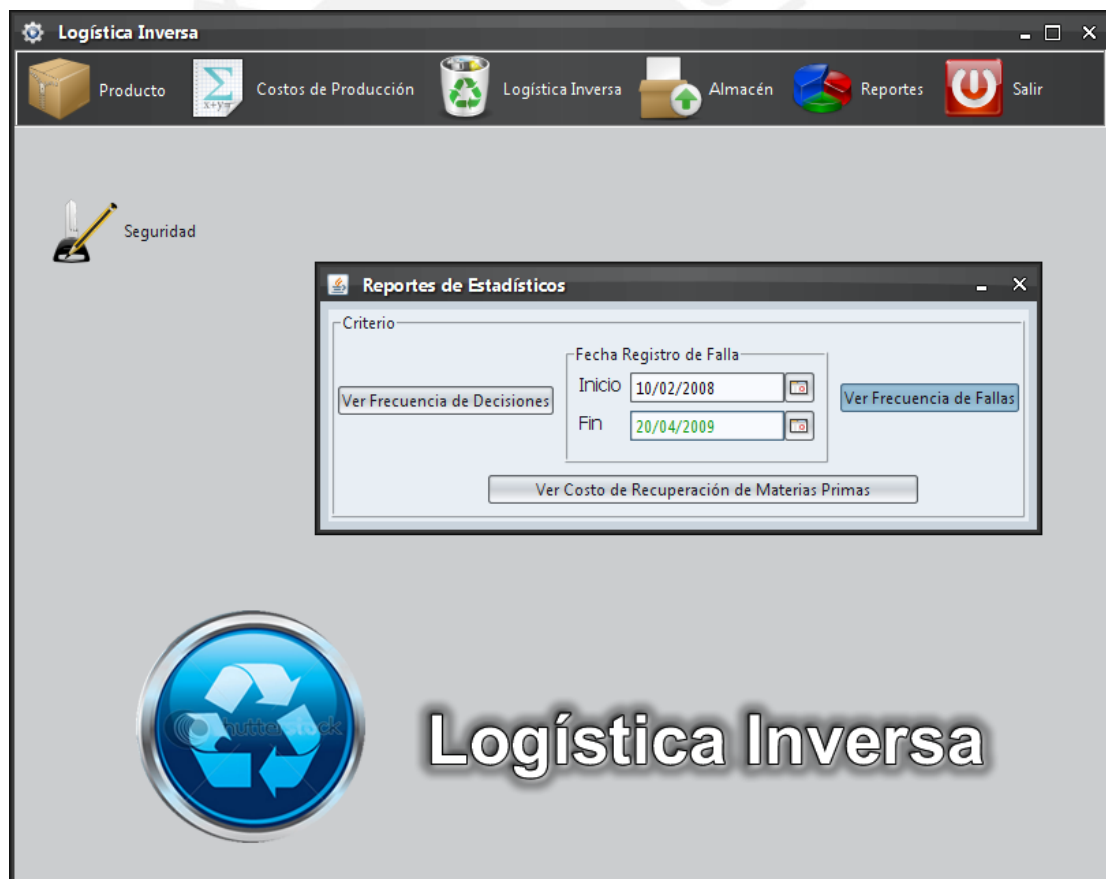


Figura 4.27: Prototipo de la generación del reporte estadístico

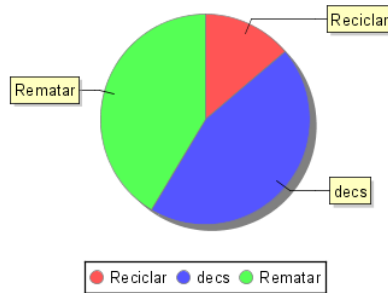
FRECUENCIA DE TOMA DE DECISIONES

Fecha de Registro de Falla

Fecha Inicio: 10/02/2008

Fecha Fin: 20/04/2009

Decisión	Nombre	Frecuencia	%	Fecha de Registro
DECI00001	Reciclar	4	13.7931	14/12/2008
DECI00003	decs	13	44.8275	04/01/2009
DECI00002	Rematar	12	41.3793	14/12/2008



12/04/2009 12.39 AM

Page 1 of 1

Figura 4.28: Prototipo del reporte de frecuencia de decisiones tomadas.

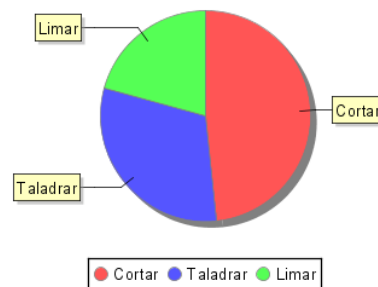
FRECUENCIA FALLAS

Fecha de Registro de Falla

Fecha Inicio: 10/02/2008

Fecha Fin: 20/04/2009

Actividad	Nombre	Frecuencia	%	Fecha de Registro
ACTI00001	Cortar	14	48.27586	13/12/2008
ACTI00003	Taladrar	9	31.03448	13/12/2008
ACTI00002	Limar	6	20.68965	13/12/2008



12/04/2009 12.40 AM

Page 1 of 1

Figura 4.29: Prototipo de la frecuencia de las actividades falladas.



Figura 4.30: Prototipo del reporte de materias primas recuperadas.

Para observar los demás reportes, ver anexos sección 10. Plantillas de reportes.

A continuación se muestra las pantallas Web del módulo de reportes.



Figura 4.31: Resultado de la búsqueda de órdenes vía Web

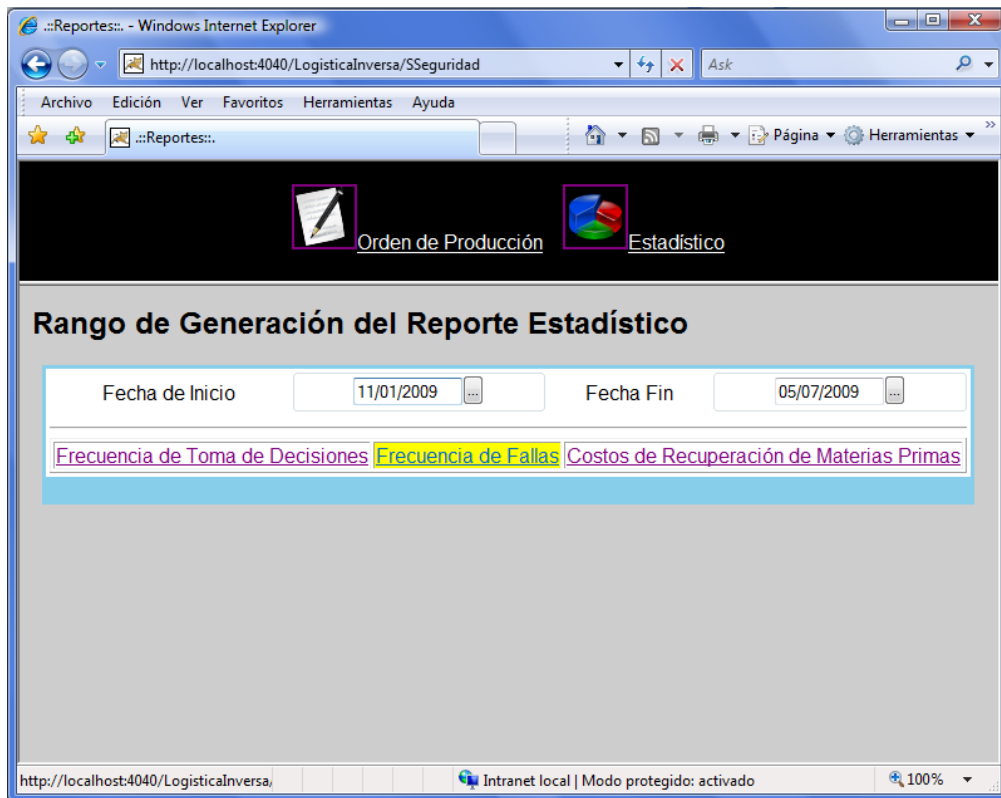


Figura 4.32: Pantalla de la generación de reportes estadísticos vía Web

4.1.6. Procedimientos almacenados

Se establece que para la identificación de los diversos campos principales del sistema, es decir los códigos de las materias primas, maquinarias, productos, planes de producción, etc. Estos se realicen mediante la generación automática del código y en forma correlativa.

A continuación se presenta un ejemplo:

Para el código del producto se sigue el siguiente formato: código identificador 'PROD' + número correlativo en el formato '00000'.

Códigos válidos del producto: PROD00001, PROD00002, PROD000003, etc.

Seguido, se muestra los identificadores que poseen este tipo de formato:

- Producto: PROD00000
- Materia prima: MATE00000
- Maquinaria: MAQU00000
- Empleado: EMPL00000
- Plan de producción: PLAN00000

- Fase: FASE00000
- Actividad: ACTI00000
- Orden de producción: ORDE00000
- Decisión: DECI00000
- Parámetro costo PARA00000

Por ello, para la generación automática de los códigos correlativos se utiliza el lenguaje PL/SQL proporcionado por el motor de base de datos Oracle, mediante los denominados Stores Procedures. Un ejemplo de este código se muestra a continuación para la generación y registro de un nuevo código de producto:

Store procedure “INSERTA_PRODUCTO”

```

create or replace procedure "INSERTA_PRODUCTO"
(nombre IN VARCHAR2,
descripcion IN VARCHAR2,
estado IN VARCHAR2,
imagen IN BLOB,
idProducto OUT VARCHAR2)
is

CURSOR C1 IS select STR_CODIGO from PRODUCTO;
Maximo_ID INTEGER(12):=-1;
Aux_ID INTEGER(12);
Nuevo_ID VARCHAR2(12);

BEGIN

FOR c1rec IN C1 Loop
  Aux_ID:=TO_NUMBER(SUBSTR(c1rec.STR_CODIGO,5,5));
  IF Aux_ID>Maximo_ID THEN
    Maximo_ID:=Aux_ID;
  END IF;
END Loop;

IF Maximo_ID<0 THEN
  Maximo_ID:=1;
ELSE
  Maximo_ID:=Maximo_ID+1;
END IF;

Nuevo_ID :=TO_CHAR(Maximo_ID,'00000');
Nuevo_ID:=SUBSTR(Nuevo_ID, 2, 5);
Nuevo_ID:=CONCAT('PROD',Nuevo_ID);
idProducto:=Nuevo_ID;

INSERT INTO PRODUCTO VALUES(Nuevo_ID, nombre, descripcion ,estado,
SYSDATE(), imagen);
COMMIT;

```

Paralelamente, se utiliza también otros stores procedures que no generan códigos, sino que se encargan de realizar transacciones como por ejemplo el registro de precios.

Store procedure “INSERTA_PRECIO”

```
create or replace procedure "INSERTAR_PRECIO"
(codigo IN VARCHAR2,
precio IN FLOAT)
is
Maximo_ID INTEGER(12):=-1;
BEGIN
INSERT INTO PRECIO (STR_CODIGO, DBL_PRECIO, FEC_REGISTRO,
FEC_FIN, STR_ESTADO) VALUES(codigo, precio, SYSDATE(), null, 'Habilitado');
COMMIT;
END;
```

Para observar todos los stores procedures y los scripts del modelo de datos, ver en anexos la sección 9 Modelo de base de datos.

Finalmente, es importante mencionar que en la sección 6 de los anexos se encuentran los estándares de programación seguidos en el desarrollo del sistema.

4.2. Pruebas

Las pruebas unitarias permiten verificar la funcionalidad y estructura de cada componente individualmente una vez que ha sido codificado. En este sistema en particular se elige las pruebas del tipo “Caja negra” (ver figura 4.25) que evalúa en base a entradas y salidas exclusivamente, siendo de una rápida comprobación en comparación por ejemplo con las pruebas de “Caja blanca” (ver figura 4.26), que implica una verificación de la estructura interna, en todo momento, lo cual significa mayor tiempo [35].

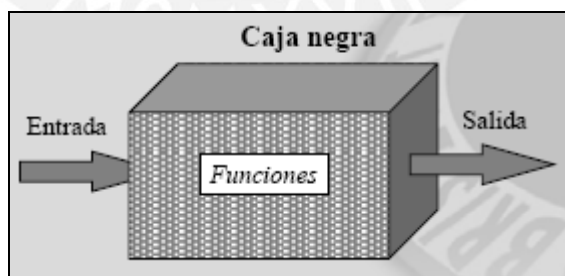


Figura 4.33: Flujo de las pruebas de caja negra [35]

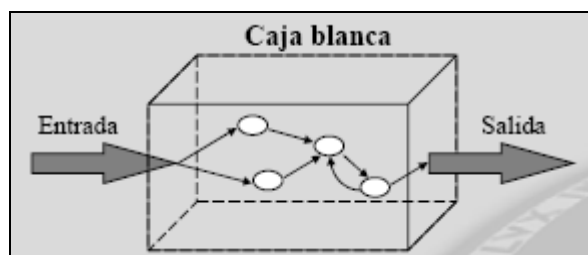


Figura 4.34: Flujo de las pruebas de caja blanca [35]

4.2.1. Clases de equivalencia

Para las pruebas se utilizarán las clases de equivalencia, las cuales representan al conjunto de datos que tienen un comportamiento similar a cualquier miembro de un conjunto determinado (ver figura 4.27).

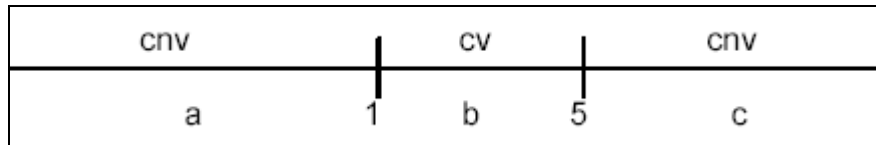


Figura 4.35: Ejemplo de clases equivalentes [35]

Para la elaboración de las clases de equivalencia se tomará en cuenta tres valores:

- Condición de entrada
- Clases válidas
- Clases no válidas

Ejemplos:

Mantener Plan de Producción

Condición de entrada	Clases validas	Clases no validas
Campo Nombre	1. Nombre entre 4 y 15 caracteres. 2. Caracteres únicamente del alfabeto de la 'A' a la 'Z' y de la 'a' a la 'z'	3. Nombre menor a 4 caracteres. 4. Nombre mayor a 15 caracteres. 5. Caracteres que no pertenecen al alfabeto, p.e: "" @\$%&/#" *+ ' etc.
Campo Descripción	6. Descripción entre 6 y 30 caracteres. 7. Caracteres únicamente del alfabeto de la 'A' a la 'Z' y de la 'a' a la 'z'	8. Descripción menor a 6 caracteres. 9. Descripción mayor de 30 caracteres. 10. Caracteres que no pertenecen al alfabeto, p.e: "" @\$%&/#" *+ ' etc.
Campo Fecha inicio	11. Fecha en el formato dd/mm/aaaa. 12. 1<=mm<=12 13. 1980<=aaaa<=9999 14. 0<dd y con valor máximo de acuerdo al mes.	15. Fecha que no cumpla con el formato dd/mm/aaaa. 16. mm<1 17. mm>12 18. aaaa<1980 19. aaaa>9999
Campo Fecha fin	20. Fecha en el formato dd/mm/aaaa. 21. 1<=mm<=12 22. 1980<=aaaa<=9999 23. 0<dd y con valor máximo de acuerdo al	24. Fecha que no cumpla con el formato dd/mm/aaaa. 25. Fecha fin menor que el campo Fecha inicio. 26. mm<1

	mes.	27. mm>12 28. aaaa<1980 29. aaaa>9999
Campo Tiempo ejecución	30. Número real mayor que 0 y menor que 10 000. 31. máximo 2 decimales.	32. Número real menor que 0. 33. Número real mayor que 10 000. 34. mayor de dos decimales.

Tabla 4.2: Cuadro de clases de equivalencia 1.

Registro de Orden de Producción

Condición de entrada	Clases validas	Clases no validas
Campo Código de Producto.	1. Alfanumérico de 6 caracteres con el formato "PRODNNNNN". 2. 00000<=NNNNN<=99999	3. Alfanumérico mayor de 6 caracteres. 4. Alfanumérico menor de 6 caracteres. 5. Alfanumérico que no cumpla con el formato "PRODNNNNN".
Campo Cantidad	6. Numero entero mayor que 0	7. Número menor que 0. 8. Alfanumérico

Tabla 4.3: Cuadro de clases de equivalencia 2.

Consultar Decisiones

Condición de entrada	Clases validas	Clases no validas
Campo Nombre	1. Nombre entre 4 y 15 caracteres. 2. Caracteres únicamente del alfabeto de la 'A' a la 'Z' y de la 'a' a la 'z'	3. Nombre menor a 4 caracteres. 4. Nombre mayor a 15 caracteres. 5. Caracteres que no pertenecen al alfabeto, p.e: ""@\$/%&/#"*'+ etc.
Campo Descripción	6. Descripción entre 6 y 30 caracteres. 7. Caracteres únicamente del alfabeto de la 'A' a la 'Z' y de la 'a' a la 'z'	8. Descripción entre 0 y 6 caracteres. 9. Descripción mayor de 30 caracteres. 10. Caracteres que no pertenecen al alfabeto, p.e: ""@\$/%&/#"*'+ etc.
Campo Código	11. Alfanumérico de 9 caracteres con el formato "DECISNNNNN". 12. 00000<=NNNNN<=99999	13. Alfanumérico mayor de 9 caracteres. 14. Alfanumérico menor de 9 caracteres. 15. Alfanumérico que no cumpla con el formato "DECISNNNNN".
Campo Fecha Inicio	16. Fecha en el formato dd/mm/aaaa 17. 1<=mm<=12 18. 1980<=aaaa<=9999	20. Fecha que no cumpla con el formato dd/mm/aaaa. 21. mm<1

	19. 0<dd y con valor máximo de acuerdo al mes.	22. mm>12 23. aaaa<1980 24. aaaa>9999
Campo Fecha Fin	25. Fecha en el formato dd/mm/aaaa 26. 1<=mm<=12 27. 1980<=aaaa<=9999 28. 0<dd y con valor máximo de acuerdo al mes.	29. Fecha que no cumpla con el formato dd/mm/aaaa. 30. Fecha fin menor que el campo Fecha inicio. 31. mm<1 32. mm>12 33. aaaa<1980

Tabla 4.4: Cuadro de clases de equivalencia 3.

Para visualizar las demás clases de equivalencia del sistema, ver anexos sección 7.

4.2.2. Catálogo de pruebas

Asimismo, se elaborará un catálogo de pruebas, por pantalla, como los que se muestran a continuación:

Pantalla de registro de orden de producción

Pre requisitos de la prueba:

Haber registrado al menos un producto y su plan de producción con estado activo. Además, para el plan de producción se debe de tener registrado todo los precios de la maquinaria y la mano de obra necesaria. Asimismo, se debe de tener stock de materia prima disponible para la orden de producción a registrar.

Identificador de Prueba		Pantalla de Registro de Orden de Producción	
Objetivo de Prueba		Validar su correcto funcionamiento	
Paso	Acción	Resultado Esperado	Visto
Ingreso al registro de Orden de Producción			
1.	Ingresar al sistema Logística Inversa.	Se muestra la pantalla de inicio de sesión.	<input type="checkbox"/> √
2.	Ingresar el usuario: "admin" y la contraseña: "admin."	Se muestra la pantalla principal del sistema Logística Inversa.	<input type="checkbox"/> √
3.	En el menú seleccionar "Producto", y luego "Orden de Producción".	Se muestra la pantalla de Orden de Producción.	<input type="checkbox"/> √
Registrar Orden de Producción			
4.	Hacer clic en el botón Nuevo.	Se muestra un tab del Detalle de Orden.	<input type="checkbox"/> √
5.	Se ingresa el código del Producto. Hacer click en el botón Añadir.	Se añade el producto a la grilla de la parte inferior.	<input type="checkbox"/> √
6.	Se ingresa una cantidad en la grilla para el producto ingresado.	Se añade la cantidad.	<input type="checkbox"/> √

	Se ingresa la tecla enter		
Registrar Costos Indirectos			
7.	Hacer clic en el Tab de Costos Indirectos.	Se muestra el Tab de Costos Indirectos, con tres radio button de los inductores: Unidades de Producción, Horas Maquina, Costo de Mano de Obra de Directa. Se muestra además en la grilla los costos indirectos del sistema.	<input checked="" type="checkbox"/>
8.	Se selecciona un radio Button. Se ingresa una cantidad para cada Costo indirecto de cada grilla. Hacer clic en el botón Grabar.	Se muestra una pantalla de confirmación de Registro de Orden de Producción.	<input checked="" type="checkbox"/>
9.	Aceptar la confirmación.	Se registra la Orden de Producción y se muestra una Orden de Producción.	<input checked="" type="checkbox"/>
Finalizar Orden Producción			
10.	Ingresar los datos que sirva de filtro a la búsqueda de la orden de producción (código, estado, fecha inicio, fecha fin). Hacer clic en el botón Buscar.	Se muestra una grilla con los datos que cumplen con el criterio de búsqueda.	<input checked="" type="checkbox"/>
11.	Se selecciona una o más filas de la grilla. Seleccionar el Tab Detalle Orden.	Se muestra el Tab de Detalle Orden, con los productos y cantidades de la Orden de Producción.	<input checked="" type="checkbox"/>
12.	Se cambia el combobox del estado de la Orden de Producción a Finalizado. Hacer click en el botón Grabar.	Se muestra una pantalla de confirmación de Modificación de Orden de Producción.	<input checked="" type="checkbox"/>
13.	Aceptar la confirmación.	Se finaliza la Orden de Producción y se muestra una ventana informativa.	<input checked="" type="checkbox"/>
Anular Orden Producción			
14.	Ingresar los datos que sirva de filtro a la búsqueda de la orden de producción (código, estado, fecha inicio, fecha fin). Hacer clic en el botón Buscar.	Se muestra una grilla con los datos que cumplen con el criterio de búsqueda.	<input checked="" type="checkbox"/>
15.	Se selecciona una o más filas de la grilla. Seleccionar el Tab Detalle Orden.	Se muestra el Tab de Detalle Orden, con los productos y cantidades de la Orden de Producción.	<input checked="" type="checkbox"/>
16.	Se cambia el combobox del estado de la Orden de Producción a Anulado. Hacer click en el botón Grabar.	Se muestra una pantalla de confirmación de Modificación de Orden de Producción.	<input checked="" type="checkbox"/>
17.	Aceptar la confirmación.	Se anula la Orden de Producción y se muestra una	<input checked="" type="checkbox"/>

		ventana informativa.	
--	--	----------------------	--

Tabla 4.5: Cuadro de plan de prueba 1.

Resultado de la prueba:

El registro de la orden de producción y los costos indirectos se realizó correctamente, luego se realizó la finalización de la orden de producción también de forma satisfactoria.

La anulación de la orden de producción no fue satisfactoria en un inicio, se tuvo que registrar una nueva de orden de producción he inmediatamente después proceder con la anulación de la orden de producción (no realizar la finalización) y la prueba tuvo un resultado satisfactorio.

Pantalla del registro de fallasPre requisitos de la prueba:

Haber registrado al menos una orden de producción y que se encuentre en estado en proceso. Además, se debe de tener registrado al menos una decisión.

Identificador de Prueba		Pantalla de Registro de Fallas	
Objetivo de Prueba		Validar su correcto funcionamiento	
Paso	Acción	Resultado Esperado	Visto
Ingreso al Registro de Fallas			
1.	Ingresar al sistema Logística Inversa.	Se muestra la pantalla de inicio de sesión.	<input checked="" type="checkbox"/>
2.	Ingresar el usuario: "admin" y la contraseña: "admin.".	Se muestra la pantalla principal del sistema Logística Inversa.	<input checked="" type="checkbox"/>
3.	En el menú seleccionar "Logística Inversa", y luego "Registro Fallas".	Se muestra la pantalla del registro de Fallas, listando en una grilla todas las ordenes de producción en proceso.	<input checked="" type="checkbox"/>
Registrar una Falla			
4.	Seleccionar una Fila de la grilla de órdenes de producción. Seleccionar el Tab Detalle Orden.	Se muestra una grilla con los productos que conforman la orden de producción.	<input checked="" type="checkbox"/>
5.	Seleccionar una Fila de la grilla de productos. Seleccionar el Tab Fallas.	Se muestra una grilla con las Fases del plan de producción del producto seleccionado.	<input checked="" type="checkbox"/>
6.	Seleccionar una Fase.	Se muestra una grilla con las actividades que comprenden la Fase.	<input checked="" type="checkbox"/>
7.	Se selecciona el check box de Falló. Se ingresa la cantidad de productos afectados.	Se muestra la cantidad de productos afectados de forma temporal.	<input checked="" type="checkbox"/>

Registrar Decisión			
8.	Seleccionar el Tab de Decisiones.	Se muestra una grilla con la lista de las Actividades a las cuales se les asignó el check box de Falló.	<input checked="" type="checkbox"/>
9.	Seleccionar una Actividad.	Se muestra en una grilla las decisiones registradas en el sistema.	<input checked="" type="checkbox"/>
10.	Hacer click en el botón mejor decisión.	Se muestra una ventana con un ranking de la mejor decisión a tomar.	<input checked="" type="checkbox"/>
11.	Seleccionar el una decisión en el check box de Seleccionar. Hacer clic en Grabar.	Se graba satisfactoriamente todas las fallas registradas y se muestra una ventana informativa.	<input checked="" type="checkbox"/>
Asignar Costos de Recuperación			
12.	Hacer click en el botón de Recuperación de Materias Primas.	Se muestra una ventana de Recuperación de materias Primas, mostrando un grilla con las actividades Falladas.	<input checked="" type="checkbox"/>
13.	Seleccionar una Actividad. Seleccionar el Tab de Recuperación.	Se muestra un panel con los datos de la actividad y una grilla con las materias primas de la actividad.	<input checked="" type="checkbox"/>
14.	Se selecciona una materia Prima Se ingresa una cantidad y un precio.	Se muestra la cantidad y el precio ingresado de forma temporal.	<input checked="" type="checkbox"/>
15.	Se da clic en el botón Grabar	Se muestra una ventana de confirmación de la grabación de recuperación.	<input checked="" type="checkbox"/>
16.	Aceptar la confirmación	Se registra la recuperación de materia prima y se genera nuevos lotes en el almacén, luego se muestra una ventana informativa	<input checked="" type="checkbox"/>
Ver Lotes Generados			
17.	Hacer click en el botón de Lotes	Se muestra una ventana con una grilla de los lotes generados de la grabación de recuperación de Materias Primas.	<input checked="" type="checkbox"/>

Tabla 4.6: Cuadro de plan de prueba 2.

Resultado de la prueba:

La prueba del registro de falla se realizó correctamente, así como registro de la decisión, los costos de recuperación y se visualizó correctamente los lotes generados producto del registro de la materia prima recuperada.

Para visualizar el catálogo de pruebas del sistema, ver anexos sección 8.

5. Observaciones, conclusiones y recomendaciones

5.1. Observaciones

Del proyecto desarrollado, se debe tener en cuenta las siguientes observaciones:

- El desarrollo del sistema de logística inversa abarca únicamente la aplicación en la etapa productiva, con conceptos de contabilidad de costos tanto directos como indirectos, que permiten demostrar el ahorro y la reducción de costos que genera de su aplicación.
- Se tiene en cuenta que en todo el proceso productivo existen costos que necesariamente están presentes de forma directa o indirecta, los cuales son: mano de obra, maquinaria y materia prima, por ello son considerados como costos obligatorios en el sistema y se tiene la restricción de que este presenten los tres al menos una vez cada uno de ellos.
- El sistema se desarrolla en un entorno cliente/servidor, debido a que el proceso productivo, los costos asociados a este y las fallas son informaciones internas de la empresa, por ende no se consideró

conveniente un desarrollo Web accesible a todos los usuarios vía Internet; sin embargo si se podría realizar mediante una intranet interna. Además, cabe destacar que la cantidad de usuarios que accedan a este sistema es reducido y la información que maneja es considerada como secreto industrial y/o empresarial.

5.2. Conclusiones

Según el desarrollo realizado y los resultados obtenidos se puede concluir lo siguiente:

- Se obtiene un sistema que permite registrar las fallas que suelen ocurrir en los procesos productivos, así como asignarle una decisión específica a cada una de ellas. De esta forma queda registrada información valiosa del proceso productivo que antes no se daba.
- Se concluye que una arquitectura cliente/servidor es la más adecuada para el tipo de desarrollo y el sistema esperado, debido a los pocos usuarios que presentará y la confidencialidad que se debe mantener, ya que los planes de producción detallados en cantidades y costos son y deben ser siempre de acceso restringido. Sin embargo, una aplicación Web sería también adecuada si se realiza únicamente en una Intranet de acceso restringido.
- Se comprobó el correcto funcionamiento del lenguaje de programación Java, mediante el uso del IDE NetBeans, con el motor de base de datos Oracle Express.
- El proyecto tomó unas 564.3 horas-hombre para su realización.

5.3. Recomendaciones y trabajos futuros

A continuación se lista una serie de recomendaciones y trabajos futuros en relación a la tesis descrita y desarrollada en los capítulos anteriores.

- Para un futuro desarrollo y ampliación del trabajo, se recomienda realizar el levantamiento de información en empresas de producción de tamaño medio y/o grande, debido a que la aplicación del concepto de logística inversa se da en la recuperación de grandes cantidades de materiales, que son

producidas precisamente por empresas de producción cuyo volumen productivo es considerable.

- El desarrollo de la tesis abarcó únicamente una solución para la aplicación de logística inversa en los planes de producción. Sin embargo, un trabajo futuro sería el desarrollo de un aplicativo que involucre las otras etapas de logística inversa: reciclaje y devoluciones de productos terminados.
- Para trabajos futuros se recomienda también aplicar conceptos de inventarios, stocks, notas de ingresos, salidas, etc., que permitan un mejor desarrollo de la aplicación de logística inversa de reciclaje y devoluciones.
- Asimismo, se recomienda tener siempre presente los conceptos de contabilidad de costos en desarrollos futuros ya que precisamente lo que se busca, mediante logística inversa, es generar una reducción de estos y para lo cual se necesita precisamente de resultados y valores numéricos que lo demuestren.
- Además, se recomienda realizar una aplicación Web para el desarrollo de trabajos futuros, especialmente en la etapa de reciclaje y devolución ya que este tipo de logística inversa se da de forma externa, es decir que proviene de lugares externos a la empresa productiva que lo aplica, por ende una aplicación Web daría lugar a tener la información registrada en línea y permitiría estar preparados para el aprovisionamiento de materiales.
- Se recalca que la arquitectura utilizada en el proyecto permite la realización de trabajos futuros mencionados en los párrafos anteriores, esto debido a la aplicación de capas.
- Se recomienda como trabajo futuro, la aplicación del proceso de jerarquía (AHP) de una manera más general, es decir poder realizar el cálculo ya no con una sola matriz, sino permitir el ingreso de varias matrices por parte del usuario. Para mayor información consultar la fuente número tres de la bibliografía.

- Asimismo, como una ampliación del proyecto se recomienda la carga de los planes del plan de producción, es decir los datos como actividades, tiempo de actividad, actividades por fase, fases, tiempo del plan de producción, etc. Se realice mediante una carga de archivos XML con una estructura definida.
- Se recomienda como trabajo futuro el desarrollo de interfaces con ERPs, ya que de este modo se integraría de una mejor forma al trabajo actual de las empresas de producción.



Bibliografía

Libros

- [1] Jacobson, G.Booch, J.Rumbaugh, “El proceso unificado de desarrollo de software”, Pearson Education, ISBN: 84-7829-036-2, 2000.
- [2] Charles T. Horngren, “Contabilidad de costos, un enfoque gerencial”, Pearson – Prentice Hall, ISBN: 970-26-0761-2, 2007.
- [3] Hamdy A. Taha, “Investigación de Operaciones”, Pearson, ISBN: 9702604982, 2004.
- [4] Roger S. Pressman, “Ingeniería de Software, un enfoque práctico”, Mc Graw Hill, ISBN: 84-481-3214-9, 2002.
- [5] Project Management Institute, Inc, “Guía de los Fundamentos de la Dirección de Proyectos”, Four Campus Boulevard, ISBN: 1-930699-73-5, 2004.

Revistas

- [6] Bruce Cadwell, “Untapped opportunities exist in returned products, a side of logistics few businesses have thought about—until now”, Information Week 729, 1999.

Conferencias publicadas, y no publicadas

- [7] Julio César Ángulo, “Artículo de Logística”, 2003.
- [8] Rogers & Tibben Lembke, “Going backwards: Reverse logistics trends and practices”, Reverse - Logistics Executive Council, Reno - Nevada University, 1998.

Reportes de investigación

- [9] Rubí Lacoba, S. y T. Bañegil, “Diseño de la función inversa de la Logística”: aspectos Estratégicos, Tácticos y Operativos, XI congreso ACEDE, 2003.
- [10] Arnulfo Arturo García, “Recomendaciones táctico-operativas para implementar un programa de Logística Inversa: Estudio de caso en la Industria del reciclaje de plásticos”, Editado por eumed.net, 2003.
- [11] Stock J. R, “Reverse Logistics”, CLM, Oak brook IL, 1992.
- [12] Grupo Europeo de Logística Inversa, RevLog, 1998.
- [13] “La implantación de la logística inversa en una multinacional de la distribución. Experiencia Carrefour”, Carrfour, Foro Internacional Pilot, Zaragoza España. Mayo 2001.
- [14] De Brito Marisa P., Dekker Rommert, “Reverse Logistics – a framework”, Erasmus University Rotterdam, Holland, Econometric Institute Report EI, 2002-38.

Referencias de fuentes electrónicas

- [15] Imagen y descripción del sistema de información de logística inversa NetReturn de FEDEX. Disponible en: www.fedex.com.
- [16] Imágenes de bombas centrífugas y helicoidales del catálogo de productos de la empresa Hidrostal Perú S.A. Disponible en: www.hidrostal.com.pe.

- [17] Definición de RUP y las mejores prácticas para el desarrollo de sistemas e implementación efectiva de la administración de proyectos de software, según IBM. Disponible en: <http://www-306.ibm.com/software/awdtools/rup/>
- [18] Gráfico de las fases RUP. Disponible en: <http://www.mugperu.com/portal/ForosdeDiscusi%C3%B3n/tabid/104/forumid/21/postid/1756/view/topic/Default.aspx>
- [19] Diagrama de procesos según el PMBOK del Project Management Institute. Disponible en: http://www.12manage.com/methods_pmi_pmbok_es.html
- [20] Especificaciones de las características tecnológicas de motores de base de datos según builder au, Australia. Disponible en: <http://www.builder.au.com.au/architect/database/soa/Road-test-Four-databases-tested/0,339024547,339224962-7,00.htm>
- [21] Artículo de costos ocultos de los procesos logísticos, Instituto Politécnico de Setúbal- Escola Superior de Ciências Empresarias, Estefanilha – Setúbal – Portugal. Disponible en: http://www.esce.ips.pt/departamentos/eg/Publica%C3%A7%C3%B5es/bmoura_ENGI_06.pdf
- [22] Gráfico de los motivos de ingreso de materia prima a la cadena productiva, texto en portugués, encuentro nacional de ingenieros de producción, Ouro Preto - Brasil, 2003. Disponible en: http://www.abepro.org.br/biblioteca/ENEGEP2003_TR0112_0090.pdf
- [23] Definición actual del concepto de logística inversa por parte del “grupo europeo de logística inversa”. Disponible en: <http://www.reverselogisticstrends.com/reverse-logistics.php>
- [24] Artículo de “Computer Hardware Reverse Logistic: A Field Study of Canadian Facilities”, Interuniversity Research Centre of Enterprise Networks, Logistics and Transportation, Suzanne Marcotte, Marie-Eve Hallé, Benoit Montreuil, 2008, Canada. Disponible en: <http://www.cirrelt.ca/DocumentsTravail/CIRRELT-2008-41.pdf>
- [25] Artículo de “Reverse logistics network design with stochastic lead times”, Faculty of Applied Economics, University of Antwerp, Kris Lieckens, Nico Vandaele, 2005, Belgium. Disponible en: http://www.sciencedirect.com/science?_ob=ArticleURL&_udi=B6VC5-4G1V66P-2&_user=10&_rdoc=1&_fmt=&_orig=search&_sort=d&_view=c&_version=1&_urlVersion=0&_userid=10&md5=6b00e3794b67b3ff59543433c0617a19
- [26] Folleto, “10° Congresso de Logística, Responsabilidade e Eficiência na Logística”, A APOLOG – Associação Portuguesa de Logística, Lisboa - Portugal, 2007. Disponible en: <http://www.aplog.pt/modules/aplog/public/docs/10CongressoFolheto.pdf>
- [27] Empresa dedicada a soluciones de la administración de logística inversa para empresas de producción, en la etapa de reciclaje. Disponible en: <http://www.reverselogistic.com/>
- [28] Artículo de conceptos y definiciones de órdenes de producción en empresas de producción, Lic. Cecilia Rasgado Luis, Instituto Tecnológico del Istmo, México. Disponible en: <http://www.itistmo.edu.mx/pagina%20de%20la%20lic%20Cecilia%20Rasgado%20Luis/COSTOS%20I%20UNI5.html>
- [29] Definición del concepto de costo. Disponible en: <http://definicion.de/costo/>

- [30] Definición del concepto de análisis de sistemas según DAEDALUS, empresa española especializada en tecnologías de información y comunicaciones. Disponible en: <http://www.daedalus.es/inteligencia-de-negocio/sistemas-complejos/ingenieria-de-sistemas/analisis-de-sistemas/>
- [31] Tutorial de UML, Patricio Salinas, Universidad de Chile. Disponible en: <http://www.dcc.uchile.cl/~psalinas/uml/casosuso.html>
- [32] Precio comercial del motor de base de datos Oracle Estandar Edition 10g. Disponible en: <http://www.oracle.com/corporate/pricing/technology-price-list.pdf>
- [33] Precio comercial del servidor HP Proliant ML-110 G5. Disponible en: <http://www.mesajilhnos.com/dproductos.php?ip=6403>
- [34] Precio comercial de Microsoft Windows XP Profesional Español OEM. Disponible en: <http://www.mesajilhnos.com/lproductoss.php?is=81>

Otros

- [35] Curso de Ingeniería de Software 2007-2, Ingeniería Informática - PUCP.
- [36] Curso de Gerencia de Proyectos y Calidad 2009-1, IC-PUCP.
- [37] Figura de Reciclaje disponible en: http://www.ecologismo.com/wp-content/uploads/2008/11/simbolos_reciclaje.jpg

