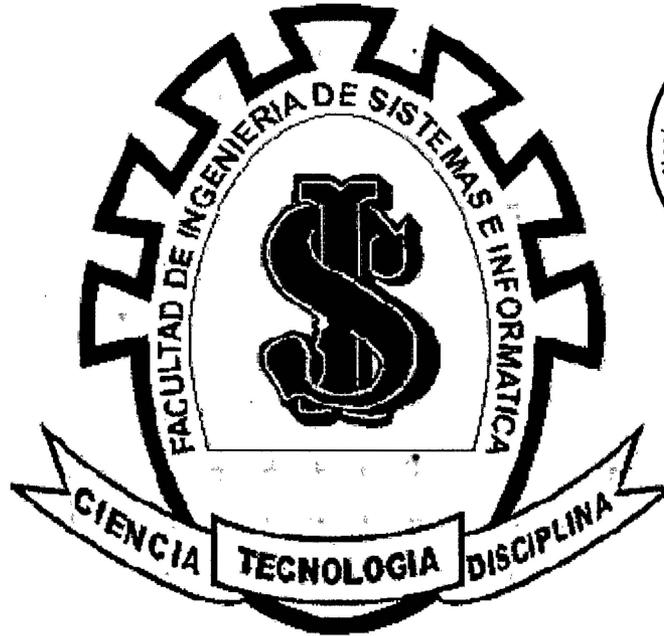


**NO SALE A
DOMICILIO**

UNIVERSIDAD NACIONAL DE LA AMAZONÍA PERUANA

**FACULTAD DE INGENIERÍA
DE SISTEMAS E INFORMÁTICA**



**“SISTEMA DE INFORMACIÓN DE ALMACÉN PARA
PYME ‘TRIUNFO SAC’ IQUITOS”**

INFORME PRÁCTICO DE SUFICIENCIA

**PARA OPTAR EL TÍTULO PROFESIONAL DE:
INGENIERO DE SISTEMAS E INFORMÁTICA**

Presentado por el Bachiller:
Roussell Eduardo Ramírez Álvarez

Asesor:
Ing. Luis Honorato Pita Astengo

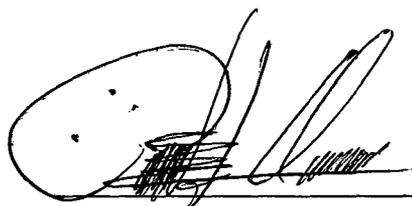
**IQUITOS – PERÚ
2011**

DONADO POR:

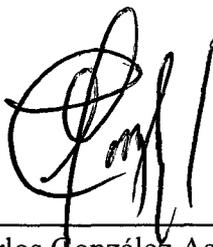
Roussell E. Ramirez Alvarez

Iquitos, 22 de 9 de 2012

INFORME TÉCNICO DEL EXÁMEN DE SUFICIENCIA PREVIA
ACTUALIZACIÓN ACADÉMICA APROBADO EN SUSTENTACIÓN PÚBLICA,
POR EL JURADO EXAMINADOR DESIGNADO POR EL COORDINADOR DE LA
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA DE LA
UNIVERSIDAD NACIONAL DE LA AMAZONÍA PERUANA.



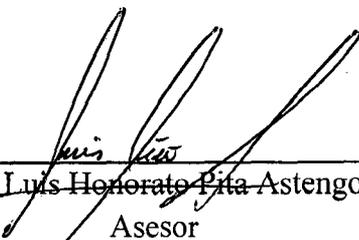
Ing. Carlos Iván García Gómez
Presidente



Ing. Carlos González Aspajo
1° Miembro



Ing. Roberto Martín Tuesta Pereyra
2° Miembro



Ing. Luis Honorato Pita Astengo
Asesor

DEDICO este trabajo a:

Mi madre Pochita, que me ama tanto y por ende la única que aún cree en mis mentiras;

Mi padre Carlos al único hombre que amo más que a mí mismo;

*Mis hermanas Zully, Nady, Gise, Karen y Vanessa las únicas mujeres a las que confiaría
mi vida;*

*Mis tíos Jorge, Hersón, César y Javier que fueron un soporte económico escondido en mi vida
universitaria y un apoyo espiritual en otros casos;*

Mi abuelita Juanita que me acompañó y aconsejó a cada paso de este arduo y duro camino.

AGRADEZCO a:

Mi amigo y colaborador Jorge quien estuvo en el cierre de este último capítulo universitario;

Mi amigo y asesor Luis por su paciencia y compromiso;

La empresa Triunfo SAC por su acogedora estancia durante el desarrollo de este trabajo.

Resumen

El presente trabajo fue elaborado por el Bach. Roussell Eduardo Ramírez Álvarez; y comprende el Sistema de Información en plataforma Windows para la gestión de Almacén, desarrollado para la sub oficina de Almacén de la empresa Triunfo SAC en la ciudad de Iquitos en las cuales se presentaban las problemáticas de transacciones, errores de carácter humano y demoras en los tiempos de respuestas en generación reportes realizados por el personal encargado de la sub área; bajo la Gerencia General de la Lic. Clara Elena Álvarez Ruiz de Ramírez quien puso a disponibilidad del Autor a todo personal involucrado y/o encargado del negocio – de ser – necesario para la culminación satisfactoria del presente trabajo.

SisTriunfo es un sistema de información centrada en solucionar y mitigar la mayor cantidad de amenazas en las que pueda estar incurriendo en la sub oficina de Almacén; aplicando las técnicas y herramientas aprendidas durante mi estancia universitaria en nuestra alma mater como es la Universidad Nacional de la Amazonia Peruana en su Facultad de Ingeniería de Sistemas e Informática. El tiempo de duración para el desarrollo de la aplicación será de ocho (08) semanas calendario iniciándose el lunes 13 de diciembre y terminando el viernes 4 de febrero.

En base a los resultados mostrados hemos concluido que se ha logrado los objetivos específicos del presente informe – en torno a los indicadores de control, tasa de errores y tiempos eficientes en la generación de reportes – descritos en las conclusiones.

A continuación se resumen los capítulos de la Sección II, en el cual se desarrollarán los temas específicos a tratar:

En el capítulo I, hace referencia a la problemática encontrada respecto a los procesos observados así como sus actividades que se desarrollan dentro de cada una de ellas; planteándose los objetivos de carácter general y específicos ahondado a lo que el presente trabajo apunta su norte.

En el Capítulo II, se estructura de forma tal que se muestre la metodología para definir la aplicación del presente trabajo, en este caso Rational Unified Proccess RUP será la encargada de seguir con la estricta secuencia para la culminación satisfactoria del presente trabajo.

En el Capítulo III, se describen los procesos globales y específicos en los que la sub oficina de Almacén está inmersa; y de esta forma dar inicio a la aplicación según la metodología de desarrollo de software seleccionada.

Se culmina el presente trabajo con los capítulos IV, V y VI donde se muestran los resultados, la discusión de resultados, las conclusiones y recomendaciones respectivamente.

Palabras Claves

- Sistema, conjunto de procesos integrados entre sí con la finalidad de cumplir un objetivo en común.
- Almacén, área o unidad dentro de una empresa con el objetivo de controlar la entrada y salida de productos.
- Triunfo SAC, pequeña empresa de rubro comercial, ubicada en la ciudad de Iquitos, Ca. Samanez Ocampo 895.
- Iquitos, capital del departamento de Loreto localizada al noreste del Perú.
- FISI, Facultad de Ingeniería de Sistemas e Informática.
- UNAP, Universidad Nacional de la Amazonia Peruana.

Abstract

This work was written by Bach. Roussell Eduardo Alvarez Ramírez, and includes the Information System on the Windows platform for warehouse management, developed for the sub office Triunfo Company Store SAC in the city of Iquitos in which the transaction had the problems, errors character human and delays in response times on reports made by staff of the sub area, under the General Management of Ms. Alvarez Clara Elena Ramirez, who made available to the Author to all staff involved and / or business manager - to be - necessary for the successful completion of this work.

SisTriunfo is an information system focused on fixing and mitigate as many threats that may be making the sub office warehouse, applying the techniques and tools learned during my academic stay at our alma mater as the National University Peruvian Amazon in the Faculty of Engineering and Computer Systems. The time period for the development of the application shall be eight (08) calendar weeks beginning on Monday December 13 and ending on Friday 4 February.

Based on the results shown we have concluded that it has been the satisfaction of the specific objectives of this report - about the indicators of control, rate and time-efficient Erros report generation -.

The following summarizes the chapters of Section II, which will be developed to address specific issues:

In Chapter I, refers to the problems encountered with respect to the observed processes and activities taking place within each one of them, considering the objectives of general and specific that deepened to its present work points north.

In Chapter II, is structured in a way that shows the methodology for determining the application of this work, in this case Unifeed processed Rational RUP will be responsible to follow the strict sequence to the successful completion of this work.

Chapter III describes the overall and specific processes in the sub office warehouse are immersed, and thus begin the application according to the software development methodology selected.

This work culminates in chapters IV, V and VI which shows the results, discussion of findings, conclusions and recommendations respectively.

Keywords

- System, integrated set of processes together in order to fulfill a common goal.
- Warehouse area or unit within a company with the aim of controlling the entry and exit of products.
- Triunfo SAC, small business line of business, located in the city of Iquitos, ca. Samanez Ocampo 895. Ocampo Samanez 895.
- Iquitos, capital of Loreto is located in northeastern Peru.
- FISI, Faculty of Engineering and Computer Systems.
- UNAP, Universidad Nacional de la Amazonía Peruana.

Índice General

Resumen	i
Abstract.....	ii
Índice General.....	iii
Índice de Tablas y Cuadros	vi
Índice de Ilustraciones	vii
Sesión I: Datos Generales.....	1
1. Titulo de la Tesina	1
2. Áreas de Desarrollo	1
3. Generalidades de la Institución.....	1
3.1. Razón Social	1
3.2. Ubicación de la Empresa	1
3.3. Organigrama Funcional	2
3.4. Funciones de la Oficina o Área	2
4. Bachiller.....	3
5. Asesor	3
6. Colaboradores.....	3
7. Duración estimada de ejecución del Proyecto	3
8. Presupuesto Estimado.....	3
Sesión II: Desarrollo del Tema	4
Capítulo 1: Introducción	4
1.1. Contexto.....	4
1.2. Problemática Objeto de la Aplicación	5
1.3. Objetivos del Proyecto.....	6
1.3.1. Objetivo General.....	6
1.3.2. Objetivos Específicos	6
Capítulo II: Descripción del Diseño de la Solución	7
2.1. Técnicas de Recolección de Datos	7
2.2. Metodología y Herramienta a emplear	8
2.2.1. Metodologías	8
2.2.2. Herramientas.....	9
2.3. Descripción del desarrollo de la Solución	10
2.4. Indicadores de Evaluación de la Solución	10

2.5. Relación de Entregables	10
Capítulo III: Desarrollo de la Solución de la Propuesta	11
3.1. Modelado del Negocio.....	11
3.1.1. Caso de Uso del Negocio.....	11
3.1.2. Escenarios y Procesos.....	12
3.1.3. Modelo de Objetos del Negocio	16
3.1.4. Modelo de Dominio.....	17
3.2. Modelado de Requerimientos	18
3.2.1. Propósito.....	18
3.2.2. Alcance	18
3.2.3. Descripción de Stakeholders y Usuarios	18
3.2.4. Descripción Global del Producto.....	18
3.2.5. Caso de Uso de Requerimiento.....	19
3.2.6. Especificaciones de Caso de Uso.....	22
3.3. Análisis	23
3.3.1. Diagrama de Colaboración	23
3.3.2. Diagrama de Secuencia.....	26
3.3.3. Diagrama de Clases	29
3.4. Modelado de Diseño.....	30
3.4.1. Diseño de Interfaz.....	30
Diseño de la Base de Datos	34
3.5. Implementación	36
3.5.1. Diagrama de Componentes.....	36
3.5.2. Diagrama de Despliegue.....	37
Capítulo IV: Resultados y su Discusiones.....	38
Capítulo V: Conclusiones.....	39
Capítulo VI: Recomendaciones	40
Anexos.....	42
Anexo 01: Encuestas a profesionales.	42
Anexo 02: Entrevista a Dueño de Proceso de Almacén	48
Anexo 03: Resultado de Encuestas a Profesionales.....	49
Anexo 04: Presupuesto Estimado	50
Anexo 05: Vistas de Boletas, Facturas y Guías de Remisión.....	51
Anexo 06: Diccionario de Datos de Base de Datos SisTriunfo.....	54
Anexo 07: Resultados de Indicadores.....	56

Anexo 08: Glosario de Términos.....	60
Anexo 09: Manual Técnico	61
Anexo 10: Manual de Usuario.....	65
Anexo 11: Código Fuente.....	77

Índice de Tablas y Cuadros

TABLA 1: CRONOGRAMA DE ACTIVIDADES - SISTRIUNFO	3
TABLA 2: LISTA DE ENTREGABLES	10
TABLA 3: LISTA DE WORKERS Y OBJETOS DE MODELO DE OBJETOS SISTRIUNFO	16
TABLA 4: RESUMEN DE STAKEHOLDERS	18
TABLA 5: RESUMEN DE USUARIOS	18
TABLA 6: ESPECIFICACIÓN DE CASO DE USO	22
TABLA 7: RELACIÓN DE PROFESIONALES ENCUESTADOS	47
TABLA 8: RESULTADO DE ENCUESTAS A PROFESIONALES	49
TABLA 9: PRESUPUESTO HARDWARE.....	50
TABLA 10: PRESUPUESTO SOFTWARE	50
TABLA 11: PRESUPUESTO INSUMOS, SERVICIOS Y MATERIALES DE ESCRITORIO.....	50
TABLA 12: DICCIONARIO DE DATOS BASE DE DATOS SISTRIUNFO.....	54
TABLA 13: DE INDICADORES RESPECTO A LA REDUCCIÓN DE TIEMPO EN LAS TRANSACCIONES DE PRODUCTOS	56
TABLA 14: RESULTADO DE INDICADORES RESPECTO A LA REDUCCIÓN DE TASA DE ERROR EN LAS TRANSACCIONES DE PRODUCTOS.....	56
TABLA 15: RESULTADO DE INDICADORES RESPECTO A LA REDUCCIÓN DE TASA DE ERROR EN LAS TRANSACCIONES DE PROVEEDORES Y TRANSPORTISTAS	57
TABLA 16: DE INDICADORES RESPECTO A LA REDUCCIÓN DE TASA DE ERROR EN LAS TRANSACCIONES DE PROVEEDORES Y TRANSPORTISTAS.....	57
TABLA 17: RESULTADO DE INDICADORES RESPECTO A LA REDUCCIÓN DE TIEMPO EN LAS TRANSACCIONES DE COMPRAS	58
TABLA 18: RESULTADO DE INDICADORES RESPECTO A LA REDUCCIÓN DE TASA DE ERROR EN LAS TRANSACCIONES DE COMPRAS	58
TABLA 19: RESULTADO DE INDICADORES RESPECTO A LA REDUCCIÓN DE TIEMPO EN LA GENERACIÓN DE REPORTES	59

Índice de Ilustraciones

ILUSTRACIÓN 1: PLANO UBICACIÓN TRIUNFO SAC	1
ILUSTRACIÓN 2: ORGANIGRAMA FUNCIONAL PYME TRIUNFO SAC.....	2
ILUSTRACIÓN 3: CASO DE USO DEL NEGOCIO – TRIUNFO	11
ILUSTRACIÓN 4: DIAGRAMA DE ACTIVIDADES REGISTRANDO COMPRA	13
ILUSTRACIÓN 5: DIAGRAMA DE ACTIVIDADES TRANSFIRIENDO MERCADERÍA.....	14
ILUSTRACIÓN 6: DIAGRAMA DE ACTIVIDADES ADMINISTRANDO SISTRIUNFO	15
ILUSTRACIÓN 7: MODELO DE OBJETOS DE SISTRIUNFO	16
ILUSTRACIÓN 8: MODELO DE DOMINIO SISTRIUNFO.....	17
ILUSTRACIÓN 9: CASO DE USO DE REQUERIMIENTO GESTIONANDO TRANSACCIÓN DE ALMACÉN	19
ILUSTRACIÓN 10: CASO DE USO DE REQUERIMIENTO TRANSFERIR MERCADERÍA	20
ILUSTRACIÓN 11: CASO DE USO DE REQUERIMIENTO ADMINISTRAR SISTRIUNFO	21
ILUSTRACIÓN 12: DIAGRAMA DE COLABORACIÓN REGISTRAR COMPRA	23
ILUSTRACIÓN 13: DIAGRAMA DE COLABORACIÓN GESTIONAR PROVEEDOR	24
ILUSTRACIÓN 14: DIAGRAMA DE COLABORACIÓN GESTIONAR PRODUCTO	25
ILUSTRACIÓN 15: DIAGRAMA DE SECUENCIA REGISTRAR COMPRA.....	26
ILUSTRACIÓN 16: DIAGRAMA DE SECUENCIAS GESTIONAR PROVEEDOR	27
ILUSTRACIÓN 17: DIAGRAMA DE SECUENCIA GESTIONAR PRODUCTO.....	28
ILUSTRACIÓN 18: DIAGRAMA DE CLASES SISTRIUNFO	29
ILUSTRACIÓN 19: LOGIN SISTRIUNFO.	30
ILUSTRACIÓN 20: INTERFAZ PRINCIPAL ALMACÉN	30
ILUSTRACIÓN 21: REGISTRAR COMPRA.....	31
ILUSTRACIÓN 22: REGISTRAR GUÍA DE REMISIÓN	31
ILUSTRACIÓN 23: VISTA DE REPORTE SISTRIUNFO	32
ILUSTRACIÓN 24: INTERFAZ PRINCIPAL SISTEMAS.....	33
ILUSTRACIÓN 25: TRANSACCIÓN PERSONA.....	33
ILUSTRACIÓN 26: DIAGRAMA FÍSICO BASE DE DATOS TRIUNFO.....	34
ILUSTRACIÓN 27: DIAGRAMA LÓGICO BASE DE DATOS TRIUNFO.....	35
ILUSTRACIÓN 28: DIAGRAMA DE COMPONENTES SISTRIUNFO.....	36
ILUSTRACIÓN 29: DIAGRAMA DE DESPLIEGUE SISTRIUNFO.....	37
ILUSTRACIÓN 30: IMAGEN GUÍA DE REMISIÓN.....	51
ILUSTRACIÓN 31: IMAGEN FACTURA TRIUNFO SAC.....	52
ILUSTRACIÓN 32: IMAGEN BOLETA TRIUNFO SAC	53

Sesión I: Datos Generales

1. Título de la Tesina

“Sistema de Información de Almacén para la Pyme ‘Triunfo SAC’ Iquitos”

2. Áreas de Desarrollo

Desarrollo de Sistemas de Información / Desarrollo de Software.

3. Generalidades de la Institución

3.1. Razón Social

PYME TRIUNFO S.A.C., es una empresa de Líneas de Vestir y Calzados tiene las necesidades de entrada y salida de mercadería; sin sucursales en su cobertura ni procesos definidos en sus actividades.

3.2. Ubicación de la Empresa

Samanez Ocampo # 895, ciudad de Iquitos, provincia de Maynas, del departamento de Loreto.

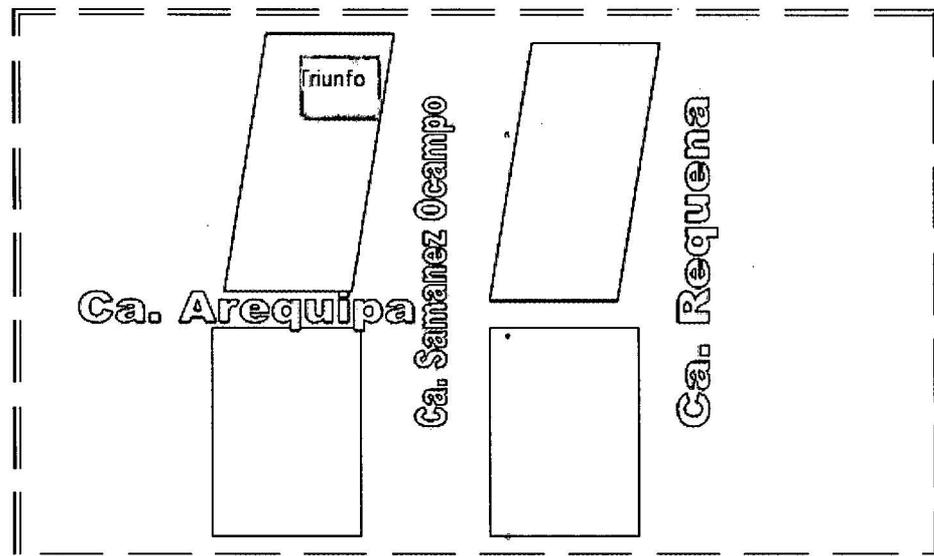


Ilustración 1: Plano Ubicación Triunfo SAC
Fuente: Pyme Triunfo SAC

3.3. Organigrama Funcional

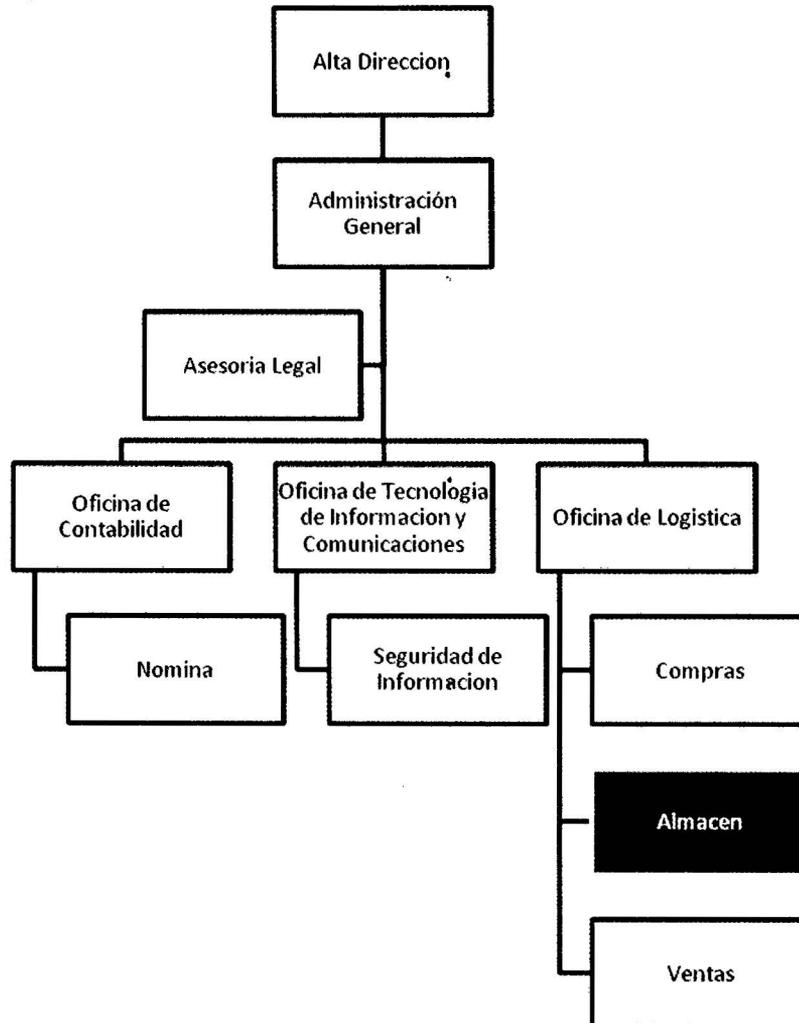


Ilustración 2: Organigrama Funcional Pyme Triunfo SAC
Fuente: Pyme Triunfo SAC

3.4. Funciones de la Oficina o Área

La sub oficina Almacén se encuentra ubicada dentro de la oficina de Logística y cumple las siguientes funciones:

- Transacción de productos (registros, actualizaciones, dar de baja productos deteriorados y consultas)
- Transacción de proveedores (registros, actualizaciones, dar de baja y consultas)
- Transacción de transportistas (registros, actualizaciones, dar de baja y consultas)
- Transacción de compras (registros, actualizaciones y consultas).
- Transacción de notas de crédito por mercadería en mal estado (registros, actualizaciones y consultas).
- Transacción de Guía Remisión (registros, actualizaciones y consultas).
- Reportes de productos, proveedores, transportistas, stock productos, compras, guías de remisión, notas de crédito y kardex de almacén.

4. Bachiller

Roussell Eduardo Ramírez Álvarez

5. Asesor

Ing. Luis Honorato Pita Astengo

6. Colaboradores

Bach. Jorge Walter Zvietcovich Diaz

7. Duración estimada de ejecución del Proyecto

La duración estimada para su producción y posterior pase ejecución es de ocho (08) semanas (56 días) calendario, a partir de lunes 13 de diciembre de 2010.

Tabla 1: Cronograma de Actividades - SisTriunfo

Fuente: Elaboración Propia

Cronograma De Actividades – Diagrama De Gantt										
ID	Actividades	Semanas								
		Dic.		Ene.				Feb.		
		1	2	3	4	5	6	7	8	9
	I. Inicio									
1	Recopilación de información.									
2	Análisis y Planteamiento del problema.									
3	Modelado de Negocio.									
4	Modelado de requerimiento									
	II. Elaboración									
5	Modelado del análisis.									
6	Modelado de Diseño.									
	• Elaboración del modelo lógico de la BD.									
	• Elaboración del modelo físico de la BD.									
	III. Construcción									
7	Elaboración del diagrama de componentes.									
8	Elaboración del diagrama de despliegue.									
9	Desarrollo del Sistema de Registro de Denuncia de las Deficiencias del Servicio de Alumbrado Público.									
	IV. Transición									
10	Realización de las pruebas respectivas.									
11	Elaboración de los manuales.									
	• Manual Técnico									
	• Manual Usuario									
12	Elaboración de la documentación final del proyecto.									

El presente diagrama de Gantt está elaborado en el marco del proyecto SisTriunfo, para la elaboración de la tesis de aplicación para optar por el grado de ingeniero de sistemas e informática

8. Presupuesto Estimado

El presupuesto estimado para la ejecución del siguiente proyecto es de S/. 1265.00 (Ver Anexo 4)

Sesión II: Desarrollo del Tema

Capítulo 1: Introducción

1.1. Contexto

Triunfo SAC, es una empresa legalmente constituida ante los órganos de control gubernamentales bajo la denominación de Pequeña y Micro Empresa (Pyme), fundada el 10 de octubre de 2010; inicia sus actividades el 01 de noviembre del mismo año en la Ca. Samanez Ocampo 895, ciudad de Iquitos - Perú.

Que cuenta con grandes volúmenes de mercadería; la Alta Dirección se ha visto en la necesidad de plantearse las opciones de adquisición o desarrollo de un Sistema de Información capaz de manejar y controlar todas y cada unas de las actividades que sus procesos tiene a buen llevar, fue a esta interrogante que se desarrolló el presente proyecto denominado “Sistema de Información de Almacén para la Pyme Triunfo SAC” o comercialmente conocido como SisTriunfo. Aplicando la metodología Rational Unified Process (RUP) – que representa, mejores prácticas en el desarrollo de Sistemas de Información de calidad, tales como una mejor Administración de Requerimientos, Arquitectura Basada en Componentes de Uso y Modelos de Software Visuales, para gestionar de manera eficiente los tiempos y los recursos con los que se cuenta dentro de la Organización – para la construcción de software de alta calidad con el objetivo de controlar los procesos de almacén, asegurar la integridad de la información registrada y obtener la máxima eficiencia en los tiempos de respuesta durante las solicitudes de reportes que apoyen en la toma de decisiones en la oficina de Logística (Compra, Almacén y Venta) sean las más adecuadas y correctas durante el ciclo de vida del negocio.

La sub oficina Almacén, en el cual se centrará el objetivo del estudio del presente informe posee un trabajador que controla la entrada de las mercaderías recién llegadas, quien a su vez se encarga de elaborar en un cuaderno en forma detallada las especificaciones de los comprobantes que llegan junto a los paquetes – fardos de mercadería – para su control en dicho depósito. Así mismo, es el que controla si los productos están en condiciones óptimas, y de no ser el caso éste procede a contactar al proveedor para el cambio del producto o devolución del dinero, debido a que es el primero en encontrar anomalías durante la revisión de los paquetes y la llamada será transferida al personal Contable para su mejor gestión y control durante dicha transacción. Almacén también controla el despacho de los productos para tienda y se pueda dar pase a su venta al público. SisTriunfo, brindará los beneficios de control, eficiencia e integridad en la información el cual promoverá el cambio ansiado dentro de los procesos de negocio que esta empresa posee.

El presente proyecto fue desarrollado en ocho (08) semanas calendario empezando el lunes 13 de diciembre del año 2010 y culminando el 04 de febrero del año 2011 en la sub oficina de Almacén. Dicha sub oficina se rige de la oficina de Logística en Ca. Samanez Ocampo 895 2do piso, Iquitos - Perú, quien es la encargada de validar las operaciones que ellos manejen comprobando así la calidad en sus actividades y desempeño profesional.

1.2. Problemática Objeto de la Aplicación

A pesar de ser una Pequeña y Micro Empresa posee las mismas fallas estructurales de una empresa de mayor ámbito – Corporación PALMEZ ubicada en av. Grau 1055 y con RUC n° 20528233075; quien tiene en el mercado más de 15 años en el mismo rubro también carece de una adecuada estandarización en sus procesos de almacén es un claro ejemplo a ello – el porqué es muy sencillo de responder, la Alta Dirección conoce empíricamente de las actividades en sus procesos y esto es aún peor que no conocerlas siendo la frase principal en ellos *¿debe ser así...?* Generándose dudas entre sus encargados y no *¿cómo podemos hacer para mejorar...?* y realizar un análisis de brechas en el entorno de empresas que poseen éxito en calidad de sus procesos.

Almacén, que representa una parte importante para el funcionamiento de la empresa tiene en sus actividades las siguientes problemáticas actuales:

1. Control deficiente en:

- Los productos y sus stocks.
- Los proveedores y transportistas.
- El registro de compra.
- La salida de mercadería.

Que impide al dueño de proceso controlar la información en tiempo real (por ejemplo: para gestionar las compras y el envío de mercadería debido a los costos involucrados en tal o cual proveedor y/o transportista al momento de realizar una nueva transacción –compra o transporte de mercadería–).

2. Errores de carácter humano en:

- Los productos y sus stocks.
- Los proveedores y transportistas.
- El registro de compra.
- La salida de mercadería.

Que impiden al dueño de proceso verificar si la información registrada es de carácter confiable (por ejemplo: al registrar y modificar constantemente el cuaderno de stock de almacén, el cual pueden generar un déficit o incremento en almacén o tienda convirtiéndose en malas tomas de decisiones o inventarios inconsistentes dentro de la empresa).

3. Demoras en:

- La generación de reportes y tiempos de respuesta a solicitudes de reportes principales (compras, guías de remisión, notas de crédito y kardex de almacén), teniendo en cuenta que almacén representa para Triunfo SAC la columna vertebral de las futuras operaciones de procesos organizativos.

1.3. Objetivos del Proyecto

1.3.1. *Objetivo General*

Desarrollar un Sistema de Información, que automatice los procesos internos de la Organización, el cual dará solución a los problemas estructurales de la sub Oficina Almacén el cual permitirá a Triunfo seguir evolucionando con otros módulos que permitan la integración plena de todos sus procesos de ingreso y salida de mercadería.

1.3.2. *Objetivos Específicos*

- Tener un estricto control en los procesos e integridad en la información en los productos y sus stocks que se generen.
- Tener un estricto control en los procesos e integridad en la información en las transacciones de proveedores y transportistas que se generen, y de esta forma gestionar de la manera más apropiada los recursos (económicos o de factor tiempo) que cuenta la empresa.
- Tener un estricto control en los procesos e integridad en la información en las compras que se generen, permitiendo de esta forma la confiabilidad irrestrictita en la entrada de productos a almacén.
- Tener un estricto control en los procesos e integridad en la información en las transferencias que se generen, permitiendo de esta forma la confiabilidad irrestrictita en la salida de productos a almacén.
- Generar reportes eficientes que apoyen en la toma de decisiones que Triunfo SAC necesita.

Capítulo II: Descripción del Diseño de la Solución

2.1. Técnicas de Recolección de Datos

De acuerdo a una encuesta realizada a 15 profesionales (Ver Anexo 3), se determinaron 3 métodos para el correcto levantamiento de información, que son los siguientes:

- 1) Recopilación Documentaria: Fotocopiar y escanear cada uno de los documentos importantes – tales como boletas, facturas, notas de crédito, registro de compras y ventas, guías de remisión y otros reportes de carácter indispensable – empleados en todos los procesos funcionales por los dueños de procesos.
- 2) Entrevistas y encuestas: Con los dueños de Procesos, para determinar el nivel de criticidad en los procesos, de manera tal que se pueda enfocar los más importantes e ir de menos a más durante la fase de producción.
- 3) Observación Directa: Visitas constantes en horarios laborales permitirán un mejor entendimiento de la problemática y una mejor visión de la propuesta de solución.

Para validar los indicadores se tomo en cuenta la siguiente técnica de recolección de datos:

- 1) Entrevistas y encuestas: con el dueño de proceso de almacén, para demostrar el grado de confiabilidad y rendimiento mostrado por el Sistema de Información SisTriunfo.

2.2. Metodología y Herramienta a emplear

2.2.1. Metodologías

De acuerdo a una encuesta realizada a 15 profesionales (Ver Anexo 3), se determinó la Metodología Procesos Integrados Racionalmente (Rational Unified Process RUP) que ayudará a la pronta culminación y solución de los procesos más críticos de la Organización en este corto periodo de tiempo.

Definición RUP

RUP es una metodología de Ingeniería de Software. Provista de una disciplina aproximada para asignar tareas y responsabilidades dentro de una organización en perfeccionamiento. Es excelente para asegurar la producción de software de alta calidad que reúne todas las necesidades de los usuarios finales, dentro de un entorno limitado.

RUP es una guía para el cómo usar efectivamente el Lenguaje de Modelado Integrado. UML es un lenguaje estándar que nos permite comunicarnos claramente entre los requerimientos, análisis y diseño. UML fue creado por *Rational Software*, y ahora pertenece a la organización de estándares de Grupos de Administración de Objetos (Object Management Group OMG).

The Rational Unified Process describe cómo desplegar efectivamente aproximaciones del negocio durante el desarrollo de software para que el equipo de desarrollo realice tareas precisas. RUP aplica en su metodología 6 mejores prácticas para el desarrollo de software de alta calidad y son las siguientes:

- 1ero.- Software de Desarrollo Iterativo
- 2do.- Administración de requerimientos.
- 3ero. – Arquitectura Basada en Componentes de Uso.
- 4to.- Modelos de Software Visuales.
- 5to.- Verificación de Calidad de Software.
- 6to. – Control de Cambios para el Software.

De la metodología RUP nos enfocaremos en el 2do, 3ero y 4to puntos de desarrollo de buenas prácticas al momento del análisis, diseño y producción:

2do.- Administración de requerimientos: RUP describe cómo elegir, organizar y documentar la funcionalidad y contrastes de un requerimiento; controla la toma de decisiones durante los procesos del negocio, y fácilmente captura y comunica los requerimientos del negocio. Las ideas o pensamientos de escenarios y casos de uso restringidos en los procesos y demostrarán la forma de capturar los requerimientos funcionales y harán que se controle en tiempo de diseño, implementación y pruebas de software, haciéndolo más parecido al producto final que el usuario necesita.

3ero. – Arquitectura Basada en Componentes de Uso: Este proceso se enfoca en desarrollar de manera fácil una arquitectura ejecutable robusta, priorizando recursos para un desarrollo escalable. Describe cómo se diseñará una arquitectura con características flexibles, capaz de adaptarse a los cambios, es intuitivo y entendible, además de promover el re-uso efectivo del software o de alguno de sus componentes. RUP soporta *desarrollo basado en Componentes*. Estos componentes son módulos no triviales, subsistemas que se ejecutan en una función específica. RUP provee una aproximación sistemática a la definición de una arquitectura usando nuevos y existentes componentes. Estos son ensamblados con una arquitectura bien definida y personalizada o en una infraestructura En Línea, para que la industria del re-uso emerja.

4to. - Modelos de Software Visuales: Este proceso muestra cómo un modelo de software es capturado estructuralmente. Permite mostrar detalles y escribir códigos usando “Edificios Gráficos”. Abstracciones gráficas que ayudan a comunicar los diferentes aspectos del Software; ver cómo los elementos del sistema trabajan juntos. Confirma que la creación del Edificio (estereotipo) es consistente con el código, mantiene consistencia entre el diseño y la implementación y evita que la comunicación sea ambigua.

2.2.2. Herramientas

- Rational Rose Enterprise Edition, para el desarrollo del modelado del Sistema de Información.
- Microsoft SQL Server 2008, para el desarrollo de la arquitectura de la Base de Datos del Sistema de Información
- Microsoft Visual Studio 2008, en su lenguaje de programación Visual Basic para el desarrollo del Sistema de Información.
- Platinum Erwin ERX 3.5.2., para diseñar los diagramas lógicos y físicos de la Base de Datos.

2.3. Descripción del desarrollo de la Solución

El desarrollo del presente sistema de información se enfoca a solucionar aspectos críticos principales que se cierne dentro de la Organización y posibles amenazas que a futuro pueden repercutir en los procesos del negocio, se mencionan de la siguiente manera:

- Generar en cortos periodos de tiempos reportes principales para el control del almacén (productos, proveedores, transportistas, compras, guías de remisión, notas de crédito y kardex) ;
- SisTriunfo cuenta con una base de datos normalizada para almacén, ello brindará la integridad y la confianza necesaria a la Alta Gerencia, así como al personal que labora en ella;
- Flexibilidad y control en todas las transacciones principales de registro, actualización y consultas, a la gestión de proveedores, transportistas, usuarios, productos y registro de compras;
- Control de stock y revisiones precisas dentro del almacén;
- Al centralizar la información, las políticas de seguridad de TI harán que impidan la fuga de información impertinente por empleados que laboran en dicho almacén;
- Se propone la automatización e integración de cada una de las áreas involucradas en el Negocio en un mediano plazo;

2.4. Indicadores de Evaluación de la Solución

- Tiempo en las transacciones de almacén (productos, proveedores, transportistas, compras, guías de remisión y notas de crédito).
- Tasa de errores en las transacciones de almacén (productos, proveedores, transportistas, compras, guías de remisión y notas de crédito).
- Tiempos de respuesta a solicitudes de reportes.

2.5. Relación de Entregables

Tabla 2: Lista de Entregables

Ítem	Entregables	Estado
01	Informe de Trabajo Práctico	Entregado
02	Código Impreso del Sistema de Información SisTriunfo	Entregado
03	Ejecutable de Sistema de Información SisTriunfo	Entregado
04	Diseño Lógico y Físico de Base de Datos SisTriunfo	Entregado

Fuente: Elaboración Propia

Capítulo III: Desarrollo de la Solución de la Propuesta

3.1. Modelado del Negocio

3.1.1. Caso de Uso del Negocio

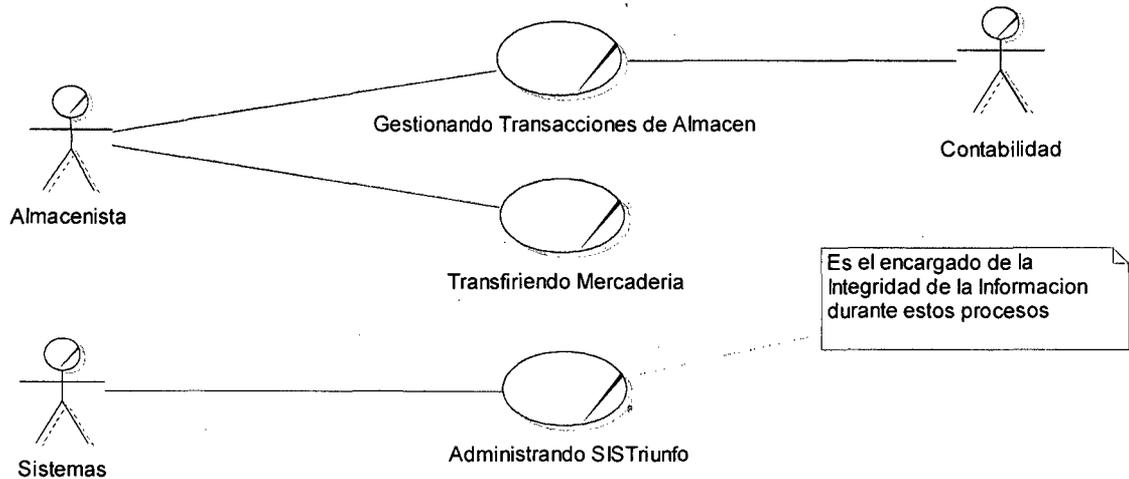


Ilustración 3: Caso de Uso del Negocio – Triunfo
Fuente: Elaboración Propia

Procesos:

- a) **Gestionando Transacciones de Almacén:** Es el proceso principal por el cual se derivan los siguientes subprocesos:
 - **Registro de Compra:** Es el proceso mediante el cual el almacenista verificará con Guía de Remisión y Comprobante (Factura o Boleta) en mano el ingreso de los productos al almacén de la empresa. Se registrará en ella además de las cantidades, los precios y totales.
 - **Gestión de Proveedores:** Son las actividades de registro, actualización, consulta, dar de baja a un proveedor.
 - **Gestión de Transportistas:** Son las actividades de registro, actualización, consulta, dar de baja a un Transportista.
 - **Gestión de Productos:** Son las actividades de registro, actualización, consulta, dar de baja a Productos.
 - **Nota de Crédito:** Es el proceso mediante el cual se da a conocer la devolución de parte o el total de mercadería en mal estado.
- b) **Transfiriendo Mercadería:**
 - **Registrar una Guía de Remisión:** sus actividades comienzan enviando una solicitud de aceptación para la transferencia de mercadería a la tienda, sin embargo, si la tienda encuentra inconsistencia durante esta actividad rechazará la solicitud y la guía no será registrada hasta sanear dicho impase entre la tienda y almacén.

- c) Administrando SisTriunfo: Es el proceso por el cual se gestionan los procesos de personal y usuarios al sistema SisTriunfo.
 - Gestión de Personal Son las actividades de registro, actualización, consulta, dar de baja a un Personal.
 - Gestión de Usuario Son las actividades de registro, actualización, consulta, dar de baja a un Usuario.

3.1.2. *Escenarios y Procesos*

Registro de Compra: El encargado de almacén primero registra al proveedor, luego procede a registrar los datos del transportista; una vez realizado ambos caso procede con la revisión de la mercadería llegada, si la mercadería llega en mal estado, el encargado procede a comunicarse con el proveedor para que se genere una Nota de Crédito; dicha Nota de Crédito validará el cambio del producto o la devolución del dinero por los productos que se encuentran en mal estado; saneado este impase o no habiéndose encontrado productos en mal estado se procede a ingresar los comprobantes y los detalles de las compras; de haberse encontrado con algún producto que no esté registrado en SisTriunfo, el encargado podrá acceder a un formulario para su correcto llenado del producto no encontrado; terminando esta actividad el encargado coteja la información de orden de compra con los productos adquiridos y registra que productos no han sido comprados para su posterior análisis en Logística. Además se verifica si el presupuesto estimado en la orden de compra refleja el 100% del valor total de la compra de no ser así el encargado registra el monto real no consumido durante todo el proceso; para finalizar el proceso de registro de compras el encargado actualiza los Stock del Almacén para futuras transferencias a Tienda.

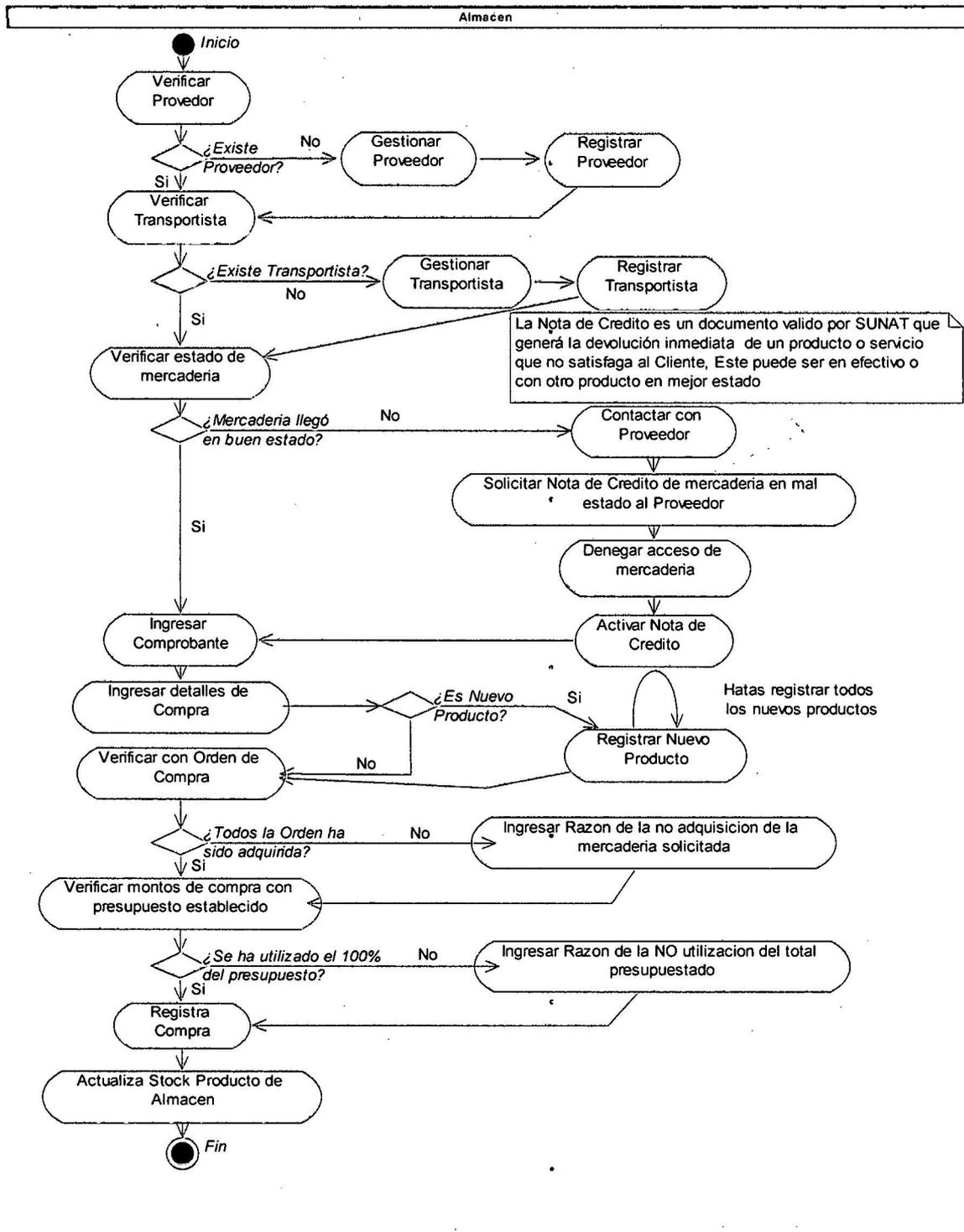


Ilustración 4: Diagrama de Actividades Registrando Compra
 Fuente: Elaboración Propia

Transfiriendo Mercadería: Para realizar la transferencia de mercadería a la tienda se hará uso de una Guía de Remisión, el cual es un documento válido por SUNAT, de lo contrario puede ser embargada y puesto a disposición de éste organismo gubernamental, en primer lugar se seleccionan los productos y las cantidades a ser enviadas; estableciendo los posibles precios para ponerlos a la venta, seleccionamos el destino de arribo, y procedemos a consultar al encargado de tienda si acepta el envío de mercadería, las razones por las cuales el encargado de tienda no las acepte pueden ser por el mal estado en los productos o inconsistencia en la cantidad registrada y los productos reales, una vez saneada el impase generado se procede a registrar la Guía de Remisión original culminando de esta manera dicho proceso

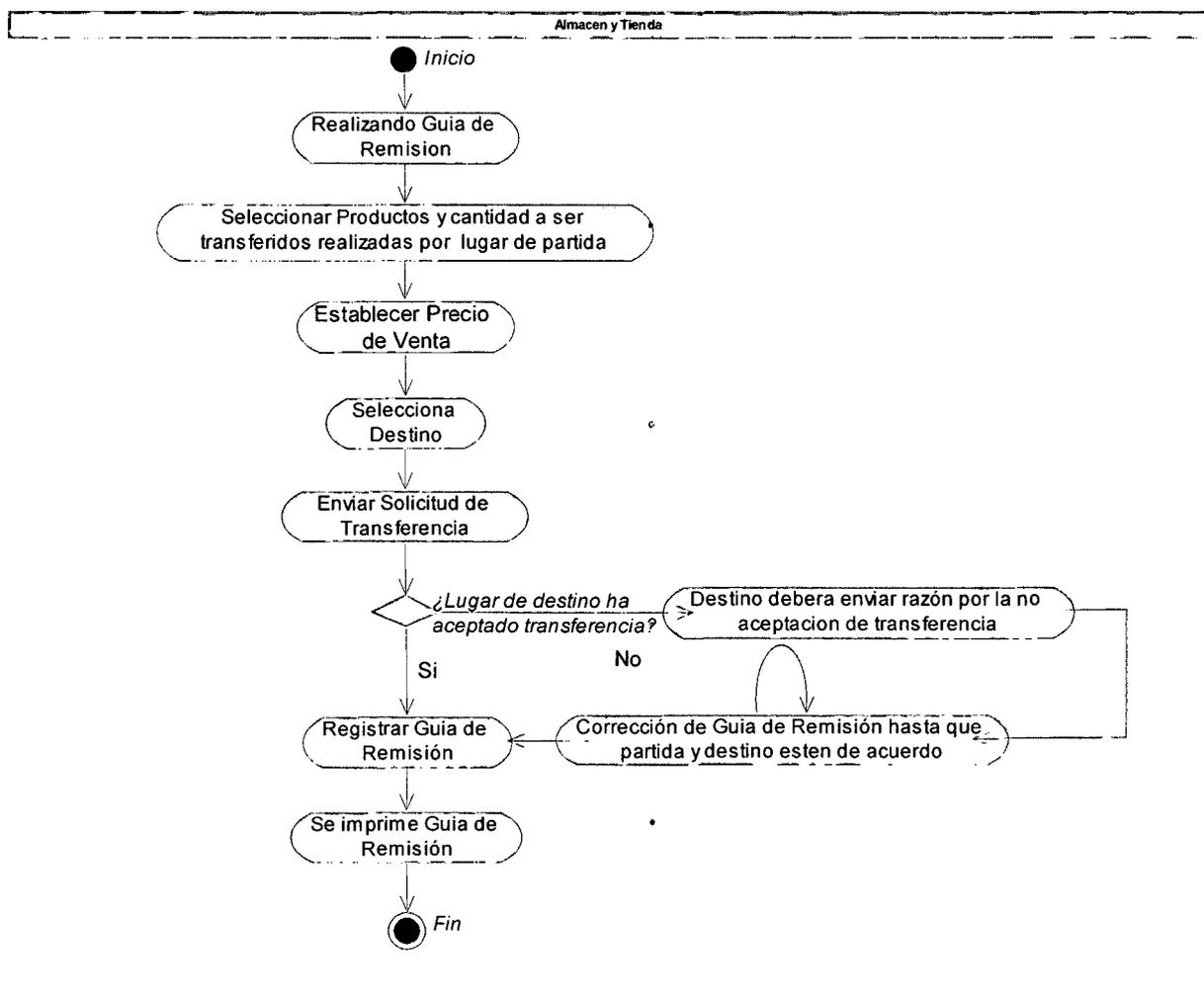


Ilustración 5: Diagrama de Actividades Transfiriendo Mercadería
Fuente: Elaboración Propia

Administrando SisTriunfo: La oficina de TIC, se encarga de velar por la integridad de la información realizando función de control y seguimiento de posibles inconsistencias durante el registro de las compras y las transferencias realizadas corroborando que toda la información almacenada sea íntegra y confiable.

Como Triunfo SAC no posee gestión de personal la oficina de TIC velará por el mejor control de sus empleados y usuarios de SisTriunfo.

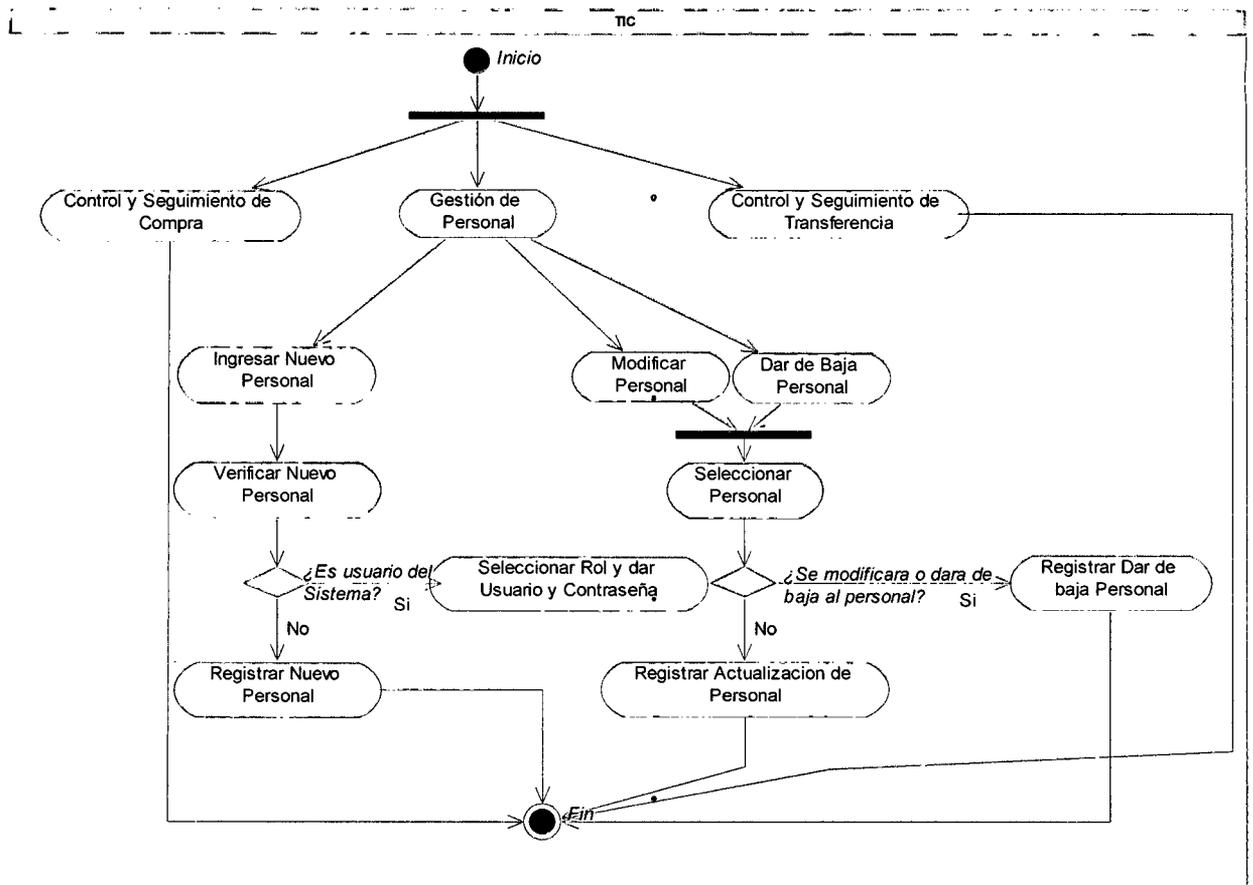


Ilustración 6: Diagrama de Actividades Administrando SisTriunfo
Fuente: Elaboración Propia

3.1.3. *Modelo de Objetos del Negocio*

Tabla 3: Lista de Workers y Objetos de Modelo de Objetos SisTriunfo

Trabajadores del Negocio (Workers)	Objetos Principales	Objetos Dependientes
Sistemas	Personal	Cargo
		Usuario
	Guía Remisión	
Contador	Compras	Comprobantes
		Proveedor
	Guía Remisión	
Almacenista	Compras	Comprobantes
		Proveedor
	Producto	Marca
		Modelo
		TipoProducto
		Proveedor
		Compras
	Guía Remisión	
	Stock Producto	Guía Remisión
	Nota de Crédito	Comprobaste

Fuente: Elaboración Propia

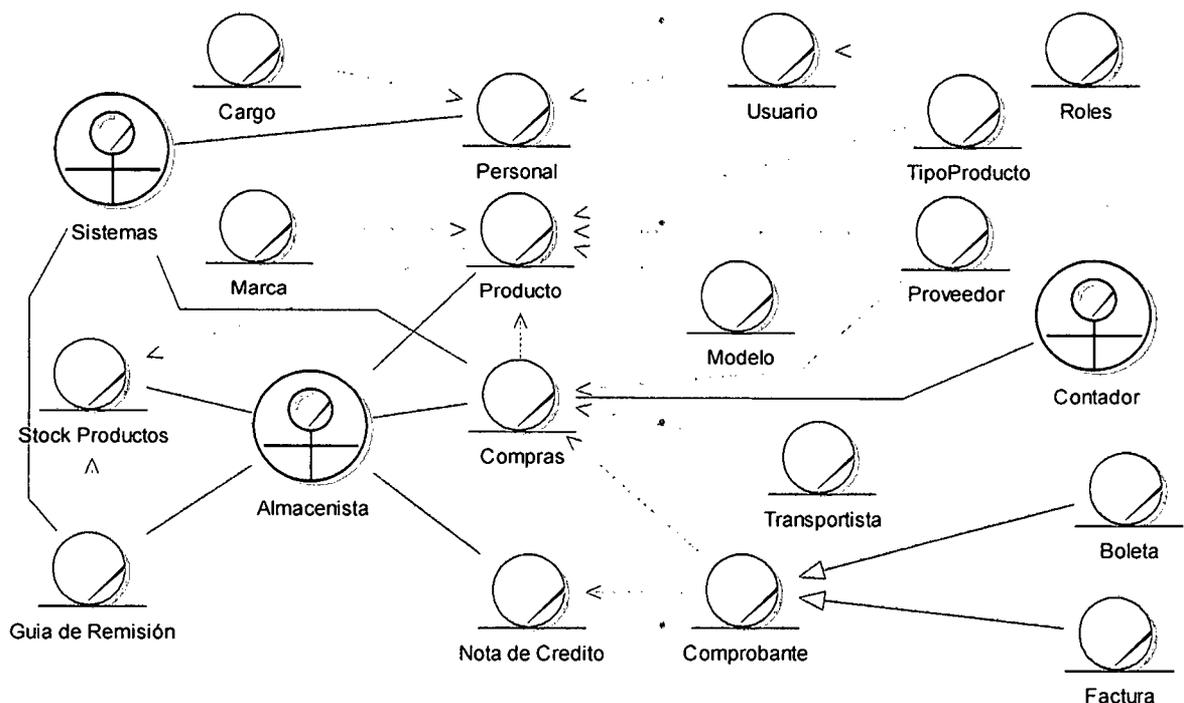


Ilustración 7: Modelo de Objetos de SisTriunfo
 Fuente: Elaboración Propia

3.1.4. Modelo de Dominio

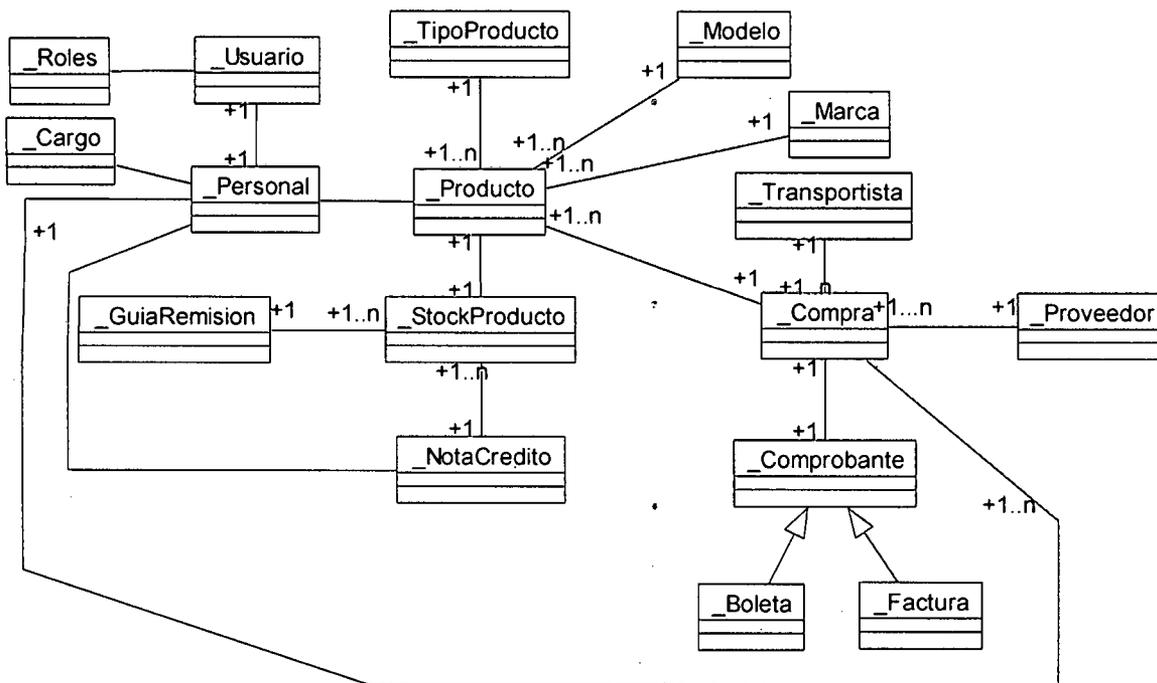


Ilustración 8: Modelo de Dominio SisTriunfo
Fuente: Elaboración Propia

3.2. Modelado de Requerimientos

3.2.1. Propósito

Identificar de modo gráfico los requerimientos principales para el buen desempeño de los procesos del negocio que la sub Oficina de Almacén maneja.

3.2.2. Alcance

Abarcar todos los aspectos que la sub Oficina de Almacén controla dividiendo sus actividades e integrando sus procesos.

3.2.3. Descripción de Stakeholders y Usuarios

3.2.3.1. Resumen de Stakeholders

Tabla 4: Resumen de Stakeholders

Stakeholders		
Gerente General	Lic. Clara Elena Álvarez Ruiz de Ramírez	Dueña del Negocio y principal sponsor para la implementación del Sistema de Información
Dueño de Proceso Almacén	Sr. Edwin Murrieta Rojas	Encargado del Almacén Triunfo
Dueño de Proceso Contable	CPC. Carlos Ramírez Hidalgo	Encargado del control de procesos contables en la Organización.

Fuente: Elaboración Propia

3.2.3.2. Resumen de Usuarios

Tabla 5: Resumen de Usuarios

Usuarios	
Almacenista	Usuario encargado de los procesos realizados dentro del almacén
Administrador Sistema	Usuario encargado de administrar al personal y usuarios del sistema

Fuente: Elaboración Propia

3.2.4. Descripción Global del Producto

3.2.4.1. Perspectiva del Producto

Controlar los procesos de la sub oficina de Almacén, permitiendo a la Oficina de Logística manejar las otras sub Oficinas de la Organización.

3.2.4.2. Resumen de Características

- SisTriunfo encaja perfectamente en las necesidades que Almacén ha solicitado.
- Los reportes producidos por el sistema facilitarán las labores de los miembros de logística.

3.2.5. *Caso de Uso de Requerimiento*

a) Caso de Uso de Requerimiento Gestionando Transacción de Almacén

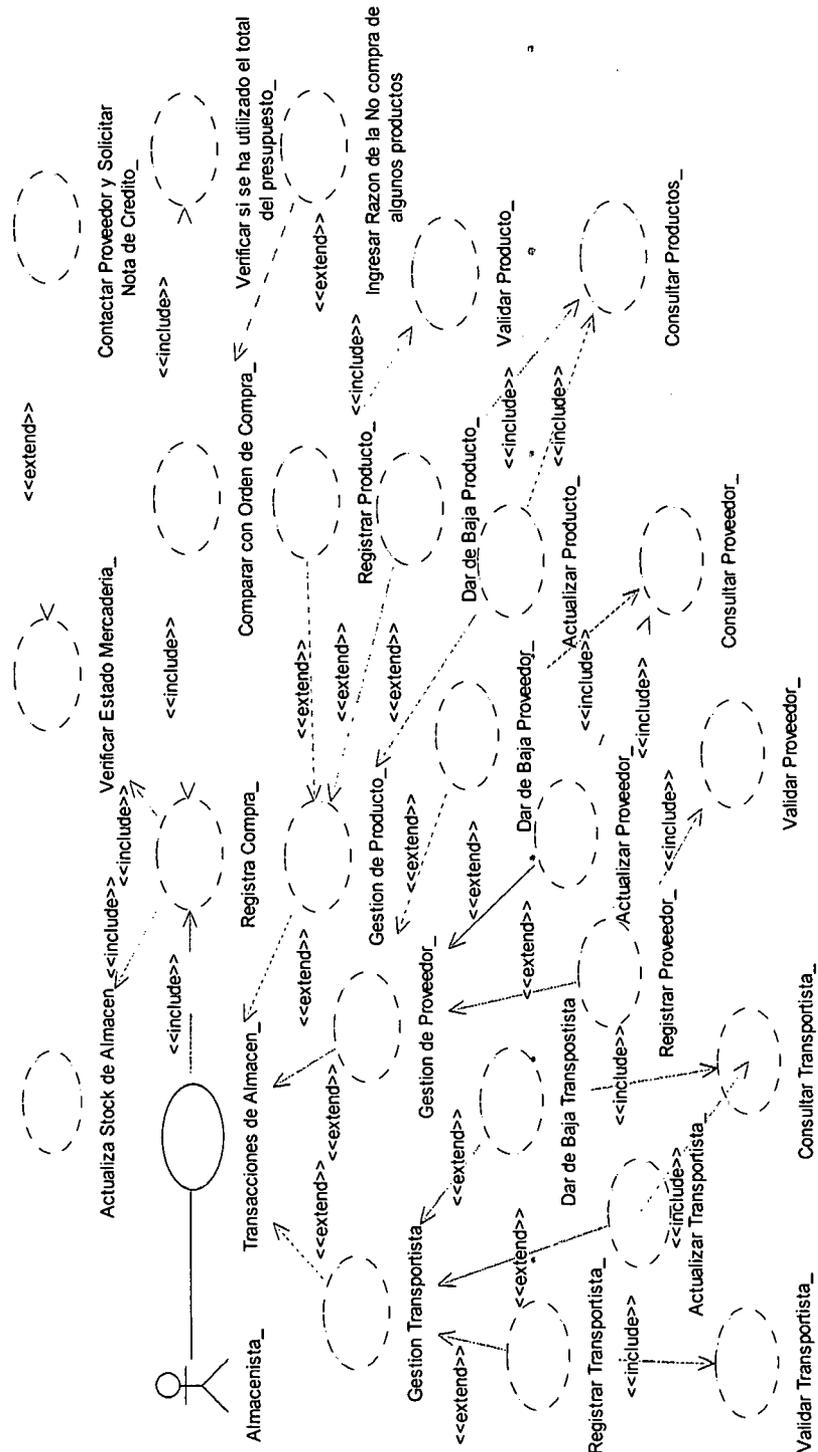


Ilustración 9: Caso de Uso de Requerimiento Gestionando Transacción de Almacén
 Fuente: Elaboración Propia

b) Transferir Mercadería

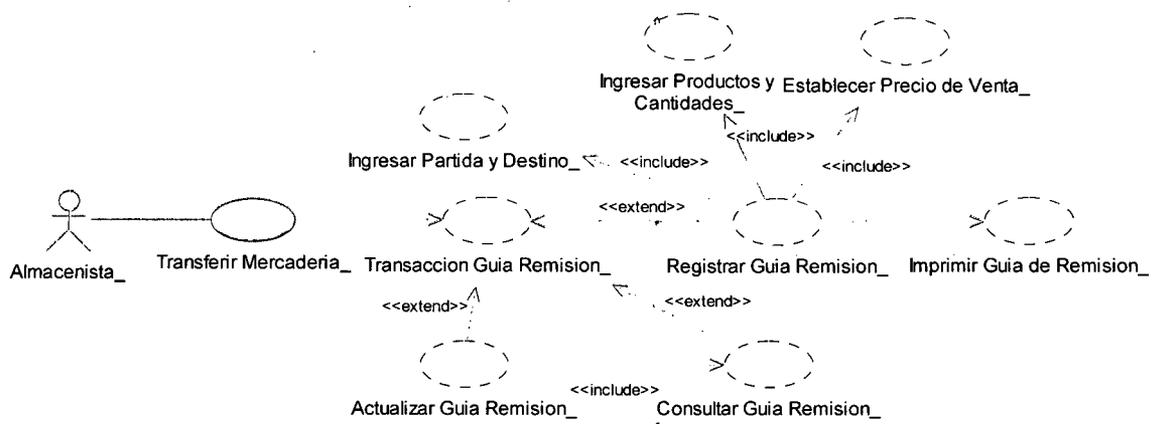


Ilustración 10: Caso de Uso de Requerimiento Transferir Mercadería
Fuente: Elaboración Propia

c) Administrando SisTriunfo

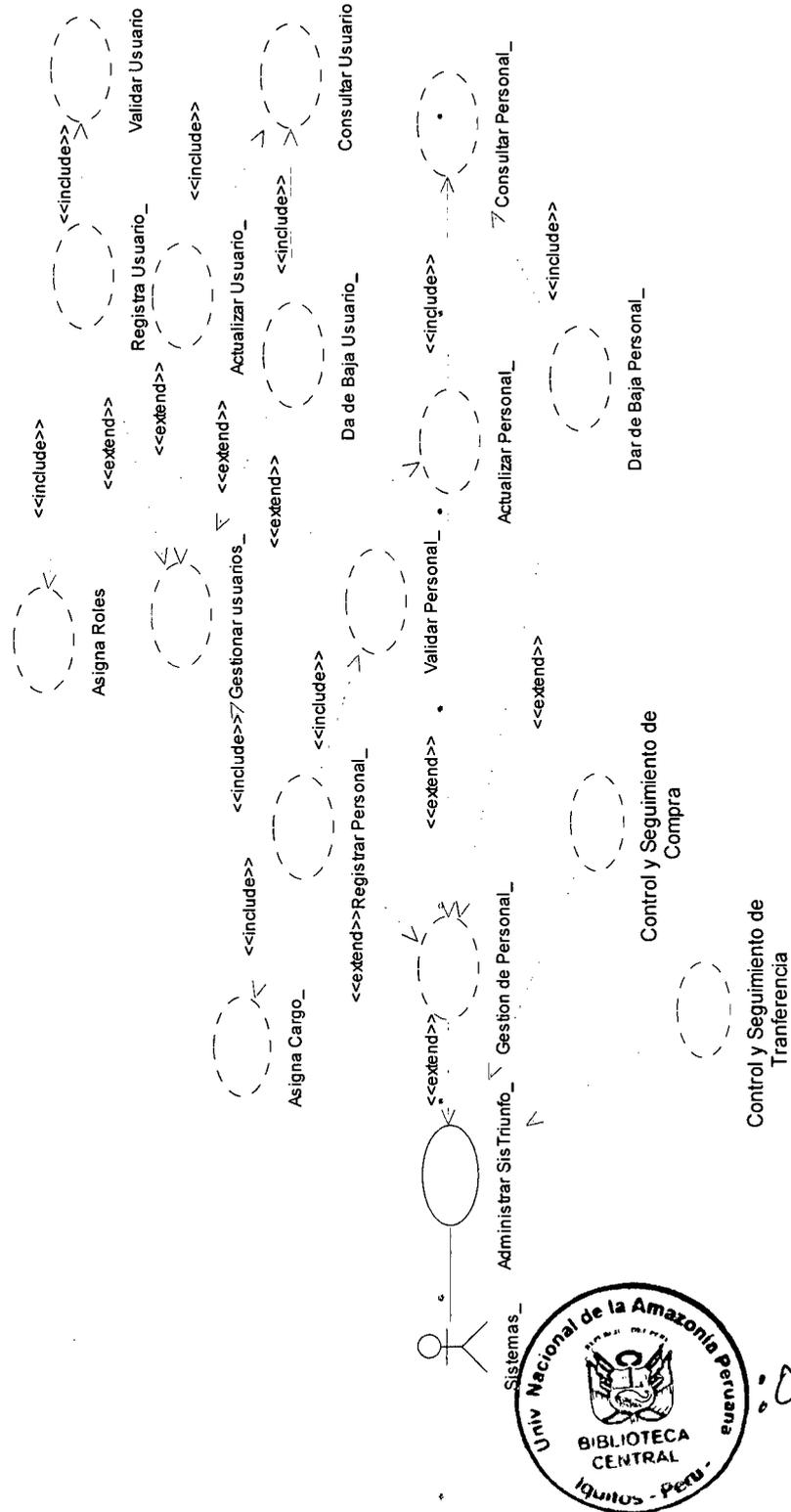


Ilustración 11: Caso de Uso de Requerimiento Administrar SisTriunfo
 Fuente: Elaboración Propia



00072

3.2.6. Especificaciones de Caso de Uso

Tabla 6: Especificación de Caso de Uso

Caso de Uso	Caso de Uso de Requerimiento	Especificación
Transacción Almacén	Transacción Compra_	Este proceso es alimentado por los subprocesos de proveedores, transportistas y productos.
	Transacción Proveedor_	Este proceso de transacciones genera los subprocesos de Registro, Actualización, Consulta y Dar de Baja a un Proveedor
	Transacción Transportista_	Este proceso de transacciones genera los subprocesos de Registro, Actualización, Consulta y Dar de Baja a un Transportista.
	Transacción Producto_	Este proceso de transacciones genera los subprocesos de Registro, Actualización, Consulta y Dar de Baja a un Producto.
	Transacción Nota Crédito_	Este proceso de transacciones genera los subprocesos de Registro, Actualización y Consulta a una Nota de Crédito.
Transfiriendo Mercadería	Transacción Guía Remisión_	Este proceso de transacciones genera los subprocesos de Registro, Actualización y Consulta a una Guía de Remisión
Administrando SisTriunfo	Gestión de Personal_	Se encarga de la gestión de personal este proceso de transacciones genera los subprocesos de Registro, Actualización y Consulta a un Personal.
	Gestión de Usuario_	Se encarga de la gestión de Usuarios este proceso de transacciones genera los subprocesos de Registro, Actualización y Consulta a un Usuario.
	Control y Seguimiento de Transferencia_	Se encarga control y seguimiento de transferencias este proceso entregara información del historial de las transferencias realizadas.
	Control y Seguimientos de Compras	Se encarga control y seguimiento de compras este proceso entregara información del historial de las compras realizadas.

Fuente: Elaboración Propia

3.3. Análisis

3.3.1. Diagrama de Colaboración

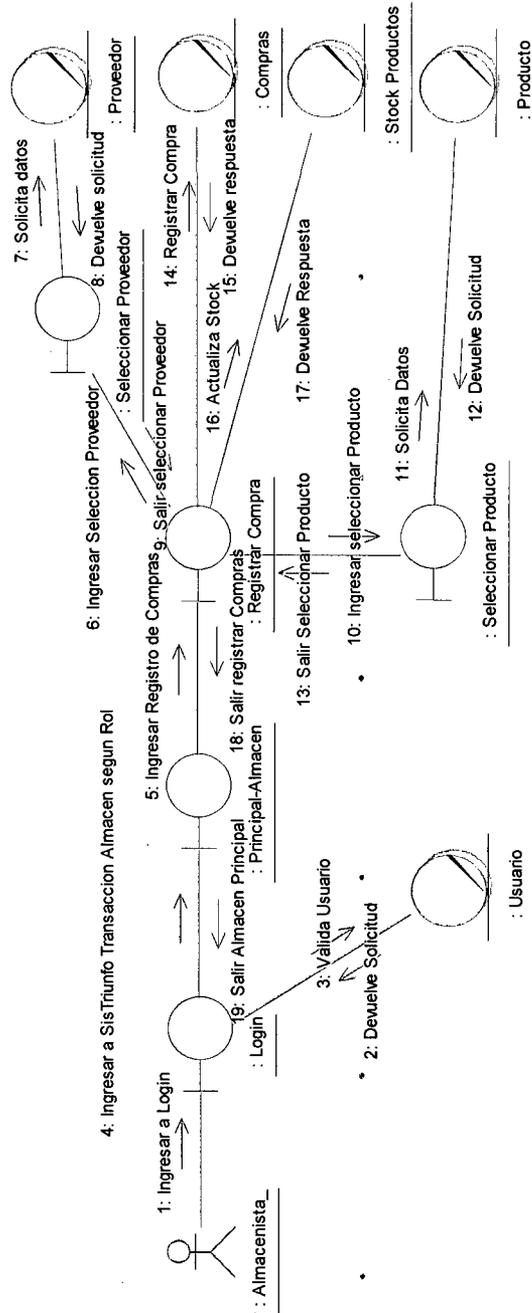


Ilustración 12: Diagrama de Colaboración Registrar Compra
 Fuente: Elaboración Propia

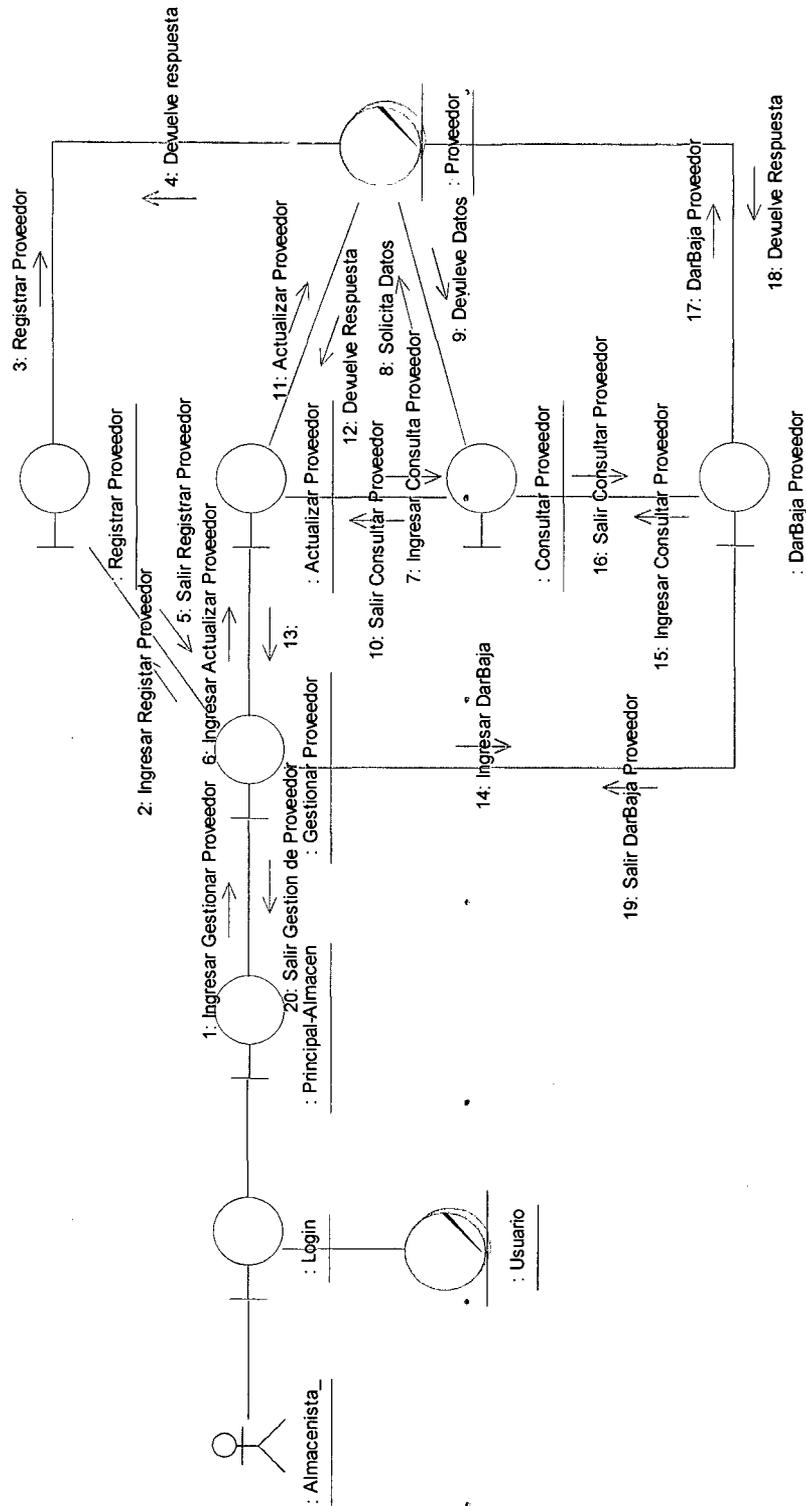


Ilustración 13: Diagrama de Colaboración Gestionar Proveedor
 Fuente: Elaboración Propia

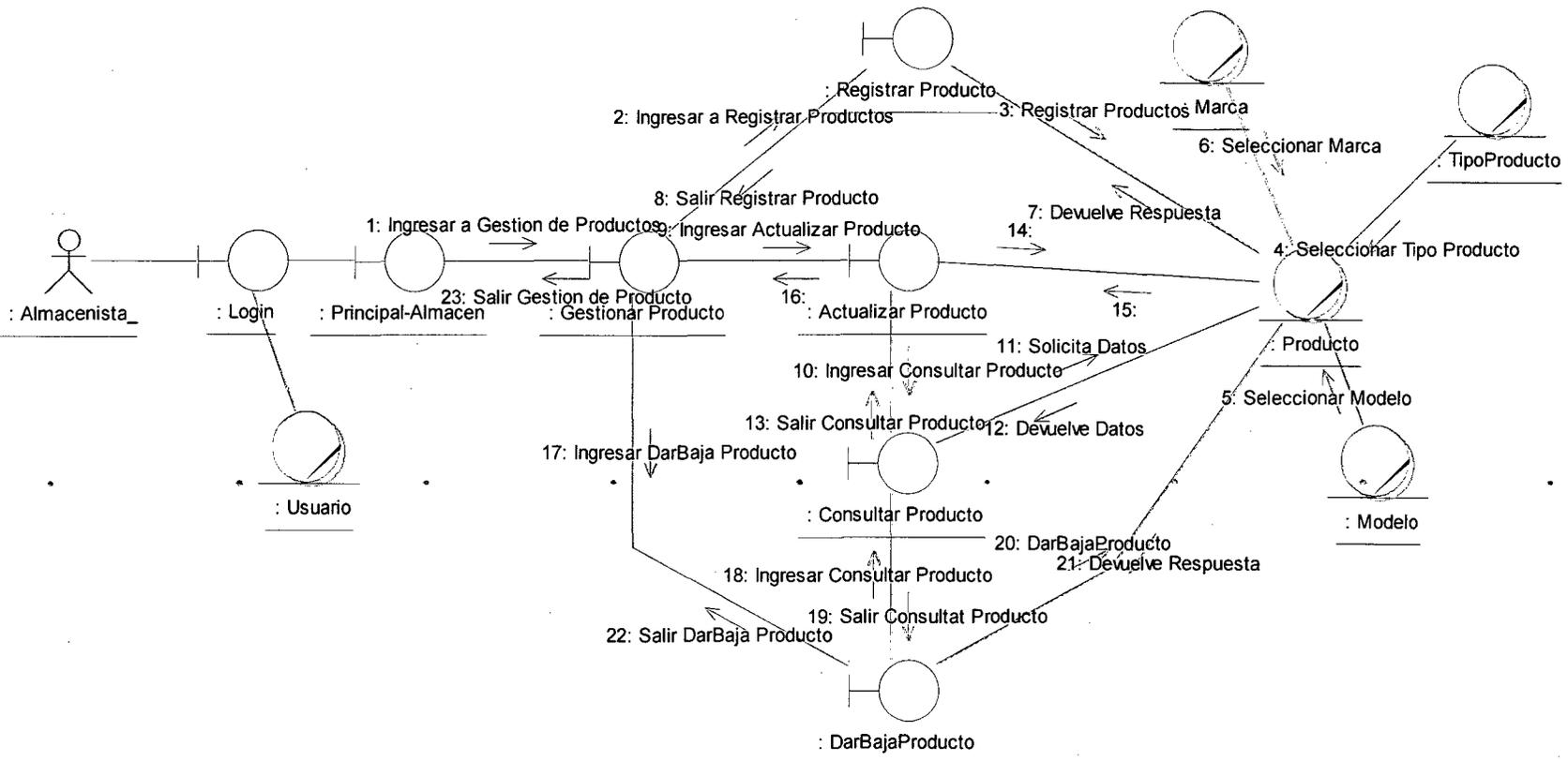


Ilustración 14: Diagrama de Colaboración Gestionar Producto
 Fuente: Elaboración Propia

3.3.2. Diagrama de Secuencia

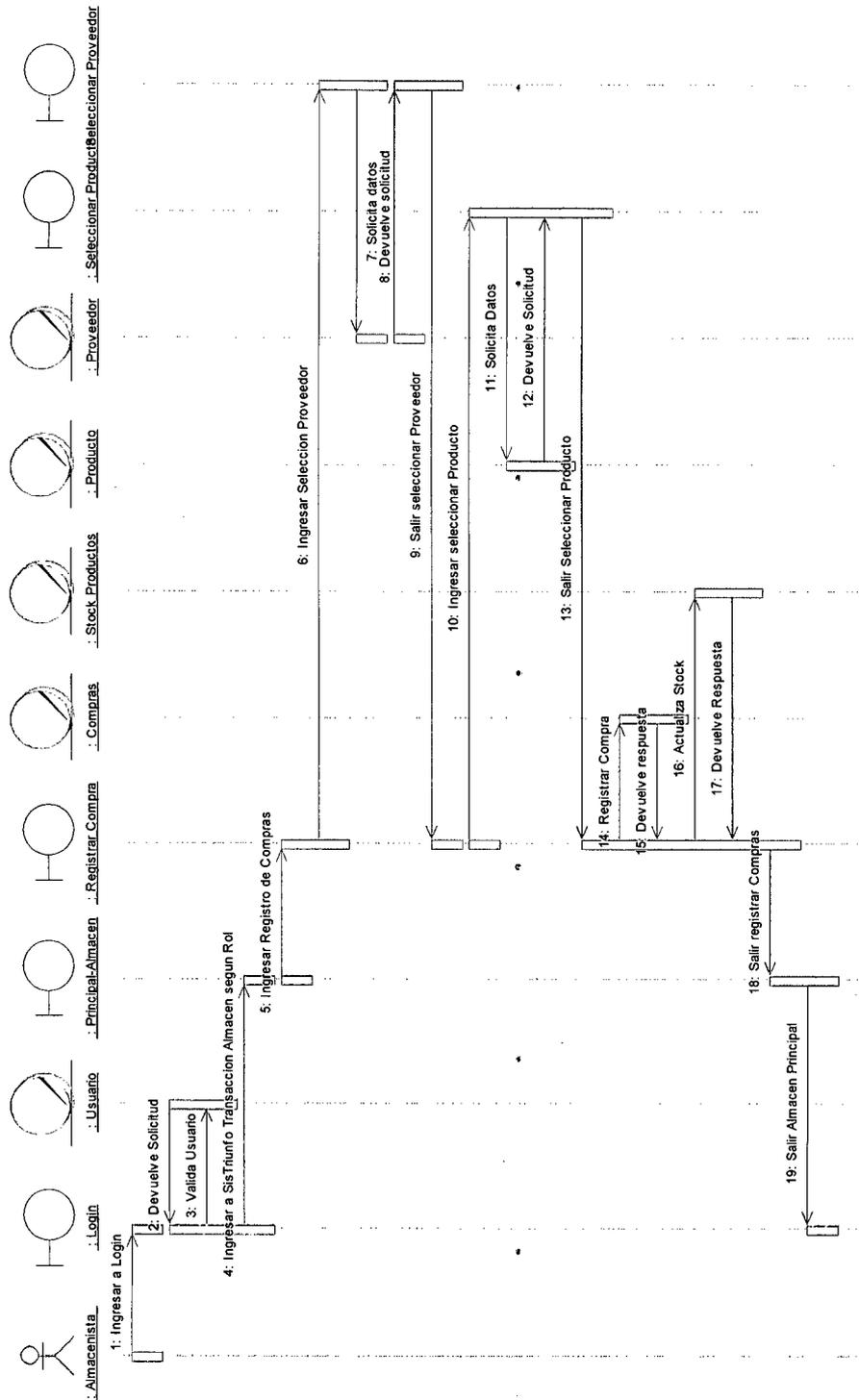


Ilustración 15: Diagrama de Secuencia Registrar Compra
 Fuente: Elaboración Propia

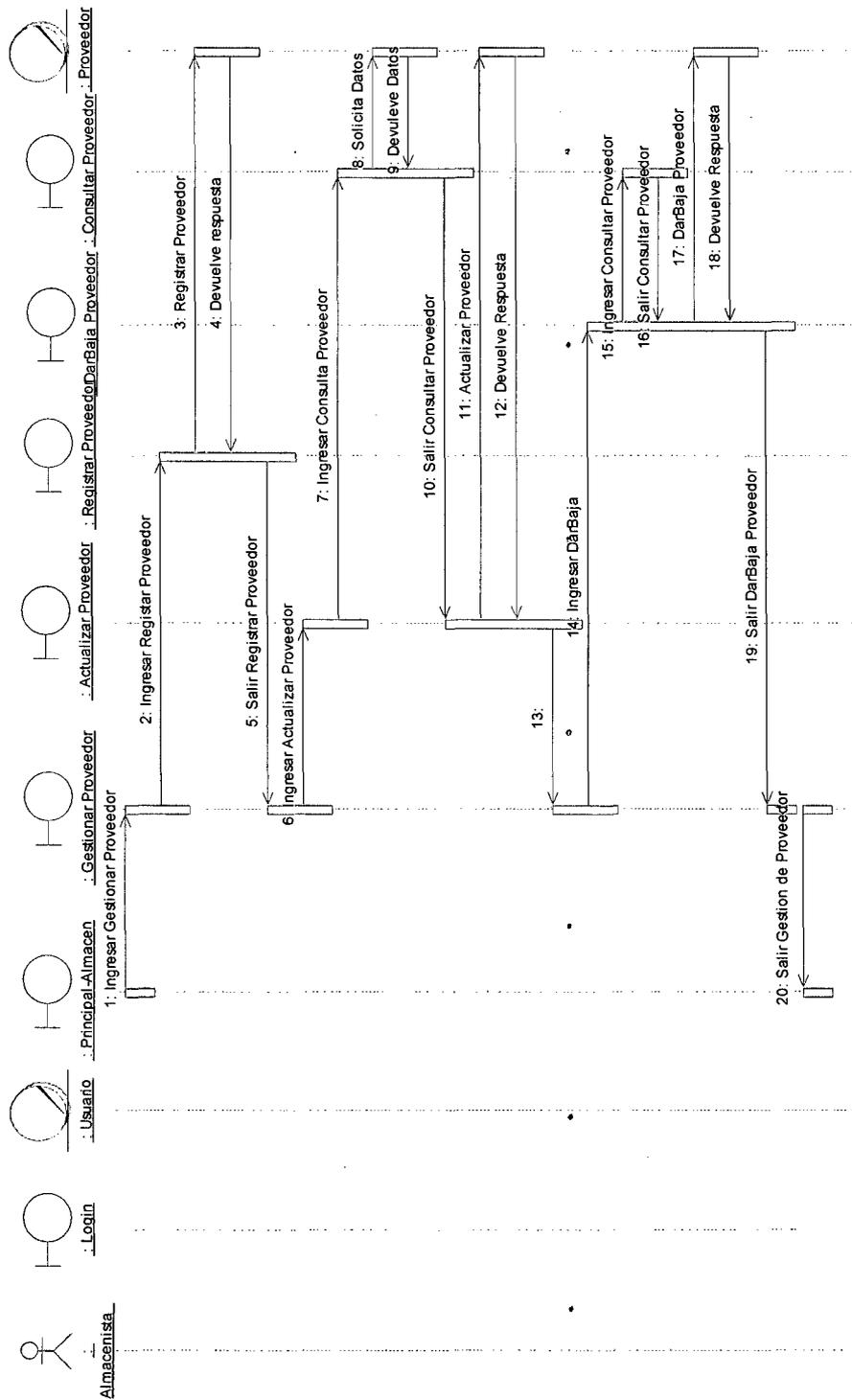


Ilustración 16: Diagrama de Secuencias Gestionar Proveedor
 Fuente: Elaboración Propia

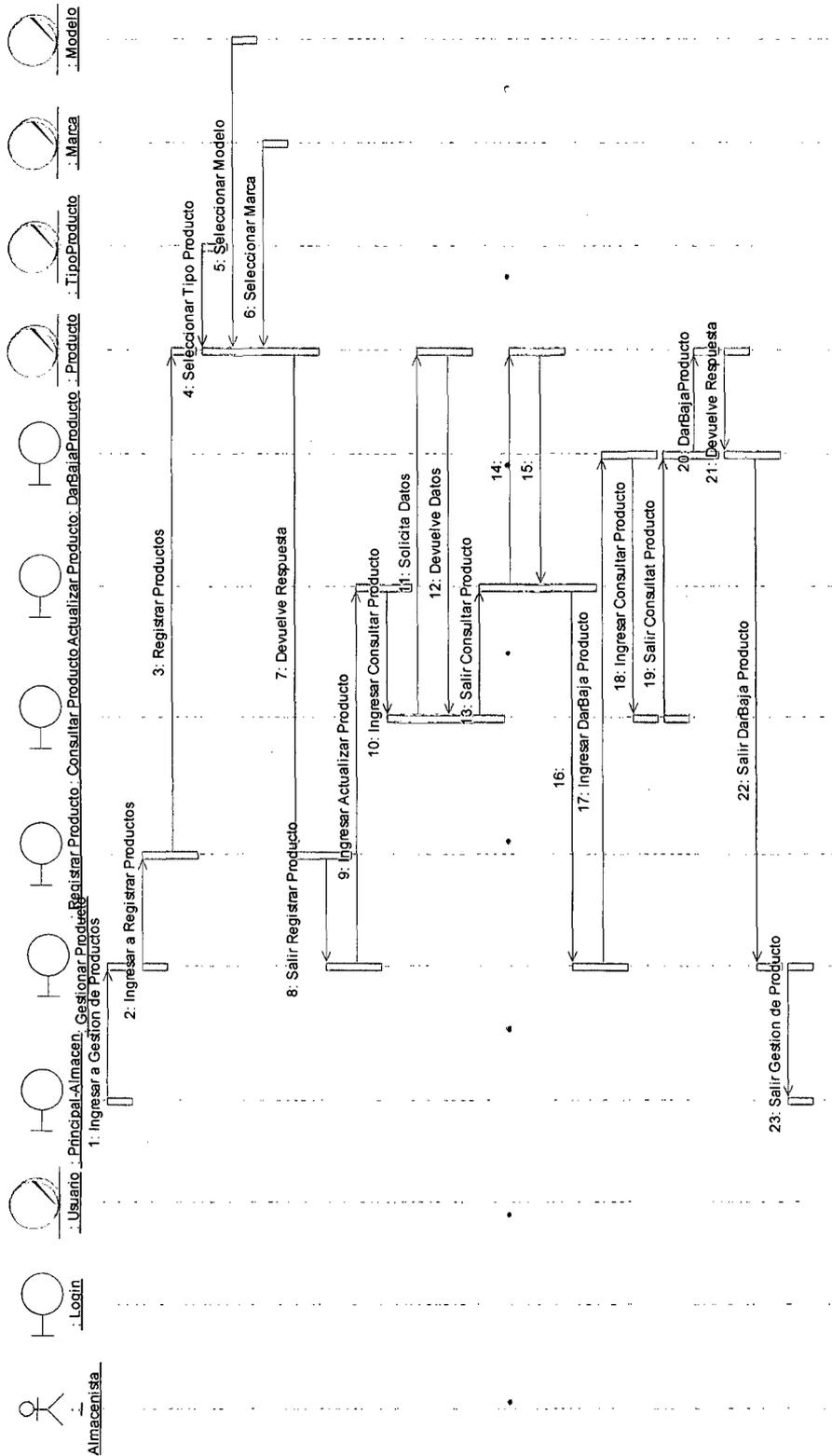


Ilustración 17: Diagrama de Secuencia Gestionar Producto
 Fuente: Elaboración Propia

3.3.3. Diagrama de Clases

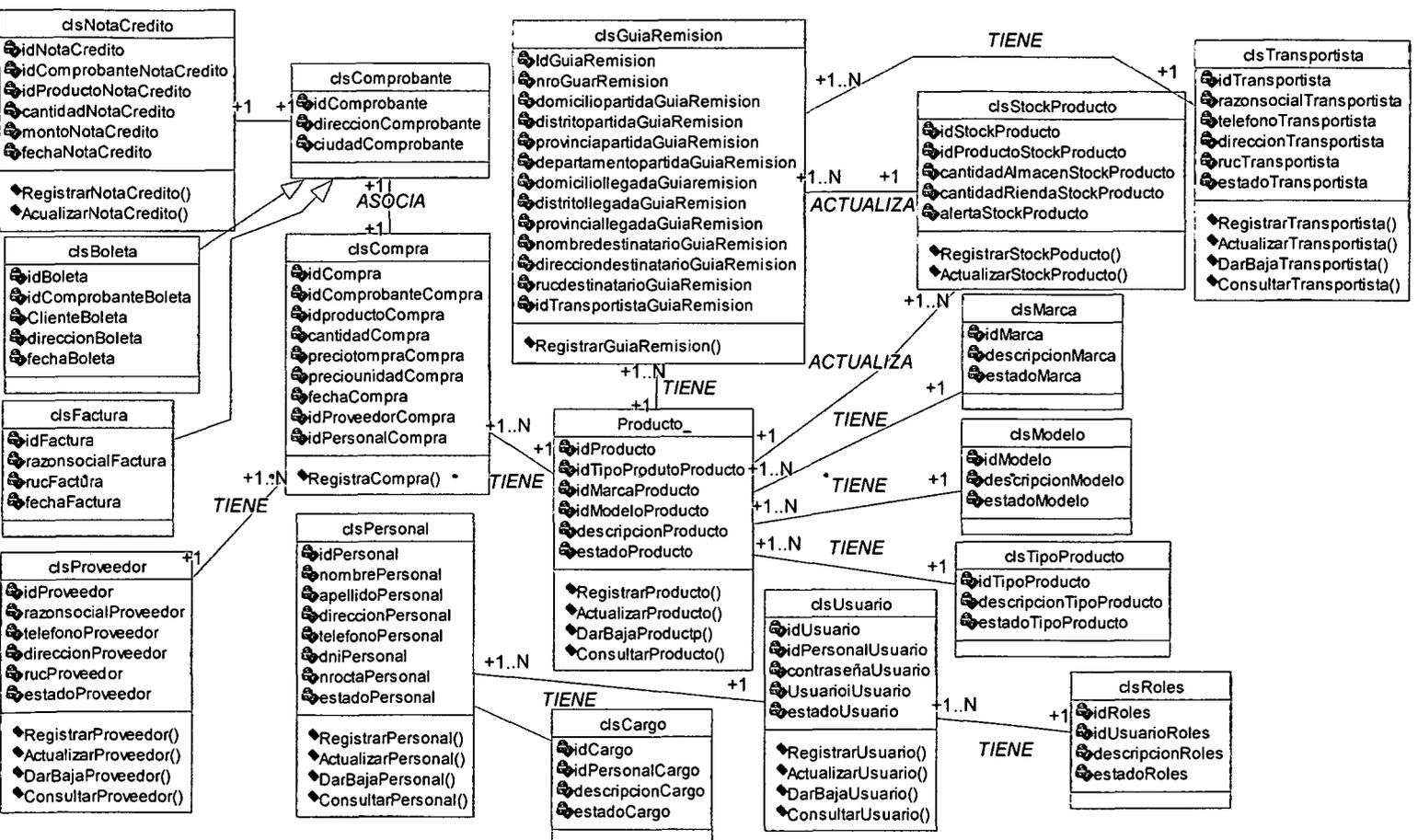


Ilustración 18: Diagrama de Clases SisTriunfo
 Fuente: Elaboración Propia

3.4. Modelado de Diseño

3.4.1. Diseño de Interfaz

Módulos Almacén



Ilustración 19: Login SisTriunfo.
Fuente: Elaboración Propia

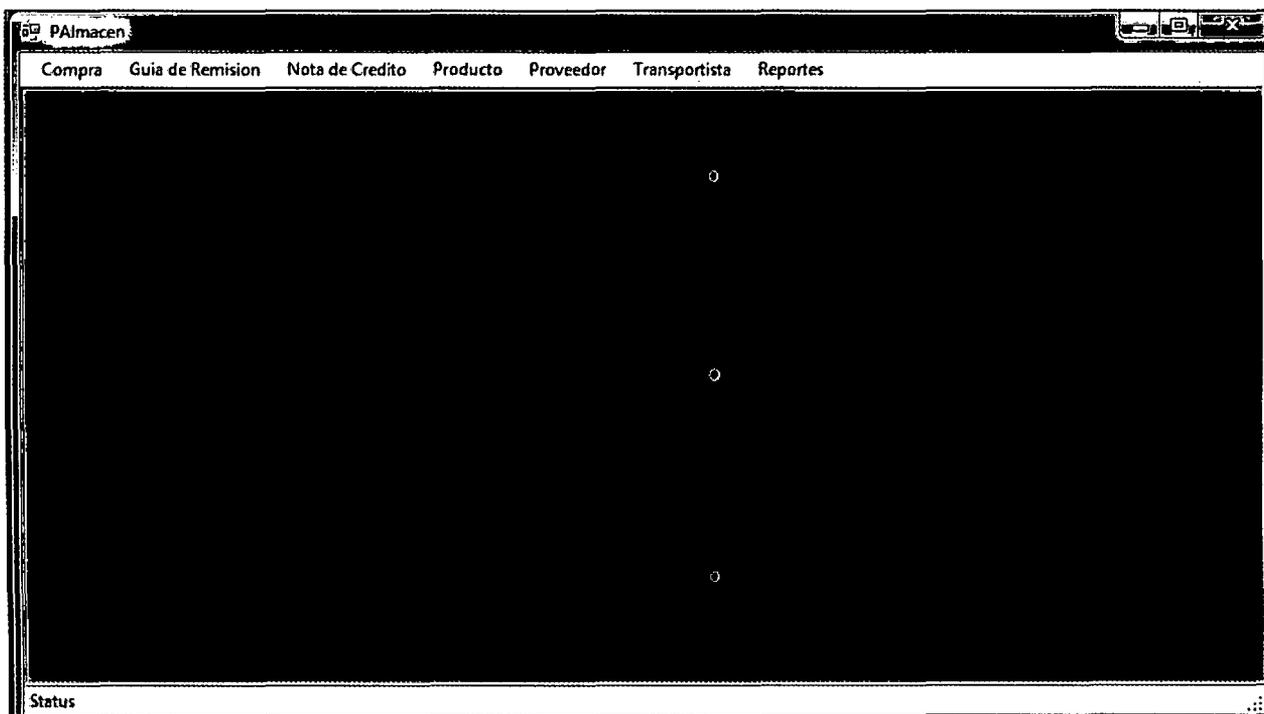


Ilustración 20: Interfaz Principal Almacén
Fuente: Elaboración Propia

COMPRA DIRECTA

Comprobante
Tipo Comprobante: Serie: Número:
Proveedor: Descripción: Compra:
[Registrar] [Limpiar]

Detalle Comprobante

Cantidad	Descripción	Precio	Importe
----------	-------------	--------	---------

[Actualizar] [Eliminar]

Producto: 0.00
Compra: Precio: Importe: Stock:

Ilustración 21: Registrar Compra
Fuente: Elaboración Propia

GUIA REMISIÓN

Guía Remisión
Tienda: Serie: Número: Transportista: Descripción:
Punto Partida: Punto Llegada: Fecha:
[Registrar] [Limpiar]

Detalle Guia Remisión

Cantidad	Descripción	Precio	Importe
----------	-------------	--------	---------

[Actualizar] [Eliminar]

Producto: 0.00
Compra: Precio: Importe: Stock:

Ilustración 22: Registrar Guía de Remisión
Fuente: Elaboración Propia

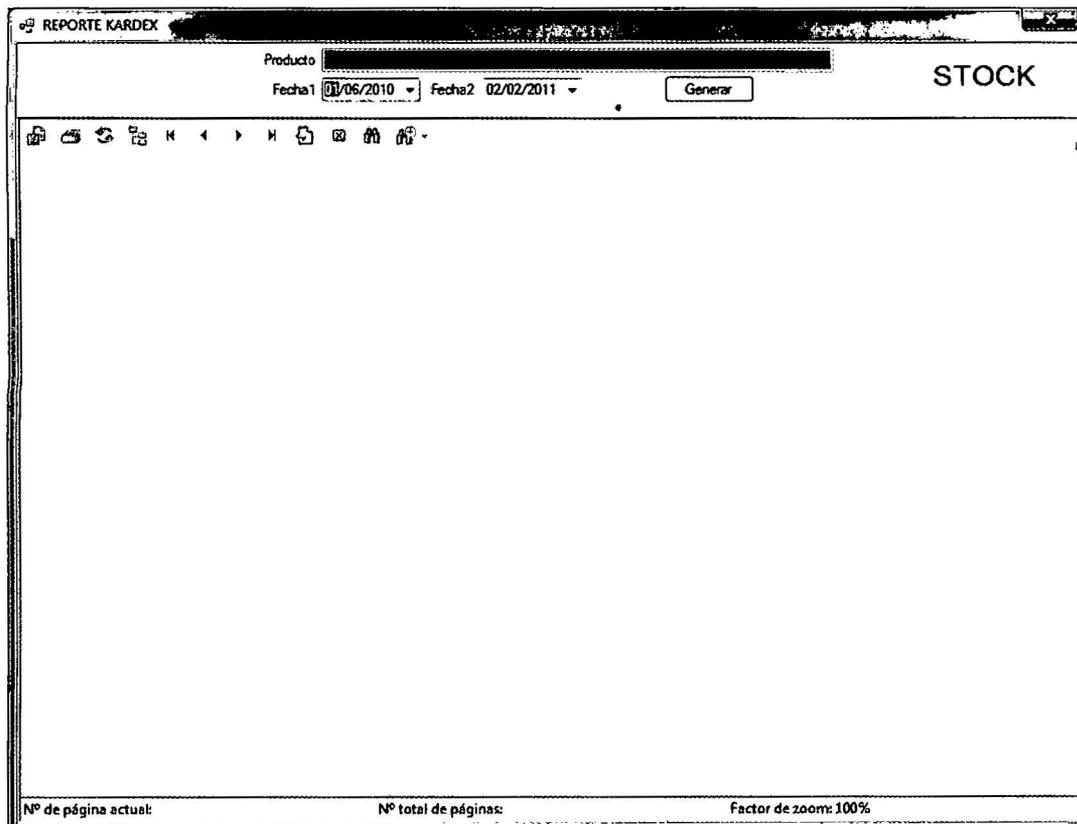


Ilustración 23: Vista de Reporte SisTriunfo
Fuente: Elaboración Propia

Módulo Sistemas

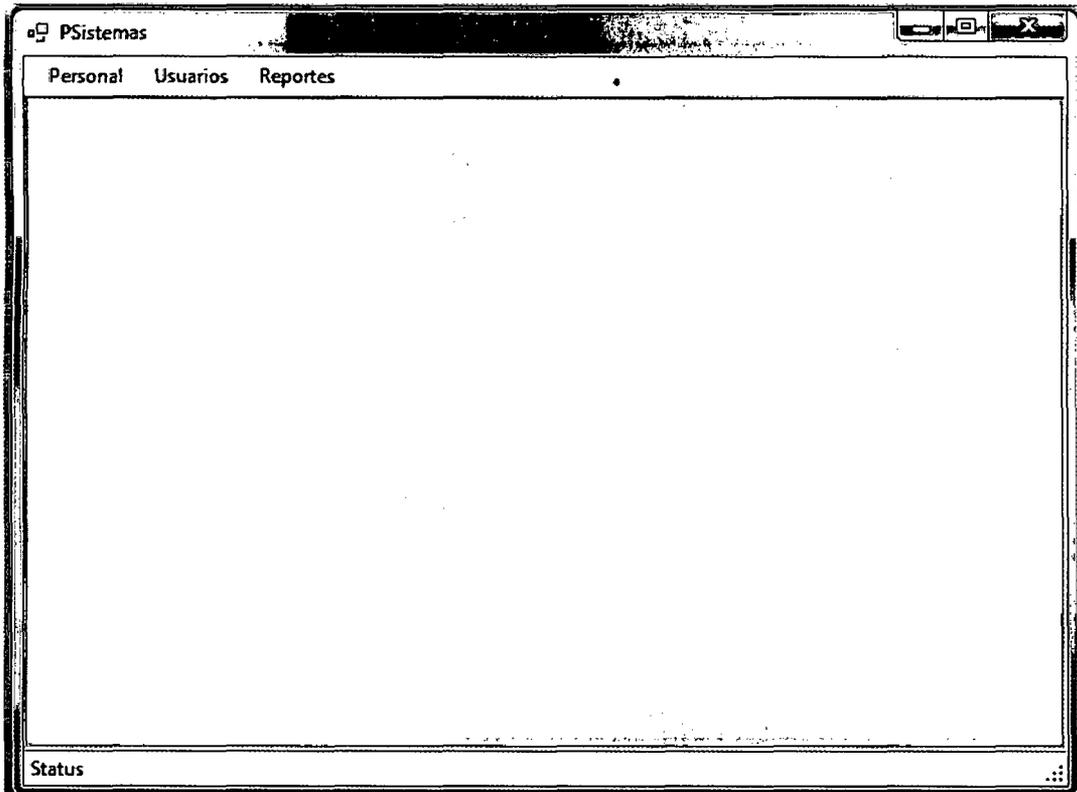


Ilustración 24: Interfaz Principal Sistemas
 Fuente: Elaboración Propia

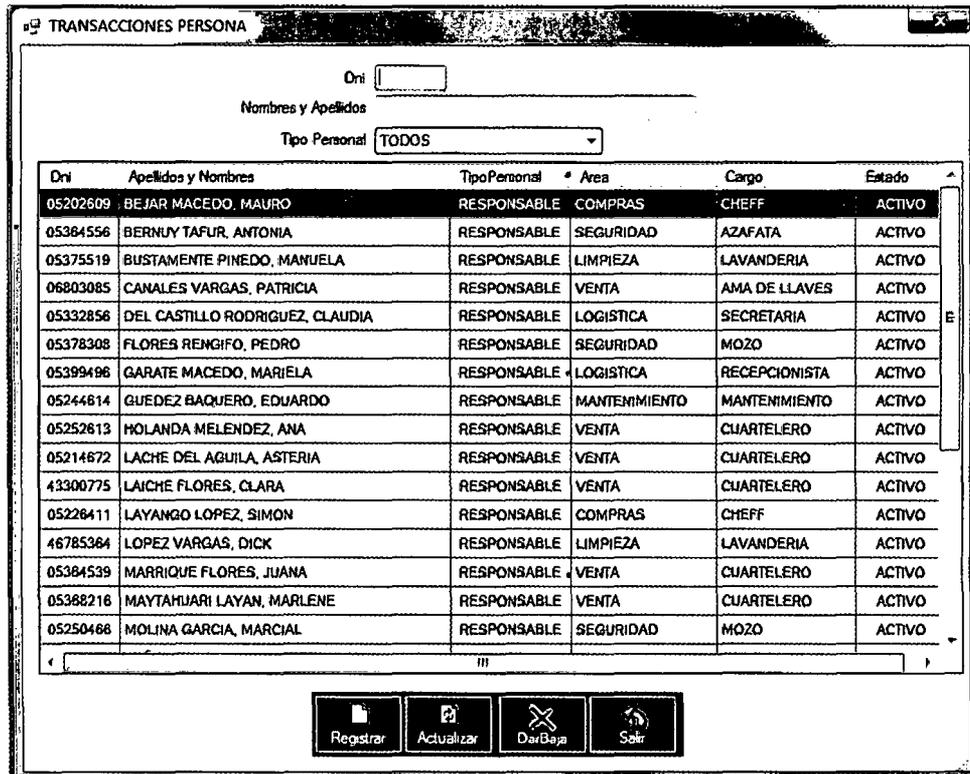


Ilustración 25: Transacción Persona
 Fuente: Elaboración Propia

3.4.1.2. Modelo Físico de la Base de Datos

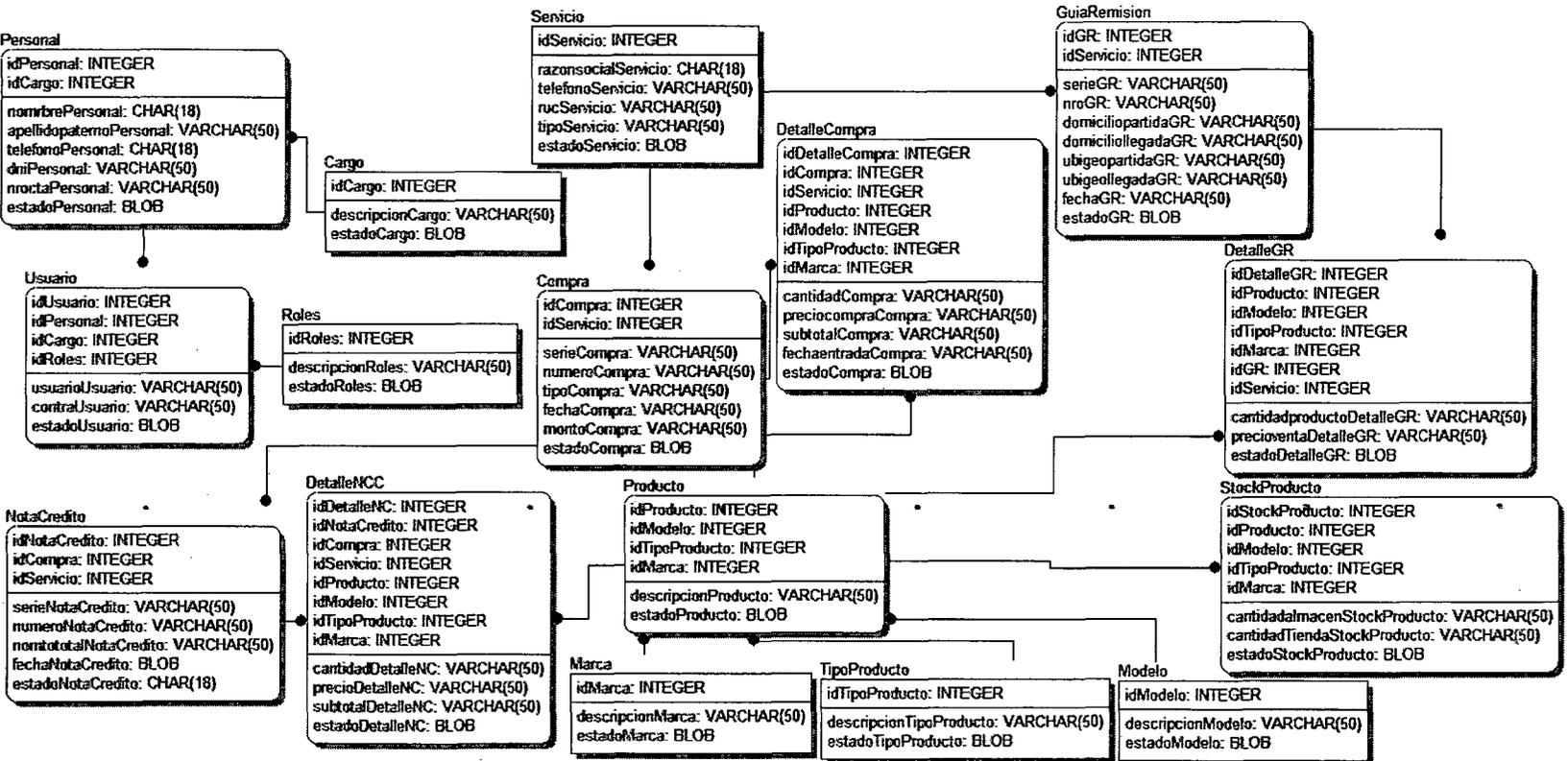


Ilustración 27: Diagrama Lógico Base de Datos Triunfo
 Fuente: Elaboración Propia

3.5. Implementación

3.5.1. Diagrama de Componentes

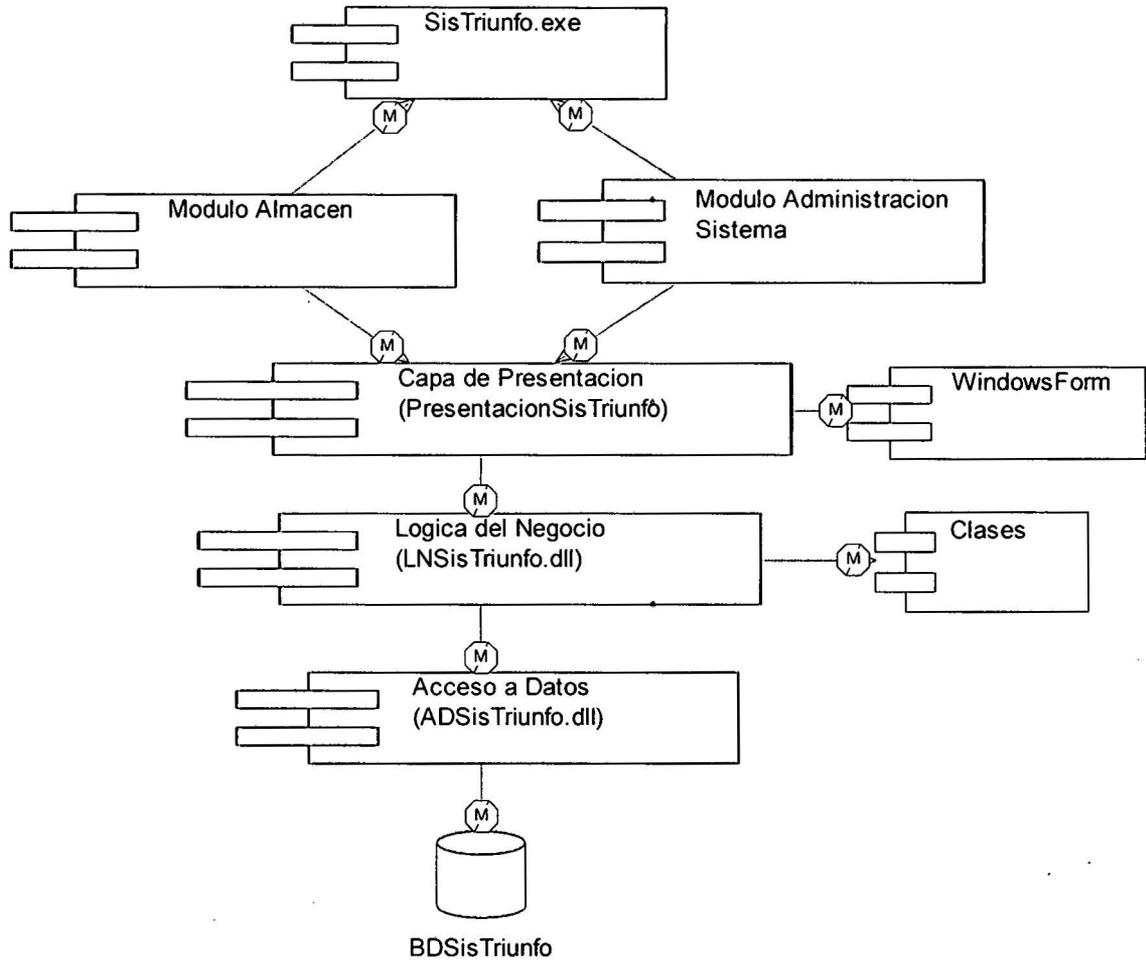


Ilustración 28: Diagrama de Componentes SisTriunfo
Fuente: Elaboración Propia

3.5.2. Diagrama de Despliegue

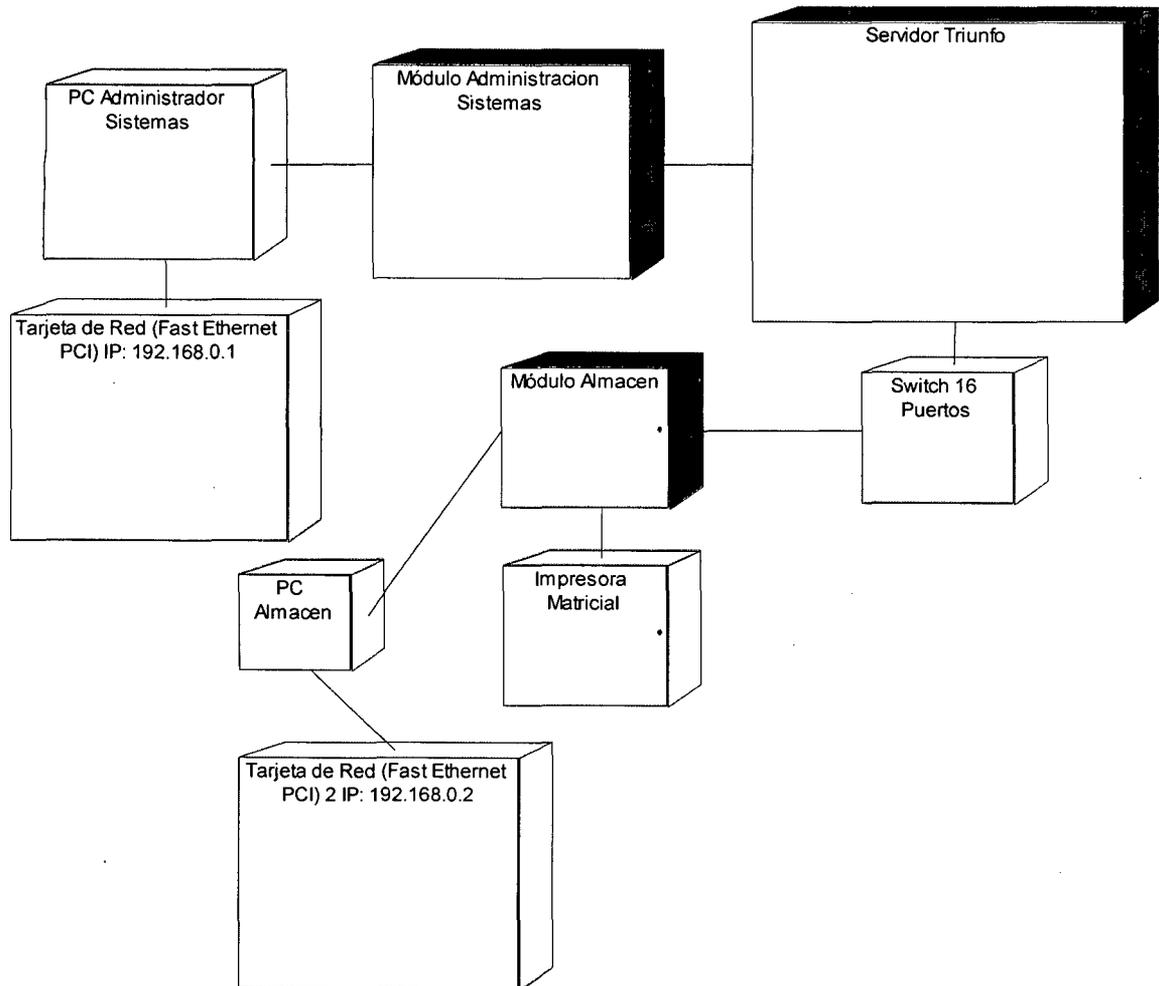


Ilustración 29: Diagrama de Despliegue SisTriunfo
Fuente: Elaboración Propia

Capítulo IV: Resultados y su Discusiones

1. Al automatizar los procesos de control documentario en Productos, Proveedores, Transportistas y Compras, el dueño de proceso (Almacenista) logra controlar desde su computador el negocio y sus actividades desde el módulo de almacén del Sistema de Información de SisTriunfo.
2. Reducción en los tiempos de registros de:
 - Productos, al haber realizado una muestra (ver anexo 7, tabla 13) de estos nos demuestra una reducción de hasta un 54.70% en los registros de productos.
 - Proveedores y transportistas, al haber realizado una muestra (ver anexo 7, tabla 15) de estos nos demuestra una reducción de hasta un 30.33% en los registros de productos.
 - Compras, al haber realizado una muestra (ver anexo 7, tabla 17) de estos nos demuestra una reducción de hasta un 36.7% en los registros de productos.
3. Reducción en Tasas de Errores de:
 - Productos, al haber realizado una muestra (ver anexo 7, tabla 14) de estos nos demuestra una reducción de hasta un 100% en la tasa de error en los registros de productos.
 - Proveedores y transportistas, al haber realizado una muestra (ver anexo 7, tabla 16) de estos nos demuestra una reducción de hasta un 33.3% en los registros de productos.
 - Compras, al haber realizado una muestra (ver anexo 7, tabla 18) de estos nos demuestra una reducción de hasta un 87.2% en los registros de productos.
4. Reducción en los tiempos de generación de reportes de control (ver anexo 7, tabla 19):
 - De las especificaciones de los productos de hasta un 98.61%.
 - De las especificaciones de los Transportistas de hasta un 97.22%
 - De volúmenes de stock Productos de hasta un 99.72%
 - De los detalles de Compras de hasta un 99.17% en los registros de ingreso de mercadería.
 - De las especificaciones de las Guías de Remisión de hasta un 95.83%
 - De las especificaciones de las Notas de Crédito de hasta un 95.83%
 - De volúmenes de Kardex de Almacén de un 99.86%

Capítulo V: Conclusiones

Se concluye que:

1. Culminado el Sistema de Información SisTriunfo utilizando la metodología Rational Unified Process (RUP), desarrollado para pyme Triunfo SAC, se logró dar solución a los problemas estructurales administrativos de la sub Oficina de Almacén el cual debe permitir a Triunfo, seguir evolucionando en este rubro comercial y maximizar el control y la eficiencia dentro de su ámbito laboral.
2. Se logró tener un estricto control en los procesos (entrada mercadería a almacén y salida de mercadería a tienda) e integridad en la información (se logró la reducción en las tasas de errores permitiendo la integridad plena de la información) en los productos y sus stocks que se generan.
3. Se logró tener un estricto control en los procesos (registro, actualización y dar de baja a proveedores) e integridad en la información (se logró la reducción en las tasas de errores permitiendo la integridad plena de los datos) en las transacciones de proveedores y transportistas que se generen, y de esta forma gestionar de la manera más apropiada los recursos (económicos o de factor tiempo) que cuenta la empresa.
4. Se logró tener un estricto control en los procesos (registro, actualización y dar de baja) e integridad en la información (se logró la reducción en las tasas de errores permitiendo la integridad plena de los datos) en las compras que se generen, permitiendo de esta forma la confiabilidad irrestrictita en la entrada de productos a almacén.
5. Se logró tener un estricto control en los procesos (registro, actualización y dar de baja) e integridad información (se logró la reducción en las tasas de errores permitiendo la integridad plena de los datos) en la información en las transferencias que se generen, permitiendo de esta forma la confiabilidad irrestrictita en la salida de productos a almacén.
6. Con el Sistema de Información SisTriunfo desarrollado para pyme Triunfo SAC, se logró la *reducción en los tiempos de generación de reportes* principales en la sub oficina de almacén mostrando de esta manera su eficiencia y dinamismo para la sub oficina de almacén.

Capítulo VI: Recomendaciones

Se recomienda:

1. Implementar la aplicación dentro de la infraestructura tecnológica recomendada en el diagrama de despliegue (ver Capítulo III, 3.5. Implementación, 3.5.2. Diagrama de Despliegue, Ilustración 29).
2. Desarrollar los módulos de Orden de Compra y Venta, para la gestión plena de la Pyme Triunfo SAC.
3. Al personal encargado, validar la información haciendo uso de los reportes que SisTriunfo posee con la información documentada, para que de esta manera depure información no válida.

Bibliografías

- [Fowler] Martin Fowler *Is Design dead?*
<http://www.martinfowler.com/articles/designDead.html>
- [Harrison] Peter Harrison *Evolutionary Programming*,
<http://www.devcentre.org/research/evoprogramming.htm>
- Memoria UML, Manuel Lluesma Camps - Páginas del 66 al 87.
- Ingeniería de Software, Ian Sommerville – Capítulos 1 y 2.
- El Modelo de Componentes, Geoffrey Sparks – Páginas del 4 al 11.
- Aprendizaje UML en 24. Joseph Schmuller – Páginas del 24 al 36 y de pág. 168 al 191.
- Clase Modelado IS2, Anónimo
- Modelado Sistemas UML, Popkin Software and Systems
- Ingeniería de Software un Enfoque Práctico, Roger S. Pressman - Parte Cuarta: Ingeniería Del Software Orientada A Objeto. Páginas del 343 al 420.

Anexos

Anexo 01: Encuestas a profesionales.

Apellidos y Nombres del Encuestado
Cargo que desempeña Actualmente
País

Técnica en Levantamiento de Información

¿Qué nivel de eficiencia – en adquisición de requerimientos - ha mostrado esta técnica en anteriores proyectos?

1) Recolección de documentos principales

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

2) Entrevistas

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

3) Encuestas

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

4) Reuniones

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

5) Análisis de Brechas

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

Metodología en Análisis y Diseño

¿Qué nivel de acoplamiento posee esta metodología para realizar un proyecto de desarrollo?

A) XP

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

B) RUP

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

C) Métrica

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

D) Orientado a Objetos

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

Soporte de lenguajes

Gestión de proyectos posee Componente anteriores – en otros – en dichos lenguajes que estén disponibles?

A) PHP

- A: Todos
- B: Algunos
- C: Ninguno

B) Visual Basic

- A: Todos
- B: Algunos
- C: Ninguno

C) Visual C++

- A: Todos
- B: Algunos
- C: Ninguno

D) Visual C#

- A: Todos
- B: Algunos
- C: Ninguno

E) Power Builder

- A: Todos
- B: Algunos
- C: Ninguno

E) Java

- A: Todos
- B: Algunos
- C: Ninguno

¿El lenguaje a utilizarse es multiplataforma?

A) PHP

- A: Si
- B: No

B) Visual Basic

- A: Si
- B: No

C) Visual C++

- A: Si
- B: No

D) Visual C#

- A: Si
- B: No

E) Power Builder

- A: Si
- B: No

E) Java

- A: Si
- B: No

¿Qué nivel de soporte existe con respecto al lenguaje a emplear (Existe un proveedor determinado o es de uso libre)?

A) PHP

- A: Si
- B: No

B) Visual Basic

- A: Si
- B: No

C) Visual C++

- A: Si
- B: No

D) Visual C#

- A: Si
- B: No

E) Power Builder

- A: Si
- B: No

E) Java

- A: Si
- B: No

Metodología de programación

¿Existe disponibilidad en el mercado de programadores aplicando la metodología de programación?

A) Orientado a Objetos

- A: Existe
- B: Muy Poco
- C: No Existe

B) Estructurada

- A: Existe
- B: Muy Poco
- C: No Existe

¿Qué nivel de rendimiento ha mostrado la metodología a emplearse en anteriores proyectos?

A) Orientado a Objetos

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

B) Estructurado

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

Sistema Operativo

¿Qué nivel de respaldo posee el SO ante múltiples usuarios y/o nivel de transacciones de información, cortes de energía, etc.?

A) Windows 2003 Server

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

B) Windows 2005 Server

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

C) Windows 2008 Server

- A: Excelente
- B: Muy Buena
- C: Buena
- D: Adecuada
- E: Deficiente

D) Unix

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

E) Linux

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

Administración de Datos

¿Qué nivel de gestión de tráfico en las transacción, almacenamiento, seguridad y administración de datos posee?

A) Oracle

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

B) MySql

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

C) DB2

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

D) Microsoft SQL Server 2005

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

E) Microsoft Office Access

- A: Excelente**
- B: Muy Buena**
- C: Buena**
- D: Adecuada**
- E: Deficiente**

Relación de profesionales encuestados

Tabla 7: Relación de Profesionales encuestados.

Nombre y Apellido	Cargo que desempeñan en la actualidad
Bryand Ferdinand Hidalgo	Especialista en Sistemas de Información, BIOINFO – IIAP
Annie Reyes Perez	Soprote Informático, Caja Municipal de Maynas – Iquitos.
Herón Meza Barrera	Especialista en Sistemas de Información, SICON – IIAP
Alcibiades Palestini Marinho	Analista en Sistemas de Información, SUNAT – Iquitos
Jorge Zvietcovich Díaz	Programador, Hospital Regional de Loreto-Iquitos
Dinno Ríos Yumbato	Analista de Negocios
Henry Rengifo Montoya	Analista de Sistemas, Graña & Montero – Lima
Jesica Salazar Piña	Especialista en Sistemas de Información, Academia de la Magistratura – Lima
Mary Roylith Tenazoa Rivera	Administrativo OGGA – UNAP Iquitos
Juan A. Vela Reyna	Analista Programador en Sistemas de Información.
Yno Gómez Díaz	Sin Cargo.
Enrique Díaz Linares	Analista Programador en Sistema de Información.
Isaac Ocampo Yahuarcani	Jefe SITEC IIAP
Karen Quispe Grandez	Especialista en Sistemas de Información Georeferencial

Anexo 02: Entrevista a Dueño de Proceso de Almacén

Fecha: 16 de diciembre de 2010

Hora: 10:30 am.

Lugar: Almacén Triunfo SAC, Samanez Ocampo 895, 2do piso.

¿Explique claramente sus funciones dentro almacén?

Me encargo de elaborar en un cuaderno en forma detallada las especificaciones de los comprobantes que llegan junto a los paquetes – fardos de mercadería – para su control en dicho depósito. También soy el que controla si los productos están en condiciones óptimas, y de no ser así procedo a contactar al proveedor para su cambio o devolución del dinero, también controlo el despacho de los productos para tienda y se pueda dar pase a su venta al público.

¿Cuáles son sus operaciones más dentro del almacén?

Realizo el registro de todos los productos que llegan y el proveedor. Luego guardo todos los comprobantes en un cuaderno donde irán apuntadas todas las compras hechas. Cuando un lote de mercadería sale a tienda realizo una guía de remisión, pero como el local está en el mismo lugar que el almacén aun no hice uso de ese documento.

¿Realiza Usted alguna operación extra fuera de almacén?

Soy el encargado de ir hasta el aeropuerto a revisar la llegada de mercadería junto al Contador.

¿Cuáles son los mayores problemas que posee dentro de Almacén?

La mayor parte del tiempo la tienda se queja por que se anota mal las facturas de compra y contabilidad siempre esta quejándose por eso, además de que cuando piden un reporte de los productos o del stock, o si se ha enviando algo a tienda me demoro mucho en revisar los libros de salida.

¿Tiene Usted alguna sugerencia de cómo podría agilizarse mejor estas operaciones?

No, ninguna.

¿Cree Usted que con la implementación de un Sistema de Almacén estas operaciones (procesos) puedan mejorar?

Sí, siempre y cuando me capaciten para utilizar el sistema, ya que nunca tuve estudios en computación.

¿De darle la adecuada capacitación cree usted poder dominar el sistema?

Desde luego, además de aprender mantendré mi trabajo.

Anexo 03: Resultado de Encuestas a Profesionales

Tabla 8: Resultado de Encuestas a Profesionales

Atributo	Comparación	Confiabilidad	Robustez	Reutilización	Eficiencia	Portabilidad	Facilidad de Uso
Técnicas en Levantamiento de Información	Recolección de documentos principales				Excelente		
	Entrevistas				Buena		
	Encuestas				Buena		
	Reuniones				Muy Buena		
	Análisis de Brechas (Comparaciones con el entorno)				Adecuada		
Metodología en Análisis y Diseño	XP	Excelente					
	RUP	Adecuada					
	Métrica	Deficiente					
	Orientado a Objetos	Adecuada					
Soporte de lenguajes	PHP			Ninguno		Si	No
	Visual Basic			Algunos		No	Si
	Visual C#			Algunos		No	Si
	Visual C++			Ninguno		No	Si
	Power Bulder			Ninguno		No	Si
	Java			Ninguno		Si	No
Metodología de programación	Orientado a Objetos	Existe			Excelente		
	Estructurado	Existe			Buena		
Sistema Operativo	Windows 2003 Server	Buena					
	Windows 2005 Server	Buena					
	Windows 2008 Server	Muy Buena					
	Unix	Excelente					
Administración de Datos	Linux	Muy Buena					
	Oracle	Excelente					
	MySql	Adecuada					
	DB2	Deficiente					
	Microsoft SQL Server 2005	Muy Buena					
	Microsoft SQL Server 2008	Muy Buena					
	Microsoft Office Access	Deficiente					

Anexo 04: Presupuesto Estimado

Tabla 9: Presupuesto Hardware

Hardware	Cantidad	Precio Real	Monto Invertido
Computadora Personal (Laptop)	01	S/.2100	S/.0
Impresora Matricial o Multifuncional	01	S/.800	S/.0
Computadoras Pentium IV	02	S/.2000	S/.1200
Sub Total		S/.4000	S/.1200

Tabla 10: Presupuesto Software

Software	Cantidad	Precio Real	Monto Invertido
Microsoft Visual Studio 2008	01	S/.1400	S/.0
PhotoFilter	01	S/.0	S/.0
SO Windows XP	02	S/.900	S/.0
Servidor Windows 2008 Server	01	S/.1400	S/.0
Microsoft SQL Server 2008	03	S/.1400	S/.0
Internet Information Server	03	S/.0	S/.0
Microsoft Office 2007	01	S/.400	S/.0
Adobe Reader 8	03	\$0	S/.0
Sub Total		S/.5500	S/.0

Tabla 11: Presupuesto Insumos, Servicios y Materiales de Escritorio

Insumos, Servicios y Materiales de Escritorio	Cantidad	Precio Real	Monto Invertido
Papel	1000	S/.23	S/.23
Compact Disk	25	S/.5	S/.5
Lapiceros	24	S/.12	S/.12
Fotocopias	360	S/.25	S/.25
Anillados	04	S/.0	S/.0
Sub Total		S/.65	S/.65

Nota: Sólo se hará uso del *Monto Invertido*, debido a que se cuenta con la mayor parte de los recursos en hardware y software en la empresa Triunfo SAC.

Anexo 05: Vistas de Boletas, Facturas y Guías de Remisión

Guía de Remisión

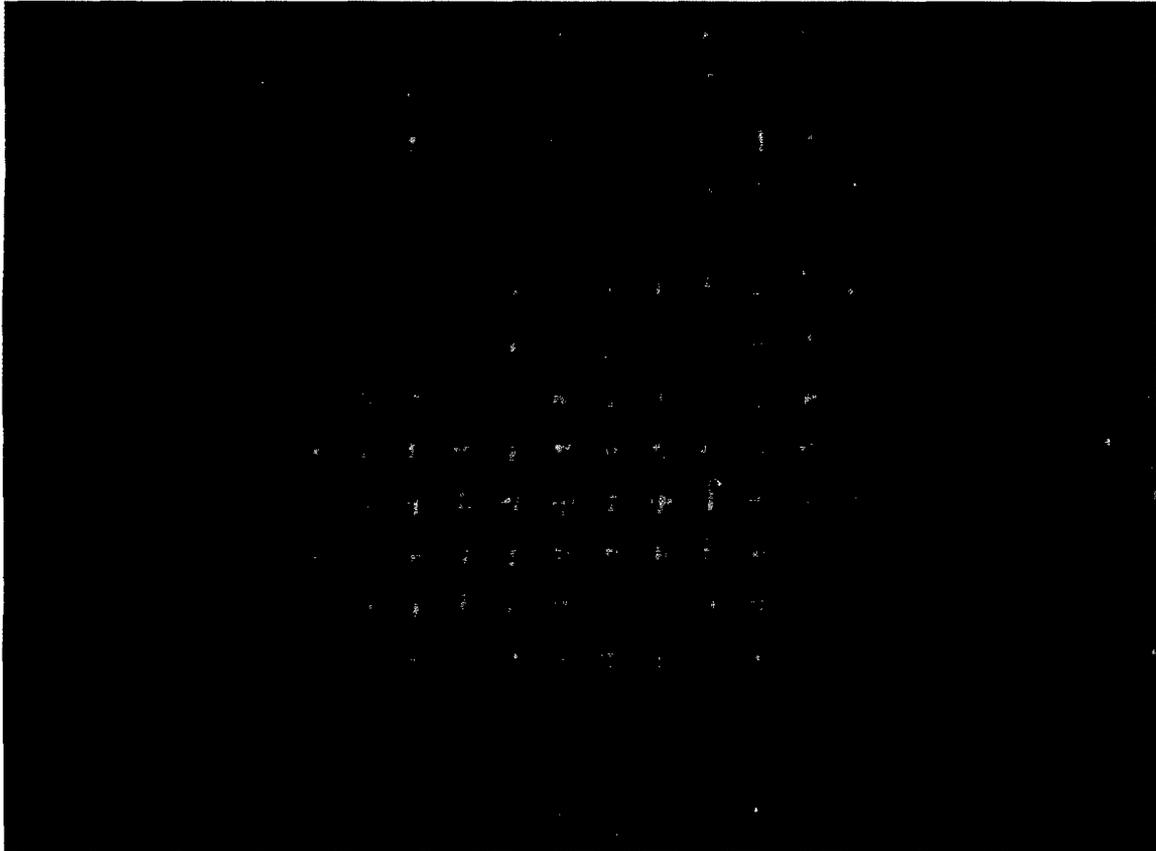


Ilustración 30: Imagen Guía de Remisión

Factura

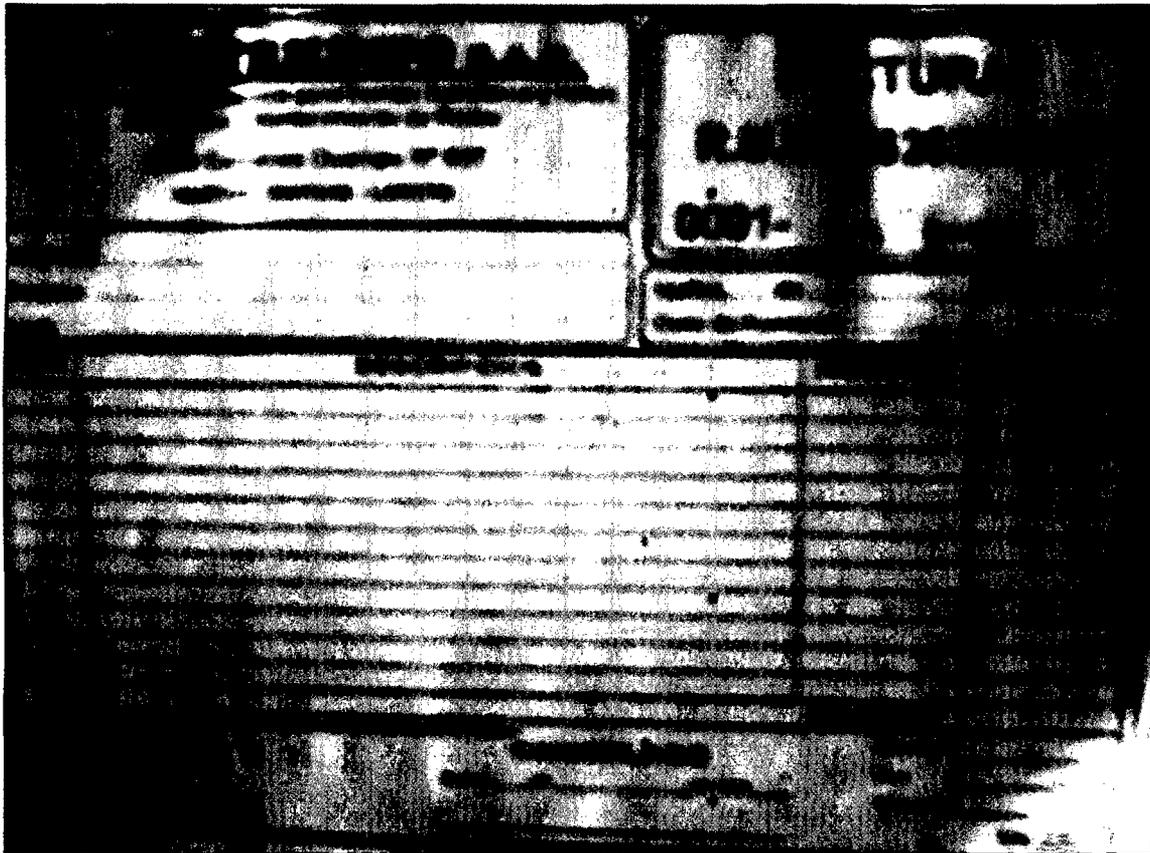


Ilustración 31: Imagen Factura Triunfo SAC

Boleta

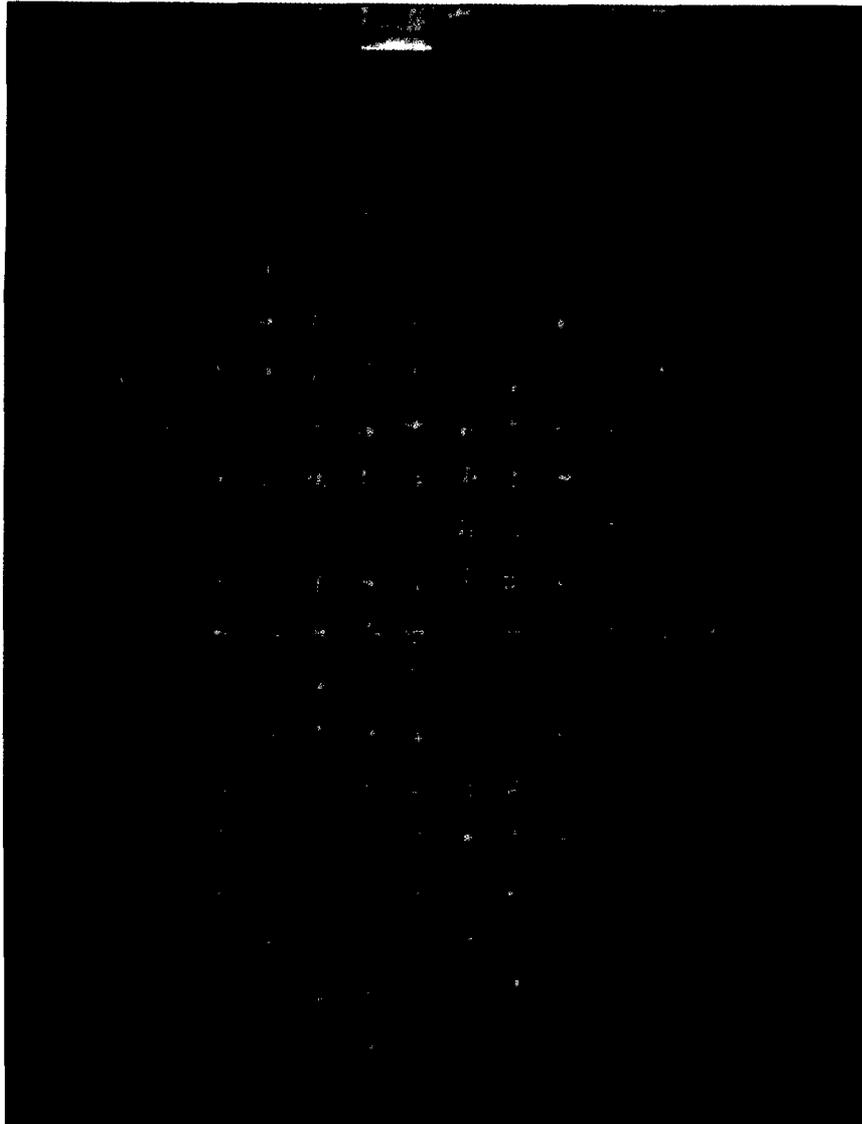


Ilustración 32: Imagen Boleta Triunfo SAC

Anexo 06: Diccionario de Datos de Base de Datos SisTriunfo

Tabla 12: Diccionario de Datos Base de Datos SisTriunfo

Tabla	Atributo	Tipo de Dato	Null Option
Cargo	descripcionCargo	VARCHAR(50)	NULL
Cargo	estadoCargo	BLOB	NULL
Cargo	idCargo	INTEGER	NOT NULL
Compra	estadoCompra	BLOB	NULL
Compra	fechaCompra	VARCHAR(50)	NULL
Compra	idCompra	INTEGER	NOT NULL
Compra	idServicio	INTEGER	NOT NULL
Compra	montoCompra	VARCHAR(50)	NULL
Compra	numeroCompra	VARCHAR(50)	NULL
Compra	serieCompra	VARCHAR(50)	NULL
Compra	tipoCompra	VARCHAR(50)	NULL
DetalleCompra	cantidadCompra	VARCHAR(50)	NULL
DetalleCompra	estadoDetalleCompra	BLOB	NULL
DetalleCompra	fechaentradaCompra	VARCHAR(50)	NULL
DetalleCompra	idCompra	INTEGER	NOT NULL
DetalleCompra	idDetalleCompra	INTEGER	NOT NULL
DetalleCompra	idMarca	INTEGER	NOT NULL
DetalleCompra	idModelo	INTEGER	NOT NULL
DetalleCompra	idProducto	INTEGER	NOT NULL
DetalleCompra	idServicio	INTEGER	NOT NULL
DetalleCompra	idTipoProducto	INTEGER	NOT NULL
DetalleCompra	preciocompraCompra	VARCHAR(50)	NULL
DetalleCompra	subtotalCompra	VARCHAR(50)	NULL
DetalleGR	cantidadproductoDetalleGR	VARCHAR(50)	NULL
DetalleGR	estadoDetalleGR	BLOB	NULL
DetalleGR	idDetalleGR	INTEGER	NOT NULL
DetalleGR	idGR	INTEGER	NOT NULL
DetalleGR	idMarca	INTEGER	NOT NULL
DetalleGR	idModelo	INTEGER	NOT NULL
DetalleGR	idProducto	INTEGER	NOT NULL
DetalleGR	idServicio	INTEGER	NOT NULL
DetalleGR	idTipoProducto	INTEGER	NOT NULL
DetalleGR	precioventaDetalleGR	VARCHAR(50)	NULL
DetalleNCC	cantidadDetalleNC	VARCHAR(50)	NULL
DetalleNCC	estadoDetalleNC	BLOB	NULL
DetalleNCC	idCompra	INTEGER	NOT NULL
DetalleNCC	idDetalleNC	INTEGER	NOT NULL
DetalleNCC	idMarca	INTEGER	NOT NULL
DetalleNCC	idModelo	INTEGER	NOT NULL
DetalleNCC	idNotaCredito	INTEGER	NOT NULL
DetalleNCC	idProducto	INTEGER	NOT NULL
DetalleNCC	idServicio	INTEGER	NOT NULL
DetalleNCC	idTipoProducto	INTEGER	NOT NULL
DetalleNCC	precioDetalleNC	VARCHAR(50)	NULL
DetalleNCC	subtotalDetalleNC	VARCHAR(50)	NULL
GuiaRemision	domiciliollegadaGR	VARCHAR(50)	NULL
GuiaRemision	domiciliopartidaGR	VARCHAR(50)	NULL

Tabla	Atributo	Tipo de Dato	Null Option
GuiaRemision	estadoGR	BLOB	NULL
GuiaRemision	fechaGR	VARCHAR(50)	NULL
GuiaRemision	idGR	INTEGER	NOT NULL
GuiaRemision	idServicio	INTEGER	NOT NULL
GuiaRemision	nroGR	VARCHAR(50)	NULL
GuiaRemision	serieGR	VARCHAR(50)	NULL
GuiaRemision	ubigeollegadaGR	VARCHAR(50)	NULL
GuiaRemision	ubigeopartidaGR	VARCHAR(50)	NULL
Marca	descripcionMarca	VARCHAR(50)	NULL
Marca	estadoMarca	BLOB	NULL
Marca	idMarca	INTEGER	NOT NULL
Modelo	descripcionModelo	VARCHAR(50)	NULL
Modelo	estadoModelo	BLOB	NULL
Modelo	idModelo	INTEGER	NOT NULL
NotaCredito	estadoNotaCredito	CHAR(18)	NULL
NotaCredito	fechaNotaCredito	VARCHAR(50)	NULL
NotaCredito	idCompra	INTEGER	NOT NULL
NotaCredito	idNotaCredito	INTEGER	NOT NULL
NotaCredito	idServicio	INTEGER	NOT NULL
NotaCredito	nomtotalNotaCredito	VARCHAR(50)	NULL
NotaCredito	numeroNotaCredito	VARCHAR(50)	NULL
NotaCredito	serieNotaCredito	VARCHAR(50)	NULL
Personal	apellidopaternoPersonal	VARCHAR(50)	NULL
Personal	dniPersonal	VARCHAR(50)	NULL
Personal	estadoPersonal	BLOB	NULL
Personal	idCargo	INTEGER	NOT NULL
Personal	idPersonal	INTEGER	NOT NULL
Personal	nomrbrePersonal	CHAR(18)	NULL
Personal	nroctaPersonal	VARCHAR(50)	NULL
Personal	telefonoPersonal	CHAR(18)	NULL
Producto	descripcionProducto	VARCHAR(50)	NULL
Producto	estadoProducto	BLOB	NULL
Producto	idMarca	INTEGER	NOT NULL
Producto	idModelo	INTEGER	NOT NULL
Producto	idProducto	INTEGER	NOT NULL
Producto	idTipoProducto	INTEGER	NOT NULL
Roles	descripcionRoles	VARCHAR(50)	NULL
Roles	estadoRoles	BLOB	NULL
Roles	idRoles	INTEGER	NOT NULL
Servicio	estadoServicio	BLOB	NULL
Servicio	idServicio	INTEGER	NOT NULL
Servicio	razonsocialServicio	CHAR(18)	NULL
Servicio	rucServicio	VARCHAR(50)	NULL
Servicio	telefonoServicio	VARCHAR(50)	NULL
Servicio	tipoServicio	VARCHAR(50)	NULL
StockProducto	cantidadalmacenStockProducto	VARCHAR(50)	NULL
StockProducto	cantidadTiendaStockProducto	VARCHAR(50)	NULL
StockProducto	estadoStockProducto	BLOB	NULL

Anexo 07: Resultados de Indicadores

Se realizó una muestra de 10 registros de productos, 10 registros entre proveedores y transportistas, 10 registros compras, 5 registros de guías de remisión de salida de almacén y las solicitudes de reportes, para realizar las pruebas comparativas – y medir el tiempo y la tasa de error - entre las transacciones hechas sin y con el SisTriunfo respectivamente y la eficiencia mostrada en la generación de reportes. A continuación se muestra una tabla comparativa entre estas 2 muestras:

Tabla 13: de Indicadores respecto a la reducción de tiempo en las transacciones de Productos

Producto	Tiempo Registro de Productos sin SisTriunfo	Tiempo Registro de Productos con SisTriunfo
1	50''	12''
2	45''	15''
3	35''	19''
4	1'10''	22''
5	48''	18''
6	32''	22''
7	49''	17''
8	26''	15''
9	43''	17''
10	52''	16''
Total	6'10''	2'53''
Promedio	45''	17.3''

Se contabiliza una reducción en el tiempo de registro de productos de hasta un 54.70% con SisTriunfo en referencia al proceso realizado sin SisTriunfo.

Tabla 14: Resultado de Indicadores respecto a la reducción de tasa de error en las transacciones de Productos

Producto	Tasa de Errores Registro de Productos sin SisTriunfo	Tasa de Errores Registro de Productos con SisTriunfo
1	0	0
2	0	0
3	1	0
4	0	0
5	1	0
6	1	0
7	1	0
8	0	0
9	1	0
10	0	0
Tasa de Error	5	0

Se contabiliza una reducción en la tasa de errores en el registro de productos de hasta un 100% con SisTriunfo en referencia al proceso realizado sin SisTriunfo.

Tabla 15: Resultado de Indicadores respecto a la reducción de tasa de error en las transacciones de Proveedores y Transportistas

Proveedores y Transportistas	Tiempo Registro de Proveedores y Transportistas sin Sis-Triunfo	Tiempo Registro de Proveedores y Transportistas con Sis-Triunfo
1	38''	28''
2	28''	22''
3	34''	23''
4	42''	21''
5	45''	25''
6	29''	26''
7	19''	12''
8	35''	30''
9	28''	19''
10	37''	24''
Total	5'35	3'50''
Promedio	33.5''	23''

Se contabiliza una reducción en el tiempo de registro de proveedores y transportistas de hasta un 30.33% con SisTriunfo en referencia al proceso realizado sin SisTriunfo.

Tabla 16: de Indicadores respecto a la reducción de tasa de error en las transacciones de Proveedores y Transportistas.

Proveedores y Transportistas	Tasa de Errores Registro de Productos sin SisTriunfo	Tasa de Errores Registro de Productos con SisTriunfo
1	0	0
2	0	0
3	0	0
4	0	1
5	1	0
6	1	0
7	0	0
8	0	0
9	1	0
10	0	0
Tasa de Error	3	1

Se contabiliza una reducción en la tasa de errores en el registro de proveedores y transportistas de hasta un 33.3% con SisTriunfo en referencia al proceso realizado sin SisTriunfo.

Tabla 17: Resultado de Indicadores respecto a la reducción de tiempo en las transacciones de Compras

Número Comprobante	Tiempo Registro de Compras sin SisTriunfo	Tiempo Registro de Compras con SisTriunfo
1	3'42''	1'55''
2	2'39''	1'13''
3	5'57''	3'22''
4	1'41''	1'12''
5	3'54''	1'58''
6	4'23''	3'50''
7	3'23''	2'20''
8	4'52''	2'48''
9	6'53''	4'52''
10	4'23''	2'58''
Total	41'37''	26'28''
Promedio	4'10''	2'39''

Se contabiliza una reducción en el tiempo de registro de compras de hasta un 36.7% con SisTriunfo en referencia al proceso realizado sin SisTriunfo.

Tabla 18: Resultado de Indicadores respecto a la reducción de tasa de error en las transacciones de Compras

Número Comprobante	Tasa de Errores Registro de Compras sin SisTriunfo	Tasa de Errores Registro de Compras con SisTriunfo
1	0	0
2	0	0
3	3	0
4	0	0
5	1	0
6	1	1
7	1	0
8	0	0
9	2	0
10	0	0
Tasa de Error	8	1

Se contabiliza una reducción en la tasa de errores del registro de compras de hasta un 87.2% con SisTriunfo en referencia al proceso realizado sin SisTriunfo.

Tabla 19: Resultado de Indicadores respecto a la reducción de tiempo en la generación de Reportes

Reportes	Tiempo de Generación de Reportes sin SisTriunfo	Tiempo de Generación de Reportes con SisTriunfo	Reducción de Tiempo en Porcentaje
Productos	6'	5''	98.61%
Transportistas	3'	5''	97.22%
Stock Productos	30'	5''	99.72%
Compras	10'	5''	99.17%
Guías de Remisión	2'	5''	95.83%
Notas de Crédito	2'	5''	95.83%
Kardex de Almacén	1h aproximadamente	5''	99.86%

Se contabiliza una reducción en el tiempo de la generación de reportes de:

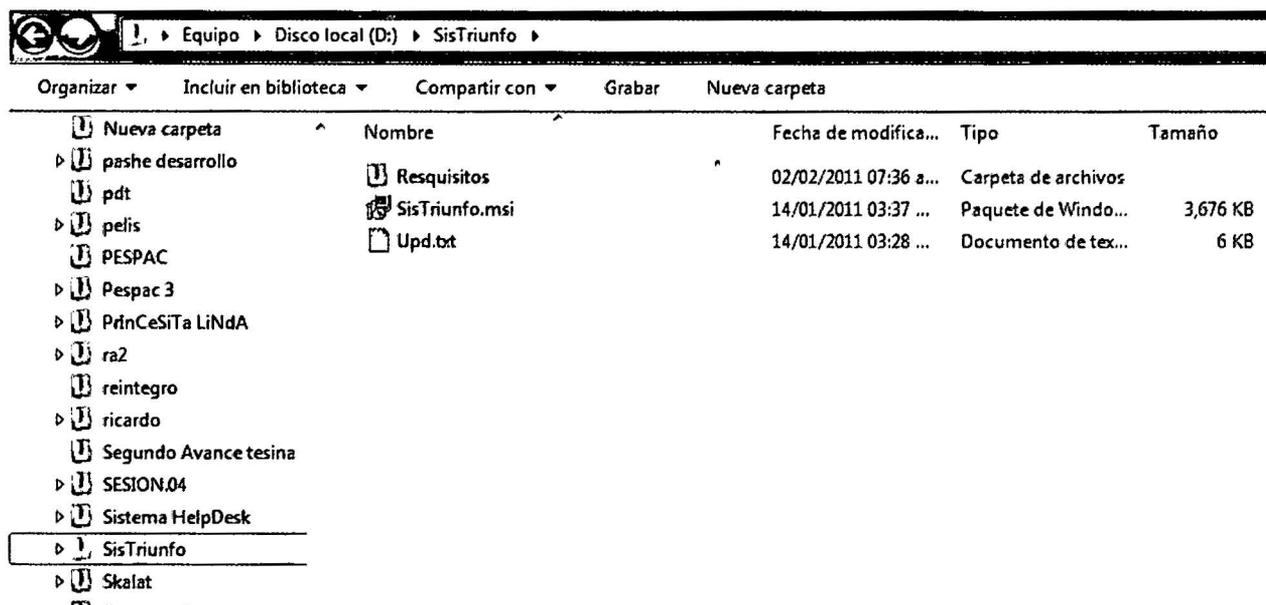
- Producto de hasta un 98.61%.
- Transportistas de hasta un 97.22%
- Stock Productos de hasta un 99.72%
- Compras de hasta un 99.17%
- Guías de Remisión de hasta un 95.83%
- Notas de Crédito de hasta un 95.83%
- Kardex de Almacén de un 99.86%

Anexo 08: Glosario de Términos

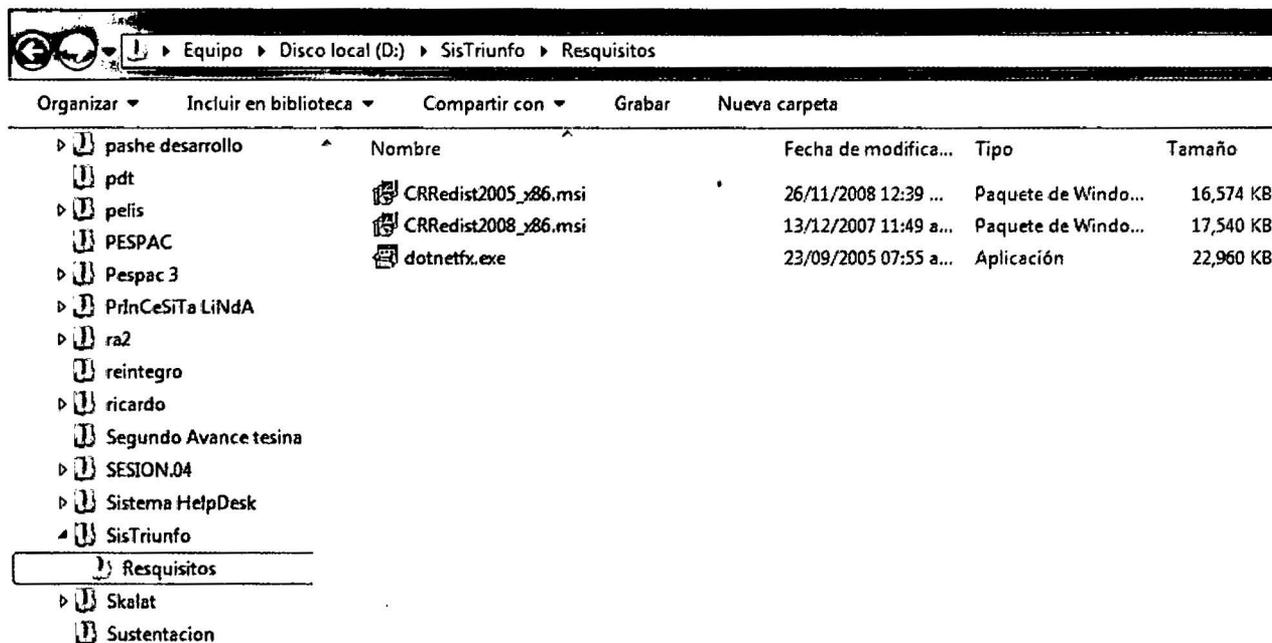
1. **El Comprobante de Pago:** Es un documento que acredita la transferencia de bienes, la entrega en uso o la prestación de servicios [*Artículo 1° de la Resolución de Superintendencia N° 007-99/SUNAT*].
2. **Factura:** Es el comprobante de pago que, por lo general, es utilizado en transacciones entre empresas o personas que necesitan sustentar el pago del IGV en sus adquisiciones, a fin de utilizar el denominado "crédito fiscal", así como gasto o costo para efecto tributario [*Numeral 1 del Artículo 4° de la Resolución de Superintendencia N° 007-99/SUNAT*].
3. **Boleta de Venta:** Es el comprobante de pago que se emiten en operaciones con consumidores o usuarios finales, y que no permite ejercer el derecho al *crédito fiscal*, ni sustentar gasto o costo para efecto tributario. Cuando el importe de la venta o servicio prestado supere un (01) U.I.T. por operación, será necesario consignar los siguientes datos de identificación del adquirente o usuario: apellidos y nombres, dirección y número de su documento de identidad [*Numeral 3.10 del Artículo 8° de la Resolución de Superintendencia N° 007-99/SUNAT*].
4. **Guía de Remisión:** Es el documento que emite el Remitente para sustentar el traslado de bienes con motivo de su compra o venta y la prestación de servicios que involucran o no la transformación de bienes, cesión en uso, consignaciones y remisiones entre establecimientos de una misma empresa y otros. Por remitente se entiende al propietario o poseedor de los bienes al inicio del traslado, al prestador de servicios, como: servicios de mantenimiento, reparación de bienes, entre otros; a la agencia de aduanas, al almacén aduanero, tanto en el caso del traslado de mercancía nacional como de mercancía extranjera; al consignador [*Capítulo V de la Resolución de Superintendencia N° 007-99/SUNAT modificado por las Resoluciones de Superintendencia N° 004 -2003/SUNAT, N° 028-2003/SUNAT y N° 219-2004/SUNAT*].
5. **Nota de Crédito:** Es emitida por los contribuyentes que están acogidos al Régimen General o Especial del Impuesto a la Renta por las anulaciones, descuentos o bonificaciones, devoluciones y otros. En el caso de descuentos o bonificaciones sólo podrán modificar comprobantes de pago que otorguen derecho a crédito fiscal o sustenten gasto o costo para efecto tributario. Tratándose de operaciones con consumidores finales, los descuentos o bonificaciones deberán constar en el mismo comprobante de pago.

Anexo 09: Manual Técnico

La aplicación SisTriunfo puede ser instalada en cualquier unidad dentro del Servidor, como se observa en la siguiente ilustración, a continuación se muestra el contenido del instalador para su posterior ejecución:



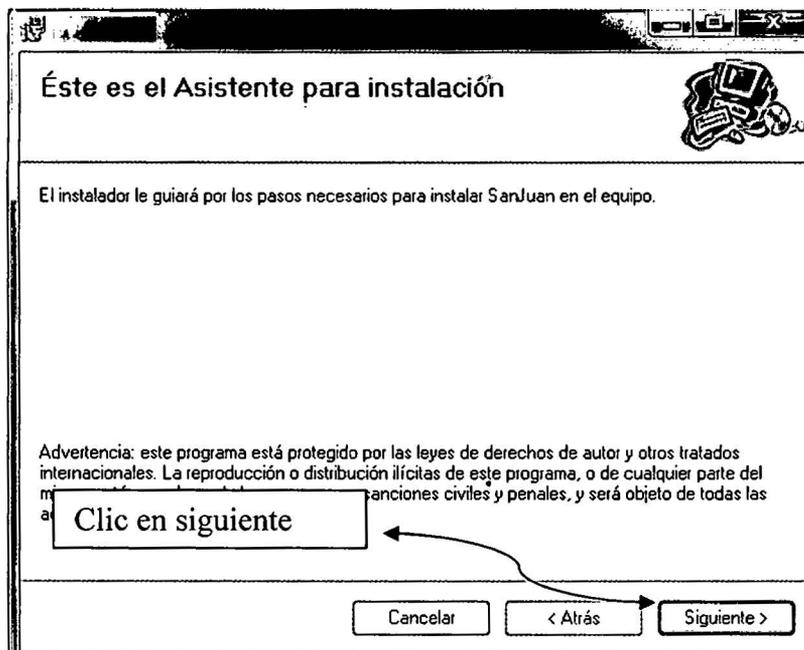
Dentro de la carpeta SisTriunfo se encontrará el instalador del Sistema de Información denominado SisTriunfo.msi, y una carpeta adicional denominada Requisitos.



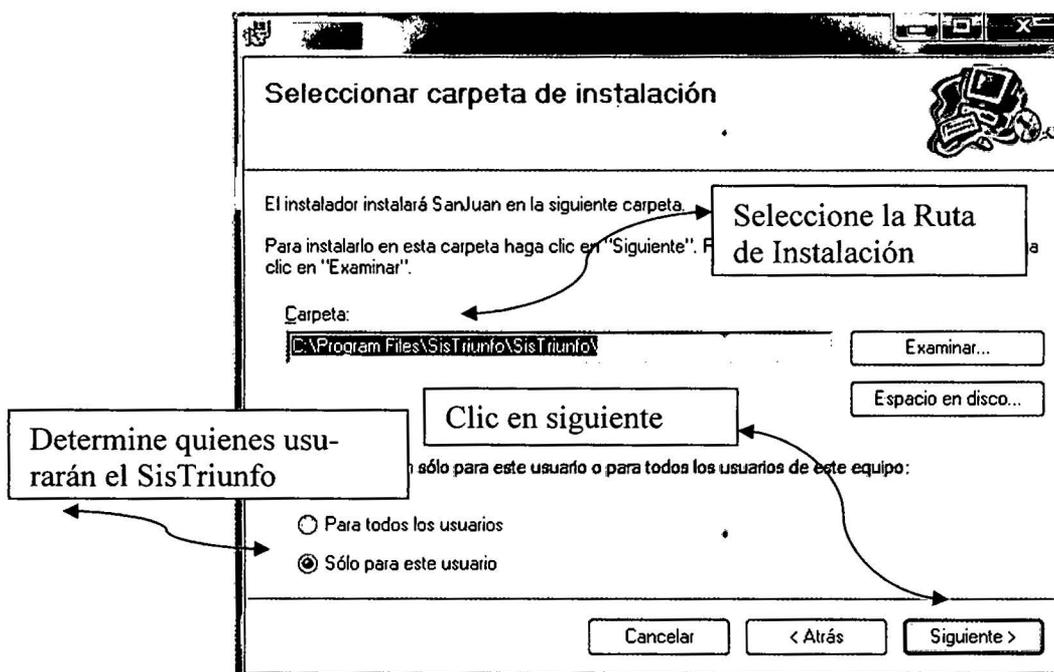
Dentro de requisitos se necesitarán instalar las aplicaciones CRRedist2008_x86.msi para visualizar los reportes en Cristal Report 2008 y dotnetfx.exe, para que levante la aplicación en plataforma .NET.

Instalación SisTriunfo

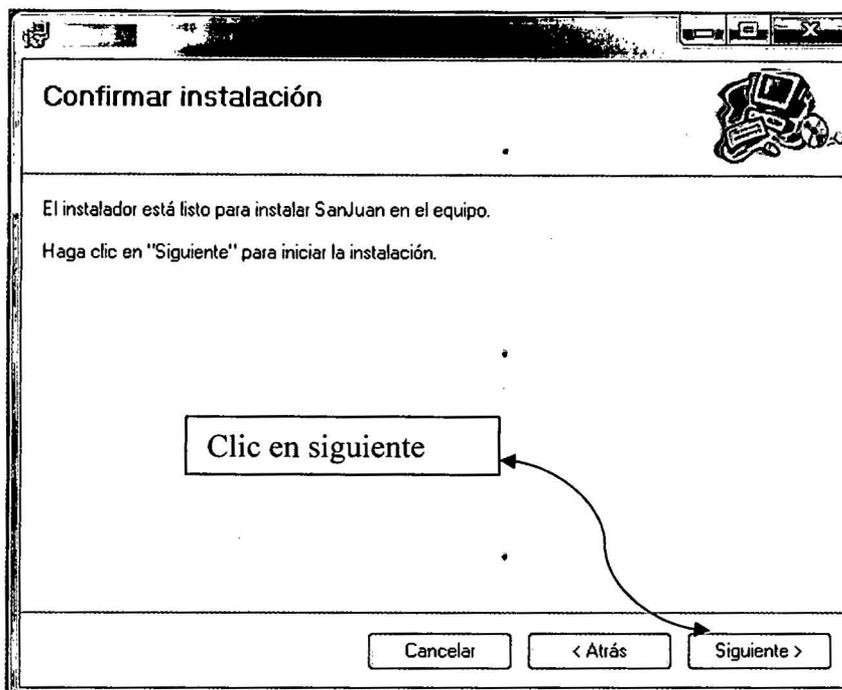
Paso 1:



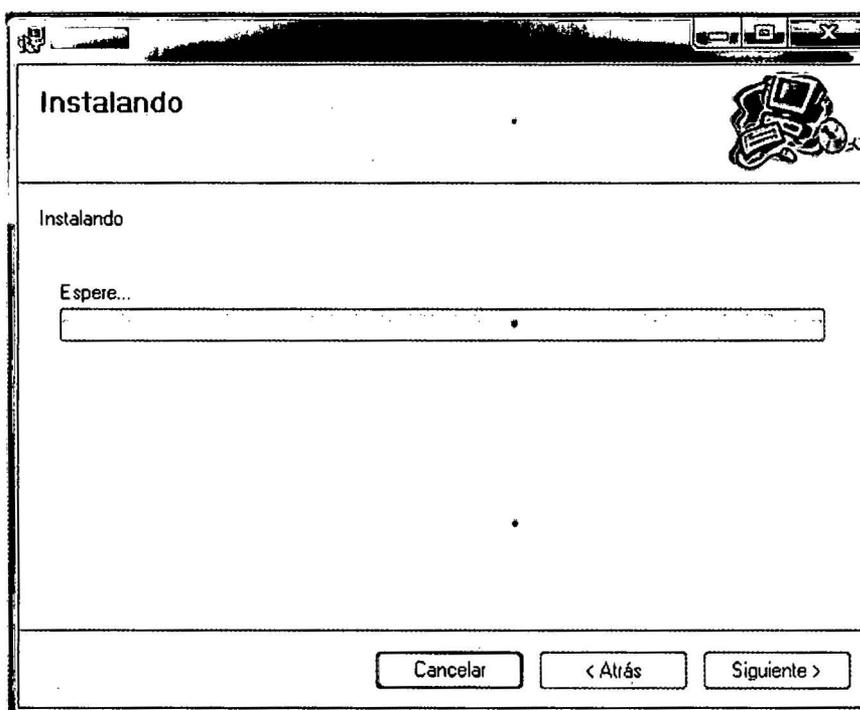
Paso 2:



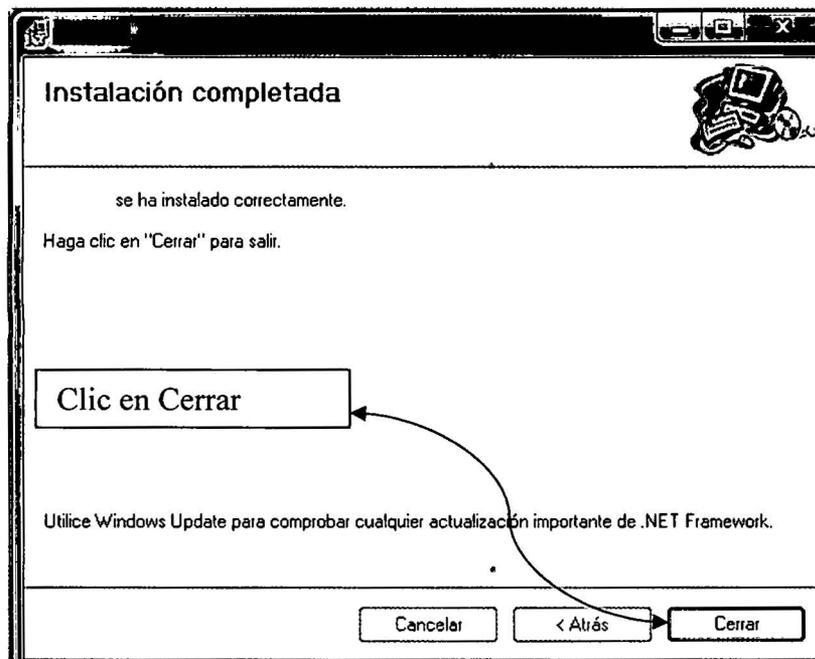
Paso 3:



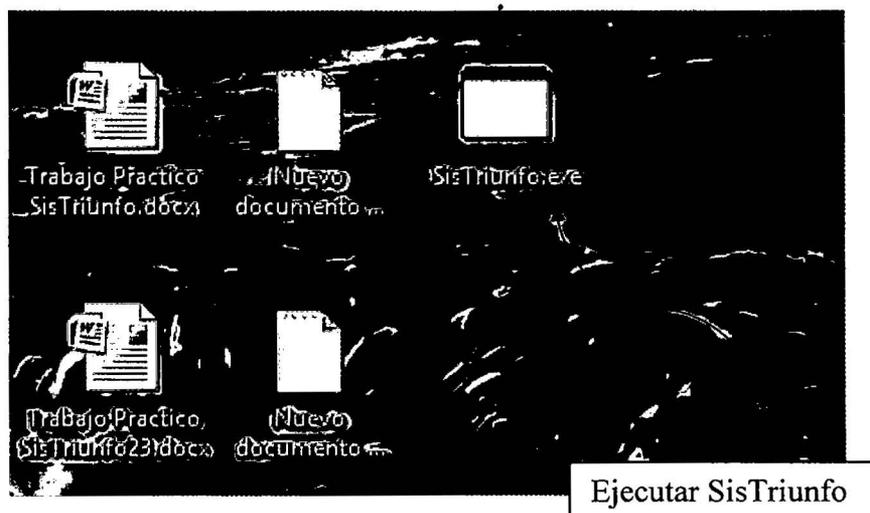
Paso 4:



Paso 5:



Paso 6:



Anexo 10: Manual de Usuario

SisTriunfo cuenta con dos (02) módulos integrados que cumplen funciones separadas para el control de procesos.

El primero es el módulo de Sistemas, el cual permitirá la administración de los Usuarios al SisTriunfo, el segundo será el módulo de Almacén el cual controlará los procesos que se lleven a cabo dentro de esa sub oficina. A continuación se presentará ambos manuales para la administración de SisTriunfo:

1. Módulo Sistemas:

Transacciones Persona, Consultar Personal, gestionará las búsquedas por 3 tipos:

Dni	Apellidos y Nombres	Tipo Personal	Area	Cargo	Estado
05202609	BEJAR MACEDO, MAURO	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
05364556	BERNUY TAFUR, ANTONIA	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
05375519	BUSTAMENTE PINEDO, MANUELA	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
06803085	CANALES VARGAS, PATRICIA	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
05332856	DEL CASTILLO RODRIGUEZ, CLAUDIA	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
05378308	FLORES RENGIFO, PEDRO	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
05399496	GARATE MACEDO, MARIELA	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
05244614	GUEDEZ BAQUERO, EDUARDO	RESPONSABLE	MANTENIMIENTO	MANTENIMIENTO	ACTIVO
				CUARTELERO	ACTIVO
				CUARTELERO	ACTIVO
				CUARTELERO	ACTIVO
05228411	LAYANGO LOPEZ, SIMON	RESPONSABLE	COMPRAS	CHEFF	ACTIVO
46785364	LOPEZ VARGAS, DICK	RESPONSABLE	LIMPIEZA	LAVANDERIA	ACTIVO
05364539	MARRIQUE FLORES, JUANA	RESPONSABLE	VENTA	CUARTELERO	ACTIVO
05368216	MAYTAHUARI LAYAN, MARLENE	RESPONSABLE	VENTA	CUARTELERO	ACTIVO
05250468	MOLINA GARCIA, MARCIAL	RESPONSABLE	SEGURIDAD	MOZO	ACTIVO

El Usuario del módulo será el Administrador general del Sistema de Información, podrá ejercer control sobre el personal de la empresa y asignar Usuarios a dicho personal de ser necesario. El Administrador podrá: Registrar, Actualizar y Dar de Baja, así mismo al generar las transacciones de Usuarios.

2. Módulo de Almacén, para las transacciones de Productos tenemos:

Registro de Productos:

Para Registra Producto, se debe seleccionar la Marca, el Modelo, el Tipo de Producto y de ser necesaria la descripción del mismo. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda

Actualizar Productos:

Para Actualizar Producto, se debe seleccionar el código del producto y automáticamente se cargarán los datos del producto. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código

Consultar Productos:

Código	Producto	Modelo	Talla	Precio	Stock
000202	BALON - RESTAURANT	MANGA LARGA	15.00	30.00	5.000
000249	BALON - RESTAURANT	MANGA LARGA	15.00	138.50	0.000
000119	BOLSA - ALOJAMIENTO	MANGA LARGA	0.00	0.09	90.000
000120	BOLSA - ALOJAMIENTO	MANGA LARGA	15.00	0.03	315.000
000712	BOLSA - ALOJAMIENTO	MANGA LARGA	15.00	2.88	0.000
000801	BOLSA - ALOJAMIENTO	MANGA LARGA	15.00	2.07	0.000
001070	BOLSA - ALOJAMIENTO	MANGA LARGA	0.00	0.03	280.000
001135	BOLSA - ALOJAMIENTO	MANGA LARGA	2.00	3.00	4.000
001136	BOLSA - ALOJAMIENTO	MANGA LARGA	2.00	3.00	4.000
000799	BOLSA - BAR Y RSS	MANGA LARGA	15.00	2.05	0.000

Para Consultar Producto, se debe ingresar la descripción y automáticamente la grilla irá actualizándose hasta gestionar la búsqueda.

Dar de Baja un Producto:

Producto

Código: 000125

Marca: 10 VARIOS

Modelo: 2 MANGA LARGA

Tipo Producto: 1 UNIDAD

Descripción: FOCO CLARO CHIYODA 40 WTS

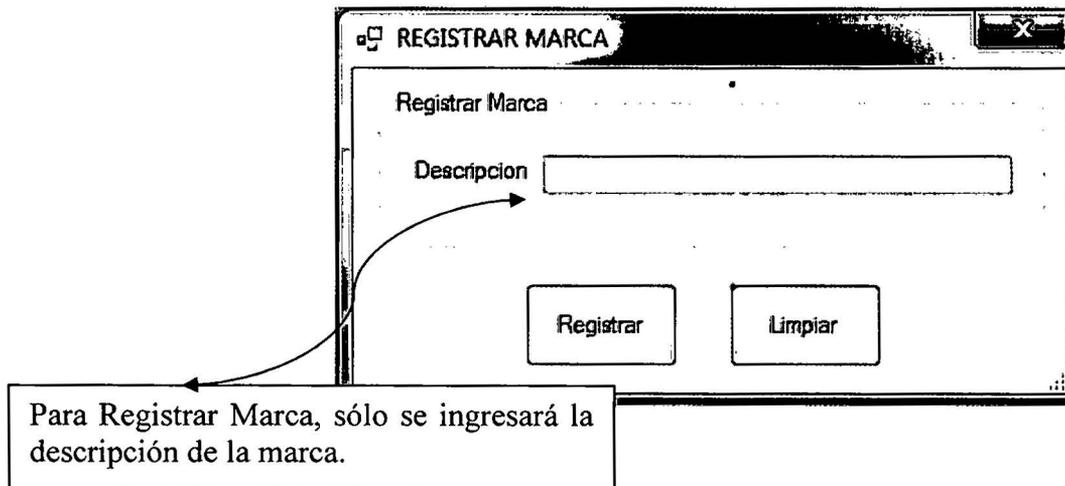
Talla: 15.00 Precio: 1.50 Stock: 0.000

DAR BAJA

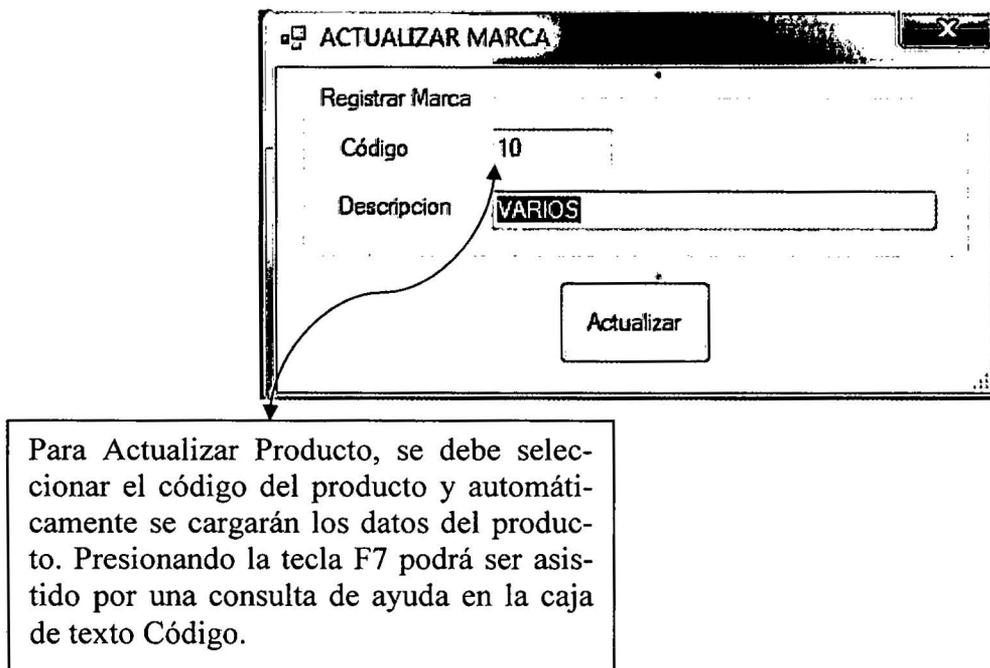
Para Dar de Baja Producto, se debe seleccionar el código del producto y automáticamente se cargarán los datos del producto. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código

A continuación se muestra las transacciones de Marca, Modelo y Tipo de Producto que tienen la misma estructura:

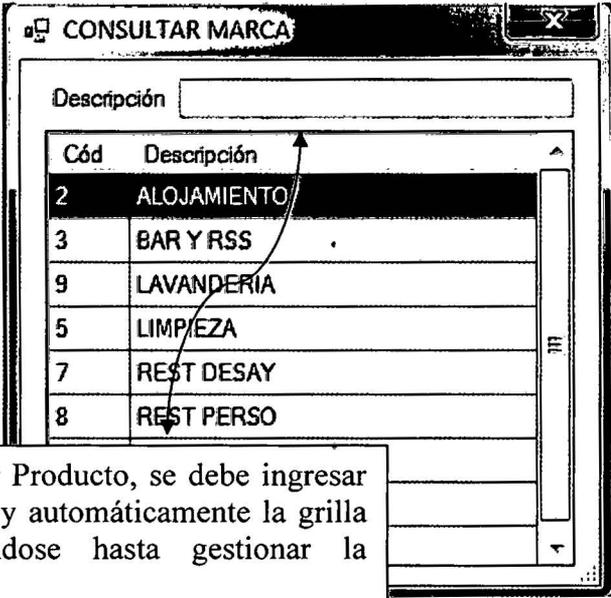
Registro Marca/Modelo/Tipo de Producto:



Actualizar Marca/Modelo/Tipo de Producto:



Consultar Marca/Modelo/Tipo de Producto:



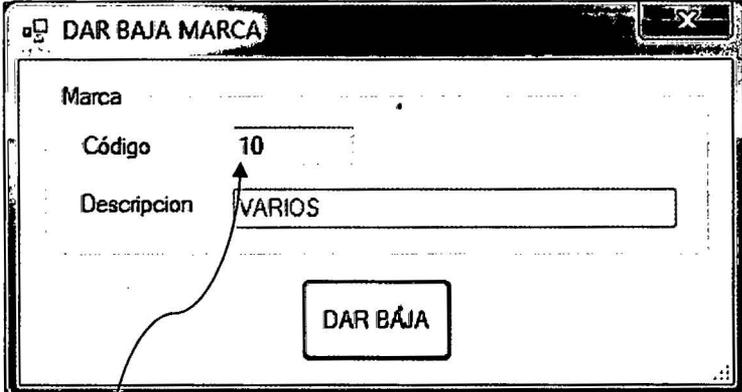
The screenshot shows a window titled "CONSULTAR MARCA". At the top, there is a text input field labeled "Descripción". Below it is a table with two columns: "Cód" and "Descripción". The table contains the following data:

Cód	Descripción
2	ALOJAMIENTO
3	BAR Y RSS
9	LAVANDERIA
5	LIMPIEZA
7	REST DESAY
8	REST PERSO

Below the table is a text box containing the following text:

Para Consultar Producto, se debe ingresar la descripción y automáticamente la grilla irá actualizándose hasta gestionar la búsqueda.

Dar de Baja Marca/Modelo/Tipo de Producto:



The screenshot shows a window titled "DAR BAJA MARCA". It contains a form with the following fields:

- Marca: (empty)
- Código: 10
- Descripcion: VARIOS

At the bottom of the form is a button labeled "DAR BAJA".

Para Dar de Baja Producto, se debe seleccionar el código de la Marca y automáticamente se cargarán los datos de la Marca. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código

Transacciones de Proveedor:

Registrar Proveedor:

REGISTRAR PROVEEDOR

Registrar Proveedor

Ruc

Descripción

Teléfono

Dirección

Registrar Limpiar

Para Registrar Proveedor, se ingresará el RUC, la Descripción (Razón Social), Teléfono y Dirección del Proveedor.

Actualizar Proveedor:

ACTUALIZAR PROVEEDOR

Registrar Proveedor

Ruc

10053007762

Descripción

"LA MIELE"

Teléfono

23-4424

Dirección

YANAMONO-RIO AMAZONAS

Actualizar

Para Actualizar Proveedor, se debe seleccionar el RUC del proveedor y automáticamente se cargarán los datos del producto. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código.

Consultar Proveedor:

CONSULTAR PROVEEDOR

Descripción:

Ruc	Descripción	Telefono	Direccion
20493322975	"FERRIMAQ"	222822	
10165320706	"FRUTAS DE LA COSTA"	226147	CALLE ARICA 1114
10053007762	"LA MIELE"	23-4424	YANAMONO-RIO AMAZONAS
10052904451	"MI NATURALEZA AMAZONICA"		GARCIA SANZ 342
20528317091	2000% GRADOS EIRL	223137	CALLE E. AGUIRRE 701
20528105310	A & B ITALY SRL	234375	CALLE ARICA 1059
10448428147	A Y S ALIMENTOS Y SERVICIOS	0429788072	CALLE LAS PALMERAS 509
20103802947	A.F. BOULLOSA S.R.LTDA.	264592	AV GRAU N° 1876 - 1880
			ALFONSO UGARTE 127

Para Consultar Proveedor, se debe ingresar la descripción y automáticamente la grilla irá actualizándose hasta gestionar la búsqueda.

Dar de Baja Proveedor:

DAR BAJA PROVEEDOR

Proveedor

Ruc o Dni
20493322975

Descripción
"FERRIMAQ"

Telefono
222822

Dirección

Para Dar de Baja Proveedor, se debe seleccionar el RUC del Proveedor y automáticamente se cargarán los datos de la Marca. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código

Transacciones Transportistas:

Registrar Transportista:

REGISTRAR TRANSPORTISTA

Registrar Proveedor

Ruc

Descripción

Brevete

Teléfono

Dirección

Registrar

Limpiar

Para Registrar Proveedor, se ingresar el RUC, la Descripción (Razón Social), Teléfono, Brevete y Dirección del Proveedor.

Actualizar Transportista:

ACTUALIZAR TRANSPORTISTA

Actualizar Proveedor

Ruc

12341324325

Descripción

ALCY PALESTINI PASHE

Brevete

MY-1234

Teléfono

23232332

Dirección

DASFADSFA

Para Actualizar Transportista, se debe seleccionar el RUC del proveedor y automáticamente se cargarán los datos del producto. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código.

Consultar Transportista:

Ruc	Descripción	Brevete	Telefono	Dirección
12341324325	ALCY PALESTINI PASHE	MY-1234	23232332	DASFADSFSA

Para Consultar Transportista, se debe ingresar la descripción y automáticamente la grilla irá actualizándose hasta gestionar la búsqueda.

Dar de Baja Transportista:

Transportista

Ruc
12341324325

Descripción
ALCY PALESTINI PASHE

Brevete
MY-1234

Teléfono
23232332

Dirección
DASFADSFSA

DAR BAJA

Para Dar de Baja Transportista, se debe seleccionar el RUC del Transportista y automáticamente se cargarán los datos de la Marca. Presionando la tecla F7 podrá ser asistido por una consulta de ayuda en la caja de texto Código

Registrar Compra

COMPRAS DIRECTAS

Selecciona el Tipo de Comprobante: Factura o Boleta

Comprobante
Tipo Comprobante: BOLETA Serie Número

Registrar Limpiar

Proveedor Descripción Compra: 01/02/2011

Detalle Comprobante

Ingresa la Serie y el Número del Comprobante

Actualizar Eliminar

Selecciona al Proveedor

Selecciona al Producto

Ingresa Precio de Compra

Producto: 0.00

Compra Precio Importe Stock

Botones:

Registrar

Registra la cabecera de la compra y los detalles de la compra.

Limpiar

Limpiar las cajas de texto ingresadas actualmente por el usuario, siempre y cuando el usuario aun no las haya registrado.

Actualizar

Actualiza los detalles de las Compras Efectuadas. Se tiene que seleccionar una fila de la grilla al cual se le hace mención.

Eliminar

Elimina un detalle de una Compra. Se tiene que seleccionar una fila de la grilla al cual se le hace mención.

Registrar Guía de Remisión

Botones:

Registrar

Registra la cabecera de la compra y los detalles de la Guía de Remisión.

Limpiar

Limpiar las cajas de texto ingresadas actualmente por el usuario, siempre y cuando el usuario aun no las haya registrado.

Actualizar

Actualiza los detalles de las Guías de Remisión Efectuadas. Se tiene que seleccionar una fila de la grilla al cual se le hace mención.

Eliminar

Elimina un detalle de una Guía de Remisión. Se tiene que seleccionar una fila de la grilla al cual se le hace mención.

Anexo 11: Código Fuente

Módulo Almacén: A continuación se mostrará los principales registros de SisTriunfo

Registrar Producto

```
Imports SisAlmacenLN
Public Class RegistrarProducto
#Region "Propiedades"
    Dim dtCategoria, dtModelo, dtUnidadMedida As New DataTable
    Public Shared opc As String = 0
#End Region
#Region "Objetos"
    Dim objProducto As New Producto
    Dim objCategoria As New Categoria
    Dim objModelo As New Modelo
    Dim objUnidadMedida As New UnidadMedida
#End Region
#Region "Procedimiento"
    Sub Limpiar()
        txtIdMarca.Text = ""
        txtIdMarca.BackColor = Color.White
        txtMarca.Text = ""
        txtIdModelo.Text = ""
        txtIdModelo.BackColor = Color.White
        txtModelo.Text = ""
        txtIdTipoProducto.Text = ""
        txtTipoProducto.Text = ""
        txtProducto.Text = ""
        txtProducto.BackColor = Color.White
        txtTalla.Text = ""
        txtPrecio.Text = ""
        txtStock.Text = ""
        txtIdMarca.Focus()
    End Sub
#End Region
#Region "Evento_Load"
    Private Sub RegistrarProducto_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

    End Sub
#End Region
#Region "Evento_Click"
    Private Sub btnRegistrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
    Try
        Dim Resultado As MsgBoxResult = MessageBox.Show("Deseas Guardar Datos?", "Mensaje", MessageBoxButtons.YesNo,
        MessageBoxIcon.Information)
        If Resultado = MsgBoxResult.Yes Then
            If txtIdMarca.Text <> "" Then
                dtCategoria = objCategoria.ListarIdEstado(txtIdMarca.Text, 1)
                If dtCategoria.Rows.Count <> 0 Then
                    txtIdMarca.BackColor = Color.White
                    txtMarca.Text = dtCategoria.Rows(0).Item("Descripcion").ToString()
                If txtIdModelo.Text <> "" Then
                    dtModelo = objModelo.ListarIdEstado(txtIdModelo.Text, 1)
                    If dtModelo.Rows.Count <> 0 Then
                        txtIdModelo.BackColor = Color.White
                        txtModelo.Text = dtModelo.Rows(0).Item("Descripcion").ToString()
                    If txtIdTipoProducto.Text <> "" Then
                        dtUnidadMedida = objUnidadMedida.ListarIdEstado(txtIdTipoProducto.Text, 1)
                        If dtUnidadMedida.Rows.Count <> 0 Then
                            txtIdTipoProducto.BackColor = Color.White
                            txtTipoProducto.Text = dtUnidadMedida.Rows(0).Item("Descripcion").ToString()
                        If txtTalla.Text <> "" Then
                            txtTalla.BackColor = Color.White
                        If txtPrecio.Text <> "" Then
                            txtPrecio.BackColor = Color.White
                            txtPrecio.Text = Format(CDb1(txtPrecio.Text), "#,##0.00")
                        If txtStock.Text <> "" Then
                            txtStock.BackColor = Color.White
                            txtStock.Text = Format(CDb1(txtStock.Text), "#,##0.000")
                            objProducto.Ingresar(Format(CDb1(objProducto.ObtenerId + 1), "000000"), txtProducto.Text,
                            txtIdTipoProducto.Text, txtIdMarca.Text, txtIdModelo.Text, txtTalla.Text, txtPrecio.Text, txtStock.Text, 1)
                            Limpiar()
                        End If
                    End If
                End If
            End If
        End If
    End Try
End Sub
#End Region
```

```
        MsgBox("Datos Registrados Correctamente", MsgBoxStyle.Information, "Mensaje")
        opc = "1"
    Else
        txtStock.BackColor = Color.Pink
        txtStock.Focus()
    End If
    Else
        txtPrecio.BackColor = Color.Pink
        txtPrecio.Focus()
    End If
    Else
        txtTalla.BackColor = Color.Pink
        txtTalla.Focus()
    End If
    Else
        txtIdTipoProducto.BackColor = Color.Pink
        txtIdTipoProducto.Text = ""
        txtIdTipoProducto.Focus()
    End If
    dtUnidadMedida.Clear()
    Else
        txtIdTipoProducto.BackColor = Color.Pink
        txtIdTipoProducto.Text = ""
        txtIdTipoProducto.Focus()
    End If
    Else
        txtIdModelo.BackColor = Color.Pink
        txtIdModelo.Text = ""
        txtIdModelo.Focus()
    End If
    dtModelo.Clear()
    Else
        txtIdModelo.BackColor = Color.Pink
        txtIdModelo.Text = ""
        txtIdModelo.Focus()
    End If
    Else
        txtIdMarca.BackColor = Color.Pink
        txtIdMarca.Text = ""
        txtIdMarca.Focus()
    End If
    dtCategoria.Clear()
    Else
        txtIdMarca.BackColor = Color.Pink
        txtIdMarca.Text = ""
        txtIdMarca.Focus()
    End If
    End If
Catch ex As Exception
    MsgBox("Error: " + ex.Message)
End Try
End Sub
Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub
#End Region
#Region "Evento_KeyPress"
Private Sub txtIdCategoria_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtIdMarca.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtIdMarca.Text <> "" Then
            dtCategoria = objCategoria.ListarIdEstado(txtIdMarca.Text, 1)
            If dtCategoria.Rows.Count <> 0 Then
                txtIdMarca.BackColor = Color.White
                txtIdMarca.Text = dtCategoria.Rows(0).Item("Descripcion").ToString()
                txtIdModelo.Focus()
            Else
                txtIdMarca.BackColor = Color.Pink
                txtIdMarca.Text = ""
                txtIdMarca.Focus()
            End If
        End If
    End If
End Sub
```

```
        End If
        dtCategoria.Clear()
    Else
        txtIdMarca.BackColor = Color.Pink
        txtIdMarca.Text = ""
        txtIdMarca.Focus()
    End If
End If
End Sub
Private Sub txtIdModelo_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtIdModelo.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtIdModelo.Text <> "" Then
            dtModelo = objModelo.ListarIdEstado(txtIdModelo.Text, 1)
            If dtModelo.Rows.Count <> 0 Then
                txtIdModelo.BackColor = Color.White
                txtModelo.Text = dtModelo.Rows(0).Item("Descripcion").ToString()
                txtIdTipoProducto.Focus()
            Else
                txtIdModelo.BackColor = Color.Pink
                txtIdModelo.Text = ""
                txtIdModelo.Focus()
            End If
            dtModelo.Clear()
        Else
            txtIdModelo.BackColor = Color.Pink
            txtIdModelo.Text = ""
            txtIdModelo.Focus()
        End If
    End If
End Sub
Private Sub txtIdTipoProducto_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtIdTipoProducto.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtIdTipoProducto.Text <> "" Then
            dtUnidadMedida = objUnidadMedida.ListarIdEstado(txtIdTipoProducto.Text, 1)
            If dtUnidadMedida.Rows.Count <> 0 Then
                txtIdTipoProducto.BackColor = Color.White
                txtTipoProducto.Text = dtUnidadMedida.Rows(0).Item("Descripcion").ToString()
                txtProducto.Focus()
            Else
                txtIdTipoProducto.BackColor = Color.Pink
                txtIdTipoProducto.Text = ""
                txtIdTipoProducto.Focus()
            End If
            dtUnidadMedida.Clear()
        Else
            txtIdTipoProducto.BackColor = Color.Pink
            txtIdTipoProducto.Text = ""
            txtIdTipoProducto.Focus()
        End If
    End If
End Sub
Private Sub txtProducto_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtProducto.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtProducto.Text <> "" Then
            txtProducto.BackColor = Color.White
            txtTalla.Focus()
        Else
            txtProducto.BackColor = Color.Cyan
            txtTalla.Focus()
        End If
    End If
End Sub
```

```
End If
End Sub
Private Sub txtTalla_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtTalla.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtTalla.Text <> "" Then
            txtTalla.BackColor = Color.White
            txtTalla.Text = Format(CDbl(txtTalla.Text), "###0.00")
            txtPrecio.Focus()
        Else
            txtTalla.BackColor = Color.Pink
            txtTalla.Focus()
        End If
    End If
End Sub
Private Sub txtPrecio_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtPrecio.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtPrecio.Text <> "" Then
            txtPrecio.BackColor = Color.White
            txtPrecio.Text = Format(CDbl(txtPrecio.Text), "###0.00")
            txtStock.Focus()
        Else
            txtPrecio.BackColor = Color.Pink
            txtPrecio.Focus()
        End If
    End If
End Sub
Private Sub txtStock_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtStock.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtStock.Text <> "" Then
            txtStock.BackColor = Color.White
            txtStock.Text = Format(CDbl(txtStock.Text), "###0.000")
            btnRegistrar.Focus()
        Else
            txtStock.BackColor = Color.Pink
            txtStock.Focus()
        End If
    End If
End Sub
#End Region
#Region "Evento_KeyDown"
Private Sub txtIdMarca_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txtIdMarca.KeyDown
    If e.KeyCode = Keys.F7 Then
        Dim objConsultarMarca As New ConsultarMarcaActualizar
        objConsultarMarca.ShowDialog()
        If ConsultarMarcaActualizar.opc = "1" Then
            txtIdMarca.BackColor = Color.White
            ConsultarMarcaActualizar.opc = ""
            txtIdMarca.Text = ConsultarMarcaActualizar.IdMarca
            txtMarca.Text = ConsultarMarcaActualizar.Descripcion
            ConsultarMarcaActualizar.IdMarca = ""
            ConsultarMarcaActualizar.Descripcion = ""
            txtIdModelo.Focus()
        End If
    End If
End Sub
Private Sub txtIdModelo_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txtIdModelo.KeyDown
    If e.KeyCode = Keys.F7 Then
        Dim objConsultarModelo As New ConsultarModeloActualizar
        objConsultarModelo.ShowDialog()
        If ConsultarModeloActualizar.opc = "1" Then
            txtIdModelo.BackColor = Color.White
            ConsultarModeloActualizar.opc = ""
            txtIdModelo.Text = ConsultarModeloActualizar.IdModelo
            txtModelo.Text = ConsultarModeloActualizar.Descripcion
            ConsultarModeloActualizar.IdModelo = ""
            ConsultarModeloActualizar.Descripcion = ""
            txtIdTipoProducto.Focus()
        End If
    End If
End Sub
Private Sub txtIdTipoProducto_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txtIdTipoProducto.KeyDown
    If e.KeyCode = Keys.F7 Then
```

```
Dim objConsultarTipoProducto As New ConsultarTipoProductoActualizar
objConsultarTipoProducto.ShowDialog()
If ConsultarTipoProductoActualizar.opc = "1" Then
    txtIdTipoProducto.BackColor = Color.White
    ConsultarTipoProductoActualizar.opc = ""
    txtIdTipoProducto.Text = ConsultarTipoProductoActualizar.IdTipoProducto
    txtTipoProducto.Text = ConsultarTipoProductoActualizar.Descripcion
    ConsultarTipoProductoActualizar.IdTipoProducto = ""
    ConsultarTipoProductoActualizar.Descripcion = ""
    txtProducto.Focus()
End If
End If
End Sub
Private Sub RegistrarProducto_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
Me.KeyDown
    If e.KeyCode = Keys.Escape Then
        If opc <> 1 Then
            opc = 0
        End If
    End If
    Close()
End If
End Sub
#End Region
#Region "Evento_FormClosing"
Private Sub RegistrarProducto_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles
Me.FormClosing
    If opc <> 1 Then
        opc = 0
    End If
    e.Cancel = False
End Sub
#End Region

End Class
```

Registrar Proveedor

```
Imports SisAlmacenLN
Public Class RegistrarProveedor
#Region "Propiedades"
    Public Shared Ruc As String
    Public Shared opc As String = 0
#End Region
#Region "Objetos"
    Dim objProveedor As New Proveedor
#End Region
#Region "Procedimiento"
    Sub Limpiar()
        txtRuc.Text = ""
        txtDescripcion.Text = ""
        txtDescripcion.BackColor = Color.White
        txtTelf.Text = ""
        txtTelf.BackColor = Color.White
        txtDireccion.Text = ""
        txtDireccion.BackColor = Color.White
        txtRuc.Focus()
    End Sub
#End Region
#Region "Evento_Load"
    Private Sub RegistrarClientes_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

        End Sub
#End Region
#Region "Evento_Click"
    Private Sub btnRegistrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
        Try
            Dim Resultado As MsgBoxResult = MessageBox.Show("Deseas Guardar Datos?", "Mensaje", MessageBoxButtons.YesNo,
                MessageBoxIcon.Information)
            If Resultado = MsgBoxResult.Yes Then
                If txtRuc.Text <> "" Then
                    txtRuc.BackColor = Color.White
                    If txtDescripcion.Text <> "" Then
                        txtDescripcion.BackColor = Color.White
                        If txtDireccion.Text <> "" Then
                            txtDireccion.BackColor = Color.White
                            objProveedor.Ingresar(txtRuc.Text, Trim(txtDescripcion.Text), "", txtTelf.Text, Trim(txtDireccion.Text), "Proveedor")
                            MsgBox("Datos Registrados Correctamente", MsgBoxStyle.Information, "Mensaje")
                            Limpiar()
                        Else
                            txtDireccion.BackColor = Color.Pink
                            txtDireccion.Focus()
                        End If
                    Else
                        txtDescripcion.BackColor = Color.Pink
                        txtDescripcion.Focus()
                    End If
                Else
                    txtRuc.BackColor = Color.Pink
                    txtRuc.Focus()
                End If
            End If
            Catch ex As Exception
                MsgBox("Error: " + ex.Message)
            End Try
        End Sub
    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub
#End Region
#Region "Evento_KeyPress"
    Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
        Try
            If Char.IsDigit(e.KeyChar) Then
                e.Handled = False
            ElseIf Char.IsControl(e.KeyChar) Then
                e.Handled = False
            Else
                e.Handled = True
            End If
            If Asc(e.KeyChar) = 13 Then
                If txtRuc.Text <> "" Then
                    txtRuc.BackColor = Color.White
                End If
            End If
        End Try
    End Sub
End Class
```

```
        txtDescripcion.Focus()
    Else
        txtRuc.BackColor = Color.Pink
        txtRuc.Focus()
    End If
End If
Catch ex As Exception
    MsgBox("Error: " + ex.Message)
End Try
End Sub
Private Sub txtDescripcion_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtDescripcion.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtDescripcion.Text <> "" Then
            txtDescripcion.BackColor = Color.White
            txtTelf.Focus()
        Else
            txtDescripcion.BackColor = Color.Pink
            txtDescripcion.Focus()
        End If
    End If
End Sub
Private Sub txtTelf_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtTelf.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtTelf.Text <> "" Then
            txtTelf.BackColor = Color.White
            txtDireccion.Focus()
        Else
            txtTelf.BackColor = Color.Cyan
            txtDireccion.Focus()
        End If
    End If
End Sub
Private Sub txtDireccion_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtDireccion.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtDireccion.Text <> "" Then
            txtDireccion.BackColor = Color.White
            btnRegistrar.Focus()
        Else
            txtDireccion.BackColor = Color.Pink
            txtDireccion.Focus()
        End If
    End If
End Sub
#End Region
#Region "Evento_KeyDown"
Private Sub RegistrarClientes_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
Me.KeyDown
    If e.KeyCode = Keys.Escape Then
        If opc <> 1 Then
            opc = 0
        End If
        Close()
    End If
End Sub
#End Region
#Region "Evento_FormClosing"
Private Sub RegistrarClientes_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs) Handles
Me.FormClosing
    If opc <> 1 Then
        opc = 0
    End If
    e.Cancel = False
End Sub
#End Region
End Class
```

Registrar Transportista

```
Imports SisAlmacenLN
Public Class RegistrarTransportista
#Region "Propiedades"
    Public Shared Ruc As String
    Public Shared opc As String = 0
#End Region
#Region "Objetos"
    Dim objTransportista As New Transportista
#End Region
#Region "Procedimiento"
    Sub Limpiar()
        txtRuc.Text = ""
        txtRuc.BackColor = Color.White
        txtDescripcion.Text = ""
        txtDescripcion.BackColor = Color.White
        txtBrevete.Text = ""
        txtBrevete.BackColor = Color.White
        txtTelf.Text = ""
        txtTelf.BackColor = Color.White
        txtDireccion.Text = ""
        txtDireccion.BackColor = Color.White
        txtRuc.Focus()
    End Sub
#End Region
#Region "Evento_Load"
    Private Sub RegistrarTransportista_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

        End Sub
#End Region
#Region "Evento_Click"
    Private Sub btnRegistrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
        Try
            Dim Resultado As MsgBoxResult = MessageBox.Show("Deseas Guardar Datos?", "Mensaje", MessageBoxButtons.YesNo,
                MessageBoxIcon.Information)
            If Resultado = MsgBoxResult.Yes Then
                If txtRuc.Text <> "" Then
                    txtRuc.BackColor = Color.White
                If txtDescripcion.Text <> "" Then
                    txtDescripcion.BackColor = Color.White
                If txtBrevete.Text <> "" Then
                    txtBrevete.BackColor = Color.White
                If txtDireccion.Text <> "" Then
                    txtDireccion.BackColor = Color.White
                objProveedor.Ingresar(txtRuc.Text, Trim(txtDescripcion.Text), Trim(txtBrevete.Text), txtTelf.Text, Trim(txtDireccion.Text),
"Transportista")
                MsgBox("Datos Registrados Correctamente", MsgBoxStyle.Information, "Mensaje")
                Limpiar()
            Else
                txtDireccion.BackColor = Color.Pink
                txtDireccion.Focus()
            End If
            Else
                txtBrevete.BackColor = Color.Pink
                txtBrevete.Focus()
            End If
            Else
                txtDescripcion.BackColor = Color.Pink
                txtDescripcion.Focus()
            End If
            Else
                txtRuc.BackColor = Color.Pink
                txtRuc.Focus()
            End If
        End If
    Catch ex As Exception
        MsgBox("Error: " + ex.Message)
    End Try
    End Sub
    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub
#End Region
#Region "Evento_KeyPress"
    Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
        Try
```

```
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtRuc.Text <> "" Then
            txtRuc.BackColor = Color.White
            txtDescripcion.Focus()
        Else
            txtRuc.BackColor = Color.Pink
            txtRuc.Focus()
        End If
    End If
    Catch ex As Exception
        MsgBox("Error: " + ex.Message)
    End Try
End Sub
Private Sub txtDescripcion_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtDescripcion.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtDescripcion.Text <> "" Then
            txtDescripcion.BackColor = Color.White
            txtBrevete.Focus()
        Else
            txtDescripcion.BackColor = Color.Pink
            txtDescripcion.Focus()
        End If
    End If
End Sub
Private Sub txtBrevete_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtBrevete.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtBrevete.Text <> "" Then
            txtBrevete.BackColor = Color.White
            txtTelf.Focus()
        Else
            txtBrevete.BackColor = Color.Pink
            txtBrevete.Focus()
        End If
    End If
End Sub
Private Sub txtTelf_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtTelf.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtTelf.Text <> "" Then
            txtTelf.BackColor = Color.White
            txtDireccion.Focus()
        Else
            txtTelf.BackColor = Color.Cyan
            txtDireccion.Focus()
        End If
    End If
End Sub
Private Sub txtDireccion_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtDireccion.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtDireccion.Text <> "" Then
            txtDireccion.BackColor = Color.White
            btnRegistrar.Focus()
        Else
            txtDireccion.BackColor = Color.Pink
            txtDireccion.Focus()
        End If
    End If
End Sub
#End Region
#Region "Evento_KeyDown"
Private Sub RegistrarTransportista_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
Me.KeyDown
    If e.KeyCode = Keys.Escape Then
        If opc <> 1 Then
            opc = 0
        End If
        Close()
    End If
End Sub
```

```
End Sub
#End Region
#Region "Evento_FormClosing"
Private Sub RegistrarTransportista_FormClosing(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosingEventArgs)
Handles Me.FormClosing
    If opc <> 1 Then
        opc = 0
    End If
    e.Cancel = False
End Sub
#End Region
End Class
```

Registrar Compra

```
Imports SisAlmacenLN
Public Class CompraDirecta
#Region "Propiedades"
    Dim Array As New ArrayList
    Dim Ruc, FechaActual, HoraActual, Descripcion As String
    Dim IdComprobante, IdDetallePedido, r, varUdp, j As Integer, i As Integer = 0
    Dim udp As Boolean = False
    Dim Stock, Total As Decimal
    Dim dtProducto, dtDPedido, dtProveedor, dtConfiguracion As New DataTable
#End Region
#Region "Objetos"
    Dim objComprobante As New Comprobante
    Dim objProducto As New Producto
    Dim objProveedor As New Proveedor
    Dim objPedido As New Pedido
    Dim objArea As New Area
    Dim objKardex As New Kardex
#End Region
#Region "Procedimiento"
    Sub CargarCombo()
        cmbTipoComprobante.DataSource = objComprobante.ListarTipo()
        cmbTipoComprobante.ValueMember = "IdTipoComprobante"
        cmbTipoComprobante.DisplayMember = "Descripcion"
    End Sub
    Sub LimpiarDataGridView()
        For r = 0 To dgvProductos.Rows.Count - 1
            dgvProductos.Rows.Remove(dgvProductos.CurrentRow)
        Next r
        i = 0
    End Sub
    Sub CargarDgvVentas()
        Try
            If txtCodigo.Text <> "" Then
                dgvProductos.Rows.Add(1)
                dgvProductos.Rows(i).Cells(0).Value = txtCodigo.Text
                dgvProductos.Rows(i).Cells(1).Value = txtCompro.Text
                dgvProductos.Rows(i).Cells(2).Value = txtDescripcion.Text
                dgvProductos.Rows(i).Cells(3).Value = txtPrecioCompro.Text
                dgvProductos.Rows(i).Cells(4).Value = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                dgvProductos.Rows(i).Cells(5).Value = txtStock.Text
                dgvProductos.Rows(i).Cells(6).Value = txtPrecioComprar.Text
                txtTotal.Text = Format(CDbl(Convert.ToDecimal(txtTotal.Text) + Convert.ToDecimal(dgvProductos.Rows(i).Cells(4).Value)),
                "#,###,##0.00")
                i = i + 1
            End If
        Catch ex As Exception
            MsgBox("Error: " + ex.Message)
        End Try
    End Sub
    Sub TextBoxActivado()
        txtCompro.Enabled = True
        txtPrecioCompro.Enabled = True
        txtImporte.Enabled = True
    End Sub
    Sub TextBoxDesActivado()
        txtCompro.Enabled = False
        txtPrecioCompro.Enabled = False
        txtImporte.Enabled = False
    End Sub
    Sub Limpiar()
        txtCodigo.Text = ""
        txtCompro.Text = ""
        txtDescripcion.Text = ""
        txtPrecioCompro.Text = ""
        txtImporte.Text = ""
        txtStock.Text = ""
        txtPrecioComprar.Text = ""
        udp = False
        txtCompro.Focus()
    End Sub
#End Region
#Region "Evento_Load"
    Private Sub CompraDirecta_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        dgvProductos.SelectionMode = DataGridViewSelectionMode.FullRowSelect
        CargarCombo()
    End Sub
End Class
```

```
txtCompro.Text = ""
txtPrecioCompro.Text = ""
txtImporte.Text = ""
txtTotal.Text = "0.00"
End Sub
#End Region
#Region "Evento_Click"
Private Sub btnRegistrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
Try
Dim Resultado As MsgBoxResult = MessageBox.Show("Deseas Guardar Datos?", "Mensaje", MessageBoxButtons.YesNo,
MessageBoxIcon.Information)
If Resultado = MsgBoxResult.Yes Then
If txtSerie.Text <> "" Then
txtSerie.Text = Format(CDbl(txtSerie.Text), "000")
txtSerie.BackColor = Color.White
If txtNumero.Text <> "" Then
txtNumero.Text = Format(CDbl(txtNumero.Text), "000000")
txtNumero.BackColor = Color.White
If txtProveedorRucDni.Text <> "" Then
If txtProveedorRucDni.Text.Length = 8 Or txtProveedorRucDni.Text.Length = 11 Then
dtProveedor = objProveedor.ListarRucEstado(txtProveedorRucDni.Text, 1)
If dtProveedor.Rows.Count <> 0 Then
txtProveedorRucDni.BackColor = Color.White
txtProveedorDescripcion.Text = dtProveedor.Rows(0).Item("Descripcion").ToString()
dtProveedor.Clear()
If dgvProductos.RowCount <> 0 Then
Dim p As Integer
objComprobante.Ingresar(txtSerie.Text, txtNumero.Text, cmbTipoComprobante.Selected.Value, 2,
txtProveedorRucDni.Text, txtFechaCompra.Text, 1)
IdComprobante = objComprobante.ObtenerId
For p = 0 To dgvProductos.Rows.Count - 1
objProducto.ActualizarStock(dgvProductos.Rows(p).Cells(0).Value, dgvProductos.Rows(p).Cells(1).Value)
Stock = objProducto.ObtenerStock(dgvProductos.Rows(p).Cells(0).Value)
If Convert.ToDecimal(dgvProductos.Rows(p).Cells(3).Value) <>
Convert.ToDecimal(dgvProductos.Rows(p).Cells(6).Value) Then
objProducto.ActualizarPrecio(dgvProductos.Rows(p).Cells(0).Value, dgvProductos.Rows(p).Cells(3).Value)
End If
objComprobante.IngresarComprobanteDetallePedido(IdComprobante, dgvProductos.Rows(p).Cells(0).Value,
dgvProductos.Rows(p).Cells(1).Value, dgvProductos.Rows(p).Cells(3).Value, 1)
objKardex.Ingresar(txtFechaCompra.Text, dgvProductos.Rows(p).Cells(0).Value, "C", cmbTipoComprobante.Text,
txtSerie.Text + "-" + txtNumero.Text, txtProveedorDescripcion.Text, "HVR-ALMACEN", "", dgvProductos.Rows(p).Cells(3).Value,
dgvProductos.Rows(p).Cells(1).Value, 0, Stock, 1)
Next p
Limpiar()
LimpiarDataGridView()
cmbTipoComprobante.SelectedIndex = 0
txtFechaCompra.Text = DateAndTime.Now.ToShortDateString
txtSerie.Text = ""
txtNumero.Text = ""
txtProveedorRucDni.Text = ""
txtProveedorDescripcion.Text = ""
txtTotal.Text = "0.00"
cmbTipoComprobante.Focus()
MsgBox("Comprobante Registrado Correctamente", MsgBoxStyle.Information, "Mensaje")
Else
MsgBox("Ingrese al Menos un Item", MsgBoxStyle.Exclamation, "Mensaje")
End If
Else
dtProveedor.Clear()
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
Else
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
Else
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
Else
txtNumero.BackColor = Color.Pink
txtNumero.Focus()
End If
End If
```

```
        Else
            txtSerie.BackColor = Color.Pink
            txtSerie.Focus()
        End If
    End If
Catch ex As Exception
    MsgBox("Error: " + ex.Message)
End Try
End Sub
Private Sub btnActualizar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnActualizar.Click
    If dgvProductos.Rows.Count <> 0 Then
        txtCodigo.Text = dgvProductos.Item(0, dgvProductos.CurrentRow.Index).Value
        txtCompro.Text = dgvProductos.Item(1, dgvProductos.CurrentRow.Index).Value
        txtDescripcion.Text = dgvProductos.Item(2, dgvProductos.CurrentRow.Index).Value
        txtPrecioCompro.Text = dgvProductos.Item(3, dgvProductos.CurrentRow.Index).Value
        txtImporte.Text = dgvProductos.Item(4, dgvProductos.CurrentRow.Index).Value
        txtStock.Text = dgvProductos.Item(5, dgvProductos.CurrentRow.Index).Value
        varUdp = dgvProductos.CurrentRow.Index
        udp = True
        TextBoxActivado()
        txtCompro.Select(0, 10)
        txtCompro.Focus()
    End If
End Sub
Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnEliminar.Click
    Dim Resultado As MsgBoxResult = MessageBox.Show("Desea Eliminar Item?", "Mensaje", MessageBoxButtons.YesNo,
    MessageBoxIcon.Information)
    If Resultado = MsgBoxResult.Yes Then
        If dgvProductos.Rows.Count <> 0 Then
            txtTotal.Text = Format(CDbl(txtTotal.Text - dgvProductos.Item(6, dgvProductos.CurrentRow.Index).Value), "#,###,##0.00")
            i = i - 1
            dgvProductos.Rows.Remove(dgvProductos.CurrentRow)
        End If
    End If
    TextBoxActivado()
    Limpiar()
End Sub
#End Region
#Region "Evento_KeyPress"
Private Sub cmbTipoComprobante_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
cmbTipoComprobante.KeyPress
    If Asc(e.KeyChar) = 13 Then
        txtSerie.Focus()
    End If
End Sub
Private Sub txtSerie_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtSerie.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtSerie.Text <> "" Then
            txtSerie.Text = Format(CDbl(txtSerie.Text), "000")
            txtSerie.BackColor = Color.White
            txtNumero.Focus()
        Else
            txtSerie.BackColor = Color.Pink
            txtSerie.Focus()
        End If
    End If
End Sub
Private Sub txtNumero_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtNumero.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtNumero.Text <> "" Then
            txtNumero.Text = Format(CDbl(txtNumero.Text), "000000")
            txtNumero.BackColor = Color.White
        End If
    End If
End Sub
```

```
        txtProveedorRucDni.Focus()
    Else
        txtNumero.BackColor = Color.Pink
        txtNumero.Focus()
    End If
End If
End Sub
Private Sub txtProveedorRucDni_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtProveedorRucDni.KeyPress
    Try
        If Char.IsDigit(e.KeyChar) Then
            e.Handled = False
        ElseIf Char.IsControl(e.KeyChar) Then
            e.Handled = False
        Else
            e.Handled = True
        End If
        If Asc(e.KeyChar) = 13 Then
            If txtProveedorRucDni.Text <> "" Then
                If txtProveedorRucDni.Text.Length = 8 Or txtProveedorRucDni.Text.Length = 11 Then
                    dtProveedor = objProveedor.ListarRucEstado(txtProveedorRucDni.Text, 1)
                    If dtProveedor.Rows.Count <> 0 Then
                        txtProveedorRucDni.BackColor = Color.White
                        txtProveedorDescripcion.Text = dtProveedor.Rows(0).Item("Descripcion").ToString()
                        txtFechaCompra.Focus()
                    Else
                        txtProveedorRucDni.BackColor = Color.Pink
                        txtProveedorDescripcion.Text = ""
                        txtProveedorRucDni.Focus()
                    End If
                    dtProveedor.Clear()
                Else
                    txtProveedorRucDni.BackColor = Color.Pink
                    txtProveedorDescripcion.Text = ""
                    txtProveedorRucDni.Focus()
                End If
            Else
                txtProveedorRucDni.BackColor = Color.Pink
                txtProveedorDescripcion.Text = ""
                txtProveedorRucDni.Focus()
            End If
        End If
    Catch ex As Exception
        MsgBox("Error: " + ex.Message)
    End Try
End Sub
Private Sub txtFechaCompra_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtFechaCompra.KeyPress
    If Asc(e.KeyChar) = 13 Then
        txtCompro.Focus()
    End If
End Sub
Private Sub txtCompro_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtCompro.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtCompro.Text <> "" And txtCompro.Text <> "0.000" Then
            txtCompro.BackColor = Color.White
            If txtPrecioCompro.Text <> "" And txtPrecioCompro.Text <> "0.00" Then
                txtPrecioCompro.BackColor = Color.White
                txtImporte.Text = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                txtPrecioCompro.Select(0, 10)
                txtPrecioCompro.Focus()
            Else
                txtPrecioCompro.BackColor = Color.Pink
                txtPrecioCompro.Text = "0"
                txtImporte.Text = "0"
                txtPrecioCompro.Select(0, 10)
                txtPrecioCompro.Focus()
            End If
        End If
    Else
        txtCompro.BackColor = Color.Pink
        txtCompro.Text = "0"
        txtImporte.Text = "0"
        txtCompro.Select(0, 10)
        txtCompro.Focus()
    End If
End If
End If
```

```
End Sub
Private Sub txtPrecioCompro_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtPrecioCompro.KeyPress
    Try
        If Asc(e.KeyChar) = 13 Then
            If txtCompro.Text <> "" And txtCompro.Text <> "0.000" Then
                txtCompro.BackColor = Color.White
            If txtPrecioCompro.Text <> "" And txtPrecioCompro.Text <> "0.00" Then
                txtPrecioCompro.BackColor = Color.White
                txtImporte.Text = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                txtImporte.Select(0, 10)
                txtImporte.Focus()
            Else
                txtPrecioCompro.BackColor = Color.Pink
                txtPrecioCompro.Text = "0"
                txtImporte.Text = "0"
                txtPrecioCompro.Select(0, 10)
                txtPrecioCompro.Focus()
            End If
        End If
    Else
        txtCompro.BackColor = Color.Pink
        txtCompro.Text = "0"
        txtImporte.Text = "0"
        txtCompro.Select(0, 10)
        txtCompro.Focus()
    End If
End If
Catch ex As Exception
    MsgBox("Error: " + ex.Message)
End Try
End Sub
Private Sub txtImporte_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtImporte.KeyPress
    Try
        If Asc(e.KeyChar) = 13 Then
            If txtDescripcion.Text <> "" Then
                If txtCompro.Text <> "" And txtCompro.Text <> "0.000" Then
                    txtCompro.BackColor = Color.White
                If txtPrecioCompro.Text <> "" And txtPrecioCompro.Text <> "0.00" Then
                    txtPrecioCompro.BackColor = Color.White
                    txtImporte.Text = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                If udp Then
                    txtTotal.Text = Format(CDbl(Convert.ToDecimal(txtTotal.Text) -
Convert.ToDecimal(dgvProductos.Rows(varUdp).Cells(4).Value)), "#,###,##0.00")
                    dgvProductos.Rows(varUdp).Cells(0).Value = txtCodigo.Text
                    dgvProductos.Rows(varUdp).Cells(1).Value = txtCompro.Text
                    dgvProductos.Rows(varUdp).Cells(2).Value = txtDescripcion.Text
                    dgvProductos.Rows(varUdp).Cells(3).Value = txtPrecioCompro.Text
                    dgvProductos.Rows(varUdp).Cells(4).Value = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                    dgvProductos.Rows(varUdp).Cells(5).Value = txtStock.Text
                    dgvProductos.Rows(varUdp).Cells(6).Value = txtPrecioComprar.Text
                    txtTotal.Text = Format(CDbl(Convert.ToDecimal(txtTotal.Text) +
Convert.ToDecimal(dgvProductos.Rows(varUdp).Cells(4).Value)), "#,###,##0.00")
                    Limpiar()
                Else
                    CargarDgvVentas()
                    Limpiar()
                    Dim objBuscarProductos As New BuscarProductos
                    objBuscarProductos.ShowDialog()
                    If BuscarProductos.opc = "1" Then
                        txtCodigo.Text = BuscarProductos.IdProducto
                        txtDescripcion.Text = BuscarProductos.Descripcion
                        txtPrecioCompro.Text = BuscarProductos.Precio
                        txtStock.Text = BuscarProductos.Stock
                        txtPrecioComprar.Text = BuscarProductos.Precio

                        BuscarProductos.opc = 0
                        BuscarProductos.IdProducto = ""
                        BuscarProductos.Descripcion = ""
                        BuscarProductos.Precio = ""
                        BuscarProductos.Stock = ""
                        txtCompro.Focus()
                    End If
                End If
            Else
                txtPrecioCompro.BackColor = Color.Pink
                txtPrecioCompro.Text = "0"
            End If
        End Try
    End Sub
```

```
        txtImporte.Text = "0"  
        txtPrecioCompro.Select(0, 10)  
        txtPrecioCompro.Focus()  
    End If  
Else  
    txtCompro.BackColor = Color.Pink  
    txtCompro.Text = "0"  
    txtImporte.Text = "0"  
    txtCompro.Select(0, 10)  
    txtCompro.Focus()  
End If  
Else  
    MsgBox("Seleccione un Producto", MsgBoxStyle.Exclamation, "Mensaje")  
    txtCompro.Select(0, 10)  
    txtCompro.Focus()  
End If  
End If  
Catch ex As Exception  
    MsgBox("Error: " + ex.Message)  
End Try  
End Sub  
#End Region  
#Region "Evento_KeyDown"  
Private Sub txtProveedorRucDni_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles  
txtProveedorRucDni.KeyDown  
    Try  
        If e.KeyCode = Keys.F7 Then  
            Dim objBuscarClientes As New BuscarProveedor  
            objBuscarClientes.ShowDialog()  
            If BuscarProveedor.Temp = 1 Then  
                txtProveedorRucDni.BackColor = Color.White  
                txtProveedorRucDni.Text = BuscarProveedor.Ruc  
                txtProveedorDescripcion.Text = BuscarProveedor.Descripcion  
  
                BuscarProveedor.Temp = 0  
                BuscarProveedor.Ruc = ""  
                BuscarProveedor.Descripcion = ""  
                txtFechaCompra.Focus()  
            End If  
        End If  
    Catch ex As Exception  
        MsgBox("Error: " + ex.Message)  
    End Try  
End Sub  
Private Sub ComprasSinPedido_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles  
Me.KeyDown  
    Try  
        If e.KeyCode = Keys.F7 Then  
            If txtProveedorRucDni.Focused = 0 Then  
                Dim objConsultarProductoActualizar As New ConsultarProductoActualizar  
                objConsultarProductoActualizar.ShowDialog()  
                If ConsultarProductoActualizar.opc = "1" Then  
                    txtCodigo.Text = ConsultarProductoActualizar.IdProducto  
                    txtDescripcion.Text = ConsultarProductoActualizar.Producto  
                    txtPrecioCompro.Text = ConsultarProductoActualizar.Precio  
                    txtStock.Text = ConsultarProductoActualizar.Stock  
                    txtPrecioComprar.Text = ConsultarProductoActualizar.Precio  
  
                    BuscarProductos.opc = 0  
                    ConsultarProductoActualizar.IdProducto = ""  
                    ConsultarProductoActualizar.Producto = ""  
                    ConsultarProductoActualizar.Precio = ""  
                    ConsultarProductoActualizar.Stock = ""  
                    txtCompro.Focus()  
                End If  
            End If  
        End If  
        If e.KeyCode = Keys.Escape Then  
            Close()  
        End If  
    Catch ex As Exception  
        MsgBox("Error: " + ex.Message)  
    End Try  
End Sub  
#End Region  
End Class
```

Registrar Guía de Remisión

```
Imports SisAlmacenLN
Public Class RegistrarGuiaRemision
#Region "Propiedades"
    Dim Array As New ArrayList
    Dim Ruc, FechaActual, HoraActual, Descripcion As String
    Dim IdGuiaRemision, IdDetallePedido, r, varUdp, j As Integer, i As Integer = 0
    Dim udp As Boolean = False
    Dim Stock, Total As Decimal
    Dim dtProducto, dtDPedido, dtProveedor, dtConfiguracion As New DataTable
#End Region
#Region "Objetos"
    Dim objGuiaRemision As New GuiaRemision
    Dim objDetalleGuiaRemision As New DetalleGuiaRemision
    Dim objProducto As New Producto
    Dim objProveedor As New Proveedor
    Dim objPedido As New Pedido
    Dim objArea As New Area
    Dim objEstablecimiento As New Establecimiento
    Dim objKardex As New Kardex
#End Region
#Region "Procedimiento"
    Sub CargarCombo()
        cmbTienda.DataSource = objEstablecimiento.Listar
        cmbTienda.ValueMember = "IdEstablecimiento"
        cmbTienda.DisplayMember = "Descripcion"
    End Sub
    Sub LimpiarDataGridView()
        For r = 0 To dgvProductos.Rows.Count - 1
            dgvProductos.Rows.Remove(dgvProductos.CurrentRow)
        Next r
        i = 0
    End Sub
    Sub CargarDgvVentas()
        Try
            If txtCodigo.Text <> "" Then
                dgvProductos.Rows.Add(1)
                dgvProductos.Rows(i).Cells(0).Value = txtCodigo.Text
                dgvProductos.Rows(i).Cells(1).Value = txtCompro.Text
                dgvProductos.Rows(i).Cells(2).Value = txtDescripcion.Text
                dgvProductos.Rows(i).Cells(3).Value = txtPrecioCompro.Text
                dgvProductos.Rows(i).Cells(4).Value = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                dgvProductos.Rows(i).Cells(5).Value = txtStock.Text
                dgvProductos.Rows(i).Cells(6).Value = txtPrecioComprar.Text
                txtTotal.Text = Format(CDbl(Convert.ToDecimal(txtTotal.Text) + Convert.ToDecimal(dgvProductos.Rows(i).Cells(4).Value)),
                "#,###,##0.00")
                i = i + 1
            End If
        Catch ex As Exception
            MsgBox("Error: " + ex.Message)
        End Try
    End Sub
    Sub TextBoxActivado()
        txtCompro.Enabled = True
        txtPrecioCompro.Enabled = True
        txtImporte.Enabled = True
    End Sub
    Sub TextBoxDesActivado()
        txtCompro.Enabled = False
        txtPrecioCompro.Enabled = False
        txtImporte.Enabled = False
    End Sub
    Sub Limpiar()
        txtCodigo.Text = ""
        txtCompro.Text = ""
        txtDescripcion.Text = ""
        txtPrecioCompro.Text = ""
        txtImporte.Text = ""
        txtStock.Text = ""
        txtPrecioComprar.Text = ""
        udp = False
        txtCompro.Focus()
    End Sub
#End Region
#Region "Evento_Load"
    Private Sub CRegistrarGuiaRemision_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
```

```
dgvProductos.SelectionMode = DataGridViewSelectionMode.FullRowSelect
CargarCombo()
txtCompro.Text = ""
txtPrecioCompro.Text = ""
txtImporte.Text = ""
txtTotal.Text = "0.00"
End Sub
#End Region
#Region "Evento_Click"
Private Sub btnRegistrar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnRegistrar.Click
Try
Dim Resultado As MsgBoxResult = MessageBox.Show("Deseas Guardar Datos?", "Mensaje", MessageBoxButtons.YesNo,
MessageBoxIcon.Information)
If Resultado = MsgBoxResult.Yes Then
If txtSerie.Text <> "" Then
txtSerie.Text = Format(CDbl(txtSerie.Text), "000")
txtSerie.BackColor = Color.White
If txtNumero.Text <> "" Then
txtNumero.Text = Format(CDbl(txtNumero.Text), "000000")
txtNumero.BackColor = Color.White
If txtProveedorRucDni.Text <> "" Then
If txtProveedorRucDni.Text.Length = 8 Or txtProveedorRucDni.Text.Length = 11 Then
dtProveedor = objProveedor.ListarRucEstado(txtProveedorRucDni.Text, 1)
If dtProveedor.Rows.Count <> 0 Then
txtProveedorRucDni.BackColor = Color.White
txtProveedorDescripcion.Text = dtProveedor.Rows(0).Item("Descripcion").ToString()
dtProveedor.Clear()
If dgvProductos.RowCount <> 0 Then
Dim p As Integer
objGuiaRemision.Ingresar(cmbTienda.Selected.Value, txtFecha.Text, txtProveedorRucDni.Text, txtSerie.Text,
txtNumero.Text, txtPuntoPartida.Text, txtPuntoLlegada.Text)
IdGuiaRemision = objGuiaRemision.ObtenerId

For p = 0 To dgvProductos.Rows.Count - 1
If dgvProductos.Rows(p).Cells(5).Value >= dgvProductos.Rows(p).Cells(1).Value Then
objProducto.ActualizarStockNotaSalida(dgvProductos.Rows(p).Cells(0).Value,
dgvProductos.Rows(p).Cells(5).Value, dgvProductos.Rows(p).Cells(1).Value)
Else
If Convert.ToString(dgvProductos.Rows(p).Cells(5).Value) <> "0.000" Then
objProducto.ActualizarStockNotaSalida(dgvProductos.Rows(p).Cells(0).Value, 0, 0)
End If
End If
objDetalleGuiaRemision.Ingresar(IdGuiaRemision, dgvProductos.Rows(p).Cells(0).Value,
dgvProductos.Rows(p).Cells(1).Value, dgvProductos.Rows(p).Cells(3).Value)
Stock = objProducto.ObtenerStock(dgvProductos.Rows(p).Cells(0).Value)
objKardex.Ingresar(txtFecha.Text, dgvProductos.Rows(p).Cells(0).Value, "S", "GR", "", "", "", "",
dgvProductos.Rows(p).Cells(3).Value, 0, dgvProductos.Rows(p).Cells(1).Value, Stock, 1)
Next p

Limpiar()
LimpiarDataGridView()
cmbTienda.SelectedIndex = 0
txtFecha.Text = DateAndTime.Now.ToShortDateString
txtSerie.Text = ""
txtNumero.Text = ""
txtProveedorRucDni.Text = ""
txtProveedorDescripcion.Text = ""
txtPuntoPartida.Text = ""
txtPuntoLlegada.Text = ""
txtTotal.Text = "0.00"
cmbTienda.Focus()
MsgBox("Datos Registrado Correctamente", MsgBoxStyle.Information, "Mensaje")
Else
MsgBox("Ingrese al Menos un Item", MsgBoxStyle.Exclamation, "Mensaje")
End If
Else
dtProveedor.Clear()
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
Else
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
Else
End If
End Sub
```

```
        txtProveedorRucDni.BackColor = Color.Pink
        txtProveedorDescripcion.Text = ""
        txtProveedorRucDni.Focus()
    End If
Else
    txtNumero.BackColor = Color.Pink
    txtNumero.Focus()
End If
Else
    txtSerie.BackColor = Color.Pink
    txtSerie.Focus()
End If
End If
Catch ex As Exception
    MsgBox("Error: " + ex.Message)
End Try
End Sub
Private Sub btnActualizar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnActualizar.Click
    If dgvProductos.Rows.Count <> 0 Then
        txtCodigo.Text = dgvProductos.Item(0, dgvProductos.CurrentRow.Index).Value
        txtCompro.Text = dgvProductos.Item(1, dgvProductos.CurrentRow.Index).Value
        txtDescripcion.Text = dgvProductos.Item(2, dgvProductos.CurrentRow.Index).Value
        txtPrecioCompro.Text = dgvProductos.Item(3, dgvProductos.CurrentRow.Index).Value
        txtImporte.Text = dgvProductos.Item(4, dgvProductos.CurrentRow.Index).Value
        txtStock.Text = dgvProductos.Item(5, dgvProductos.CurrentRow.Index).Value
        varUdp = dgvProductos.CurrentRow.Index
        udp = True
        TextBoxActivado()
        txtCompro.Select(0, 10)
        txtCompro.Focus()
    End If
End Sub
Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnEliminar.Click
    Dim Resultado As MsgBoxResult = MessageBox.Show("Desea Eliminar Item?", "Mensaje", MessageBoxButtons.YesNo,
    MessageBoxIcon.Information)
    If Resultado = MsgBoxResult.Yes Then
        If dgvProductos.Rows.Count <> 0 Then
            txtTotal.Text = Format(CDbI(txtTotal.Text - dgvProductos.Item(6, dgvProductos.CurrentRow.Index).Value), "#,###,##0.00")
            i = i - 1
            dgvProductos.Rows.Remove(dgvProductos.CurrentRow)
        End If
    End If
    TextBoxActivado()
    Limpiar()
End Sub
#End Region
#Region "Evento_KeyPress"
Private Sub cmbTipoComprobante_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
cmbTienda.KeyPress
    If Asc(e.KeyChar) = 13 Then
        txtSerie.Focus()
    End If
End Sub
Private Sub txtSerie_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtSerie.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
        e.Handled = False
    Else
        e.Handled = True
    End If
    If Asc(e.KeyChar) = 13 Then
        If txtSerie.Text <> "" Then
            txtSerie.Text = Format(CDbI(txtSerie.Text), "000")
            txtSerie.BackColor = Color.White
            txtNumero.Focus()
        Else
            txtSerie.BackColor = Color.Pink
            txtSerie.Focus()
        End If
    End If
End Sub
Private Sub txtNumero_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtNumero.KeyPress
    If Char.IsDigit(e.KeyChar) Then
        e.Handled = False
    ElseIf Char.IsControl(e.KeyChar) Then
```

```
e.Handled = False
Else
e.Handled = True
End If
If Asc(e.KeyChar) = 13 Then
If txtNumero.Text <> "" Then
txtNumero.Text = Format(CDbl(txtNumero.Text), "000000")
txtNumero.BackColor = Color.White
txtProveedorRucDni.Focus()
Else
txtNumero.BackColor = Color.Pink
txtNumero.Focus()
End If
End If
End Sub
Private Sub txtProveedorRucDni_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtProveedorRucDni.KeyPress
Try
If Char.IsDigit(e.KeyChar) Then
e.Handled = False
ElseIf Char.IsControl(e.KeyChar) Then
e.Handled = False
Else
e.Handled = True
End If
If Asc(e.KeyChar) = 13 Then
If txtProveedorRucDni.Text <> "" Then
If txtProveedorRucDni.Text.Length = 8 Or txtProveedorRucDni.Text.Length = 11 Then
dtProveedor = objProveedor.ListarRucDni(txtProveedorRucDni.Text, "Transportista")
If dtProveedor.Rows.Count <> 0 Then
txtProveedorRucDni.BackColor = Color.White
txtProveedorDescripcion.Text = dtProveedor.Rows(0).Item("Descripcion").ToString()
txtPuntoPartida.Focus()
Else
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
dtProveedor.Clear()
Else
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
Else
txtProveedorRucDni.BackColor = Color.Pink
txtProveedorDescripcion.Text = ""
txtProveedorRucDni.Focus()
End If
End If
Catch ex As Exception
MsgBox("Error: " + ex.Message)
End Try
End Sub
Private Sub txtPuntoPartida_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtPuntoPartida.KeyPress
If Asc(e.KeyChar) = 13 Then
If txtPuntoPartida.Text <> "" Then
txtPuntoPartida.BackColor = Color.White
txtPuntoLlegada.Focus()
Else
txtPuntoPartida.BackColor = Color.Pink
txtPuntoPartida.Focus()
End If
End If
End Sub
Private Sub txtPuntoLlegada_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtPuntoLlegada.KeyPress
If Asc(e.KeyChar) = 13 Then
If txtPuntoLlegada.Text <> "" Then
txtPuntoLlegada.BackColor = Color.White
txtFecha.Focus()
Else
txtPuntoLlegada.BackColor = Color.Pink
txtPuntoLlegada.Focus()
End If
End If
End Sub
```

```
End Sub
Private Sub txtFechaCompra_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtFecha.KeyPress
    If Asc(e.KeyChar) = 13 Then
        txtCompro.Focus()
    End If
End Sub
Private Sub txtCompro_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtCompro.KeyPress
    If Asc(e.KeyChar) = 13 Then
        If txtCompro.Text <> "" And txtCompro.Text <> "0.000" Then
            txtCompro.BackColor = Color.White
            If txtPrecioCompro.Text <> "" And txtPrecioCompro.Text <> "0.00" Then
                txtPrecioCompro.BackColor = Color.White
                txtImporte.Text = Format(CDbI(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                txtPrecioCompro.Select(0, 10)
                txtPrecioCompro.Focus()
            Else
                txtPrecioCompro.BackColor = Color.Pink
                txtPrecioCompro.Text = "0"
                txtImporte.Text = "0"
                txtPrecioCompro.Select(0, 10)
                txtPrecioCompro.Focus()
            End If
        Else
            txtCompro.BackColor = Color.Pink
            txtCompro.Text = "0"
            txtImporte.Text = "0"
            txtCompro.Select(0, 10)
            txtCompro.Focus()
        End If
    End If
End Sub
Private Sub txtPrecioCompro_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtPrecioCompro.KeyPress
    Try
        If Asc(e.KeyChar) = 13 Then
            If txtCompro.Text <> "" And txtCompro.Text <> "0.000" Then
                txtCompro.BackColor = Color.White
                If txtPrecioCompro.Text <> "" And txtPrecioCompro.Text <> "0.00" Then
                    txtPrecioCompro.BackColor = Color.White
                    txtImporte.Text = Format(CDbI(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                    txtImporte.Select(0, 10)
                    txtImporte.Focus()
                Else
                    txtPrecioCompro.BackColor = Color.Pink
                    txtPrecioCompro.Text = "0"
                    txtImporte.Text = "0"
                    txtPrecioCompro.Select(0, 10)
                    txtPrecioCompro.Focus()
                End If
            Else
                txtCompro.BackColor = Color.Pink
                txtCompro.Text = "0"
                txtImporte.Text = "0"
                txtCompro.Select(0, 10)
                txtCompro.Focus()
            End If
        End If
    Catch ex As Exception
        MsgBox("Error: " + ex.Message)
    End Try
End Sub
Private Sub txtImporte_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles
txtImporte.KeyPress
    Try
        If Asc(e.KeyChar) = 13 Then
            If txtDescripcion.Text <> "" Then
                If txtCompro.Text <> "" And txtCompro.Text <> "0.000" Then
                    txtCompro.BackColor = Color.White
                    If txtPrecioCompro.Text <> "" And txtPrecioCompro.Text <> "0.00" Then
                        txtPrecioCompro.BackColor = Color.White
                        txtImporte.Text = Format(CDbI(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
                    End If
                    If udp Then
                        txtTotal.Text = Format(CDbI(Convert.ToDecimal(txtTotal.Text) -
Convert.ToDecimal(dgvProductos.Rows(varUdp).Cells(4).Value)), "#,###,##0.00")
                        dgvProductos.Rows(varUdp).Cells(0).Value = txtCodigo.Text
                    End If
                End If
            End If
        End If
    End Try
End Sub
```

```
dgvProductos.Rows(varUdp).Cells(1).Value = txtCompro.Text
dgvProductos.Rows(varUdp).Cells(2).Value = txtDescripcion.Text
dgvProductos.Rows(varUdp).Cells(3).Value = txtPrecioCompro.Text
dgvProductos.Rows(varUdp).Cells(4).Value = Format(CDbl(txtCompro.Text * txtPrecioCompro.Text), "#,###,##0.00")
dgvProductos.Rows(varUdp).Cells(5).Value = txtStock.Text
dgvProductos.Rows(varUdp).Cells(6).Value = txtPrecioCompr.Text
txtTotal.Text = Format(CDbl(Convert.ToDecimal(txtTotal.Text) +
Convert.ToDecimal(dgvProductos.Rows(varUdp).Cells(4).Value)), "#,###,##0.00")
Limpiar()
Else
    CargarDgvVentas()
    Limpiar()
    Dim objBuscarProductos As New BuscarProductos
    objBuscarProductos.ShowDialog()
    If BuscarProductos.opc = "1" Then
        txtCodigo.Text = BuscarProductos.IdProducto
        txtDescripcion.Text = BuscarProductos.Descripcion
        txtPrecioCompro.Text = BuscarProductos.Precio
        txtStock.Text = BuscarProductos.Stock
        txtPrecioCompr.Text = BuscarProductos.Precio

        BuscarProductos.opc = 0
        BuscarProductos.IdProducto = ""
        BuscarProductos.Descripcion = ""
        BuscarProductos.Precio = ""
        BuscarProductos.Stock = ""
        txtCompro.Focus()
    End If
End If
Else
    txtPrecioCompro.BackColor = Color.Pink
    txtPrecioCompro.Text = "0"
    txtImporte.Text = "0"
    txtPrecioCompro.Select(0, 10)
    txtPrecioCompro.Focus()
End If
Else
    txtCompro.BackColor = Color.Pink
    txtCompro.Text = "0"
    txtImporte.Text = "0"
    txtCompro.Select(0, 10)
    txtCompro.Focus()
End If
Else
    MsgBox("Seleccione un Producto", MsgBoxStyle.Exclamation, "Mensaje")
    txtCompro.Select(0, 10)
    txtCompro.Focus()
End If
End If
Catch ex As Exception
    MsgBox("Error: " + ex.Message)
End Try
End Sub
#End Region
#Region "Evento_KeyDown"
Private Sub txtProveedorRucDni_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtProveedorRucDni.KeyDown
    Try
        If e.KeyCode = Keys.F7 Then
            Dim objConsultarTransportistaActualizar As New ConsultarTransportistaActualizar
            objConsultarTransportistaActualizar.ShowDialog()
            If ConsultarTransportistaActualizar.opc = "1" Then
                txtProveedorRucDni.BackColor = Color.White
                ConsultarTransportistaActualizar.opc = ""
                txtProveedorRucDni.Text = ConsultarTransportistaActualizar.Ruc
                txtProveedorDescripcion.Text = ConsultarTransportistaActualizar.Descripcion

                ConsultarProveedorActualizar.Ruc = ""
                ConsultarProveedorActualizar.Descripcion = ""

                txtPuntoPartida.Focus()
            End If
        End If
    Catch ex As Exception
        MsgBox("Error: " + ex.Message)
    End Try
End Sub
```

```
Private Sub RegistrarGuiaRemision_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles Me.KeyDown
    Try
        If e.KeyCode = Keys.F7 Then
            If txtProveedorRucDni.Focused = 0 Then
                Dim objConsultarProductoActualizar As New ConsultarProductoActualizar
                objConsultarProductoActualizar.ShowDialog()
                If ConsultarProductoActualizar.opc = "1" Then
                    txtCodigo.Text = ConsultarProductoActualizar.IdProducto
                    txtDescripcion.Text = ConsultarProductoActualizar.Producto
                    txtPrecioCompro.Text = ConsultarProductoActualizar.Precio
                    txtStock.Text = ConsultarProductoActualizar.Stock
                    txtPrecioComprar.Text = ConsultarProductoActualizar.Precio

                    BuscarProductos.opc = 0
                    ConsultarProductoActualizar.IdProducto = ""
                    ConsultarProductoActualizar.Producto = ""
                    ConsultarProductoActualizar.Precio = ""
                    ConsultarProductoActualizar.Stock = ""
                    txtCompro.Focus()
                End If
            End If
        End If
        If e.KeyCode = Keys.Escape Then
            Close()
        End If
        Catch ex As Exception
            MsgBox("Error: " + ex.Message)
        End Try
    End Sub
#End Region

End Class
```



Reporte de Kardex de Almacén

```
Imports SisAlmacenLN
Public Class ReporteKardex
#Region "Propiedades"
    Dim dtsKardex As New DataSet
#End Region
#Region "Objetos"
    Dim objKardex As New Kardex
#End Region
#Region "Procedimiento"
#End Region
#Region "Evento_Load"
    Private Sub ReporteKardex_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        txtFecha1.Text = "01/06/2010"
    End Sub
#End Region
#Region "Evento_Click"
    Private Sub btnGenerar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGenerar.Click
        Try
            If txtDescripcion.Text <> "" Then
                If Convert.ToDateTime(txtFecha2.Text) >= Convert.ToDateTime(txtFecha1.Text) Then
                    dtsKardex = objKardex.dtsListar(txtCodigo.Text, txtFecha1.Text, txtFecha2.Text, 1)
                    Dim objrptReporteKardex As New rptReporteKardex
                    objrptReporteKardex.SetDataSource(dtsKardex.Tables(0))
                    CrystalReportViewer1.ReportSource = objrptReporteKardex
                    CrystalReportViewer1.Zoom(100)
                    CrystalReportViewer1.Show()
                    dtsKardex.Clear()
                Else
                    MsgBox("Fecha2 >= Fecha1", MsgBoxStyle.Exclamation, "Mensaje")
                End If
            Else
                MsgBox("Seleccione un Producto", MsgBoxStyle.Exclamation, "Mensaje")
            End If
        Catch ex As Exception
            MsgBox("Error: " + ex.Message)
        End Try
    End Sub
#End Region
#Region "Evento_KeyPress"

#End Region
#Region "Evento_KeyDown"
    Private Sub ReporteKardex_KeyDown(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles Me.KeyDown
        Try
            If e.KeyCode = Keys.F7 Then
                Dim objBuscarProductos As New BuscarProductos
                objBuscarProductos.ShowDialog()
                If BuscarProductos.opc = "1" Then
                    txtCodigo.Text = BuscarProductos.IdProducto
                    txtDescripcion.Text = BuscarProductos.Descripcion
                    lblStock.Text = BuscarProductos.Stock
                    BuscarProductos.opc = 0
                    BuscarProductos.IdProducto = ""
                    BuscarProductos.Descripcion = ""
                    BuscarProductos.Precio = ""
                    BuscarProductos.Stock = ""
                    btnGenerar.Focus()
                End If
            End If
            If e.KeyCode = Keys.Escape Then
                Close()
            End If
        Catch ex As Exception
            MsgBox("Error: " + ex.Message)
        End Try
    End Sub
#End Region
#Region "Evento_FormClosing"
#End Region
End Class
```