



**UNIVERSIDAD TÉCNICA DEL NORTE**  
**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**  
**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN**  
**SISTEMAS COMPUTACIONALES**

**TEMA:**

**SISTEMA DE CONTROL DE PROCESOS ADMINISTRATIVOS PARA LA EMPRESA**  
**DE TRANSPORTES ANDINA COMPAÑÍA ANÓNIMA (TACA)**

**AUTOR: ALEJANDRO DIONISIO ALMEIDA ORTIZ**

**DIRECTOR: ING. PEDRO GRANDA**

**IBARRA – ECUADOR**

**2015**

## INDICE DE CONTENIDO

### Tabla de contenido

<b>1. INTRODUCCIÓN</b> .....	3
<b>2. Herramientas y Metodología</b> .....	4
<b>2.1. Enfoques de desarrollo de software.</b> .....	4
<b>2.1.1. Cascada:</b> .....	4
El modelo más elemental que ha servido de guía para la construcción de los otros paradigmas de desarrollo; teniendo en cuenta las siguientes actividades: Ingeniería y Análisis del Sistema, Análisis de los requisitos, Diseño, Codificación, Pruebas, Mantenimiento. ....	4
<b>2.1.2. Prototipado:</b> .....	4
A diferencia del incremental, ofrece al cliente un prototipo de cómo está quedando el proyecto al momento que se va desarrollando, pero de igual manera fragmenta en fases el proyecto general teniendo en cuenta las siguientes actividades: plan rápido, modelado rápido, construcción del prototipo, desarrollo; entrega y retroalimentación, comunicación, entrega del desarrollo final. ....	4
<b>2.1.3. Incremental:</b> .....	4
<b>2.1.4. Espiral:</b> .....	5
Este modelo trabaja en función del tiempo, cada vez que sus actividades cambian, su tiempo de ejecución varía; consta de 4 actividades que son las siguientes: determinar objetivos, análisis de riesgos, desarrollar y probar, planificación. ....	5
<b>2.1.5. RAD:</b> .....	5
<b>2.2. Patrones de diseño MVC</b> .....	6
MVC, es un concepto de programación, que, desde el aparecimiento de los paradigmas de programación y entre ellos el orientado a objetos, establece otros conceptos como clases, objetos, herencia, encapsulamiento, polimorfismo; que hoy en día no son usados al 100% en la creación de aplicaciones, pero su conceptualización es útil al momento de crear una estructura de manejo de datos o información. ....	6
<b>2.3. Herramientas de desarrollo</b> .....	6

## INDICE DE ILUSTRACIONES

Ilustración 1 Ciclo de Vida XP .....	5
Ilustración 2 Patrón MVC.....	6
Ilustración 3 Diagrama del Aplicativo por capas Fuente: Autor .....	7
Ilustración 4 Diagrama proceso Ingreso Inventarios.....	7
Ilustración 5 Diagrama Proceso Egreso Inventarios .....	7
Ilustración 6 Diagrama Proceso Cuentas por Cobrar .....	7
Ilustración 7 Diagrama Procesos Cuentas por Cobrar.....	7
Ilustración 8 Diagrama Proceso Guía de Encomienda.....	7
Ilustración 9 Modelo Conceptual Base de Datos TACA.....	7

## MANUAL TÉCNICO

### 1. INTRODUCCIÓN

La Empresa de Transportes Andina (TACA)<sup>1</sup>, conformada como Compañía Limitada en el año 1970; pasó a ser Compañía Anónima el 3 de agosto de 1978<sup>2</sup>, siendo fecha especial para la misma. Por tal razón, cada año se organizan festejos que permiten estrechar lazos de amistad y compañerismo. Así mismo es el momento propicio para reafirmar la convicción de ser mejores en el servicio de transporte público.

Siendo una de las empresas de transporte más sólidas, no solo en su estructura jurídica sino también en la fórmula de servicio moderno que ofrece a los miles de usuarios, que diariamente utiliza sus unidades por las carreteras del norte de país.

Luego de más de 3 décadas, significa cambios en el aspecto de modernización del servicio, seguridad y profesionalismo. En este lapso, los socios se han reocupado por ofrecer transporte de calidad; sin embargo, más allá de mantener unidades cómodas y seguras, los socios comprenden que el usuario merece consideración y buen trato, por ello ponen énfasis en cultivar excelentes servicios y relaciones humanas.

Con el avance en las TIC, cada vez los servicios que esta ofrece, son más útiles y fáciles de utilizar; no obstante los costos para adquirirlos, de igual manera son altos y en muchos casos no se encuentran disponibles en el país. Por esta razón muchas de las empresas, ya sean públicas, privadas o mixtas, crean sus propios métodos de administración de la información, ya sean aplicativos de escritorio, herramientas de ofimática<sup>3</sup>, incluso archivadores físicos<sup>4</sup>, que hacen que los documentos en algunas ocasiones viajen por toda la empresa sin tener un lugar destinado para su reposo.

La Empresa de Transportes Andina Compañía Anónima (TACA), no cuenta con una sistematización de procesos administrativos, sin aplicar una metodología de desarrollo de software, y a su vez el uso de software libre, lo cual acarrea algunas causas y efectos que se denotan a continuación.

---

<sup>1</sup> TACA: Transportes Andina Compañía Anónima.

<sup>2</sup> 1978: año en que se constituyó la empresa TACA; obtenida de la copia de escritura de constitución de compañías.

<sup>3</sup> Ofimática: palabra que describe a las herramientas informáticas de trabajo que se utilizan en una oficina, por ejemplo, office libre office, etc.

<sup>4</sup> Archivadores físicos: contenedores que almacenan documentos tangibles, para una persona u organización.

- La información se encuentra desorganizada, esto generaría una pérdida de registros.
- Los documentos al ser impresos, con el paso del tiempo se degradan y produciría una contaminación ambiental y al mismo tiempo un gran uso de papel.
- Los espacios físicos se ven reducidos por el amontonamiento de documentos tangibles, que en muchas ocasiones existen más de un duplicado.

Las herramientas que utilizaremos para el desarrollo del sistema serán de software libre en nuestro caso usaremos herramientas como:

- ECLIPSE
- JSF
- POSTGRES
- PRIMEFACES
- CSS

## **2. Herramientas y Metodología**

### **2.1. Enfoques de desarrollo de software.**

#### **2.1.1. Cascada:**

El modelo más elemental que ha servido de guía para la construcción de los otros paradigmas de desarrollo; teniendo en cuenta las siguientes actividades: Ingeniería y Análisis del Sistema, Análisis de los requisitos, Diseño, Codificación, Pruebas, Mantenimiento.

#### **2.1.2. Prototipado:**

A diferencia del incremental, ofrece al cliente un prototipo de cómo está quedando el proyecto al momento que se va desarrollando, pero de igual manera fragmenta en fases el proyecto general teniendo en cuenta las siguientes actividades: plan rápido, modelado rápido, construcción del prototipo, desarrollo; entrega y retroalimentación, comunicación, entrega del desarrollo final.

#### **2.1.3. Incremental:**

Fragmenta en partes o fases el proyecto general y se repiten las mismas actividades en cada fase siendo las siguientes: análisis, diseño, código, prueba.

#### 2.1.4. Espiral:

Este modelo trabaja en función del tiempo, cada vez que sus actividades cambian, su tiempo de ejecución varía; consta de 4 actividades que son las siguientes: determinar objetivos, análisis de riesgos, desarrollar y probar, planificación.

#### 2.1.5. RAD:

El desarrollo rápido de aplicaciones también conocido como RAD (en inglés rapid application development) es uno de los modelos para el proceso de desarrollo de software, diseñado por James Martin en 1980. Este método comprende el desarrollo iterativo, la construcción de prototipos y el uso de herramientas CASE (Computer Aided Software Engineering) y herramientas de rápido desarrollo. Hoy en día los desarrolladores de software suelen utilizar aplicaciones que permiten realizar de forma rápida y sencilla el diseño y codificación de interfaces gráficas de usuario. Algunas de las plataformas más conocidas son Visual Studio Net, Delphi, NetBeans, entre otros.[3]

El análisis se lleva a cabo evaluando los requisitos solicitados por usuario final, para de esta manera generar las historias de usuario y tareas pertinentes.

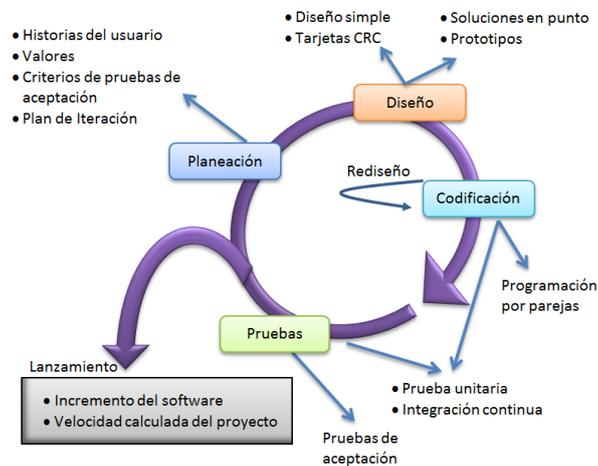


Ilustración 1 Ciclo de Vida XP

Fuente: Codejobs

Así también la elección de la metodología de desarrollo ágil; se opta o se declara que la metodología más óptima para esta investigación es la metodología XP, gracias a la reducción de tiempo en cuanto al desarrollo y planeación de actividades definidas, basándose en un patrón de desarrollo.

## 2.2. Patrones de diseño MVC<sup>5</sup>

MVC, es un concepto de programación, que, desde el apareamiento de los paradigmas de programación y entre ellos el orientado a objetos, establece otros conceptos como clases, objetos, herencia, encapsulamiento, polimorfismo; que hoy en día no son usados al 100% en la creación de aplicaciones, pero su conceptualización es útil al momento de crear una estructura de manejo de datos o información.

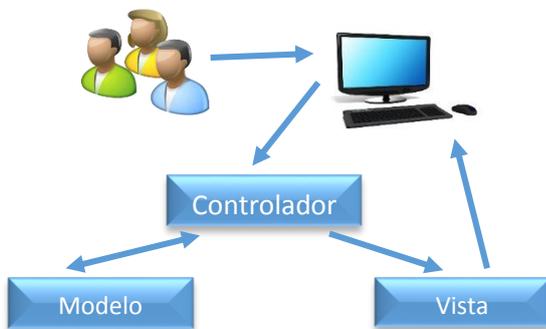


Ilustración 2 Patrón MVC

Fuente: Autor

MVC, es un patrón de diseño, que fractura el proyecto en 3 fases o módulos: modelo, vista, controlador; cuyos nombres fueron identificables gracias a sus conceptos, de tal manera que dan una claridad con respecto a las actividades en las que se va a enmarcar, para lograr el completo desarrollo del proyecto.

## 2.3. Herramientas de desarrollo

En el desarrollo de aplicativos, sistemas, y demás aplicativos, que ayudan en desempeño empresarial o institucional; no solo ofreciendo ahorro en tiempo, sino también ágil respuesta ante las peticiones del usuario, para poder tomar decisiones oportunas y de esta manera lograr que dicha empresa o institución, ya sea pública, privada o mixta, obtenga más beneficios y no solo los

---

<sup>5</sup> MVC: modelo vista controlador

ingresos aumentes, sino también sus clientes. Estas aplicaciones son desarrolladas en diferentes SW<sup>6</sup> de programación ya sean estas libres <sup>7</sup>o privativas<sup>8</sup>; estas a su vez le otorgan al programador diferentes formas de trabajo, así también limitadas formas de trabajo.

Para poder dar cumplimiento al desarrollo, se utilizaron las siguientes herramientas, con el fin de dar una solución óptima ya las necesidades de la empresa:

- **ECLIPSE:** Es una plataforma de desarrollo en Java, compuesta por una gama librerías y el conjunto de herramientas, que le permiten al desarrollador (programador) trabajar en un lenguaje de código abierto multiplataforma<sup>9</sup>a; proporcionando un IDE<sup>10</sup> que pone al frente de la plataforma toda su funcionalidad, permitiendo la facilidad de extenderse a otros lenguajes de programación, procesado de texto, aplicaciones de red y también a base de datos más conocidos como *Sistemas de Gestión de Base de Datos*.
- **JSF:** (Java Server Faces) Es una tecnología de desarrollo JAVA y a la vez un framework<sup>11</sup>, que reducen el desarrollo de interfaces de usuario, usando JSP<sup>12</sup> y de la misma manera soporta las nuevas incorporaciones de HTML.
- **JPA:** (Java Persistence API) Es una herramienta desarrollada para Java EE, que permite realizar la persistencia de datos relacionales directamente con las aplicaciones, ya sean estas de escritorio (*desktop*), también que estén disponibles en la Web, sea esta una red local o una red global.
- **POSTGRES:** Es un SGBD<sup>13</sup> relacional orientado a objetos y libre, publicado bajo la licencia BSD<sup>14</sup>(es la licencia de software otorgada principalmente para los sistemas BSD)[4]. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa y/o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre y/o apoyados por organizaciones comerciales. Dicha comunidad es denominada el PGDG [5]
- **PRIMEFACES:** Cuenta con un conjunto de componentes enriquecidos, encapsulados o disponibles en una librería, destinada para JSF; cómo es de código abierto este a su vez facilita la

---

<sup>6</sup> SW: Software (sistemas o aplicaciones que ofrecen ayudas al usuario)

<sup>7</sup> Libres: que no hay que pagar licenciamientos para su optimo uso.

<sup>8</sup> Privativo: el usuario debe adquirir una licencia por cierta cantidad de dinero para obtener una gran usabilidad.

<sup>9</sup> Multiplataforma: permite ser instalado o ejecutado en cualquier sistema operativo.

<sup>10</sup> IDE: Entorno de Desarrollo Integrado (Integrated Development Enviroment), es una aplicación informática con servicios integrales, que facilitan al programador el desarrollo de software.

<sup>11</sup> Framework: (marco de trabajo), un conjunto de conceptos estandarizados para resolver y enfrentar problemas en particular

<sup>12</sup> JSP: (Java Server Page), tecnología que permite hacer despliegues en páginas web

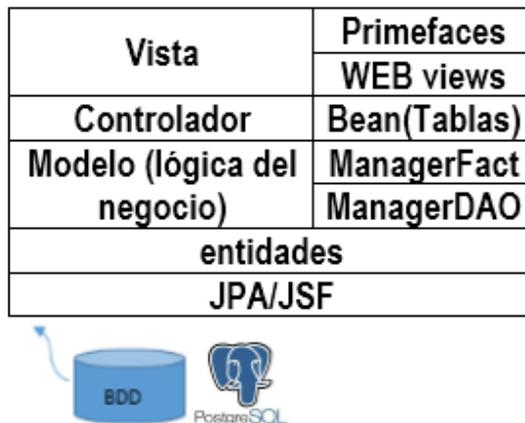
<sup>13</sup> SGDB: Sistema de Gestión de Base de Datos

<sup>14</sup> Berkeley Software Distribution

creación de aplicaciones web, amigables para el usuario con diseños o acciones innovadoras, permitiendo de esta manera, pueda hacer equipo en sí con otras librerías

### 3. PROCESOS Y AQUITECTURA

#### 3.1. ARQUITECTURA



*Ilustración 3 Diagrama del Aplicativo por capas  
Fuente: Autor*

El grafico por capas nos muestra una estructura MVC; esta a su vez, para entenderla respeta una jerarquía, para poder acceder a los datos y de la misma manera, para mostrar los datos (el análisis se lo llevará desde la base de datos, hasta las vistas de usuario; abajo – arriba), la cual cada capa trabaja de la siguiente manera:

- **BDD:** registra la información, cuando la transacción ha sido validada; esto quiere decir, que cumpla los requisitos solicitados dentro de la capa de **lógica del negocio** y la capa de **controlador**.
- **JPA / JSF:** tecnología en la cual se está desarrollando el aplicativo y pone en marcha el cumplimiento de las reglas establecidas dentro de las capas del MVC.
- **Entidades:** son clases dedicadas o clases privadas, en la cual se asignan los mismos atributos de cada tabla de la BDD, en donde se dará acceso al **controlador** y así mismo el **modelo** (lógica del negocio), obtendrá la información de la base de datos y son asignados a estas entidades; también se puede indicar que se realiza un manejo de objetos; es decir que las entidades toman la misma definición de programación orientada a objetos, en la cual los objetos y las entidades, manejan casi

la misma característica que tiene la POO<sup>15</sup>; algunos de estos son, atributos, accesorios y mutadores, manejo de datos mediante métodos.

- **Modelo (lógica del negocio):** este se sub divide en dos capas siendo: ManagerDAO, que es encargado de realizar la conexión directa a la base de datos, haciendo uso de procedimientos almacenados (SP<sup>16</sup>), más conocidos como CRUD's<sup>17</sup>; estos cuando son consultados en la base de datos, nos retorna una tabla del mismo tipo de dato (**entidad**) en la tabla que fue consultada la sentencia SQL<sup>18</sup>. Otra de las capas que existen en esta lógica es la ManagerFact, la cual se encarga de hacerle las peticiones al ManagerDAO y poder así realizar la transacción que le han designado y retornarlas a la **entidad** o directamente al **controlador**.
- **Controlador:** se encarga de recibir las peticiones de las vistas, que han sido pre programadas, para ser asignadas dentro de una clase, en variables del mismo tipo de datos y nombre tal y como fueron creadas en la base de datos.

Los procesos de administración son aquellos que permiten en una empresa planear, organizar, direccionar y controlar las funciones organizacionales que esta posea; convirtiéndose en una herramienta de gran ayuda, en la cual pueda realizar exitosamente todas sus actividades cotidianas.

Ahora se puede decir que: los procesos son un conjunto de actividades sencillas, rutinarias y descriptivas, que demuestran paso a paso el qué, cómo, dónde y cuándo se realizan las actividades en el trabajo diario. Por esta razón es indispensable la creación de manuales de funciones, procesos, calidad, políticas, organizacional, métodos, requerimientos entre otros, dependiendo de las necesidades del área, del departamento de la empresa o simplemente del departamento de Recursos Humanos.

### 3.2. Inventarios

El módulo de inventarios se encarga de manejar el ingreso y egreso de los artículos pertenecientes a la empresa T.A.C.A.; al poseer 2 acciones (ingresos y egresos), se divide en dos áreas de trabajo, permitiéndole desenvolverse como si fueran uno solo.

---

<sup>15</sup> POO: programación orientada a objetos.

<sup>16</sup> SP: por sus siglas en ingles Store Procedure o procedimientos almacenados.

<sup>17</sup> CRUD's: procedimientos almacenados que por sus siglas en ingles definen: Create, Research, Update Delete.

<sup>18</sup> SQL: es un lenguaje declarativo de acceso hacia las bases de datos

### **3.2.1. Ingreso de Inventarios**

En este Proceso se lleva un control exhaustivo, en cuanto a la administración de ingreso de artículos dentro de la empresa; contiene metodologías de control, para evitar fallos en cuanto a valores de registro.

### **3.2.2. Egreso de inventarios**

Este proceso permite llevar un control de la salida de artículos que posee la empresa dentro de sus inventarios, conteniendo eventos de control en cuanto a fallos, para evitar registros inválidos

## **3.3. Cuentas**

El módulo de cuentas permite crear, administrar y terminar las deudas que los clientes adquieren con la empresa (por cobrar) o las deudas que la empresa adquiere con los proveedores (por pagar). Al manejar dos estados, estos se dividen en dos submódulos que se revisan a continuación.

### **3.3.1. Por Cobrar**

Este submódulo permite administrar aquellas deudas que los clientes adquirieron con la empresa, lo que le permite a la empresa conocer el historial de pagos del cliente y determinar si su deuda puede aumentar o estar en espera hasta que finalice la deuda actual.

### **3.3.2. Por Pagar**

Este submódulo permite visualizar y registrar los pagos por completo o abonos a aquellas cuentas que se tiene con los proveedores.

## **3.4. Guías de Encomiendas**

Este módulo permite administrar, generar y supervisar, los envíos que realizan los clientes, a través del uso del servicio de transporte de mensajería, que ofrece la empresa T.A.C.A.

## **Proceso Ingreso Inventarios**

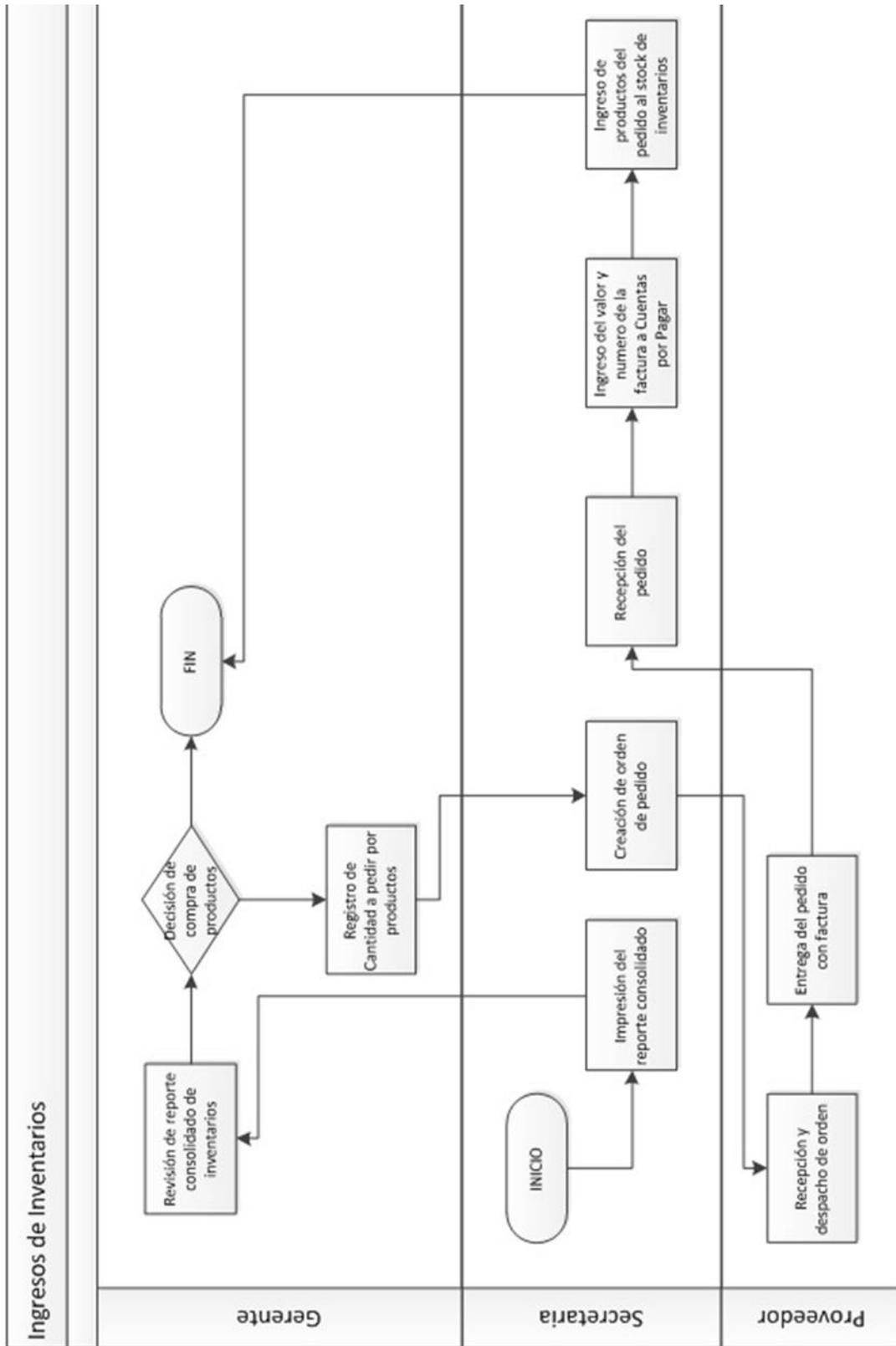
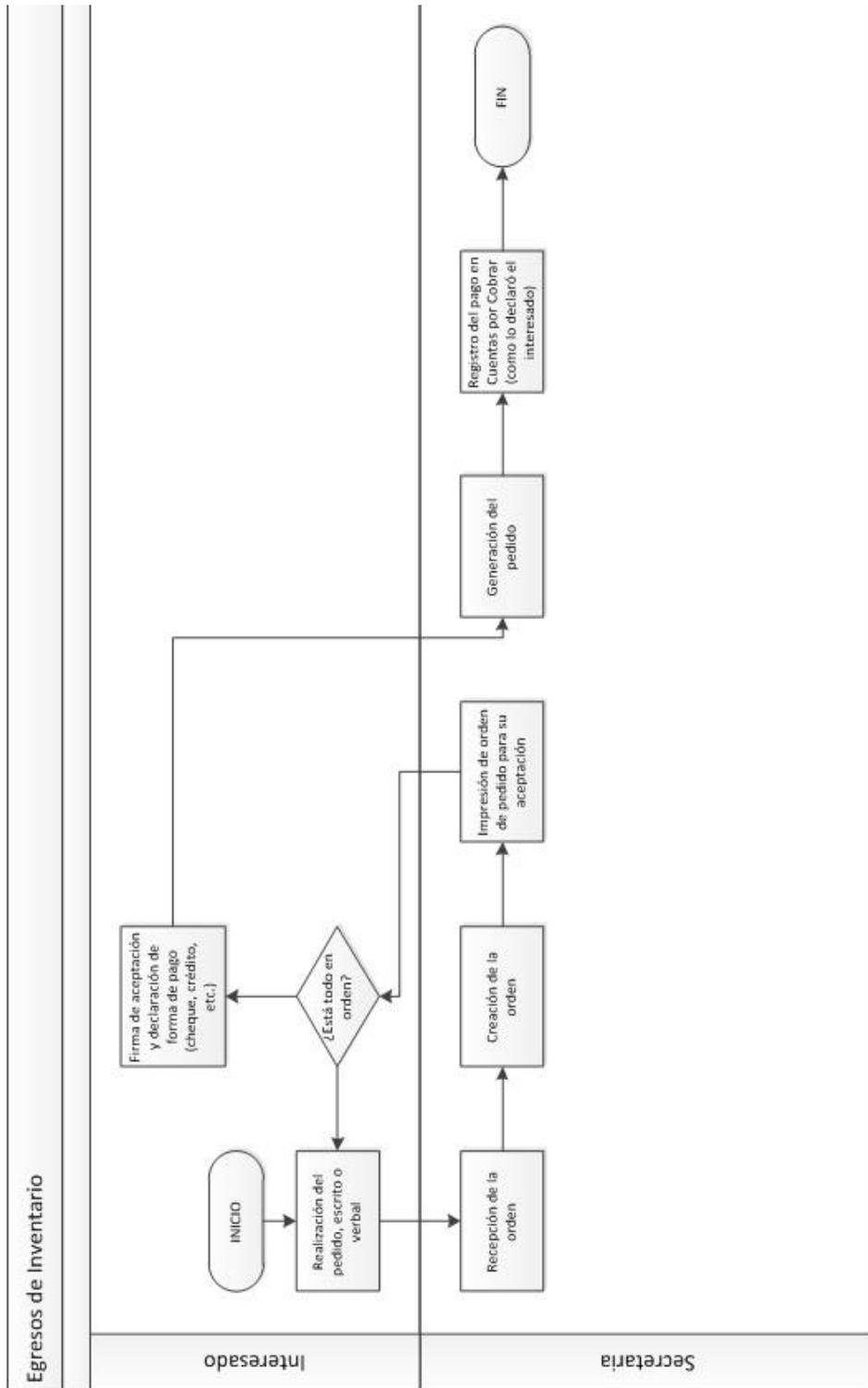


Ilustración 4 Diagrama proceso Ingreso Inventarios

Fuente: Autor



**Proceso Egreso Inventarios**



*Ilustración 5 Diagrama Proceso Egreso Inventarios*

*Fuente: Autor*

## Proceso Cuentas Por Cobrar

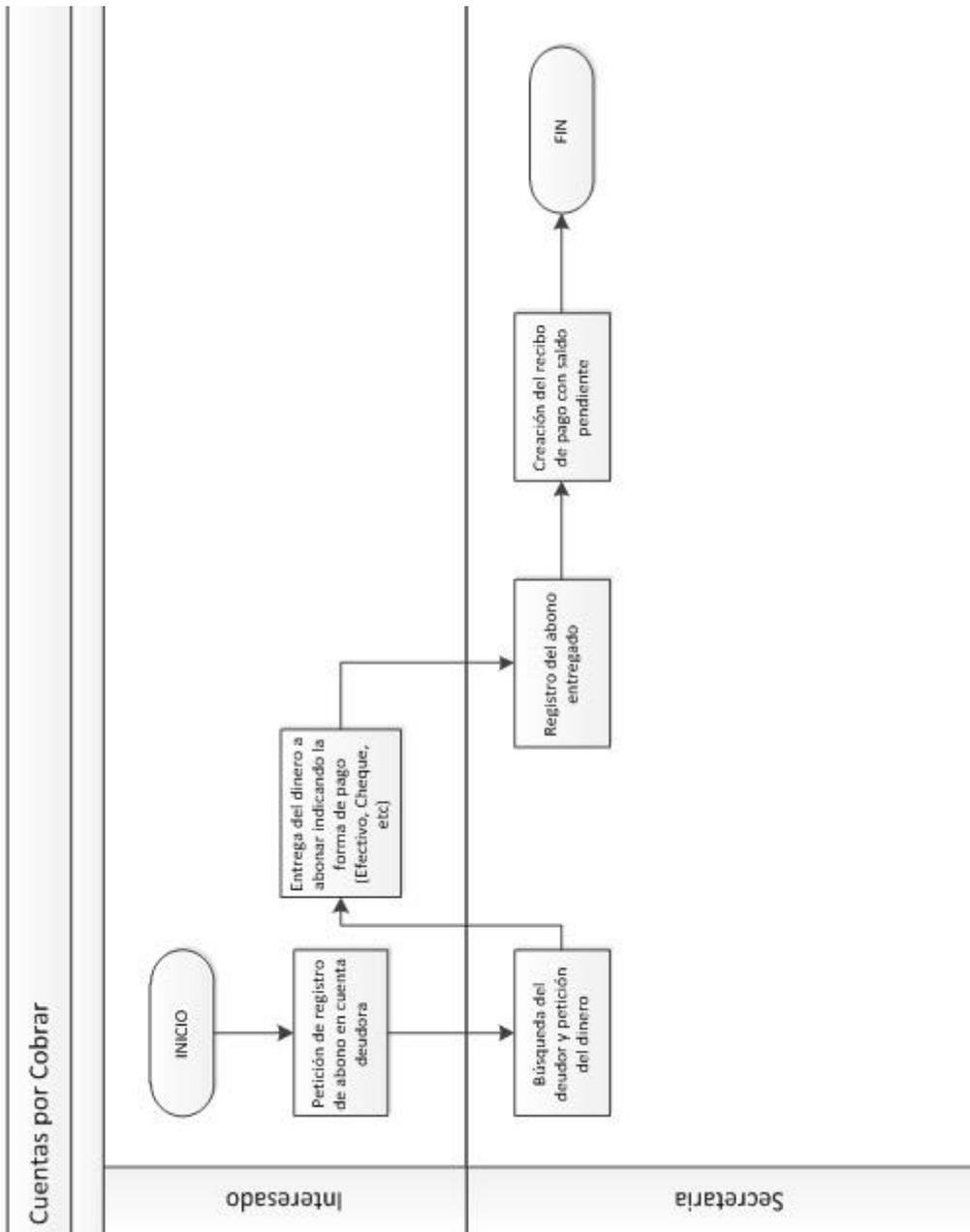
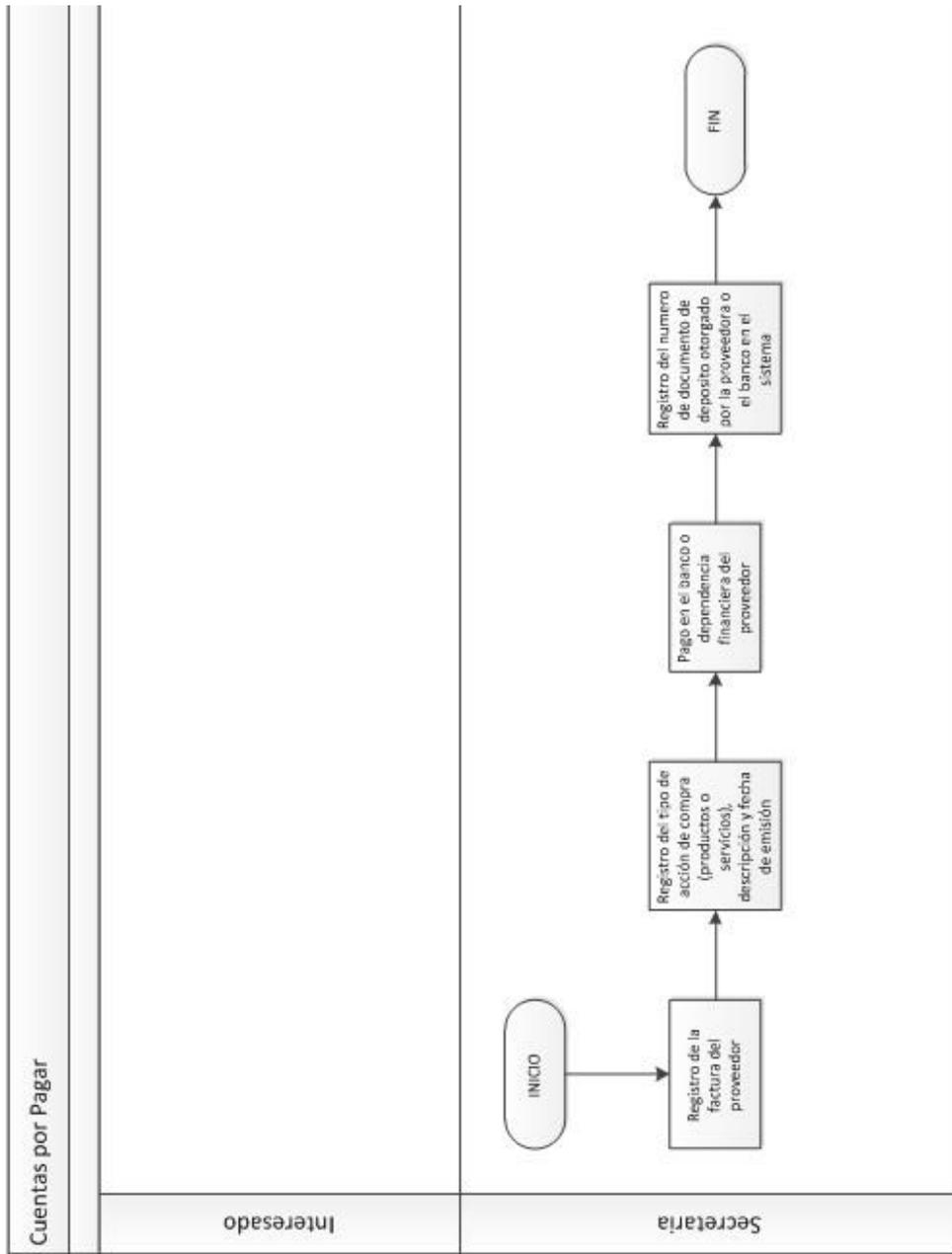


Ilustración 6 Diagrama Proceso Cuentas por Cobrar

Fuente: Autor

# Proceso Cuentas por Pagar



*Ilustración 7 Diagrama Procesos Cuentas por Cobrar*

*Fuente: Autor*

## **Proceso Guías de Encomiendas**

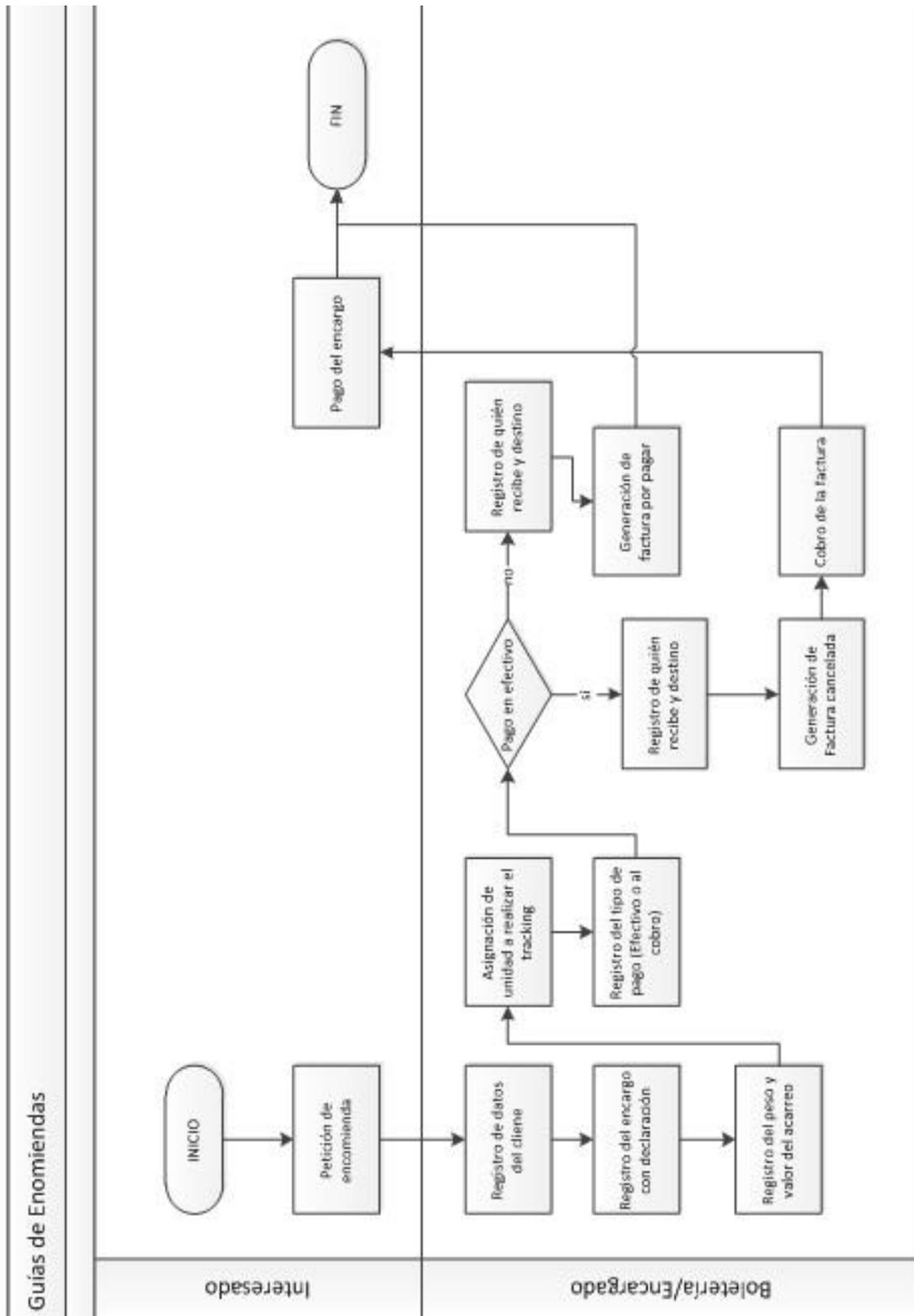


Ilustración 8 Diagrama Proceso Guía de Encomienda

Fuente Autor

**Modelo Conceptual**

**Base de datos (Transportes Andina C.A.)**

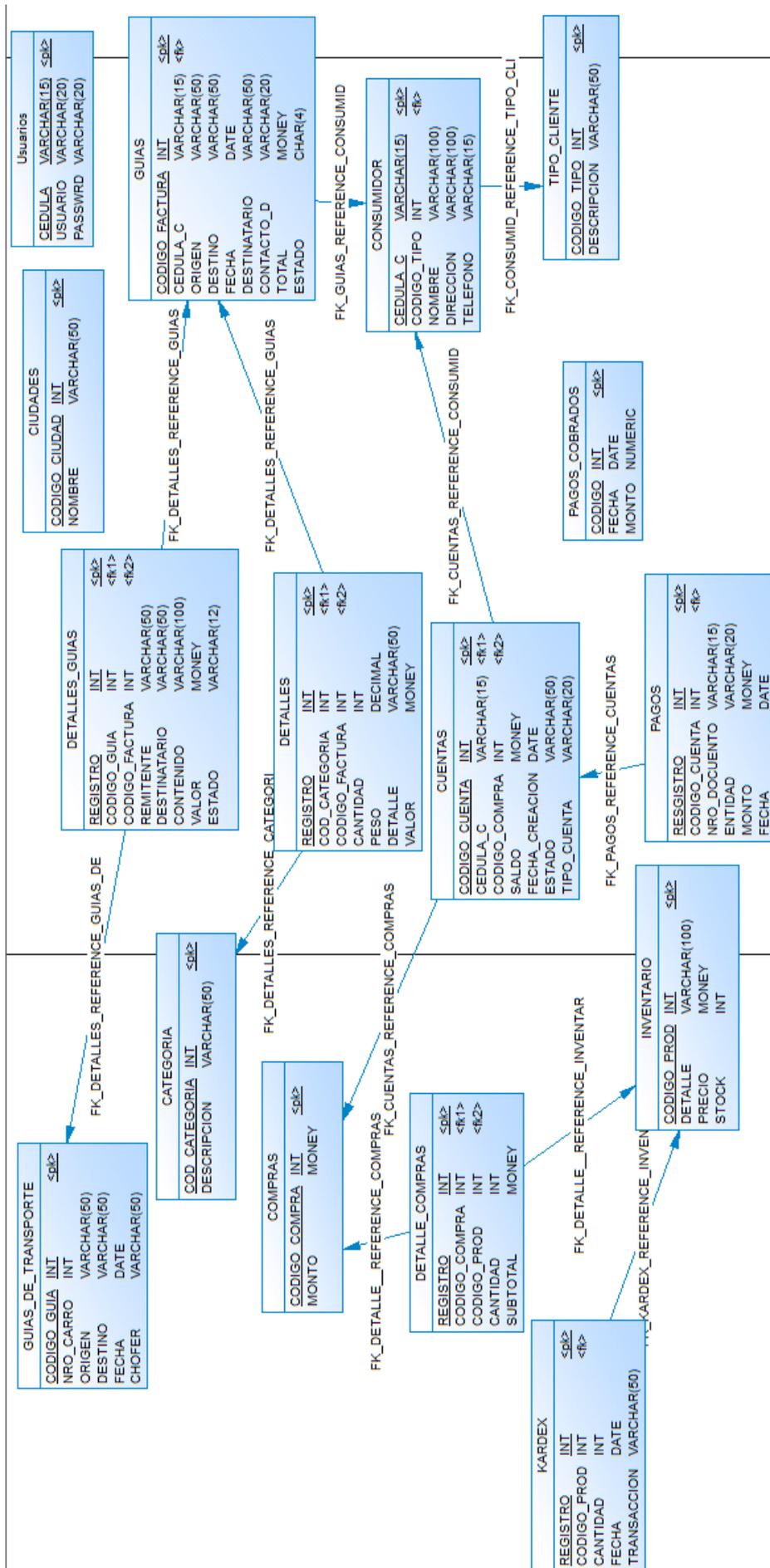


Ilustración 9 Modelo Conceptual Base de Datos TACA

Fuente: Autor

