

IMPLEMENTACION DE UN NUEVO SISTEMA DE INFORMACIÓN PARA EL  
ALMACÉN DE LA DIRECCIÓN DE IMPUESTOS Y ADUANAS SECCIONAL  
PEREIRA

DIEGO ALEJANDRO QUINTERO HOYOS

UNIVERSIDAD CATÓLICA POPULAR DEL RISARALDA  
PROGRAMA DE INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES  
PRÁCTICAS PROFESIONALES  
PEREIRA  
2009

IMPLEMENTACION DE UN NUEVO SISTEMA DE INFORMACIÓN PARA EL  
ALMACÉN DE LA DIRECCIÓN DE IMPUESTOS Y ADUANAS SECCIONAL  
PEREIRA

DIEGO ALEJANDRO QUINTERO HOYOS

INFORME DE PRÁCTICA PROFESIONAL

Tutor

RICARDO ALONSO HURTADO MOSQUERA

INGENIERO DE SISTEMAS

UNIVERSIDAD CATÓLICA POPULAR DEL RISARALDA  
PROGRAMA DE INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES  
PRÁCTICAS PROFESIONALES  
PEREIRA  
2009

## CONTENIDO

	Pág.
INTRODUCCION .....	16
1. PRESENTACION DE LA ORGANIZACIÓN.....	17
1.1 CÓMO NACIÓ LA DIAN.....	17
1.2 CUÁL ES SU NATURALEZA JURÍDICA .....	17
1.3 DÓNDE TIENE SU JURISDICCIÓN .....	17
1.4 PARA QUÉ EXISTE LA DIAN.....	18
1.5 QUIÉN ES EL REPRESENTANTE LEGAL DE LA DIAN.....	18
1.6 QUÉ FUNCIONES LE COMPETEN A LA DIAN .....	18
1.7 QUÉ PROCESOS SE DESARROLLAN EN LA DIAN.....	19
1.8 PLAN ESTRATEGICO .....	20
1.8.1 Misión:.....	20
1.8.2 Visión: .....	20
1.8.3 Política: .....	21
1.8.4 Mapa Estratégico: .....	21
1.8.5 Estrategia:.....	23
1.9 ORGANIGRAMA:.....	24
1.9.1 Organigrama Nivel Central:.....	25

1.9.2 Organigrama Direcciones Seccionales Locales.....	26
2. DEFINICIÓN LÍNEA DE INTERVENCIÓN.....	27
3. DIAGNÓSTICO DEL ÁREA DE INTERVENCIÓN .....	28
3.1 IDENTIFICACIÓN DEL PROBLEMA: .....	28
4. EJE DE INTERVENCIÓN .....	30
5. JUSTIFICACIÓN EJE DE INTERVENCIÓN .....	31
6. OBJETIVOS.....	32
6.1 OBJETIVO GENERAL .....	32
6.2 OBJETIVOS ESPECÍFICOS:.....	32
7. REFERENTE CONCEPTUAL.....	33
8. CRONOGRAMA .....	44
9. PRESENTACIÓN Y ANÁLISIS DE RESULTADOS .....	48
10. RECOMENDACIONES .....	53
CONCLUSIONES .....	54
BIBLIOGRAFÍA.....	55
ANEXOS .....	56

## LISTA DE TABLAS

	<b>Pág.</b>
TABLA 1. Cronograma Práctica Profesional	47.

## LISTA DE FIGURAS

	Pág.
FIGURA 1. Organigrama División Central	25
FIGURA 2. Organigrama Divisiones Seccionales	26
FIGURA 3. Primera Versión Modelo Entidad-Relación	39
FIGURA 4. Segunda Versión Modelo Entidad-Relación	40
FIGURA 5. Tercera Versión Modelo Entidad-Relación	42

## ANEXOS

	Pág.
ANEXO A. Descripción de Entidades y Relaciones	56

## GLOSARIO

**ATRIBUTO:** Es una característica de interés o un hecho sobre una entidad o sobre una relación. Los atributos representan las propiedades básicas de las entidades y de las relaciones. Toda la información extensiva es portada por los atributos. Gráficamente, se representan mediante bolitas que cuelgan de las entidades o relaciones a las que pertenecen.

Cada atributo tiene un conjunto de valores asociados denominado *dominio*. El dominio define todos los valores posibles que puede tomar un atributo. Puede haber varios atributos definidos sobre un mismo dominio. Los atributos pueden ser simples o compuestos. Un *atributo simple* es un atributo que tiene un solo componente, que no se puede dividir en partes más pequeñas que tengan un significado propio. Un *atributo compuesto* es un atributo con varios componentes, cada uno con un significado por sí mismo. Un grupo de atributos se representa mediante un atributo compuesto cuando tienen afinidad en cuanto a su significado, o en cuanto a su uso. Un atributo compuesto se representa gráficamente mediante un óvalo.

Los atributos también pueden clasificarse en monovalentes o polivalentes. Un *atributo monovalente* es aquel que tiene un solo valor para cada ocurrencia de la entidad o relación a la que pertenece. Un *atributo polivalente* es aquel que tiene varios valores para cada ocurrencia de la entidad o relación a la que pertenece. A estos atributos también se les denomina *multivaluados*, y pueden tener un número máximo y un número mínimo de valores. La *cardinalidad* de un atributo indica el número mínimo y el número máximo de valores que puede tomar para cada ocurrencia de la entidad o relación a la que pertenece. El valor por omisión es uno a uno.

Por último, los atributos pueden ser derivados. Un *atributo derivado* es aquel que representa un valor que se puede obtener a partir del valor de uno o varios atributos, que no necesariamente deben pertenecer a la misma entidad o relación.

**BASE DE DATOS:** Una base de datos es un conjunto de datos, que se refieren al mismo tipo de persona u objeto, y se muestran al usuario estructurados en filas y columnas. A cada estructura de filas y columnas se le denomina tabla. Los sistemas de gestión de bases de datos que se pueden adquirir no sólo cumplen esta definición sino que además presentan muy diversas utilidades, entre las que

destacan las de creación de tablas, vistas, formularios e informes mediante interfaces gráficas. También contienen herramientas y lenguajes para la definición, descripción y manipulación de datos, para la creación de programas, funciones y procedimientos, así como elementos de gestión de usuarios: creación, definición y modificación de usuarios y permisos de acceso a datos o programas, que garanticen la seguridad de la base de datos.

**BASE DE DATOS RELACIONAL** Tipo de base de datos o sistema de administración de bases de datos, que almacena la información en varias tablas (filas y columnas de datos) o ficheros independientes y realiza búsquedas que permiten relacionar datos que han sido almacenados en más de una tabla. El término fue acuñado en 1970 por el investigador británico Edgar F. Codd.

En las tablas de una base de datos relacional, las filas representan registros (conjuntos de datos acerca de individuos o elementos separados) y las columnas representan campos (atributos particulares de un registro). Las tablas o ficheros de la base de datos relacional deben tener un campo común, es decir, un campo que almacena, en cada una de ellas, la misma información para cada registro y que va a ser el que permita establecer la relación al realizar las consultas. Por ejemplo, si una tabla contiene los campos NÚM-EMPLEADO, APELLIDO, NOMBRE y ANTIGÜEDAD, otra tabla puede contener los campos DEPARTAMENTO, NÚM-EMPLEADO y SALARIO; la base de datos relacional utilizará el campo NÚM-EMPLEADO de las dos tablas para encontrar información relativa a un empleado, y que estaba almacenada en alguna de las dos tablas, como por ejemplo los nombres de los empleados que ganan un cierto salario o los departamentos de todos los empleados contratados a partir de un día determinado. El resultado de la búsqueda produce una tercera tabla que combina los datos solicitados de ambas tablas. En otras palabras, una base de datos relacional utiliza los valores coincidentes de campos comunes de dos tablas para relacionar información de ambas.

**CARDINALIDAD:** Forma en que cada entidad se relaciona con las demás entidades.

**CLASE:** un tipo de datos definido por el usuario que especifica un conjunto de objetos que comparten las mismas características

**DATO:** Un dato es una representación simbólica (numérica, alfabética), atributo o característica de una entidad. El dato no tiene valor semántico en sí mismo, pero convenientemente tratado se puede utilizar en la realización de cálculos o toma de decisiones.

**DRIVER:** Un driver o controlador posibilita que el sistema operativo de una computadora pueda entenderse con un equipamiento periférico, como es el caso de una impresora, una placa de video, un mouse, un módem, etc. Para poder interactuar con el periférico, el sistema operativo debe realizar una abstracción del hardware brindando una forma de manipularlo mediante una interface, esto es, algún mecanismo que permita controlar su funcionamiento, un conjunto de instrucciones que indican de qué manera debería comunicarse con tal o cual dispositivo.

**ENTIDAD:** Cualquier tipo de objeto o concepto sobre el que se recoge información: cosa, persona, concepto abstracto o suceso. Por ejemplo: coches, casas, empleados, clientes, empresas, oficios, diseños de productos, conciertos, excursiones, etc. Las entidades se representan gráficamente mediante rectángulos y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Hay dos tipos de entidades: fuertes y débiles. Una *entidad débil* es una entidad cuya existencia depende de la existencia de otra entidad. Una *entidad fuerte* es una entidad que no es débil.

**ERROR:** es una acción humana que resulta en que un software contenga una falta, por lo cual un *error* puede significar la inclusión de una *falta* en el programa, haciendo que el sistema *falle*.

**FALLA:** ocurre cuando un programa no se comporta bien, siendo la falla una propiedad (estadística) de un sistema en ejecución.

**FALTA:** existe en el código del programa, el cual si se presenta, puede ocasionar una *falla* (*failure*). No puede haber una *falta* si el programa no puede *fallar* (*fail*).

**IDENTIFICADOR:** Un identificador de una entidad es un atributo o conjunto de atributos que determina de modo único cada ocurrencia de esa entidad. Un identificador de una entidad debe cumplir dos condiciones:

1. No pueden existir dos ocurrencias de la entidad con el mismo valor del identificador.
2. Si se omite cualquier atributo del identificador, la condición anterior deja de cumplirse.

Toda entidad tiene al menos un identificador y puede tener varios identificadores alternativos. Las relaciones no tienen identificadores.

**INGENIERÍA DEL SOFTWARE:** Es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software; es decir, la aplicación de la ingeniería al software.

**INTEGRIDAD REFERENCIAL:** La integridad referencial es un sistema de reglas que utilizan la mayoría de las bases de datos relacionales para asegurarse que los registros de tablas relacionadas son válidos y que no se borren o cambien datos relacionados de forma accidental produciendo errores de integridad.

**INTERFAZ GRÁFICA DE USUARIO:** Tipo de visualización que permite al usuario elegir comandos, iniciar programas y ver listas de archivos y otras opciones utilizando las representaciones visuales (iconos) y las listas de elementos del menú. Las selecciones pueden activarse bien a través del teclado o con el ratón. (Véase Interfaz de usuario). Para los autores de aplicaciones, las interfaces gráficas de usuario ofrecen un entorno que se encarga de la comunicación con el ordenador o computadora. Esto hace que el programador pueda concentrarse en la funcionalidad, ya que no está sujeto a los detalles de la visualización ni a la entrada a través del ratón o del teclado. También permite a los programadores crear programas que realicen de la misma forma las tareas más frecuentes, como guardar un archivo, porque la interfaz proporciona mecanismos estándar de control como ventanas y cuadros de diálogo. Otra ventaja es que las aplicaciones escritas para una interfaz gráfica de usuario son independientes de los dispositivos: a medida que la interfaz cambia para permitir el uso de nuevos dispositivos de entrada y salida, como un monitor de pantalla grande o un dispositivo óptico de almacenamiento, las aplicaciones pueden utilizarlos sin necesidad de cambios.

**JAVA:** Java es un lenguaje de programación con el que podemos realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Está desarrollado por la compañía Sun Microsystems con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado. Es una ventaja significativa para los desarrolladores de software, pues

antes tenían que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux, Apple, etc. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

**LLAVE FORÁNEA:** Una llave Foránea es un campo (o campos) que señala la clave primaria de otra tabla. El propósito de la clave externa es asegurar la integridad referencial de los datos. En otras palabras, sólo se permiten los valores que se esperan que aparezcan en la base de datos.<sup>1</sup>

**LLAVE PRIMARIA:** La clave primaria se utiliza para identificar en forma única cada línea en la tabla. Puede ser parte de un registro real, o puede ser un campo artificial (uno que no tiene nada que ver con el registro real). Una clave primaria puede consistir en uno o más campos en una tabla. Cuando se utilizan múltiples campos como clave primaria, se los denomina claves compuestas.<sup>2</sup>

### **MODELO ENTIDAD-RELACIÓN:**

Es un concepto de modelado para bases de datos, propuesto por Peter Chen en 1976, mediante el cual se pretende 'visualizar' los objetos que pertenecen a la Base de Datos como **entidades** (se corresponde al concepto de objeto de la Programación Orientada a Objetos) las cuales tienen unos atributos y se vinculan mediante **relaciones**.

Es una representación conceptual de la información. Mediante una serie de procedimientos se puede pasar del modelo E-R a otros, como por ejemplo el modelo relacional. El modelado entidad-relación es una *técnica* para el modelado de datos utilizando diagramas entidad relación. No es la única técnica pero sí la más utilizada.

**OBJETO:** conjunto de datos y funciones relacionadas que son instancias de clases determinadas.

**REGISTRO:** Un registro es una pequeña parte o unidad donde se almacenan cierto tipo de datos los cuales han sido destinados aguardarse en ese lugar.

**RELACIÓN (INTERRELACIÓN):** Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su función. Las

---

<sup>1</sup> <http://sql.1keydata.com/es/sql-clave-externa.php>

<sup>2</sup> <http://sql.1keydata.com/es/sql-clave-primaria.php>

relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.

Las entidades que están involucradas en una determinada relación se denominan *entidades participantes*. El número de participantes en una relación es lo que se denomina *grado* de la relación. Por lo tanto, una relación en la que participan dos entidades es una relación *binaria*; si son tres las entidades participantes, la relación es *ternaria*; etc.

Una *relación recursiva* es una relación donde la misma entidad participa más de una vez en la relación con distintos papeles. El nombre de estos papeles es importante para determinar la función de cada participación. La *cardinalidad* con la que una entidad participa en una relación especifica el número mínimo y el número máximo de correspondencias en las que puede tomar parte cada ocurrencia de dicha entidad. La participación de una entidad en una relación es *obligatoria (total)* si la existencia de cada una de sus ocurrencias requiere la existencia de, al menos, una ocurrencia de la otra entidad participante. Si no, la participación es *opcional (parcial)*. Las reglas que definen la cardinalidad de las relaciones son las *reglas de negocio*.

**SISTEMAS DE INFORMACIÓN:** Los sistemas de información son subsistemas de la empresa que facilitan la transferencia de información entre los subsistemas real, financiero y directivo, y entre los subsistemas de éstos, entendiendo como subsistema cada departamento empresarial. Un sistema de información está formado por tres tipos de elementos: recursos físicos, documentos y sus archivadores, equipos de telecomunicaciones y de informática (hardware y software); recursos humanos (usuarios y personal informático) y protocolos o normas que rigen el uso y transmisión de los flujos de información.

**TABLA:** Las tablas en las bases de datos es el medio de almacenamiento de datos recolectados por un programa. Cada tabla debe tener un nombre para así tener un fácil acceso La tabla está compuesta por campo y registro:

Campo: es el nombre que se le da a la columna este nombre debe ser único y además debe especificar el tipo de dato.

Registro: corresponde a cada fila que compone la tabla allí, es donde se almacenan los datos

**VALIDACIÓN:** Consiste en comprobar que tanto el algoritmo como el programa cumplen la especificación del problema. También es el proceso de comprobar la

precisión de los datos; conjunto de reglas que se pueden aplicar a un control para especificar el tipo y el intervalo de datos que los usuarios pueden especificar.

## RESUMEN

### Resumen

Para la Dirección de Impuestos y Aduanas Nacionales (DIAN), contar con Sistemas de Información que le permitan administrar y controlar todos los datos relacionados con las actividades de recaudo de impuestos y aduanas es vital para cumplir con su razón de ser. Es por ello que se plantea la necesidad de crear nuevos Sistemas que estén en condiciones de suplir dichos requerimientos.

Este informe de Práctica tiene como finalidad mostrar los aspectos más importantes (técnicos, teóricos y prácticos) para la implementación de un Sistema de Información para el Almacén de la DIAN, Seccional Pereira, el cual permita controlar y gestionar todos los procesos concernientes a la Entrada y Salida de Productos (Consumo y Devolutivos), desde empresas o seccionales de la DIAN en todo el país, hacia las distintas dependencias de la entidad en esta ciudad, al igual que proporcionar los elementos necesarios para el seguimiento paso a paso del proyecto.

**Descriptor:** Sistemas de Información, Bases de Datos, Modelo Entidad-Relación, Interfaz de Usuario, Migración, Ingeniería del Software, Reportes.

### Abstract

For Dirección de Impuestos y Aduanas Nacionales (DIAN), have information systems that will enable it to manage and monitor all data relating to the activities of tax collectors and customs is vital to meet its purpose. That is why there is a need to create new systems that are able to fill those requests.

This practice report has the finality aims to show the most important aspects (technical, theoretical, practical) for the implement of a Information System for Almacen of the DIAN, Sectional Pereira, which allows controlling and managing all processes relating to the entry and exit of products (Consumption and devolutive), from companies or sectionals of DIAN in all country, to different units from the institution in this city, like that provide the elements needed to monitor the project step by step.

**Descriptors:** Information System, Database, Entity Relationship Model, GUI, Migration, Software Engineering, Reports

## INTRODUCCION

El manejo de información es un proceso crítico para cualquier empresa: el ingreso, almacenamiento y procesamiento de sus datos en un Sistema de Información es un proceso prioritario para las empresas. Una compañía que no pueda acceder a sus datos de una manera fácil, rápida y confiable, ya sea para los distintos procesos que se manejan internamente o para suplir las necesidades de sus clientes, no es competitiva y está destinada al fracaso en el actual modelo económico.

Para ello, las empresas hoy en día cuentan con distintas herramientas que le permiten tener un control total sobre la información que manejan, con el fin de estar siempre actualizadas y garantizar la integridad en los datos que se manejan constantemente. Estas herramientas son llamadas Sistemas de Información: son la unión de distintos elementos (software y hardware) que permiten la automatización de distintos procesos (ingreso, salida) los cuales permiten a las compañías realizar diversas actividades sobre sus datos (consultas, ingreso y salida, borrado, estadísticas).

La Dirección de Impuestos y Aduanas Nacionales (DIAN), en su política de calidad y teniendo en cuenta los diferentes procesos para la administración, control y supervisión de las distintas actividades tributarias y aduaneras, debe contar con Sistemas de Información que le permitan conocer de una manera detallada, rápida y concreta todos los movimientos que se realicen con los registros ingresados o almacenados, con el fin de garantizar una correcta recaudación de las obligaciones fiscales (impuestos) y una buena utilización y distribución de los recursos asignados.

Por ello, la DIAN Seccional Pereira, con el fin de optimizar sus recursos y tener un mejor control sobre ellos, requiere el diseño e implementación de un Sistema de Información para la administración y control de todos los productos de consumo y devolutivos que ingresen al almacén de la institución, y los cuales son despachados a las distintas divisiones según sus necesidades. Sistema que debe suplir a cabalidad las exigencias de los usuarios y de la institución en cuanto a la presentación, procesamiento y manipulación de la información.

## **1. PRESENTACION DE LA ORGANIZACIÓN.**

### **1.1 CÓMO NACIÓ LA DIAN**

La Dirección de Impuestos y Aduanas Nacionales (DIAN) se constituyó como Unidad Administrativa Especial, mediante Decreto 2117 de 1992, cuando el 1º de junio del año 1993 se fusionó la Dirección de Impuestos Nacionales (DIN) con la Dirección de Aduanas Nacionales (DAN).

Mediante el Decreto 1071 de 1999 se da una nueva reestructuración y se organiza la Unidad Administrativa Especial Dirección de Impuestos y Aduanas Nacionales (DIAN).

De igual manera, el 22 de octubre de 2008, por medio del Decreto 4048 se modifica la estructura de la Unidad Administrativa Dirección de Impuestos y Aduanas Nacionales.

### **1.2 CUÁL ES SU NATURALEZA JURÍDICA**

La DIAN está organizada como una Unidad Administrativa Especial del orden nacional, de carácter eminentemente técnico y especializado, con personería jurídica, autonomía administrativa y presupuestal y con patrimonio propio, adscrita al Ministerio de Hacienda y Crédito Público.

### **1.3 DÓNDE TIENE SU JURISDICCIÓN**

La jurisdicción de la Unidad Administrativa Especial Dirección de Impuestos y Aduanas Nacionales comprende el territorio nacional, y su domicilio principal es la ciudad de Bogotá, D.C.

La DIAN hace presencia en **42** ciudades de Colombia: Arauca, Armenia, Barrancabermeja, Barranquilla, Bogotá, Bucaramanga, Buenaventura, Cali, Cartagena, Cartago, Cúcuta, Florencia, Girardot, Ibagué, Inírida, Ipiales, Leticia, Maicao, Manizales, Medellín, Mitú, Montería, Neiva, Palmira, Pasto, Pereira, Popayán, Puerto Asís, Puerto Carreño, Riohacha, Santa Marta, San Andrés,

Sincelejo, Sogamoso, Tumaco, Tunja, Turbo, Tuluá, Quibdó, Valledupar, Villavicencio, Yopal.

#### **1.4 PARA QUÉ EXISTE LA DIAN**

Para coadyuvar a garantizar la seguridad fiscal del Estado colombiano y la protección del orden público económico nacional, mediante la administración y control al debido cumplimiento de las obligaciones tributarias, aduaneras y cambiarias, y la facilitación de las operaciones de comercio exterior en condiciones de equidad, transparencia y legalidad.

#### **1.5 QUIÉN ES EL REPRESENTANTE LEGAL DE LA DIAN**

La representación legal de la DIAN está a cargo del Director General, quien puede delegarla de conformidad con las normas legales vigentes. El cargo de Director General es de libre nombramiento y remoción; en consecuencia se provee mediante nombramiento ordinario por el Presidente de la República.

#### **1.6 QUÉ FUNCIONES LE COMPETEN A LA DIAN**

La administración de los impuestos de renta y complementarios, de timbre nacional y sobre las ventas; los derechos de aduana y los demás impuestos internos del orden nacional cuya competencia no esté asignada a otras entidades del Estado, bien se trate de impuestos internos o al comercio exterior; así como la dirección y administración de la gestión aduanera, incluyendo la aprehensión, decomiso o declaración en abandono a favor de la Nación de mercancías y su administración y disposición.

El control y vigilancia sobre el cumplimiento del régimen cambiario en materia de importación y exportación de bienes y servicios, gastos asociados a las mismas, financiación en moneda extranjera de importaciones y exportaciones, y subfacturación y sobrefacturación de estas operaciones.

La administración de los impuestos comprende su recaudación, fiscalización, liquidación, discusión, cobro, devolución, sanción y todos los demás aspectos relacionados con el cumplimiento de las obligaciones tributarias.

La administración de los derechos de aduana y demás impuestos al comercio exterior, comprende su recaudación, fiscalización, liquidación, discusión, cobro, sanción y todos los demás aspectos relacionados con el cumplimiento de las obligaciones aduaneras.

La dirección y administración de la gestión aduanera comprende el servicio y apoyo a las operaciones de comercio exterior, la aprehensión, decomiso o declaración en abandono de mercancías a favor de la Nación, su administración, control y disposición, así como la administración y control de los Sistemas Especiales de Importación-Exportación, Zonas Francas, Zonas Económicas Especiales de Exportación y las Sociedades de Comercialización Internacional, de conformidad con la política que formule el Ministerio de Comercio, Industria y Turismo en la materia, para estos últimos, con excepción de los contratos relacionados con las Zonas Francas.

Le compete actuar como autoridad doctrinaria y estadística en materia tributaria, aduanera, de control de cambios en relación con los asuntos de su competencia, así como los atinentes a los Sistemas Especiales de Importación-Exportación, Zonas Francas, Zonas Económicas Especiales de Exportación y las Sociedades de Comercialización Internacional.

## **1.7 QUÉ PROCESOS SE DESARROLLAN EN LA DIAN**

En la DIAN existen procesos: estratégicos, misionales, de apoyo y de evaluación.

Los **procesos estratégicos** tienen como finalidad orientar a la entidad para que cumpla con su misión, visión, política y objetivos y satisfacer las necesidades de las partes interesadas (organización, persona o grupo) que tengan un interés de la entidad.

Los **procesos misionales** tienen que ver con la razón de ser y las responsabilidades de la DIAN como institución del Estado que se refleja en su misión, que comprende coadyuvar a garantizar la seguridad fiscal del Estado colombiano y la protección del orden público económico nacional, mediante la administración y control al debido cumplimiento de las obligaciones tributarias, aduaneras y cambiarias, y la facilitación de las operaciones de comercio exterior en condiciones de equidad, transparencia y legalidad.

Los **procesos de apoyo** proporcionan el soporte a los procesos estratégicos, misionales y de medición, análisis y mejora.

Los **procesos de evaluación** permiten garantizar un ejercicio de medición, retroalimentación y ajuste, de tal forma que la entidad alcance los resultados propuestos. Incluyen procesos de medición, seguimiento y auditoría interna, acciones correctivas y preventivas, y son una parte integral de los procesos estratégicos, de apoyo y los misionales.

## **1.8 PLAN ESTRATEGICO**

### **1.8.1 Misión:**

En la Dirección de Impuestos y Aduanas Nacionales de Colombia somos responsables de prestar un servicio de facilitación y control a los agentes económicos, para el cumplimiento de las normas que integran el Sistema Tributario, Aduanero y Cambiario, obedeciendo los principios constitucionales de la función administrativa, con el fin de recaudar la cantidad correcta de tributos, agilizar las operaciones de comercio exterior, propiciar condiciones de competencia leal, proveer información confiable y oportuna , y contribuir al bienestar social y económico de los colombianos.

### **1.8.2 Visión:**

En el 2010 la Dirección de Impuestos y Aduanas Nacionales de Colombia ha consolidado: la autoridad fiscal del Estado, la autonomía institucional, altos niveles de cumplimiento voluntario de las obligaciones que controla y facilita, la integralidad y aprovechamiento pleno de la información, de los procesos y de la normatividad, el aporte de elementos técnicos para la adecuación de la carga fiscal y la simplificación del sistema tributario, aduanero y cambiario. Contando con un equipo de funcionarios públicos comprometido y formado para salvaguardar los intereses del Estado con un sentido de servicio, un soporte tecnológico integral y permanentemente actualizado, y un régimen administrativo especial propio de una entidad moderna que responde a las necesidades de la sociedad, funcionarios, contribuyentes y usuarios.

### **1.8.3 Política:**

La DIAN, en el marco de su misión y visión, y consciente de la importancia que su gestión representa para el bienestar del país, asume su compromiso considerando las siguientes directrices:

- Garantizar la eficacia, eficiencia y efectividad de los servicios que brinda a sus clientes y del Sistema de Gestión de Calidad y Control Interno.
- Mejorar continuamente su desempeño organizacional, con un enfoque basado en procesos y sus respectivos mecanismos de control y evaluación.
- Brindar las condiciones laborales de salud, seguridad y desarrollo de competencias, necesarias para garantizar el crecimiento personal y profesional de los empleados.
- Soportar su gestión y la toma de decisiones en información confiable y oportuna, generada en sistemas desarrollados con tecnología avanzada.
- Asegurar el adecuado desarrollo de sus procesos de comunicación.
- Implementar los elementos y acciones necesarias para aumentar la capacidad institucional para la administración de los riesgos estratégicos y del nivel ejecutor.

### **1.8.4 Mapa Estratégico:**

El mapa estratégico identifica los temas estratégicos y determina los indicadores, metas, recursos y acciones a desarrollar, que la entidad abordará como pilares fundamentales para lograr el desarrollo de la política de gestión.

- **Alcanzar la excelencia en sus operaciones**

Con unas reglas de juego dadas, unos recursos del Estado escasos y el objetivo de maximizar la recaudación de acuerdo con esas reglas de juego, el reto para la DIAN es uno de los mayores en la gestión del Estado. Este reto debe asumirse de manera inteligente, procesando la mayor cantidad de información posible, que le permita a la entidad identificar a todos los obligados, y su estado frente al fisco.

Para ello, los contribuyentes deben encontrar transacciones fáciles y transparentes que les permitan cumplir con sus obligaciones, pero a la vez la DIAN debe fortalecer el control tributario y aduanero para garantizar el

cumplimiento de dichas obligaciones. En desarrollo de esta estrategia, la DIAN debe hacer sentir que sus clientes cuentan con el apoyo y la asesoría de ésta cuando se encuentran cumpliendo al día, así como la reacción inmediata con la fortaleza necesaria en los momentos en que no lo hacen.

- **Redireccionar la entidad hacia el servicio**

La Administración tributaria debe estar a la altura del ciudadano moderno, que exige más y mejores productos y servicios “a menor precio”. En el caso de la DIAN, esto se traduce en velar por el estricto cumplimiento de las normas vigentes, garantizando que cada contribuyente pague exactamente lo que le corresponde, y que paguen sus impuestos todos los contribuyentes que están en la obligación de hacerlo.

Para ello la estrategia debe ser integral, proveyendo servicios que faciliten la tarea para los que cumplen y contando con las herramientas adecuadas para reducir de manera significativa la evasión, el contrabando y las infracciones cambiarias. Ello implica reorientar la organización, sus procesos y su gente hacia el servicio, optimizar las operaciones de control aduanero, administrar la red de recaudación, así como optimizar, agilizar y flexibilizar los procesos logísticos que soportan la gestión de la entidad.

- **Consolidar la legitimidad y la autonomía**

Más del 80% de los ingresos recaudados por la DIAN vienen del cumplimiento voluntario (o inducido) de las obligaciones por parte de los contribuyentes. Igualmente, el pago de los impuestos es uno de los principales momentos en los que el ciudadano tiene contacto con el Estado. La manera como resulta esta transacción es fundamental para determinar la actitud presente y futura del ciudadano frente a su responsabilidad contributiva y su sentimiento frente al Estado.

Si bien la DIAN no es responsable por la gestión del Estado en general, si puede ser un vehículo para comunicar sus resultados, especialmente a aquellos ciudadanos que han contribuido con el pago de sus impuestos. Debe ser un ejemplo de buena gestión y transparencia ante la ciudadanía, no sólo en el manejo de sus recursos y de su administración en general, sino en los servicios que presta, las transacciones que se hacen con ella y la aplicación de la normatividad de manera justa y objetiva. Ello implica luchar frontalmente contra la corrupción, y crear los mecanismos efectivos que permitan blindarla frente a ella.

Se trata de fortalecer la imagen institucional para que la ciudadanía y el Estado confíen cada vez más en ella y para posibilitar el acceso a más recursos e forma tal que sea posible sostener y potenciar la inversión en ella.

- **Contribuir a la competitividad del país**

Por su naturaleza, la DIAN es una entidad con alto impacto en la competitividad nacional. La manera cómo ésta garantiza el cumplimiento y la consistencia en las reglas de juego asociadas con el sistema tributario, aduanero y cambiario es fundamental para las expectativas de inversión (nacional o extranjera), la competitividad en las operaciones de comercio exterior y la reducción del tiempo y costo asociados al cumplimiento de las obligaciones.

Por esta razón, el adecuado balance entre la facilitación y el control y la seguridad jurídica y unidad de criterio son acciones prioritarias para la DIAN, que combinadas con la búsqueda de una simplificación de la legislación y una planeación proactiva de los cambios legislativos, deben coadyuvar al desarrollo competitivo de nuestro país. La existencia y seguimiento de este principio estratégico garantiza al modelo un desarrollo equilibrado que favorece la organización en todos sus aspectos.

#### **1.8.5 Estrategia:**

El mapa estratégico se puede sintetizar en una lógica de causalidad, en la que se establece que la entidad es responsable por el mejoramiento del recaudo (finanzas), a través del manejo eficiente y eficaz de la administración tributaria, cuyos pilares son el resultado de los logros obtenidos en cada una de las perspectivas, este recaudo es el insumo esencial para financiar los gastos asociados al plan nacional de desarrollo.

Para lograr el mejoramiento del recaudo es necesario incrementar los niveles de cumplimiento de obligaciones, es decir, asegurar el cumplimiento de las obligaciones formales de informar, declarar y pagar. (Clientes)

La promoción y el fomento del cumplimiento de las obligaciones se sustenta en el equilibrio entre las actividades de educación, orientación, asistencia y servicio al contribuyente y usuario que se encuentran en la dimensión estratégica Acercamiento y Satisfacción, con aquellas referidas a los programas de control formal de presencia masiva y de generación de riesgo, incluidas en la dimensión estratégica de control. (Procesos)

La entidad logra el acercamiento y satisfacción de los clientes al igual que el control, al optimizar la Inteligencia Corporativa, es decir, la comprensión y análisis de los comportamientos de contribuyentes y usuarios, mediante el desarrollo de la gestión del conocimiento institucional y la innovación estratégica, de tal forma que permita mejorar el control y el servicio que brinda la DIAN. (Aprendizaje e innovación)

Para optimizar la Inteligencia Corporativa se deben desarrollar servicios automatizados que aseguren fluidez, rapidez, seguridad y confiabilidad de los procesos operativos y potencializar la gestión humana, tanto en el aspecto humano como en lo físico y ocupacional, de tal forma que se disponga de la información necesaria para que un talento humano competente, motivado, orientado al servicio y con excelente formación, se encamine hacia el logro de los objetivos. (Aprendizaje e innovación)

### **1.9 ORGANIGRAMA:**

En la DIAN, se manejan 2 tipos de organigramas (Nivel Central, Nivel Local) dentro de los cuales, se distribuyen las distintas dependencias que hacen parte de la entidad.

### 1.9.1 Organigrama Nivel Central:

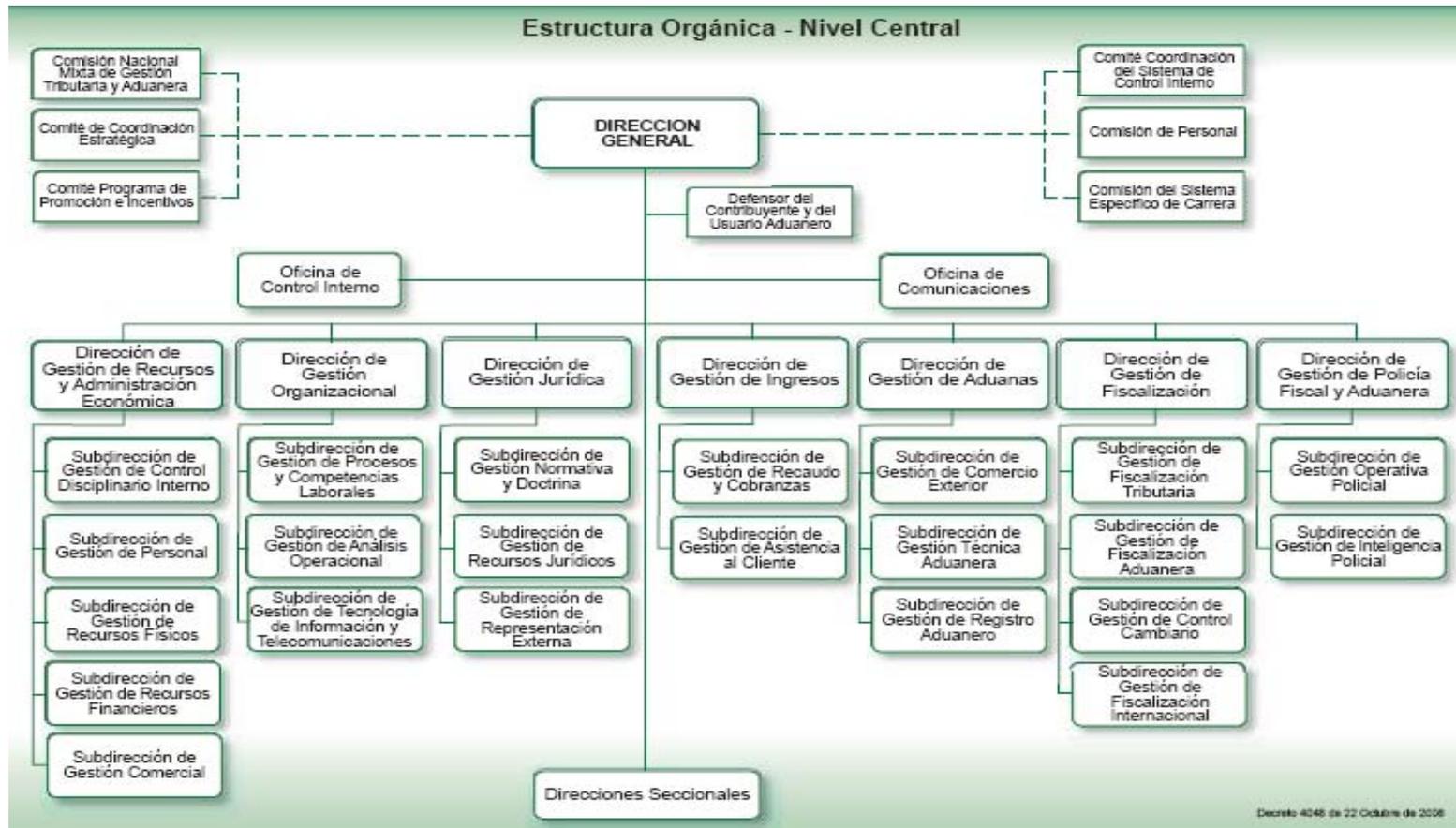


Figura 1. Organigrama Nivel Central.<sup>3</sup>

<sup>3</sup> [http://www.dian.gov.co/descargas/organigrama/Nuevos/NIVEL\\_CENTRAL.pdf](http://www.dian.gov.co/descargas/organigrama/Nuevos/NIVEL_CENTRAL.pdf)

## 1.9.2 Organigrama Direcciones Seccionales Locales

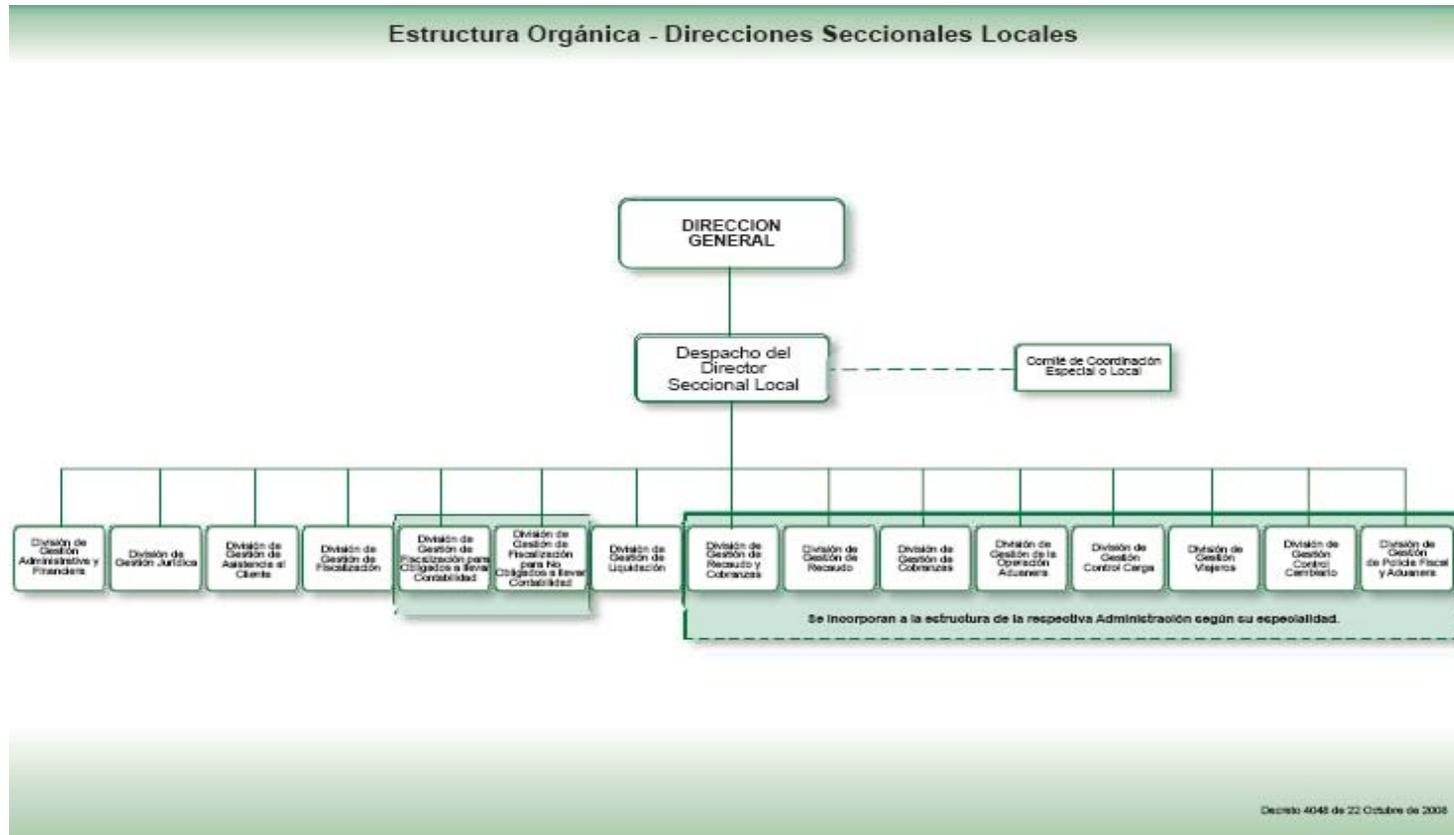


Figura 2. Organigrama Dependencias Locales<sup>4</sup>.

<sup>4</sup> [http://www.dian.gov.co/descargas/organigrama/Nuevos/Dir\\_Seccionales.pdf](http://www.dian.gov.co/descargas/organigrama/Nuevos/Dir_Seccionales.pdf)

## **2. DEFINICIÓN LÍNEA DE INTERVENCIÓN**

Para el manejo de los productos devolutivos y de consumo que entran y se retiran del Almacén de la DIAN Seccional Pereira, es necesario el uso de herramientas que permitan un control estricto sobre dichos artículos para evitar pérdidas, desfalcos o inconsistencias en la información procesada. Para cumplir con éstas exigencias, se solicitó el análisis, diseño e implementación de un Sistema de Información que permita almacenar todos los datos requeridos para dicho proceso es vital, al igual que puedan realizar distintos procesos (consultas, borrado, actualización e impresión entre otros). Teniendo en cuenta estos aspectos, la Línea de Intervención se define como Sistemas de Información.

### 3. DIAGNÓSTICO DEL ÁREA DE INTERVENCIÓN

#### 3.1 IDENTIFICACIÓN DEL PROBLEMA:

Cada una de las divisiones que conforman la Dirección de Impuestos y Aduanas Nacionales (DIAN) Seccional Pereira, requieren para su funcionamiento la utilización de distintos productos (escritorios, mesas, gasolina entre otros). Estos artículos ingresan primero al Almacén donde son registrados y después despachados. Tanto el ingreso y salida son almacenados en un Sistema de Información, el cual permite llevar un control sobre dichos movimientos mediante consultas, al igual que modificación y actualización de la información ingresada (artículos, funcionarios, dependencias entre otros)

El Sistema, aunque cumple correctamente con sus funciones y realiza todas las operaciones indicadas tiene muchas falencias con respecto a su interfaz y la manera en que llena y retorna la información. A continuación se mencionan algunos de los problemas más significativos:

- a) El Sistema fue elaborado hace más de 16 años, y no se le ha realizado ningún tipo de mejora o actualización.
- b) El software en el cual se realizó la aplicación está obsoleto y es algo lento (FoxDBase versión 2.10 de 1988); ofrece al usuario operaciones muy restringidas para el manejo de la información y permite que algunos campos sean modificados o borrados sin ningún control.
- c) La aplicación es algo compleja, ya que presenta demasiadas opciones para el usuario, muchas de ellas redundantes y otras que no utiliza.
- d) Los informes impresos sólo se pueden realizar en un tipo de Impresora determinado (EPSON), la cual por su tecnología (cinta) aumenta el tiempo de impresión demorando la presentación de los reportes.
- e) La aplicación no permite que el usuario se desenvuelva libremente a través de ella, ya que le obliga a terminar los procesos iniciados aunque éste ya no decida continuar con ellos.
- f) Ciertos reportes deben generarse a mano, perdiendo eficiencia y rapidez en el proceso de registro de los movimientos, y permitiendo también, que sean ingresados datos erróneos o inconsistentes.

Para llegar a estas conclusiones, se desarrollaron las siguientes actividades:

1. **Entrevistas con el Usuario final:** se indaga al usuario final sobre el aplicativo que actualmente está usando:
  - ¿Cuál es el proceso para el ingreso y salida de productos?
  - ¿Cuál es la finalidad del software?
  - ¿Cómo se ingresa la información al sistema?
  - ¿Cuáles son los campos utilizados, sus especificaciones y finalidad?
  - ¿Qué problemas presenta el aplicativo actual?
  - ¿Cuáles podrían ser las mejoras para dicho aplicativo?
  - ¿Qué espera del nuevo sistema?
  
2. **Observación Directa:** Se realizaron varias pruebas de ingreso y salida de artículos del almacén en donde se pudo observar cómo es la manipulación del aplicativo por parte del usuario (ingreso, modificación, borrado, actualización, impresión). Se pudo detallar, según el proceso, cuáles son los campos requeridos y de qué tipo son.
  
3. **Manipulación del Software:** se obtuvo una copia del aplicativo, sobre la cual se realizaron pruebas y análisis para entender su funcionamiento, la información requerida, los problemas presentados y las mejoras en cuanto a interfaz y manipulación de datos que sean posibles.
  
4. **Reportes:** se realizaron varias impresiones de los reportes que utiliza el aplicativo, con los cuales se obtuvo un punto de referencia para efectuar mejoras y correcciones..

#### **4. EJE DE INTERVENCIÓN**

Teniendo en cuenta los problemas antes mencionados, es necesario realizar una nueva aplicación que permitan solventar los inconvenientes presentados en la actual Sistema. El análisis, diseño e implementación del nuevo Sistema de Información para el Almacén de la DIAN, Seccional Pereira será el Eje de Intervención de la Práctica Profesional.

## 5. JUSTIFICACIÓN EJE DE INTERVENCIÓN

Para el Almacén de la DIAN Seccional Pereira es primordial el control sobre los productos que ingresen y se despachen hacia las distintas divisiones. Por ello es importante el uso de un Sistema de Información que permita la administración y supervisión de dichos productos de manera rápida, fácil y efectiva.

Aunque el Sistema actual permite el control sobre dichos productos, presenta ciertas inconsistencias (antes mencionadas) que dificultan el proceso. Como consecuencia, es necesaria la implementación de un nuevo Sistema de Información que pueda solucionar los problemas antes presentados, mejorar la interacción con el usuario y al cual se podrán añadir nuevos módulos para el procesamiento de la información.

Entre las ventajas de actualizar la aplicación están:

- a) La aplicación se desarrollaría en software ampliamente utilizados hoy en día (JAVA Y MySQL) los cuales poseen un sin fin de herramientas que permiten el desarrollo de interfaces de usuario más intuitivas y simples y mejoran enormemente el rendimiento y rapidez en el procesamiento de los datos.
- b) Dichos programas son gratuitos y no requieren de ningún tipo de licencia, por lo cual, para la Entidad representaría un reducción de gastos importante y un mejoramiento significativo en sus procesos internos.
- c) Al utilizar herramientas actuales, la configuración y la compatibilidad con diversos dispositivos es más fácil y mucho más amplia lo que permitiría agregar un sin fin de opciones de acuerdo a los requerimientos del usuario para la mejora de la aplicación.
- d) La aplicación al ser desarrollada en JAVA, es multiplataforma, es decir, es independiente del Sistema Operativo que se utilice, siendo una característica novedosa para la institución
- e) El Sistema es escalable y flexible, es decir, su modificación e inclusión de nuevos módulos es mucho más simple.

## **6. OBJETIVOS**

### **6.1 OBJETIVO GENERAL**

Analizar, Diseñar e implementar un nuevo Sistema de Información para el manejo del Almacén en la DIAN, seccional Pereira, el cual permita administrar y supervisar todas las entradas y salidas de productos de consumo y devolutivos

### **6.2 OBJETIVOS ESPECÍFICOS:**

- Generar toda la documentación necesaria que describa el proceso de desarrollo del Sistema de Información
- Realizar el modelo Entidad-Relación de la Base de Datos.
- Implementar la Base de Datos para el Almacén.
- Diseño y Construcción del Aplicativo (Software) que permita una fácil interacción entre la Base de Datos y el Usuario.
- Migración de toda la información almacenada en el anterior Sistema hacia el nuevo aplicativo.
- Elaborar los respectivos manuales (Técnico y de Usuario) de la aplicación.
- Realizar el proceso de implementación del Software.

## 7. REFERENTE CONCEPTUAL

Para la implementación del nuevo Sistema de Información para el Almacén se deben tener en cuenta aspectos como:

- Funcionalidad y Confiabilidad del Sistema Anterior.
- Herramientas de desarrollo Software a emplear
- Interacción Usuario-Aplicación (ingreso, procesamiento, y salida de información; errores e inconsistencias en los registros almacenados)
- Interfaz de Usuario (GUI Graphic User Interface), Modelo y Estructura de la Base de Datos.
- Migración de la información almacenada

El objetivo de implementar el nuevo aplicativo es poder solucionar muchos de los problemas presentados hasta ahora y mostrar al usuario una interfaz más amigable, intuitiva y fácil de usar. También, es permitir explorar nuevas alternativas que permitan mejorar el tratamiento de información mediante la inserción de nuevos módulos o mejoramiento de los ya existentes.

Este nuevo Sistema se implementará en herramientas software actuales, libres y que no incurran en costos para su utilización, lo que permite un tratamiento más rápido de la información y mejoras sustanciales de la experiencia del usuario con el aplicativo, al igual que una reducción de costos importantes para la Entidad.

EL desarrollo de este Sistema de Información se estructura en varias etapas (ver Cronograma), durante las cuales se presentan y analizan todos los requerimientos, exigencias y mejoras que debe tener el aplicativo. Estas etapas están contempladas dentro del proceso de Desarrollo Software, y están encaminadas a abordar los distintos componentes que trae consigo la construcción del programa desde diferentes puntos de vista (programador, usuario, administrador), con el fin de que el producto final pueda cumplir con su propósito de una manera eficiente, y cumpliendo las expectativas de los diferentes agentes que intervinieron en el proceso.

Dentro de la Ingeniería del Software existen varios modelos para el desarrollo de software. Cada uno de estos modelos representa una forma distinta de abordar la implementación de un aplicativo (análisis, diseño, programación, instalación). La selección de un modelo u otro, depende de las necesidades y de la naturaleza del programa:

### **Modelo de Prototipos:**

Un *prototipo* es una versión preliminar, intencionalmente incompleta o reducida de un sistema. El término *prototipaje rápido* (RAD – Rapid Application Development) se refiere al proceso de construir y evaluar uno o más prototipos rápidamente. Los prototipos son estrategias aplicadas a la mayoría de actividades del proceso de software las cuales pueden estar relacionadas con aspectos técnicos, funcionales, eficiencia o interfaces de usuario. Los prototipos rápidos permiten el desarrollo sencillo con resultados inmediatos. Ya que un prototipo se concentra en las propiedades que requieren mayor investigación, aspectos adicionales pueden dejarse de lado, siendo mostrados únicamente de forma esquemática. El propósito de los prototipos es buscar de manera preliminar información necesaria para ayudar en la toma de decisiones.

Los prototipos complementan el desarrollo de sistemas incrementales y pueden ayudar a reducir el riesgo en la especificación de requisitos o en el diseño de la arquitectura del sistema. Una ventaja de los prototipos es que sirven como medio de comunicación entre el desarrollador y el cliente, ayudando a visualizar rápidamente la dinámica del sistema. Por lo general un prototipo no es lo suficientemente robusto para ser el producto final. Sin embargo, la existencia de la demostración apoya la creencia de que el producto completo puede desarrollarse de manera satisfactoria. Si el prototipo es diseñado con cuidado puede inclusive utilizarse en el sistema final.

En nuestro caso se utilizará el Modelo de Prototipos por las siguientes razones:

- La presentación y prueba de cada prototipo al usuario, supone el mejoramiento del mismo, basado en la experiencia de éste último con el aplicativo.

- Según lo expresado por el usuario, el programador puede identificar los puntos críticos del Sistema en cuanto al procesamiento y presentación de la información, lo que ayuda a que pueda resolver de una manera más fácil, rápida y efectiva los problemas expuestos.
- La evolución de cada prototipo basada en la relación usuario-programador, hacen que el sistema final cumpla con las expectativas planteadas al principio del proceso y cumpla al 100% con el objetivo para lo cual fue implementado.

Básicamente el Sistema consta de 2 partes: una Base de Datos Relacional, y un aplicativo Software.

**Base de Datos:** La Base de Datos, es una Base de Datos Relacional: Es un sistema de administración de bases de datos, que almacena información en tablas (filas y columnas de datos) y realiza búsquedas utilizando los datos de columnas especificadas de una tabla para encontrar datos adicionales en otra tabla. En una base de datos relacional, las filas representan registros (conjuntos de datos acerca de elementos separados) y las columnas representan campos (atributos particulares de un registro). Al realizar las búsquedas, una base de datos relacional hace coincidir la información de un campo de una tabla con información en el campo correspondiente de otra tabla y con ello produce una tercera tabla que combina los datos solicitados de ambas tablas. En otras palabras, una base de datos relacional utiliza los valores coincidentes de dos tablas para relacionar información de ambas<sup>5</sup>.

Existe otro modelo: Base de Datos Orientada a Objetos. Es una base de datos en el sentido de la definición introductoria, donde los elementos de datos son objetos y las relaciones se mantienen por medio de inclusión lógica. Las entidades de aplicación están representadas como clases. La autodescripción se obtiene porque las clases son metaobjetos que contiene los nombres de atributos y métodos de señal. Una BDOO contiene un método sistemático de representación de relación, y la interfaz uniforme de usuario es un sistema de mensajes que puede explorar los objetos y sus interconexiones.

En una BDOO, las entidades de aplicación son las clases, las instancias de entidad son objetos creados desde las clases, y las relaciones se mantienen

---

<sup>5</sup> <http://www.terra.es/personal/lrmon/cat/articles/evin0045.htm>

por medio de inclusión lógica. Un sistema de señales y métodos para procesarlas contiene una interfaz uniforme para la base de datos.

Aunque el modelo Orientado a Objetos ofrece flexibilidad y mayor rendimiento en los datos y operaciones que se realicen, no se escogió este modelo ya que presenta algunas desventajas<sup>6</sup>:

- El mercado de Base de Datos Orientado a Objetos no es muy fuerte y aún se encuentra en etapa de desarrollo, por lo tanto, las herramientas para el desarrollo de este modelo no son lo suficientemente confiables para el desarrollo del Sistema de Información para el Almacén de la DIAN Seccional Pereira.
- Este modelo no se encuentra estandarizado completamente, es una tecnología nueva. Como consecuencia, se corren grandes riesgos al diseñar e implementar un Sistema de Información en él, considerando el grado de confiabilidad y robustez del mismo.

Al ser el modelo Relacional, el modelo más usado actualmente para el diseño e implementación de Bases de Datos, y teniendo en cuenta la gran documentación y herramientas existentes para su desarrollo, se seleccionó como el tipo de modelado para el desarrollo del Sistema de Información.

Ésta base está compuesta por 14 tablas, las cuales contienen todos los campos necesarios para guardar los movimientos realizados en el almacén. Fue implementada usando el Sistema Gestor de Base de Datos MySQL.

MySQL es un sistema de administración para bases de datos relacionales (rdbms) que provee una solución robusta a los usuarios con poderosas herramientas multi-usuario, soluciones de base de datos SQL (structured Query Language) multi-threaded. Es rápido, robusto y fácil de utilizar.<sup>7</sup>

---

<sup>6</sup> <http://www.dei.uc.edu.py/tai2002/BDOO/ventoo.htm>

<sup>7</sup> <http://www.terra.com.mx/articulo.aspx?articuloid=118449>

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.<sup>8</sup>

Las principales características de este gestor de bases de datos son las siguientes:

1. Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
2. Soporta gran cantidad de tipos de datos para las columnas.
3. Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc.).
4. Gran portabilidad entre sistemas.
5. Soporta hasta 32 índices por tabla.
6. Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

Todas estas características lo hacen idóneo para la implementación de la Base de Datos del Almacén, ya que garantizará la integridad de los datos, al igual que aumentará la velocidad con la cual se procesarán y accederán a ellos. Otro punto importante es el costo. MySQL al ser OpenSource y trabajar con una licencia GPL, garantiza que si el Sistema va a ser desarrollado para uso interno, su uso es gratis y no requiere la adquisición de ninguna licencia de tipo Comercial.

**Modelo Entidad-Relación:** El modelo Entidad Relación fue realizado en una herramienta llamada DeSign. Es una herramienta intuitiva de diseño de bases de datos para desarrolladores y DBA's con la cual puede crear y mantener modelos de Bases de Datos. Utiliza diagramas Entidad Relación para graficar diseño de bases de datos y generar automáticamente scripts para la construcción de la base de datos<sup>9</sup>.

DeSign soporta una gran cantidad de Bases de Datos, entre ellas están:

- Clipper

---

<sup>8</sup> [http://www.netpecos.org/docs/mysql\\_postgres/x57.html](http://www.netpecos.org/docs/mysql_postgres/x57.html)

<sup>9</sup> <http://www.datanamic.com/dezign/datasheet.html>

- dBase
- DBISAM 4
- Firebird 1.5, 2
- FoxPro
- IBM DB2 Universal DB 7, 8
- InterBase 5, 6, 7
- Informix 9
- MaxDB 7
- MS Access 95, 97, 2000, 2003, 2007
- MS SQL Server 6.5, 7, 2000, 2005
- MySQL 3, 4, 5
- NexusDB V2
- Oracle 7, 8, 9, 10, 11
- Paradox
- Pervasive v8, PSQL v9
- PostgreSQL 7, 8
- Sybase ASE 11, 12

Esta herramienta permite la creación de modelos Entidad-Relación de una manera rápida y fácil. A partir de dichos modelos, este diseñador, puede crear un Script según la Base de Datos para la cual se esté desarrollando el diseño, el cual construye la base de datos automáticamente en el SGBD utilizado, sin necesidad de escribir ninguna línea de código.

Como se mencionó anteriormente, el modelo de desarrollo de software utilizado para el Sistema de Información para Almacén es el de Prototipos. Este modelo, no sólo se ve reflejado en la realización del aplicativo (software), sino que también afecta el diseño de la Base de Datos. Es por ello, que a través de todo el proceso se presentan varias versiones del modelo Entidad-Relación, basados en el flujo de información dado entre el usuario y el programador.

A continuación se detallan las distintas versiones que ha tenido dicho modelo:

### Primera Versión:

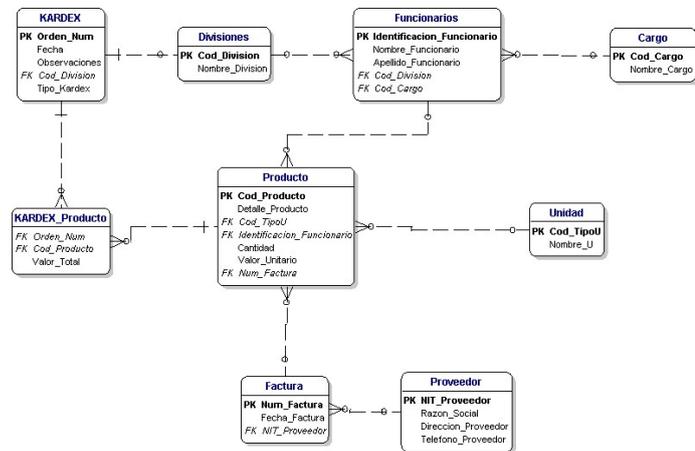


Figura 3. Primera Versión Modelo Entidad-Relación

Según la información aportada por el usuario y los datos que se recolectaron al interactuar con el anterior aplicativo, se diseñó este primero modelo. Se establece como entidad principal Productos, ya que contiene información primordial para la ingreso de los demás registros en las distintas entidades.

Al diseñar el segundo modelo se realizaron algunas mejoras:

Los campos de la Entidad Factura, contienen más información que permite detallarla mejor. La relación con la Entidad Productos de Muchos a Muchos, permite que se tenga un registro histórico de cómo ingresaron los productos (el número de la factura, el valor con el que ingresaron y la cantidad de los mismos).

Al realizar una Salida de Almacén se puede guardar el Funcionario y con éste, la División hacia donde se despacharon los productos. Este cambio se debe a que los artículos asignados, su uso y distribución son de responsabilidad del empleado y no de la División como tal.

Cuando se realiza una Entrada, es primordial registrar el número de factura con la cual se reciben los productos. Por ello se agregó el campo Num\_Factura a la Entidad Kardex. Con el anterior modelo, era imposible acceder a este registro, ya

que se debía primero buscar un producto cualquiera y conociendo el código de este, mirar el número de la factura con la cual llegó, procedimiento mucho más largo, y teniendo en cuenta que no se podía tener un registro histórico de los productos ingresados.

### Segunda Versión:

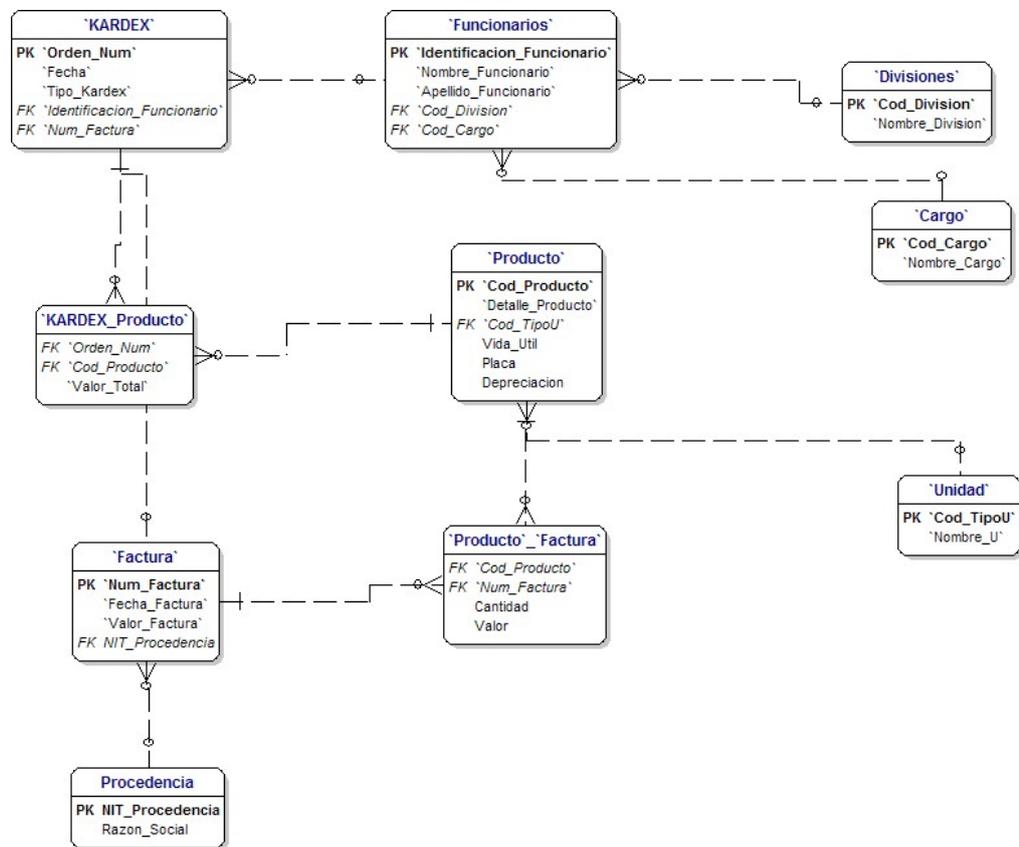


Figura 4. Segunda Versión Modelo Entidad-Relación

Uno de los cambios que se debían realizar a esta segunda versión es lo concerniente a los saldos de los productos en el Almacén. Es decir, que se pueda llevar un registro de los valores de los productos (unitario, como total teniendo en cuenta qué el valor de un producto puede variar), del stock disponible.

La herramienta Dezign no es gratuita, se debe pagar una licencia comercial por su uso. Para la realización de estas 2 primeras versiones se utilizó una versión Trial por 30 días. En vista de que la DIAN como requisito principal para la elaboración del Sistema de Información, pidió que fuera desarrollado en herramientas libres y sin costo alguno, para la realización de las siguientes versiones del modelo se utilizará MySQL WorkBench.

MySQL Workbench es una nueva generación de aplicaciones de diseño visual de base de datos que pueden ser utilizados de manera eficiente para diseñar, administrar y documentar esquemas de base de datos. Es similar a la herramienta anterior, pero está diseñada específicamente para ser utilizada con el SGBD MySQL, una gran ventaja teniendo en cuenta que es el gestor que se está utilizando. A parte de ello, es un aplicativo gratuito aunque también existen licencias de tipo comercial. En adelante esta será la herramienta a utilizar.

La segunda versión del Modelo Entidad-Relación presentaba los siguientes errores:

- No se podía distinguir si el tipo de movimiento era una Entrada o un Salida. Tampoco permitía ingresar o consultar mayor información con respecto a la actividad realizada.
- El registro de una Entrada o Salida de Almacén requería que se llenaran ciertos datos, que eran diferentes para los 2 tipos de movimiento.
- No se podía registrar el detalle de cada uno de los productos que entraran al almacén, ya que el registro se llenaba para un producto y se generalizaba para el mismo tipo de artículo que se ingresara después, teniendo en cuenta que podían tener distintos valores.
- Los productos, específicamente los devolutivos, no tenían la forma de almacenar la descripción o seriales de los mismos.
- Se requería una tabla especial para manejar los datos de los funcionarios que deben firmar los reportes generados.

### Tercera Versión:

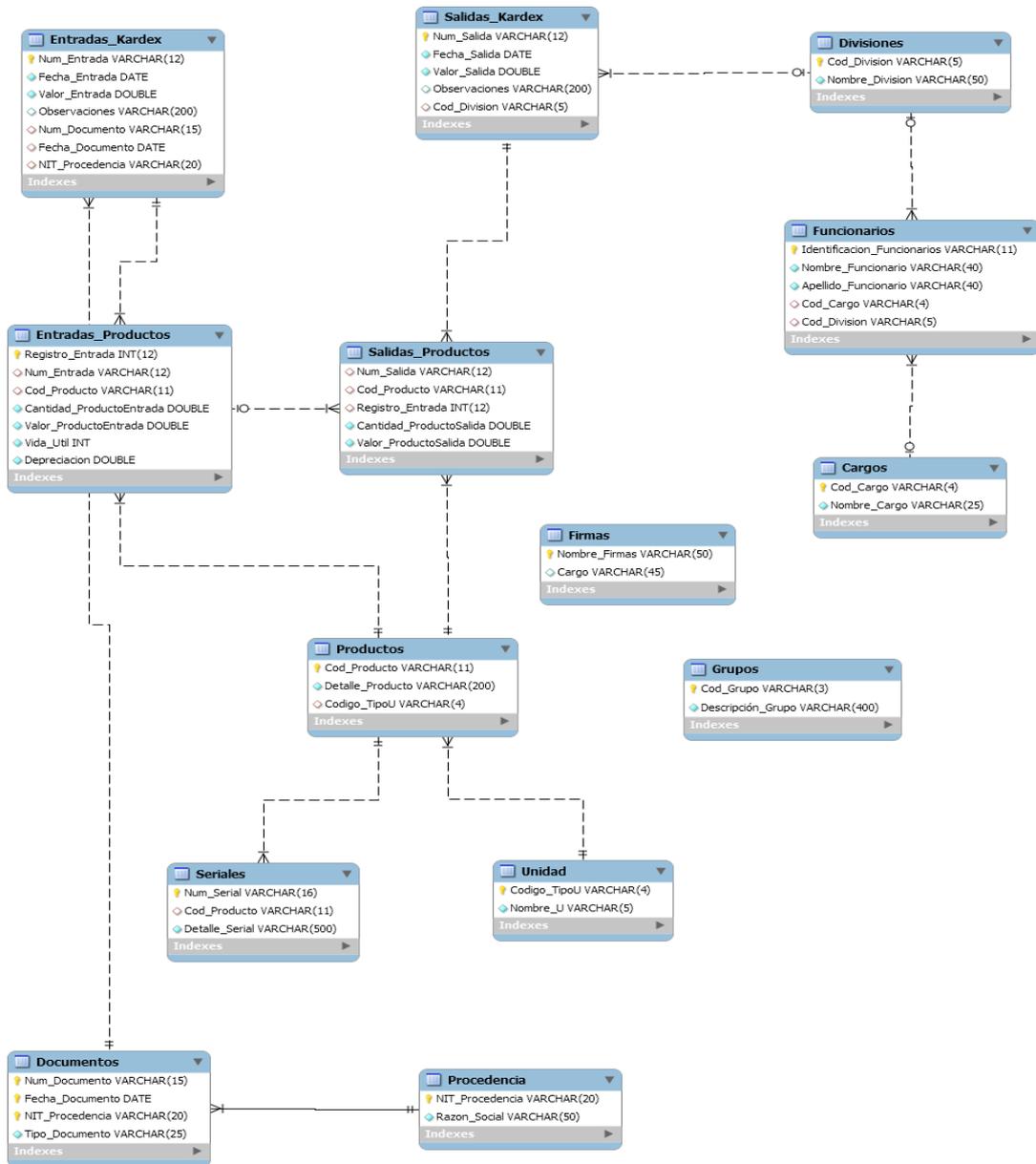


Figura 5. Tercera Versión Modelo Entidad-Relación

Este tercer modelo cambió la mayor parte de su estructura con el objetivo de solventar los problemas de los 2 anteriores y para intentar ingresar en él la mayor cantidad de información posible que permita un mejor control y manejo de las Entradas y Salidas de los artículos del almacén

Entre las mejoras están:

- La tabla Kardex se dividió en 2 diferentes: Entradas\_Kardex, Salida\_Kardex. Cada movimiento debe registrar un número consecutivo distinto y en el caso de la Entrada se debe registrar la información de dónde provino, mientras que la salida debe registrar la división hacia la cual se despacha el producto.
- Al generar estas 2 tablas, también se generan 2 tipos de relaciones con la tabla Productos: Entrada\_Productos, con la cual se registra la entrada de los productos al Almacén. Esta tabla almacena toda la información con el producto que se va a registrar: el número de la entrada asociado, el valor y la cantidad, su vida útil y depreciación. La tabla Salida\_Productos, guarda los datos del producto que será despachado.
- La salida del producto del almacén se realiza hacia una división, no hacia un funcionario específico por lo que se modificó la relación Kardex-Funcionarios a Salida\_Kardex-Divisiones.
- Se agregó la tabla seriales, la cual permite almacenar los seriales de los productos devolutivos que ingresen al almacén.
- También se agregó la tabla Firmas, para los funcionarios que deben firmar los reportes que se generen.

## 8. CRONOGRAMA

Actividades	Febrero				Marzo				Abril				Mayo				Junio				Julio			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
<b>Proyecto Aduanas</b>																								
Identificación del Problema, Análisis y Documentación																								
Modelado Entidad-Relación (Diseño, Ajustes)																								
Diseño e Implementación Base de Datos																								
Construcción Interfaz Gráfica-Conexión BD																								
<b>Proyecto Almacén</b>																								







## 9. PRESENTACIÓN Y ANÁLISIS DE RESULTADOS

Durante el diseño, construcción e implementación del Sistema de Información para el Almacén de la DIAN Seccional Pereira se elaboró un cronograma con el cual se definían cada una de las etapas que tendría el proyecto y el tiempo aproximado de cada una de ellas.

La duración de cada etapa fue definida, de tal forma que se lograría un cierto margen de ventaja con respecto a la entrega final, para lograr implementar el aplicativo y hacer todas las pruebas pertinentes para entregar un producto que pudiera satisfacer los requerimientos del cliente, y fuere totalmente confiable para su instalación.

Inicialmente se planteó la entrega para un mes antes, con el fin de migrar la información del sistema anterior al nuevo, y hacer todas las sesiones de prueba de rigor. Pero como en todo proceso de construcción software se presentan situaciones que inicialmente no se tenían estipuladas o el diseño del aplicativo y de la base de datos requería de varios cambios para satisfacer los requerimientos propuestos:

- **Reuniones con el Usuario:** Desde el comienzo del proceso, la interacción usuario-programador fue vital para el desarrollo del proyecto. Gracias a ello, se logró la completa comprensión del proceso de Entradas y Salidas de elementos (Consumos y Devolutivos) desde y hacia el Almacén.

Al comprender este proceso se empezó a diseñar cada uno de los prototipos que tuvo el proyecto. Con las entrevistas realizadas con el usuario semanalmente se logró depurar y mejorar el aplicativo, con la adición de nuevos módulos, el cambio de algunos procesos y la eliminación de otros. Por ello, constantemente se tuvo que reestructurar el Sistema de Información, invirtiendo más tiempo del inicialmente estipulado.

- **Inclusión de Herramientas Nuevas:** a medida que se iba estructurando cada uno de los prototipos, y la información aportada por el usuario y otras situaciones como el caso de licencias del software utilizado, se requirió utilizar nuevas herramientas y reemplazar otras. Estos aplicativos, que

fueron utilizados para la generación de reportes (iReport) y el diseño del Modelo Entidad-Relación (MySQL Workbench), al ser la primera vez en utilizarlos, requirió cierto tiempo para entender cómo era su funcionamiento y poder utilizarlos en el proyecto

- La generación de reportes para el aplicativo requirió cierto tiempo, ya que eran algo complejos de elaborar para poder cumplir con las necesidades del cliente. Fue una etapa que retrasó un poco el proyecto.

El proceso de migración fue algo complejo y requirió mayor tiempo del esperado ya que:

- Las 2 bases de datos tienen un manejo diferente de los registros tanto en estructura (tablas) como en el contenido de los campos (tamaño, tipos), por lo que se debía extraer la información de la base de datos anterior, tabularla para poder adecuarla a la existente.
- El nuevo Sistema almacena información nueva, es decir, información que el anterior no requería (seriales, descripciones) por lo que se tuvo que adecuar algunos campos de la BD para que almacenara valores nulos.
- La BD anterior contiene aproximadamente más de 22000 registros. Se buscaron diversas herramientas (scripts, macros entre otros) para automatizar el proceso de extracción y posterior inserción en la base de datos, pero debido a los puntos anteriores solo se pudo lograr reducir el tiempo de extracción de los registros de la base de datos.

Como resultado del proceso de migración se obtuvieron los siguientes resultados:

- Se logró migrar todos los productos (código, detalle), divisiones, cargos y funcionarios a la nueva Base de Datos.
- Todos los registros que contiene los saldos de los elementos también se logró migrarlos satisfactoriamente.
- El registro de los movimientos (entrada, salida) los cuales sumaban aproximadamente 18000, se tomó la decisión de migrar aquellos ingresados desde el año 2008 hasta la fecha, por razones de tiempo y considerando que dicha información entre ese lapso de tiempo era la requerida para implementar el nuevo sistema

## Pruebas:

Durante el desarrollo e implementación de los distintos prototipos, se realizaron pruebas de ingreso, consultas, modificación y borrado de información en cada uno de los módulos del Sistema de Información para testar su funcionalidad, nivel de seguridad y el correcto manejo de la información.

Las pruebas fueron las siguientes:

## Usuarios y Perfiles:

- Se crearon 4 tipos de usuarios con diferentes tipos de permisos.

**Administrador:** Tiene acceso total a la base de datos. Es decir, puede crear, modificar o eliminar tablas, cambiar los atributos a las entidades entre otros.

**Usuario 1:** Administrador del aplicativo . Tiene acceso a todos los módulos del aplicativo. Puede ingresar, borrar o actualizar la información.

**Usuario 2:** Para ingreso y consultas solamente. No puede actualizar, modificar o borrar información.

**Usuario 3:** Consultas solamente. Solo puede realizar consultas a la información registrada. No puede modificar, borrar, ingresar o actualizar información.

Con cada uno de los perfiles definidos para cada usuario se ingresó con cada uno de ellos, para comprobar que efectivamente solo pudieran acceder a los módulos y procedimientos habilitados para ellos. Las restricciones funcionaron para cada caso y no hubo ningún tipo de violación a la seguridad del aplicativo. También se intentó ingresar con usuarios falsos, con los cuales no se pudo acceder al programa.

## Software:

Se hicieron pruebas al diseño y distribución de los campos de texto, imágenes y demás en las distintas pantallas del aplicativo con el fin de determinar:

- Que el diseño (colores, distribución de elementos, colores, tamaño y tipo de fuentes) fuera lo mejor posible para que el usuario tuviera una mejor experiencia en el uso del aplicativo y este fuera lo más sencillo y agradable de usar.
- Que los distintos objetos incluidos, funcionaran de acuerdo a los parámetros y eventos asignados (permitir ingresar, escribir sobre ellos, mostrar información etc)
- Que pudieran funcionar sobre situaciones críticas (varias pulsaciones sobre determinados objetos, ventanas abiertas simultáneamente, entre otras)

### **Módulo de Ingresos y Salidas:**

Las pruebas fueron las siguientes:

- Dependiendo del tipo de movimiento (Entrada o Salida) que se eligiera, el consecutivo de ambos sería diferente. Se realizaron varios ingresos y varias salidas intercaladas para comprobar que efectivamente variaran según la selección. Este consecutivo reinicia cada año, por ello se cambió constantemente el año del sistema para visualizar el comportamiento del mismo.
- La fecha del documento con el cual entran los productos al almacén no puede ser posterior a la fecha en que se realiza la entrada, como mínimo debe registrar el mismo día. Por ello se ingresaron fechas posteriores, iguales y anteriores a las de la elaboración del ingreso, mostrando siempre error en las fechas posteriores.
- En una entrada o en una salida, sólo debe ingresar un tipo de producto (consumo o devolutivo). En las distintas pruebas, se introdujeron códigos de ambos tipos intercalados, mostrando errores y pidiendo que cambiemos los códigos erróneos según la selección realizada
- Se validó que los códigos ingresados siempre tuvieran 11 dígitos y sólo fueran caracteres numéricos. Y en caso tal de que no existieran, se permitía su creación.
- Para el caso de elementos devolutivos y cuando se estuviera realizando una entrada, se comprobaba que el número del serial funcionara correctamente.
- La cantidad del producto tanto para ingresar y salir, no podía ser 0 ni negativa. En caso de utilizarse valores decimales estos se deben separar por puntos no por comas. En las diferentes pruebas se ingresaron valores que no cumplían estos requisitos y el sistema no permitió su ingreso.

- Un tratamiento similar se dio a los valores de los productos: no podían ser menores de 0 o iguales. El sistema tampoco permitía un ingreso erróneo.

#### **Modulo de Consultas:**

Se muestran todas las entradas y salidas registradas, al igual que los saldos de los productos. Se procedió a imprimir cada registro para verificar la consistencia de los datos y la distribución de los reportes.

#### **Módulo de Administración:**

Las pruebas se enfocaron a la actualización, eliminación e ingreso de productos, funcionarios, divisiones entre otros. Estas modificaciones tenían que verse reflejadas en tanto en la base de datos, como en el aplicativo y en cada prueba realizada se cumplió con el objetivo.

## 10. RECOMENDACIONES

Para que el Sistema de Información pueda suplir todas las necesidades, las entrevistas con el usuario son fundamentales, ya que ellas aportan información acerca del proceso, expectativas del cliente, adiciones o modificaciones de módulos entre otras. Ésta es la única manera de que el sistema pueda cumplir con todos los requerimientos para lo cual fue diseñado.

Un buen diseño y una excelente planeación de las etapas del proyecto pueden ahorrar tiempo y gastos en la implementación del producto. Por ello es importante aplicar las distintas metodologías de la Ingeniería del Software con el fin de entender y abarcar en todo lo posible el comportamiento del sistema y sus requerimientos. Esto con el fin de que pueda facilitar futuras mejoras o inclusiones de nuevos módulos al aplicativo sin necesidad de grandes cambios al diseño o existan problemas al implementarlos.

Siempre se deben contar con imprevistos en las etapas de desarrollo del software. Ya sea por la modificación al diseño, en los requerimientos planteados inicialmente o por la necesidad de incluir nuevas herramientas para la construcción del mismo. Un buen cronograma y la inclusión de estas situaciones en el mismo ayudarán a anticipar problemas mejorando el tiempo de desarrollo del producto

## CONCLUSIONES

- La elaboración de un cronograma en el cual se estipulen las distintas etapas y los tiempos que tendrá el proceso de desarrollo de software es vital para llevar un control sobre el proceso y evitar posibles retrasos o inconvenientes.
- Todo proceso de desarrollo software debe ir acompañado por el o los usuarios finales, ya que estos son los que en definitiva, condicionan el proyecto a sus necesidades y requerimientos y dan su aprobación al mismo para su implementación y utilización.
- La generación de toda la documentación necesaria y la capacitación a los usuarios garantiza que el Sistema de Información pueda funcionar de una manera adecuada y que su mantenimiento y modificación sea un proceso rápido y eficiente.
- La estructura de la Base de Datos siempre se debe diseñar en término de eficiencia y simplicidad. En la práctica, demasiadas tablas pueden reducir la velocidad y el tiempo de consultas, inserciones y borrados de registros lo que llevaría a un uso ineficiente del Sistema de Información

## **BIBLIOGRAFÍA**

Dirección de Impuestos y Aduanas Nacionales (DIAN). Presentación de la Entidad.  
Disponible en: [www.dian.gov.co](http://www.dian.gov.co)

Sistemas Gestores de bases de datos. Gregorio Cabrera Sanchez. Cap 1 pag  
3 – 23. Pressman

Ingeniería del Software Orientada a Objetos. Weitzenfeld, Alfredo. Cap 2 y 3.  
Editorial Itam.

## ANEXOS

### ANEXO A. DESCRIPCIÓN DE ENTIDADES Y RELACIONES

**Entradas\_Kardex:** guarda información detallada de las entradas que se realizan al almacén (Orden, fecha Documento etc.)

#### **Campos:**

**Num\_Entrada:** campo tipo varchar con un tamaño máximo de 11 caracteres. Almacena el número de una entrada que se registre, es consecutivo. Llave primaria. No admite valores nulos. Está compuesto del año en el cual se realiza la entrada, y del número del consecutivo que se lleve. Cada cambio de año, reinicia la numeración.

**Fecha\_Entrada:** fecha en la cual se está registrando la entrada. Campo tipo Date (almacenar fechas). No admite valores nulos

**Valor Entrada:** Guarda el valor total de la entrada que se está registrando. Campo tipo Double (campo numérico para el manejo de valores demasiado grandes y valores decimales). No admite valores nulos

**Observaciones:** campo tipo Varchar con tamaño máximo de 200 caracteres. Permite almacenar las observaciones que se hagan al movimiento que se está realizando.

**Num\_Documento:** número de la factura o del documento en los cuales vienen detallados los productos que van a ingresar al almacén. Llave foránea. Proviene de la tabla Documento. No admite valores nulos

**Fecha Documento:** fecha que aparece en la factura o documento. Llave foránea Proviene de la Tabla Documento. No admite valores nulos

**NIT\_Procedencia:** NIT o identificación del lugar, almacén o seccional de la cual provienen los productos a ingresar. Llave foránea. No admite valores nulos

**Productos:** tabla que almacena la descripción de los productos.

**Campos:**

**Cod\_Producto:** Código del producto. Llave Primaria. Campo tipo Varchar con capacidad para 11 caracteres. No admite valores nulos

**Detalle\_Producto:** Descripción del producto. Campo tipo varchar con capacidad para 200 caracteres. No admite valores nulos

**Codigo\_TipoU:** Código del Tipo de Unidad que maneja el producto (NUM, GAL). Llave foránea. Proviene de la tabla Unidad. No admite valores nulos

**Unidad:** Tabla donde se guardan los distintos tipos de unidades con los que se categorizan los productos.

**Campos:**

**Cod\_TipoU:** Código de la Unidad. Campo Varchar con longitud máxima de 4 caracteres. Llave Primaria No admite valores nulos

**Nombre\_U:** Nombre o descripción de la unidad. Campo Varchar con longitud máxima de 5 caracteres. No admite valores nulos

**Seriales:** Tabla en la cual se almacenan los seriales para los elementos devolutivos que ingresen al almacén.

**Campos:**

**Num\_Serial:** Número con el cual se identifica el serial. Está compuesto por el número con el cual ingresa el producto al almacén

(el cual se almacena en la tabla Entradas\_Productos) y el código del producto. Campo Tipo Varchar de 16 caracteres como longitud máxima. Llave primaria No admite valores nulos

**Cod\_Producto:** Código del producto al cual corresponde el serial que se está ingresando. Campo Tipo Varchar. Proviene de la tabla Productos. No admite valores nulos

**Detalle\_Serial:** Descripción detalla del serial para el artículo que se está ingresando. Campo tipo varchar con longitud máxima de 400 caracteres.

**Documentos:** Almacena la información de la factura o documento con el cual ingresan los productos al almacén.

**Campos:**

**Num\_Documento:** número de la factura o documento. Campo tipo VARCHAR de 15 Caracteres. Llave Primaria. No admite valores nulos.

**Fecha Documento:** fecha que aparece en el documento o factura. Campo Tipo DATE. Llave Primaria. No admite valores nulos.

**NIT\_Procedencia:** NIT o Número del lugar, almacén o seccional de la cual provienen los productos. No admite valores nulos. Llave Primaria y a la vez Llave Foránea de la Tabla Procedencia.

**Tipo\_Documento:** Detalla cuál es el tipo de documento (factura, asignación etc.) Campo Tipo Varchar de 25 caracteres.

**Procedencia:** Guarda la información del origen de los productos que ingresan al almacén

**Campos:**

**NIT\_Procedencia:** NIT o Número del lugar, almacén o seccional de la cual provienen los productos. No admite valores nulos. Llave Primaria. Campo Tipo Varchar de 20 caracteres.

**Razón \_Social:** Nombre o Descripción de lugar o dependencia de donde provienen los productos. Campo Tipo Varchar de 50 caracteres

**Salidas\_Kardex:** guarda información detalla de las salidas que se realizan.

**Campos:**

**Num\_Salida:** Almacena el número de una salida que se registre, es consecutivo No admite valores nulos. Llave primaria. Campo tipo Varchar de 11 caracteres. Está compuesto del año en el cual se registra la salida, y el número del consecutivo que se lleve hasta el momento. Cada cambio de año, reinicia la numeración.

**Fecha\_Salida:** Fecha en la cual se registra la salida. Campo tipo DATE. No admite valores nulos.

**Valor\_Salida:** Valor Total de la Salida que se está registrando. Campo Tipo Double.

**Observaciones:** campo que almacena las observaciones que se le hagan a la salida que se está registrando. Campo Tipo Varchar de 200 caracteres.

**Cod\_División:** Código de la división hacia la cual se registra la salida. Llave Foránea. Proviene de la tabla Divisiones.

**Divisiones:** Guarda la información de todas las divisiones de la DIAN Seccional Pereira.

**Campos:**

**Cod\_División:** Código que identifica la división. Campo Tipo VARCHAR de 5 caracteres. Llave Primaria. No admite valores nulos.

**Nombre\_División:** Nombre de las Divisiones de la DIAN Seccional Pereira. Campo Tipo Varchar de 50 caracteres. No admite valores nulos.

**Funcionarios:** Guarda la información de los funcionarios, específicamente de los jefes de cada una de las divisiones de la DIAN.

**Campos:**

**Identificación\_Funcionarios:** Cédula del Funcionario. Llave Primaria. Campo Tipo VARCHAR de 11 caracteres. No admite valores Nulos.

**Nombre\_Funcionario:** Nombre del Funcionario. Campo Tipo Varchar de 40 caracteres de longitud. No admite valores Nulos.

**Apellido\_Funcionario:** Apellido del Funcionario. Campo Tipo VARCHAR de 40 caracteres de Longitud. No admite valores Nulos.

**Cod\_Cargo:** código que identifica el cargo que tiene el funcionario. Llave foránea. Proviene de la tabla cargos.

**Cod\_División:** código de la división a la cual pertenece el funcionario. Llave foránea. Proviene de la tabla divisiones.

**Cargos:** guarda información de los distintos cargos que existen en la DIAN.

**Cod\_Cargo:** Código que identifica el cargo. Campo tipo varchar de 4 caracteres. Llave Primaria. No admite valores nulos.

**Nombre\_Cargo:** nombre o descripción del cargo. Campo Tipo varchar de 25 caracteres.

**Firmas:** almacena la información de los funcionarios que deben firmar los reportes generados por el aplicativo.

**Campos:**

**Nombre\_Firmas:** nombres y apellidos del funcionario. Campo Tipo VARCHAR de 50 Caracteres.

**Cargo:** Cargo que tiene el funcionario. Campo Tipo Varchar de 50 Caracteres.

**Grupos:** almacena la información de los grupos a los cuales pertenecen los productos según los 3 primeros dígitos de su código.

**Campos:**

**Cod\_Grupo:** código del grupo. Llave Primaria. Campo Tipo Varchar de 3 caracteres.

**Descripción\_Grupo:** Detalle y descripción del Grupo. Campo Tipo Varchar de 400 caracteres.

**Relaciones:**

**Entradas\_Kardex – Productos:** Muchos a Muchos. Una entrada puede registrar varios productos, y un producto puede estar presente en varias entradas. Se crea una tabla:

**Entradas\_Productos:** permite relacionar la información de la entrada con los productos que ingresan en ella.

**Campos:**

**Registro\_Entrada:** número consecutivo que se le asigna a los productos que van ingresando, con el fin de poder identificarlos más fácilmente al momento de realizar una consulta o registrar una salida. Campo Tipo INT de 12 dígitos. Llave Primaria. NOT NULL.

**Num\_Entrada:** número de la entrada con la cual se van a ingresar determinados productos. Llave foránea. Proviene de la tabla Entradas\_Kardex.

**Cod\_Producto:** Código de los productos que se van a ingresar al almacén. Llave Foránea. Proviene de la tabla Productos.

**Cantidad\_ProductoEntrada:** número de unidades con que ingresa un producto. Campo tipo Double.

**Valor\_ProductoEntrada:** valor total (valor unitario x cantidad\_productoentrada) con el cual ingresa el producto.

**Vida\_Util:** vida útil que tiene el producto. Campo tipo INT.

**Depreciación:** depreciación del producto. Campo Tipo INT.

**Salidas\_Kardex – Productos- Entradas\_Productos:** muchos a muchos a muchos. Un Salida puede registrar varios productos y un producto puede estar presente en varias salidas. Además, el valor del producto con el cual sale, depende de los valores con los cuales se registró su ingreso

**Campos:**

**Num\_Salida:** número de la salida con el cual se va a registrar la salida de los productos. Llave Foránea de la tabla Salidas\_Kardex.

**Cod\_Producto:** código de cada uno de los productos que se van a registrar en la salida. Llave foránea de la tabla Productos.

**Registro\_Entrada:** número que facilita obtener los valores con los cuales ingresó el producto para registrarlos en la salida. Llave foránea de la tabla Entradas\_Productos.

**Cantidad\_ProductoSalida:** número de unidades de un producto que se van a sacar. Campo Tipo Double

**Valor\_ProductoSalida:** valor total (valor unitario x valor producto salida) del producto que se va a sacar. Campo Tipo Double

**Productos-Seriales:** de uno a muchos. Un producto puede tener varios seriales. Pero un serial solo puede pertenecer a un producto específicamente.

**Productos-Unidad:** de Muchos a uno. Un producto sólo puede tener un tipo de unidad asociado, pero un tipo de unidad puede ser para varios productos.

**Entradas\_Kardex-Documentos:** relación de uno a uno. Una entrada solo puede tener asociado un solo documento, y un documento solo puede estar asociado a una entrada.

**Documentos-Procedencia:** de muchos a uno. Un documento solo puede tener asociado una sola procedencia. Pero de una procedencia pueden venir varios productos.

**Salidas\_Kardex-Divisiones:** de muchos a uno. Una salida sólo se registra hacia una división. Pero hacia una división se pueden registrar varias salidas.

**Divisiones-Funcionarios:** de uno a muchos. Una división puede tener asociados varios funcionarios, pero un funcionario solo puede pertenecer a una división.

**Funcionarios-Cargos:** de muchos a uno. Un funcionario solo puede tener un cargo, pero un cargo puede ser asignado a varios funcionarios.