



Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

Sistema Generador de Comprobantes Fiscales Digitales:
GeComFiDi

Tesis presentada como requisito para obtener el título
de:

Licenciado en Ciencias de la Computación

Presenta

José Luis Pérez Rendón

Asesor

Graciano Cruz Almanza

Fecha

Noviembre 2009

Índice general

Introducción	III
1. Antecedentes	1
1.1. Facturas	1
1.2. Descripción del problema	3
1.3. Objetivos.	4
1.3.1. Objetivos Generales.	4
1.3.2. Objetivos Particulares.	4
1.4. Justificación.	5
1.5. Alcances.	6
1.6. Breve introducción a los Comprobantes Fiscales Digitales.	6
1.7. Soluciones comerciales	15
1.7.1. SeguriData	15
1.7.2. sign@ture	15
1.7.3. Ekomercio, Levicom y Edicom	16
1.8. Solución propuesta	20
2. Conceptos Básicos	25
2.1. Estado de la empresa	25
2.2. Equipo actual	25
2.3. Software y Hardware con el cual se cuenta	26
2.3.1. Python, Datos Generales	27
2.3.2. Filosofía	28
2.3.3. Descripción	29
3. Diseño del Sistema GeComFiDi	31
3.1. Introducción	31
3.2. Conexión a la Base de Datos	31
3.2.1. Instalación de unixODBC	32
3.2.2. Instalación del Driver de iSeriesAccess For Linux	32
3.2.3. Configuración de unixODBC	33
3.2.4. Instalación de pyODB.	39
3.3. Requerimientos legales sobre comprobantes fiscales	40

3.3.1.	Formato electrónico único	40
3.3.2.	Generación de sellos digitales para comprobantes fiscales digitales . .	42
3.3.3.	Cadena Original	42
3.3.4.	Generación del Sello Digital	46
3.3.5.	Criptografía de la Clave Pública	47
3.4.	Algoritmo Propuesto	49
3.5.	Diagramas del sistema GeComFiDi	53
3.6.	Descripción funcional	57
4.	Implementación del Sistema GeComFiDi	61
4.1.	Descripción general del Sistema GeComFiDi	61
4.1.1.	La interfaz de usuario.	62
4.1.2.	Scripts en bash	63
4.1.3.	CorePythonCFD	64
4.2.	Diagrama de Bloques de ejecución del sistema GeComFiDi	68
4.3.	Descripción del código	69
4.3.1.	Creación archivo XML	69
4.3.2.	Creación de la Cadena Original y el Sello Digital.	72
4.4.	Desarrollo, Pruebas y Resultados	73
4.4.1.	Instalación y configuración del servidor de aplicaciones Linux.	73
4.4.2.	Actualizaciones de software en el servidor de aplicaciones Linux.	73
4.4.3.	Instalación del Sistema Generador de Comprobantes Fiscales Digitales.	73
4.4.4.	Estado Actual del Trabajo de Tesis.	74
5.	Conclusiones y Perspectivas	75
A.	Instalación básica de paquetes en GNU/Linux	79
A.1.	Los paquetes rpm	79
A.2.	Paquetes bin	80
A.3.	Paquetes tgz y deb	80
A.4.	Paquetes tar.gz o tar.bz2	81
B.	Python	85
B.1.	La instalación	86
B.2.	El intérprete	86
B.3.	Hola Mundo	87
B.4.	Tipos de Datos Basicos	88
B.4.1.	Cadenas de Caracteres	88
B.4.2.	Números Enteros	89
B.4.3.	Números Reales	89
B.5.	Operadores Básicos	89
B.5.1.	Operadores aritméticos	90

Introducción

Empacadora San Marcos como toda empresa, trata de lograr un alto desempeño y mejorar día a día en todas las áreas, en el caso del departamento de Tecnologías de Información en el año de 2002, inició un proceso de migración del software administrativo y contable de la empresa. En ese entonces se trabajaba con programas comerciales de la empresa ASPEL[11], los cuales están enfocados a la pequeña empresa y con una visión muy general, sus principales módulos son: “clientes y cuentas por cobrar”, “facturas”, “remisiones”, “pedidos”, “cotizaciones y devoluciones de venta”, “vendedores”, “inventarios”, “proveedores y cuentas por pagar”, “compras”, “órdenes de compra” y “devoluciones de compra”.

Sin embargo este tipo de software administrativo no era suficiente para cubrir las necesidades de la empresa, tales como los componentes de manufactura, la cadena de valor, el control de una nómina, control de los distintos almacenes tomando en cuenta fechas de caducidad.

De ahí surge la necesidad de mirar hacia el siguiente nivel de software administrativo, se analizan varios programas denominados ERP (Enterprise Resource Planning - Planeación de Recursos Empresariales) el cual debe de estar enfocado a la empresa manufacturera y debe cubrir eficazmente puntos muy importantes como:

- Administración de Datos de Producto.
- Planeación Maestra de la Producción.
- Planeación de Materiales.
- Control de Producción en Piso.
- Costeo Basado en Órdenes de Trabajo y/o Periodos.
- Administración e inventario de distintos materiales de manufactura.
- Reportes Diarios a Ventas.

Una vez analizado el conjunto de diferentes propuestas y la relación costo-beneficio, se opta por adquirir la solución denominada MAPICS¹, de la empresa INFOR² como el ERP que controlaría los procesos de la empresa aunque no se compraron las herramientas de desarrollo,

¹MAPICS ERP es un paquete de software utilizado para controlar las operaciones de las empresas. Su nombre es un acrónimo de Manufacturing, Accounting and Production Information Control Systems.

²Infor es una empresa de software enfocado a control de recursos <http://www.latinamerica.infor.com/>

Septiembre 2002.

Por otro lado se reforma al código de comercio en materia de comercio electrónico el cual regula la expresión del consentimiento a través de medios electrónicos, el documento electrónico y su valor probatorio, publicado en el Diario Oficial de la Federación (DOF) en Mayo de 2000.

El Sistema de Administración Tributaria (SAT) y la Asociación Mexicana de Estándares para el Comercio Electrónico (AMECE) publican el 5 de Enero de 2004 en el Diario Oficial de la Federación el marco legal y jurídico sobre el uso de los comprobantes fiscales digitales y se pone en marcha su implantación en México.

Al tener Empacadora San Marcos socios comerciales los cuales están impulsando fuertemente el proyecto de Comprobantes Fiscales Digitales, es el punto de partida para iniciar las medidas para afrontar este reto, ya que el ERP no cuenta con una solución para el nuevo proceso de Comprobación Fiscal Digital.

En este trabajo de tesis, se desarrolla un sistema que genera los Comprobantes Fiscales Digitales, denominados Facturas Electrónicas que da solución al problema que enfrenta Empacadora San Marcos ya que el sistema extrae la información de ventas almacenadas en distintas tablas en el ERP, la almacena en diccionarios de datos y posteriormente crea el archivo XML³ con todas las características oficiales que la Asociación Mexicana de Estándares para el Comercio Electrónico (AMECE) en conjunto con el Sistema de Administración Tributaria (SAT) impulsan en este nuevo esquema y dando cumplimiento a los requerimientos mínimos tales como:

- Contabilidad Automatizada y Simultánea.
 - *Electrónica:* Para poder realizar una Factura Electrónica, es indispensable llevar la contabilidad a través de medios electrónicos (software administrativo y contable), no libros de mayor como la contabilidad tradicional (manual).
 - *Simultánea:* Para elaborar una Factura Electrónica primero se debe afectar el registro contable de la venta y después se genera la factura o hacerlo en un plazo no mayor de 24 hrs. a partir de la generación de la factura. Garantizar que la fecha, hora, minuto y segundo en que se afecta el registro contable sea idéntico al de la Factura Electrónica.
- *Administración de llaves y certificados* Toda herramienta que ofrezca el servicio para la generación ó recepción de Factura Electrónica, debe contar con la capacidad de administrar:

³XML, siglas en inglés de Extensible Markup Language, lenguaje de marcas extensible, es un metalenguaje extensible de etiquetas

- Llave Privada ó “bolígrafo” con el que sellará digitalmente (firmar) las facturas. La llave privada es un archivo electrónico de 1024 bits con extensión *.key, formato definido en el estándar PKCS#8⁴ y está encriptada bajo las especificaciones del estándar PKCS#1. Este “bolígrafo” se obtiene a través de una aplicación desarrollada por el SAT.
 - Certificado de Sello Digital o “identificación oficial” que identifica a su propietario, indica que el SAT le faculta para emitir facturas electrónicas y lo asocia con una llave pública (elemento de seguridad con el que se verifica si una factura fue alterada). El certificado de sello digital es un archivo electrónico con extensión *.cer en formato X509⁵ V3 el cual es otorgado por el SAT.
- *Folios*. Partiendo de dos premisas básicas: 1) El contribuyente tiene la obligación de solicitar folios electrónicos al SAT, quien otorgará un número de aprobación por rango de folios otorgados, para utilizarlos en la Factura Electrónica los cuales inician con folio 1 y en caso de utilizar series deberán ser distintas a las manejadas en papel, 2) Se deberá validar que el folio utilizado en una Factura Electrónica corresponda al número de aprobación otorgado por el SAT así como garantizar la no duplicidad de folios.

Con esta dos premisas el contribuyente deberá: Preparar sus sistemas administrativos para registrar folios y series diferentes a las actuales La solución deberá hacer las validaciones pertinentes. (duplicidad de folios y rangos)

- *Generación de Firma Electrónica Avanzada*. El Sello Digital es una serie de datos, físicamente es una cadena de caracteres, que están asociados al emisor de la factura y a los datos de la misma. El sello digital es como el “garabato” que se plasma actualmente en el papel pero con más elementos de seguridad que permiten saber si los datos de una factura fueron alterados.
- *Formato XML*. Cualquier desarrollo para la generación e interpretación así como la conservación de un comprobante fiscal digital, como lo es la Factura Electrónica, debe considerar al formato XML como único para contener la información fiscal determinada por el SAT. Este esquema XML ha sido definido y publicado en el Diario Oficial de la Federación (DOF), en su anexo 20, el 1 de Septiembre de 2004.

⁴Public-Key Cryptography Standards. En criptografía, PKCS se refiere a un grupo de estándares de criptografía de clave pública concebidos y publicados por los laboratorios de RSA, El algoritmo fue descrito en 1977 por Ron Rivest, Adi Shamir y Len Adleman en el MIT; las letras RSA son las iniciales de sus apellidos.

⁵En criptografía, X.509 es un estándar UIT-T para infraestructuras de claves públicas (en inglés, Public Key Infrastructure o PKI). X.509 especifica, entre otras cosas, formatos estándar para certificados de claves públicas y un algoritmo de validación de la ruta de certificación.

- *Impresión de la Factura Electrónica.* En el Diario Oficial de la Federación publicado el 31 de Mayo de 2004, en la regla 2.22.8, se estipula que la impresión de la Factura Electrónica, además de los requisitos especificados en el artículo 29A del Código Fiscal de la Federación, deberá contener la cadena original, el número de serie del certificado de Sello Digital, el Sello Digital y la leyenda: “*Este documento es una impresión de un comprobante fiscal digital*” como se verá mas adelante.
- *Resguardo.* El resguardo de la información es vital; sin embargo, hay que asegurarse que se realice bajo los parámetros definidos por el SAT. Esto quiere decir que toda Factura Electrónica generada y/o recibida deberá guardarse en su formato original: XML.

Cumpliendo con los anteriores requisitos, en Empacadora San Marcos se inicia el desarrollo del sistema para generación de comprobantes fiscales digitales y entrar al nuevo esquema de Facturación Electrónica.

Finalmente, el presente documento se organiza de la siguiente manera: El capítulo 1 contiene una introducción al concepto de factura y su relación con los integrantes de un ejercicio de compra - venta, el fenómeno llamado comercio electrónico entre empresas, una descripción del problema que enfrentan las empresas mexicanas y los distintos organismos del gobierno mexicano tiene para apoyar a las empresas, también se mencionan algunas soluciones comerciales y sus requerimientos técnicos para conectarse al servidor o la forma en la cual reciben la información, el costo del producto o el costo por servicios y los distintos formatos soportados para el intercambio electrónico de información, por último en este capítulo se propone desarrollar una solución que mantenga la privacidad empresarial de información tan valiosa como lo son las operaciones comerciales y que cumpla con los requerimientos legales y fiscales.

El capítulo 2 contiene el estado en el que se encuentra la empresa en relación a las tecnologías de información, su acervo en software y el hardware con el cual se encuentra trabajando, también se presenta la propuesta para desarrollar la solución y los requerimientos necesarios para su implementación y una pequeña descripción de las características, filosofía y ventajas del lenguaje de programación en el cual será desarrollada dicha solución.

El capítulo 3 describe la configuración del servidor de aplicaciones con sistema operativo Linux, el módulo necesario para que el lenguaje de programación Python pueda usar la librería unixODBC⁶ y el driver de iSeries Access para unixODBC y lograr la conexión al servidor AS/400⁷ con sistema operativo OS/400⁸ V5R2 el cual tiene a DB2⁹ como manejador de la

⁶Open Database Connectivity - ODBC. Es un estándar de acceso a Bases de datos desarrollado por Microsoft Corporation, el objetivo de ODBC es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en inglés) almacene los datos.

⁷El sistema AS/400 es un equipo de IBM de gama media y alta, para todo tipo de empresas, Se trata de un sistema multiusuario, con una interfaz controlada mediante menús y comandos

⁸OS/400 es un sistema operativo utilizado en la línea de miniordenadores AS/400

⁹DB2 es una marca comercial, propiedad de IBM, bajo la cual se comercializa el sistema de gestión de base de datos.

base de datos donde se encuentra la información. En este mismo capítulo, se describen las características técnicas que debe tener un Comprobante Fiscal Digital, sus características y reglas de los distintos campos y la forma en la cual se aplican las herramientas de cifrado para el sello digital, la criptografía de la clave pública para mantener la seguridad y autenticidad de los datos; también se detalla el algoritmo para poder conectarse a la base de datos, extraer la información, normalizar los datos, confeccionar el archivo XML y sellarlo, de acuerdo a los lineamientos legales; para finalizar se describe como funciona el Sistema Generador de Comprobantes Fiscales Digitales.

En el capítulo 4 se describe la implementación del Sistema Generador de Comprobantes Fiscales Digitales, las características de la interfaz desarrollada y su facilidad de uso, la forma en la cual es ejecutado por el usuario y un diagrama de bloque que muestra el proceso de sigue el sistema, también se describe la forma como se crea el archivo XML con todos sus elementos básicos y la forma como se obtiene la cadena original la cual será usada para sellar y darle validez al documento fiscal.

Capítulo 1

Antecedentes

1.1. Facturas

Desde el inicio de la civilización, la necesidad de intercambiar productos entre distintas personas ha desarrollado un fenómeno interesante, desde el inicio del trueque, la introducción de la moneda, la creación de rutas comerciales y fenómenos como la creación de la banca, dieron la pauta para lo que hoy conocemos como comercio.

Con el paso del tiempo, la necesidad de reglas para hacer que el comercio fuese más eficiente dio lugar a la creación de instituciones para verificar que las partes involucradas cumplieran requisitos previamente definidos y agilizar el movimiento de los productos, así es como poco a poco se van integrando distintas herramientas para tener mayor control sobre las mercancías, el vendedor y el comprador.

Uno de esos documentos de control entre el vendedor y el comprador se denomina factura, en la real academia de la lengua, se define como:

***Factura:** Relación de los objetos o artículos comprendidos en una venta, remesa u otra operación de comercio.*

La factura relaciona los objetos que intervienen entre un ejercicio de comercio, los datos de quien la emite y de quien la recibe, fecha y lugar de expedición, el monto a pagar, los precios unitarios, los precios totales, los descuentos e impuestos y elementos que le dan validez fiscal como lo es un número de folio único, una impresión del registro federal de contribuyentes y un texto con el número de aprobación que la secretaria de hacienda denominada Cédula de identificación fiscal.[4]

Se le considera el máximo justificante fiscal de la entrega de un producto o productos que hace el vendedor (emisor) al comprador (receptor), compromete a entregar el bien o servicio y obliga a realizar el pago de acuerdo a lo que establece la factura emitida. Ambas partes utilizan la factura para comprobar la transacción ante las autoridades y las revisiones de au-

Concepto		Importe	
2	750123459012v Seminario de factura electrónica	\$1,500	\$3,000
1	750123456789v Asignación de Código de barras	\$1,500	\$1,500
1	750123456790v Inscripción Club aliados	\$5,000	\$5,000
SUBTOTAL			\$9,500
IVA			\$ 450
TOTAL			\$9,950

Forma de Pago: Banco: Batch:

Elaborado Revisado Autorizado

La reproducción no autorizada de este comprobante constituye un delito en los términos de las disposiciones fiscales.

— ORIGINAL —

Figura 1.1: Factura tradicional

ditoria internas, el original se le queda al receptor, mientras que el emisor guarda una copia para su uso interno.

Por muchos años, ese ha sido el método de facturación para cualquier operación de comercio en México, pero los tiempos cambian, la incorporación de la tecnología y la necesidad de tener la información en el momento que se requiera, ha llevado al uso de nuevos métodos para hacer mas ágil el intercambio de información entre los participantes de una operación de comercio.

Poco a poco el concepto de comercio electrónico que nace a mediados de la década de los 70's y en particular del denominado comercio electrónico entre empresas conocido por sus siglas en ingles como B2B (business to business) toma mas fuerza y relevancia, con la incorporación por parte de las empresas de mayor tecnología enfocada al área de la información y pasar paulatinamente de una administración basada en libros de diario a usar los primeros desarrollos de software para llevar el control, facilitando transacciones comerciales electrónicas, normalmente utilizando tecnología como la Electronic Data Interchange (EDI, Finales de 1970) para enviar electrónicamente documentos como pedidos de compra.

1.2. Descripción del problema

Con la rápida integración de las computadoras al entorno laboral y la necesidad de intercambiar datos cada vez más valiosos, surgen a nivel mundial organismos que se esfuerzan en regular los estándares sobre los cuales el comercio electrónico basará el nuevo concepto de Factura Electrónica.

Para impulsar la factura electrónica en México, se creó un grupo de trabajo coordinado por la Asociación Mexicana de Estándares para el Comercio Electrónico¹ y conformado por un grupo de emprendedoras compañías del sector privado (Benavides, Bufete Fernandez, Casa Saba, Cemex, Chedraui, Chrysler, Colgate Palmolive, Comercial Mexicana, Danone, Estafeta, Gigante, Gillete, Grupo Gamesa, HEB, Janssen Farmacéutica, Jugos del Valle, Jumex, Kraft, Foods México, La costeña, Levi Strauss, Liverpool, Masterfoods México, Palacio de Hierro, Playtex, Procter & Gamble, Productos de Maíz, Sigma Alimentos, Nadro, Nestle, Unilever, Volkswagen, WalMart), la banca (Banamex, Banco de México, Bancomer, Cecoban, Inverlat), la autoridad fiscal, otras asociaciones (AMIPCI, CANIETI) y empresas del sector tecnologías de información (Alestra, Comité de proveedores, Telmex).

Este comité impulsó ante diversas autoridades en México la modificación y creación de leyes que dan lugar a un marco legal y fiscal en el uso de documentos electrónicos como el código de comercio, código fiscal de la federación, resolución miscelánea fiscal, norma oficial mexicana para la conservación de mensajes de datos e Infraestructura Extendida de Seguridad (IES).

En México es el Servicio de Administración Tributaria,² engloba a la factura electrónica como uno de los diversos documentos denominados Comprobantes Fiscales Digitales³ y es la presentación digital de un comprobante fiscal que utiliza los estándares publicados en el anexo 20 de la Resolución Miscelánea Fiscal del Diario Oficial de la Federación⁴ en cuanto a forma y contenido, y es transmitido a través de medios electrónicos para generar y resguardar este tipo de comprobantes fiscales digitales.

La ley obliga a todos los contribuyentes a conservar las Facturas Electrónicas por un periodo mínimo de 5 años en el formato original para efectos fiscales y para efectos comerciales deberá conservarse 10 años observando las especificaciones de la Norma de conservación de Mensajes de Datos, ó impresas en el caso de los receptores que reciban la representación impresa del comprobante fiscal digital.

Empacadora San Marcos, al ser una empresa manufacturera de productos alimenticios, tiene entre sus socios comerciales a grandes cadenas comerciales y algunas de ella son parte del comité que impulsa el uso de los comprobantes fiscales digitales, los cuales advierten que

¹AMECE

²SAT

³CFD

⁴31 mayo 2004

en un futuro cercano será necesario realizar ejercicios comerciales usando el nuevo esquema ya que representa ahorros y un mejor manejo de la información. Así surge la necesidad de cumplir los requisitos de ellos y cumplir fiscalmente para poder tener un buen funcionamiento comercial, se estima que en un periodo de 24 meses el proyecto de los comprobantes fiscales digitales esté implantado en el 85 % de las empresas mexicanas.

Es momento de analizar los alcances y requisitos para entrar en el esquema de los comprobantes fiscales digitales dentro de Empacadora San Marcos, ver el software con que se cuenta y ver las distintas opciones comerciales que se adaptan a las necesidades, la relación costo-beneficio, se convoca a proveedores de soluciones para analizar sus propuestas y métodos para obtener los datos de nuestro servidor.

1.3. Objetivos.

En este trabajo de tesis, se propone desarrollar e implementar un Sistema Generador de Comprobantes Fiscales Digitales para la empresa Empacadora San marcos, bajo los lineamientos establecidos el SAT, la AMECE y el marco jurídico establecido por el gobierno Mexicano, empleando software libre así como los recursos físicos (hardware) con que la empresa cuenta.

1.3.1. Objetivos Generales.

Desarrollar una aplicación capaz de obtener la información de un servidor AS/400 para poder generar los Comprobantes Fiscales Digitales y entrar al nuevo esquema de Facturación Electrónica con los distintos socios comerciales.

1.3.2. Objetivos Particulares.

Para poder desarrollar e implementar el Sistema Generador de Comprobantes Fiscales Digitales, debemos contar con las herramientas adecuadas, así como el entorno de software necesario. Como objetivos particulares tenemos:

- Instalar OpenSuSE Linux como plataforma para desarrollar el Sistema Generador de Comprobantes Fiscales Digitales.
- Instalar y configurar Apache Web Server en el Servidor de Aplicaciones Linux como servidor Web.
- Instalar el lenguaje de programación Python.
- Instalar y configurar unixODBC para la conexión entre el servidor Linux y la base de datos DB2 alojada en el servidor AS/400.

- Instalar y configurar pyoDB, módulo de unixODBC para establecer conexiones en aplicaciones desarrolladas en Python.
- Instalar y configurar iSeriesAccess, módulo de unixODBC para establecer conexión a servidores iSeries de IBM.
- Configurar y verificar la conexión del servidor de aplicaciones Linux con el servidor AS/400.
- Programar script en Python que obtenga información de las tablas de la Base de Datos ubicadas en el servidor AS /400.
- Crear un script que normalice los datos que son necesarios para llenar los campos del archivo XML.
- Crear un script que genere el archivo XML.
- Crear un script que use el certificado digital del SAT para sellar el archivo XML y le validez.
- Desarrollar la interfaz Web para el Sistema Generador de Comprobantes Fiscales Digitales.
- Crear un script que envíe por correo electrónico el archivo XML.
- Crear un script para el respaldo mensual de los archivos XML.

1.4. Justificación.

El constante avance de la tecnología enfocada en procesos administrativos ajustándose a las necesidades empresariales y aun más en las necesidades comerciales, ha generado la necesidad de controlar la información de forma mas eficiente. Esta situación impulsa a las empresas a mantenerse actualizadas de acuerdo a los nuevos esquemas de control electrónico y del intercambio de información.

Las reformas en materia de comercio y delitos electrónico en el código fiscal de la federación publicado en el diario oficial de la federación para el comercio electrónico, se reforma al código de comercio en materia de comercio electrónico el cual regula la expresión del consentimiento a través de medios electrónicos, el documento electrónico y su valor probatorio, el Sistema de Administración Tributaria (SAT) y la Asociación Mexicana de Estándares para el Comercio Electrónico (AMECE) publican el 5 de Enero de 2004 en el Diario Oficial de la Federación el marco legal y jurídico sobre el uso de los comprobantes fiscales digitales y se pone en marcha su implantación en México.

1.5. Alcances.

El Sistema Generador de Comprobantes Fiscales Digitales esta diseñado para ser utilizado por el departamento de facturación de la empresa Empacadora San Marcos, el diseño con el que cuenta, esta basado en la infraestructura y en cubrir las necesidades específicas de la empresa.

1.6. Breve introducción a los Comprobantes Fiscales Digitales.

¿Qué es un Comprobante Fiscal Digital? Es la representación digital de un comprobante fiscal, que utiliza los estándares definidos por el el Servicio de Administración Tributaria (SAT) definidos en el Anexo 20 de la miscelánea fiscal del diario oficial de la federación en cuanto forma y contenido para contar con validez fiscal, garantizando la integridad, autenticidad y no repudio del documento.

Cuales comprobantes fiscales pueden ser Comprobantes Fiscales Digitales?. Enumeramos los comprobantes fiscales que son validos actualmente y que pueden entrar en el nuevo esquema de Comprobantes Fiscales Digitales.

- Recibo de honorarios
- Notas de cargo
- Carta porte
- Recibo de arrendamiento
- Nota de crédito
- Factura

Características de un Comprobante Fiscal Digital. Algunas de sus características son:

- Integridad: Garantiza que la información contenida queda protegida y no puede ser manipulada o modificada.
- Autenticidad: Permite verificar la identidad del emisor y el receptor del comprobante.
- No repudio: El emisor que selle digitalmente un comprobante no podrá negar la generación del mismo.

Objetivos del Comprobante Fiscal Digital. El objetivo principal de este esquema es otorgarle validez legal tributaria a la comprobación electrónica como medio de respaldo de las operaciones comerciales entre contribuyentes y reemplazar con ello los comprobantes de papel, la solución tecnológica debería estar al alcance del mayor número de contribuyentes posible sin importar su tamaño. Uso de estándares de operación que se sujetaran a ciertos lineamientos básicos, basado en la medida de lo posible en estándares abiertos y ajustarse a las necesidades y requerimientos de la legislación tributaria aplicable.

Marco jurídico. El marco jurídico que regula todo lo relacionado al nuevo esquema sobre los Comprobantes Fiscales Digitales fue aprobado y publicado en la pagina web del la cámara de diputados del H. Congreso de la Unión y publicado de manera oficial en el Diario Oficial de la Federación (DOF). A continuación se listan los artículos y anexos relacionados.

- Art. 29 en el Código Fiscal de la Federación, publicado en el Diario Oficial de la Federación Enero 5, 2004
- Regla 2.22.6 a la 2.22.11 Resolución de la Miscelánea Fiscal, publicada en el Diario Oficial de la Federación Mayo 31, 2004
- Anexo 20 de la Miscelánea Fiscal, publicada en el Diario Oficial de la Federación Mayo 31, 2004
- Regla 2.22.12 de la Miscelánea Fiscal, publicada en el Diario Oficial de la Federación Agosto 31, 2004
- Modificación al Anexo 20 de la Miscelánea Fiscal, publicada en el Diario Oficial de la Federación Septiembre 1, 2004

Beneficios del uso de los Comprobantes Fiscales Digitales. El uso de los Comprobantes Fiscales Digitales, trae beneficios tanto al emisor como al receptor de los comprobantes, como la reducción de tiempo de revisión de factura, reducción de las aclaraciones, reducción de tiempo de proceso de facturación, reducción de tiempos de recepción del pedido hasta el anaquel. Otros beneficios son:

- *Ahorro en costos:* evitar el uso de papel y las bodegas para almacenar los documentos, el ahorro en el uso de mensajera y el tiempo en llegar a destino el documento.
- *Productividad:* el uso de Comprobantes Fiscales Digitales incrementa el tiempo de respuesta para la cobranza, es confiable, reduciendo las falsificaciones, procesos administrativos más ágiles, confidencialidad y agilidad en el acceso a la información, recepción, envío oportuno de las facturas.
- *Cumplimiento Fiscal:* Mejor control sobre los documentos al realizar los reportes mensuales al sistema de administración tributaria, permite al fisco la posibilidad de desarrollar su función de fiscalización de manera simple, inmediata y de bajo costo, tiene exactamente la misma validez que las facturas en papel facilitando proceso de auditoria, búsquedas rápidas y en un mismo lugar.

Comprobantes Fiscales Digitales, Conceptos Básicos.

- **Certificado de Sello Digital:** El certificado de sello digital es un documento electrónico proporcionado por el SAT (*.cer), que está vinculado al certificado de Firma Electrónica Avanzada (FEA) y por lo tanto a la identidad del propietario, cuya función es habilitar al titular para emitir y sellar digitalmente Comprobantes Fiscales Digitales. Este elemento es muy importante revisarlo ya que en caso de que el certificado de sello digital NO sea vigente ó válido indica que el emisor no está habilitado para generar la factura electrónica y deberás rechazarla. El certificado de sello digital se comprueba a través de la página del SAT (<http://www.sat.gob.mx/>)
- **Firma Electrónica Avanzada (FEA):** Es un conjunto de datos que se adjuntan a un mensaje electrónico, cuyo propósito es identificar al emisor del mensaje como autor legítimo de éste y garantizar la identidad de quienes realizan transacciones electrónicas con el SAT, tal y como si se tratara de una firma autógrafa.

Por sus características, la FEA brinda seguridad a las transacciones electrónicas de los contribuyentes, con su uso se puede identificar al autor del mensaje y verificar no haya sido modificado. Su diseño se basa en estándares internacionales de infraestructura de claves públicas (o PKI por sus siglas en inglés: Public Key Infrastructure) en donde se utilizan dos claves o llaves para el envío de mensajes, y nos sirve para tramitar ante el SAT el Certificado Digital.

- **Información:** Los comprobantes fiscales digitales deberán cumplir con los requisitos señalados en el Código Fiscal de la Federación así como las reglas de la Resolución de la Miscelánea Fiscal. Se deberá revisar, tal y como se hace actualmente en las facturas en papel, que sean correctos los siguientes datos:
 - Datos del Emisor: razón social, domicilio fiscal y RFC⁵. En caso de haberse emitido en un local ó establecimiento indicar el domicilio del mismo.
 - Folio y en su caso serie.
 - Lugar y fecha de expedición. La fecha deberá incluir la hora minuto y segundo en el siguiente formato: aaaa-mm-ddThh:mm:ss
 - Datos del Receptor: RFC, en caso de ser un Comprobante Fiscal Digital global que amparen operaciones efectuadas con público en general se deberá utilizar el RFC genérico (XAXX010101000).
 - Descripción de la mercancía y unidad de medida.
 - Valor unitario, importe total y monto de los impuestos.
 - En caso de ventas de primera mano de mercancía de importación deberá llevar Número y fecha del documento aduanero así como la aduana.

⁵en México el RFC es una clave que tiene toda Persona física que realice alguna actividad lícita que este obligada a pagar impuestos o toda Persona moral sin excepción. A estas personas se les llama contribuyentes

- Número de referencia bancaria y/o número de cheque con el que se efectúe el pago (opcional).
- La cadena original con la que se generó el sello digital.
- Sello digital correspondiente al comprobante fiscal digital.
- Número de serie del certificado de sello digital.
- La leyenda.
- Número y año de aprobación de los folios.

También podrá llevar información que sea útil para la operación de cada negocio como lo son: Números de proveedores, órdenes de compra, contactos, códigos de barras, descuentos, etc.


- Cadena Original y Sello Digital: Se entiende como Cadena Original a la secuencia de datos formada con la información fiscal contenida dentro de cualquier Comprobante Fiscal Digital, la cual debe ser generada bajo las especificaciones establecidas por el SAT en el Rubro C de su Anexo 20 que se verá más adelante.

Se entiende como Sello Digital al conjunto de datos asociados al emisor y a los datos del documento por lo tanto es único e irrepetible por documento. Es el elemento de seguridad en una factura ya que a través de éste se puede detectar: si un mensaje ha sido alterado (integridad), determinar quién es el autor del documento (certidumbre de origen) y capacidad de impedir que el autor del sello digital niegue haber sellado el mensaje (no repudio). Ambos elementos deberán estar presentes en la representación impresa de la factura electrónica.

- Leyenda: Todas las facturas electrónicas que sean impresas deberán portar la leyenda **“Este documento es una impresión de un comprobante fiscal digital.”**

Características de un comprobante fiscal tradicional y un Comprobante Fiscal Digital A continuación se muestra un Comprobante Fiscal Tradicional de muestra y los elementos que se requieren para hacerlo legal de acuerdo con la legislación aprobada en el Código Fiscal de la Federación.

RECIBO DE HONORARIOS



ESTADO DE IDENTIFICACIÓN FISCAL
CANTONERÍA DE IDENTIFICACIÓN FISCAL
ESTADO DE RESULTADOS DE FILIACIÓN
CÓDIGO DE VERIFICACIÓN DE FOLIOS
CÓDIGO DE VERIFICACIÓN DE FOLIOS

Camilo Cruz Estrada Folio: **AA 0012** (I)

RFC CACE-830914-602 (II)
 ESPERANZA No. 16 COL JACARANDAS
 DELEGACIÓN IZTACALCO
 CP 51530. TEL: 55 49 30 01 MÉXICO DF
 LUGAR Y FECHA DE EXPEDICIÓN (III)

RFC (De la persona a favor de quien se expida) (IV)

(A)	Honorarios \$
Por concepto de: (V)	IVA \$
Importe total (VI)	Sub Total \$
	ISR Retenido \$
	Retención IVA \$
	Total \$ (VI)

(VII) IMPRESO POR OSCAR JIMÉNEZ ESTRADA JIED700119-470 CALLE LERDO No. 50 COL. GUERRERO DEL MIGUEL HIDALGO C.P. 08000 MÉXICO DF. TEL: 55899261

(C) FECHA DE INCLUSIÓN DE LA AUTORIZACIÓN EN LA PAGINA DE INTERNET DEL SAT 6 DE MARZO DEL 2002. COMPROBANTES ELABORADOS EL 8 DE MARZO DEL 2004

(D) LA REPRODUCCIÓN NO AUTORIZADA DE ESTE COMPROBANTE CONSTITUYE UN DELITO EN LOS TÉRMINOS DE LAS DISPOSICIONES FISCALES. VIGENCIA DEL 8 DE MARZO DE 2004 AL 8 DE MARZO DE 2006

(E) NÚMERO DE APROBACIÓN DEL SISTEMA DE CONTROL DE IMPRESORES AUTORIZADOS: 720026

(IX) "EFECTOS FISCALES AL PAGO"

"IMPUESTO RETENIDO DE CONFORMIDAD CON LA LEY DE IMPUESTO SOBRE LA RENTA"

Figura 1.2: Comprobante Fiscal Tradicional

Los elementos que hacen valido al comprobante fiscal que aparece en la imagen, son los siguientes:

- A. Cedula de Identificación Fiscal.
- B. Marco Jurídico sobre las copias no autorizadas de ducho comprobante fiscal.
- C. Fecha de la inclusión de la autorización de dicho comprobante.
- D. Fecha de la elaboración de los comprobantes fiscales.
- E. Número de aprobación por parte del SAT para el impresor autorizado asignado al conjunto de folios.

Los elementos requeridos especificados en el Código Fiscal de la Federación, enumeran las partes y responsabilidades que llevan a cabo dicha operación fiscal.

- I. Folio consecutivo y en su caso serie.
- II. Registro Federal de Contribuyentes de quien emite dicho comprobante fiscal.
- III. Lugar y fecha de expedición.
- IV. Registro Federal de Contribuyentes de quien recibe el comprobante fiscal.
- V. Conceptos por el cual se esta emitiendo dicho comprobante.
- VI. Importe Total con letra y desglosado de acuerdo a las leyes fiscales.
- VII. Número de Impresor autorizado.
- VIII. Vigencia del Comprobante Fiscal.
- IX. Leyenda “Efectos Fiscales al Pago”

Como agregados podrá llevar información que sea útil para la operación de cada negocio como lo son: números de proveedores, órdenes de compra, contactos, códigos de barras, dirección de entrega, descuentos, etc.

El nuevo esquema de los Comprobantes Fiscales Digitales agrega nuevos elementos los cuales le dan la validez y seguridad a dichos comprobantes y elimina algunos de ellos.

Análisis del Comprobante Fiscal Digital. A continuación se muestran los componentes de un Comprobante Fiscal Digital con la información requerida por la Asociación Mexicana de Estándares para el Comercio Electrónico.

AMECE, A.C. RFC: AME 980602 7F9 Domicilio Fiscal: Boulevard Avila Camacho 138 Piso 9 Col. Lomas de Chapultepec C.P. 11000		No. Factura: A0001
		No. de aprobación: 20051
		No. Certificado : 000010000000000000104
Fecha : 5 de Enero de 2004 Lugar de Expedición: Mexico DF Cliente: Juan López Pérez RFC: JULP 740127 T78 Dirección: Revolución 567 Tacubaya	# O.C: 3456 No. Proveedor: 11111 Descuentos: 3%	
Cantidad	Descripción	Valor Unitario
2 cajas	Papelería	\$100.00
		Importe
		\$200.00
	Sub-Total	\$200.00
	Iva	\$30.00
	Total	\$230.00
Cadena Original		
A 001 20040105 20051 AME9806027f9 AMECEAC boulevardAvila Camacho 138 9 Lomas 000 DF Mexico JULP740127t78 ..		
Sello digital		
apqoeurnsmxncla34spdurt3532osjdmxdnjcsdj3fqfiehdfh5 adifjdfueifui7fujsikdfdf8aifsijsfidsfjoiuifheifjeifwf.....		
"Este documento es una impresión de un comprobante fiscal digital"		

Figura 1.3: Elementos de la Factura

1. Información especificada en algunas fracciones del Art. 29 del Código Fiscal de la Federación.
2. Certificado de Sello Digital.
3. Número de aprobación así como los folios y en su caso serie
4. Sello Digital.
5. Formato electrónico regulado por el SAT (XML)
6. Cadena Original y leyenda en caso de impresión.

Obligaciones de los Emisores de Comprobantes Fiscales Digitales.

- Reportar al SAT mensualmente información de los comprobantes fiscales digitales emitidos.
- Conservar en medios electrónicos actualizados los CFD y tenerlos a disposición de las autoridades fiscales
- Proveer a la autoridad de una herramienta de consulta de la información de los CFD
- Expedir comprobantes fiscales digitales globales que amparen las operaciones efectuadas con el público en general (a través del RFC genérico XAXX010101000.)

Obligaciones de los Receptores.

- Verificar ante el SAT los datos de los CFD recibidos.
- Conservar en medios electrónicos actualizados los CFD recibidos.
- Proveer a la autoridad de una herramienta de consulta de la información de los CFD.

Módulo Operativo. A continuación se muestra el esquema operativo entre el emisor y el receptor de los Comprobantes Fiscales Digitales.



Figura 1.4: Esquema Operativo

El esquema operativo para el envío y recepción de los comprobantes Fiscales Digitales es el siguiente: El emisor debe contar con sus folios, números de aprobación y el certificado digital

correspondiente, se genera la factura de modo tradicional en el sistema electrónico de contabilidad, se toma esa información y se crea el documento XML, se le extrae la cadena original y se usa el certificado digital para sellar el archivo XML, resultado así el archivo XML valido fiscalmente, es decir, ya es un Comprobante Fiscal Digital. Este documento es almacenado por el emisor por un periodo de 5 años por cualquier aclaración fiscal.

Al receptor se le hace llegar el documento de forma electrónica o magnética e incluso impresa, para la integración de sus procesos de contabilidad, el cliente tiene la obligación de verificar los datos de la factura como lo son el Folio, Serie y Vigencia del Certificado.

Requerimientos para una solución de Factura Electrónica. Requerimientos obligatorios, que deben cumplir las soluciones comerciales de generación de Comprobantes Fiscales Digitales.

- Integración.
- Administración de Certificado y Llaves Privadas.
- Generación de la Firma Electrónica
- Formato XML.
- Reporte Mensual.
- Impresión del Comprobante Fiscal Digital.
- Resguardo.
- Estándar para Datos Comerciales.
- Confidencialidad.
- Validaciones.
- Acuses Funcionales.

1.7. Soluciones comerciales

En México, la AMECE se encarga de evaluar las soluciones comerciales. AMECE no busca “certificar” a ninguna empresa, únicamente se está revisando que las soluciones cumplan con lo mínimo indispensable. Así mismo, esta actividad de revisión no está ligada ni soportada por el SAT, es una labor propia de AMECE que tiene el afán de brindar al mercado en general un servicio de valor.

Por otro lado, el proceso de revisión no asegura el correcto funcionamiento de las soluciones en las empresas. De igual forma, la estabilidad, comportamiento, soporte técnico y adecuación, entre otros; de cualquier solución revisada, es responsabilidad del proveedor, por lo tanto, AMECE no se hace responsable de los puntos mencionados.

Algunos de esos proveedores fueron: seguridata, sign@ture, ekomercio, levicom, edicom.

1.7.1. SeguriData

Es una compañía mexicana que ofrece soluciones de seguridad electrónica creadas por un equipo formado por expertos de este campo en México con la tecnología criptográfica de vanguardia y apegadas en todo momento a los estándares internacionales en el Intercambio Electrónico de Datos, Electronic Data Interchange for Administration, Transport and Commerce EDIFACT apoyado por la Organización de las Naciones Unidas ONU [9]

La solución tiene una gran integración a otras aplicaciones como lo son los ERP, software de contabilidad y aplicaciones propietarias a través de las APIS (Interfaz de Programación de Aplicaciones).

La solución ofrecida por **seguridata**, era un desarrollo en lenguaje RPG, nativo del AS/400, pero al no contar la empresa con herramientas de desarrollo, era una solución con un alto costo y sumando costo a largo plazo por distintas modificaciones o actualizaciones, era una solución altamente costosa[10].

1.7.2. sign@ture

Es una aplicación para el sistema operativo Windows construida en lenguaje C# sobre la plataforma .NET de Microsoft. Basada en servicios Windows, la arquitectura permite una integración al sistema operativo, haciendo susceptible a ataques propios de dicha plataforma.[5]

La solución presentada por la empresa **sign@ture**, consiste en dos productos o componentes que son Sign@ature xDoc y Sign@ature Server. Cada uno tiene funcionalidades bien específicas y son complementarios en el sentido de que uno interactúa con el otro para solucionar la problemática de emisión y recepción.

Sign@ature xDoc es el producto que interactúa con el sistema emisor de documentos (ERP, Facturador, etc). La gran ventaja que tiene es que ofrece una forma natural de integrarse con el sistema emisor de documentos que es a través de impresoras virtuales. Esta forma de integración permite que el impacto en los procesos tecnológicos y de negocios sean mínimos, reduciendo costos y tiempo total en el proyecto de implantación. Utilizando la herramienta Sign@print, es posible tomar cualquier spool de impresión o cola de impresión de un documento con cierta estructura base y convertirlo a un formato estándar XML previamente definido. De esta forma no se requiere ningún desarrollo adicional en el sistema emisor de documentos.

Sign@ature Server es el producto que constituye el repositorio final de los documentos electrónicos. Una vez que Sign@ature xDoc procesa los documentos, estos son enviados a los receptores finales, esta segunda parte de la solución, cuenta con una interfaz web que permite administrar los documentos así como hacer búsquedas sobre los documentos.

En general, se trata de leer el archivo spool, un archivo especial donde se encuentran todas las impresiones que se han realizado, en consecuencia, el spool de impresión o cola de impresiones es un archivo que crece en grandes proporciones y es necesario que se depure frecuentemente para evitar llenar el espacio en disco duro, cada usuario tiene su propio spool de impresión, por lo tanto a cada usuario que realiza el proceso de facturación hay que instalar la impresora virtual y así poder leer el spool correspondiente a los usuarios de los distribuidores y los usuarios que facturan en planta.

El proceso de tomar los datos del spool de impresión que propone esta solución, no es adecuado ya que en el spool contiene todos los datos de la factura tradicional y uno de los grandes problemas es coleccionar los datos que una factura electrónica tiene, ya que están dispersos en distintas tablas de la base de datos y no aparecen en el spool de impresión o cola de impresión.

1.7.3. Ekomercio, Levicom y Edicom

Las soluciones propuestas por ekomercio, edicom y levicom, presentan similitudes, ellos toman un archivo de texto plano, el cual debe contener todos los datos para poder generar la factura electrónica y una computadora donde se ejecutará una pequeña aplicación. A continuación se describen.

Ekomercio

La solución presentada por la empresa ekomercio, consiste en una aplicación instalada en una computadora la cual recibirá un archivo de texto plano, lo enviará a su red denominada Ekonet y ahí se procesará el archivo para obtener el XML sellado, posteriormente ellos lo entregarán a su respectivo destino, Ekomercio proporciona por un costo extra un buzón para almacenar una copia del archivo XML y del archivo de texto plano[12].

Los archivos de texto plano pueden contener uno o más comprobantes fiscales digitales y deben ser generados, sea con el mapa de caracteres UTF-8 o bien, con el mapa ASCII estándar, se utilizan separadores para sus secciones y campos. No es posicional y por eso no se especifica la longitud. Cada archivo de texto plano puede contener uno o más comprobantes. Se utilizará el carácter tilde (~ carácter ASCII 126) para distinguir un comprobante de otro dentro del archivo generado. Se utilizará la barra vertical o pipe (| carácter ASCII 124) para separar un campo de otro. Cada comprobante tiene tres secciones de campos:

1. Sección de campos del encabezado.
2. Sección de campos del detalle. Cada línea de detalle será separada por un retorno de carro (carácter ASCII 13 + carácter ASCII 10).
3. Sección de campos de los impuestos. Cada línea de impuesto será separada por un retorno de carro (carácter ASCII 13 + carácter ASCII 10).

Cada grupo de campos será distinguida por el carácter ¬ (código ASCII 172).

El archivo de texto plano se asemejará al formato que se muestra a continuación:

```

~[Folio] | [Nombre_Emisor] | [RFC_Emisor]
[Dom_Emisor_calle] | [Dom_Emisor_noExterior]
¬[Linea_Descripcion] | [Linea_Cantidad] | [Linea_Unidad]
¬[Linea_Descripcion] | [Linea_Cantidad] | [Linea_Unidad]

```

Características del archivo de texto plano.

Una vez confeccionado el archivo de texto plano, la aplicación se comunicara con el software alojado en su servidor denominado EDIXCFD y es donde se procesa el archivo de texto plano, es decir se crea el archivo XML, se sella y se envía al destinatario, también se realiza un resguardo y almacenamiento de los archivos XML siendo este caso un costo adicional a la licencia de uso del software y el precio anual por el resguardo de la información es variable y se basa en el número de documentos procesados en un mes, así el costo varía desde \$ 2.5 pesos por documento hasta \$ 0.50 pesos.

Al ser una aplicación instalada en una computadora, se requieren cumplir ciertas características ó requerimientos técnicos:

- El archivo en texto plano con todos los elementos.
- Sistema operativo, Windows 2000 o Windows XP.
- 128 Mb de memoria RAM.
- Espacio mínimo en disco duro 15 MB (necesario solo para la instalación).
- Acceso a Internet.
- Tener instalado el Microsoft .NET Framework 1.1.

Levicom

Levicom, una empresa 100 % mexicana, desarrolló una solución que consta de dos partes, una pequeña aplicación llamada **EDC invoice** que se encarga de recibir archivos de texto plano la cual se comunica con la segunda y más importante que es el portal **Monitor** desarrollado para que los usuarios puedan revisar el estado del documento.

EDC invoice es una aplicación desarrollada para el sistema operativo Windows y usa de manera interna Microsoft Access como motor de base datos para los folios y series que se generen, por estas características se puede instalar en un servidor compartido, EDC Invoice recibe el archivo de texto plano, almacena su folio, su serie y lo envía al servidor de Levicom para generar el archivo XML, sellarlo y enviarlo a su destinatario. Levicom proporciona un buzón para almacenar los documentos.

Algunas características de los archivos de texto plano que Levicom propone son: archivos de texto plano en codificación UTF-8, utilizan separadores para sus secciones y encabezados para los campos. No es posicional y por eso no se especifica la longitud. Cada archivo de texto plano contiene solo un comprobante.

El archivo de texto plano deberá tener el siguiente formato que se muestra a continuación:

```
H1| [serie] | [folio] | [nom_aprob] | [Date_Time_stamp] ...
H2| [nomre_empresa] | [rfc] | [cp] | [domicilio] | [n_int] ...
H4| [receptor] | [rfc_receptor] | [GLN] | [domicilio] ...
H5| [lugar_entrega] | [rfc] | [GLN] | [domicilio] ...
D| [EAN-UPC] | [descripción] | [cantidad] | [unidad_medida] ...
```

Características del archivo de texto plano para Levicom.

El costo del buzón que proporciona Levicom, tiene dos modalidades, el primer caso es por el número de facturas que se realicen al mes:

Lista aproximada de precios:

1 - 100	\$ 5.00
101 - 300	\$ 4.20
301 - 700	\$ 3.80
701 - 1,500	\$ 3.40
1,501 - 4,000	\$ 3.00
4,001 - 9,000	\$ 2.60
9,001 - 20,000	\$ 2.20
20,001 - 50,000	\$ 1.80
50,001 - mayor	a cotizar

El segundo caso está diseñado para empresas que conocen el número de facturas que emiten anualmente, esto permite hacer un pre pago por un número determinado de documentos los

cuales pueden ser utilizados en un período de 365 días.

Lista aproximada de precios:

100	\$ 500.00
300	\$ 1,260.00
700	\$ 2,660.00
1,500	\$ 3,600.00
4,000	\$ 12,000.00
9,000	\$ 23,400.60
20,000	\$ 44,000.00
50,000	\$ 90,000.00
Más de 70,000	a cotizar

Edicom

La empresa Edicom tiene un portafolio de soluciones dedicado al desarrollo e implantación de sistemas transaccionales de alto rendimiento empresa a empresa (B2B), ofreciendo así dos soluciones para la facturación electrónica. Edicom una empresa especialista en consultoría y desarrollo de software EDI⁶ e integración de datos, aplicaciones y procesos, dispone de su propia Red de Valor Añadido, para el transporte seguro de la información, así como de soluciones en el campo del Reaprovisionamiento Continuo y Trazabilidad.

Dentro de la amplia gama de soluciones, se cuenta con:

- EDIWIN4 JAVA Client, es un programa desarrollado en lenguaje JAVA que se descargará la primera vez que usted vaya a utilizarlo y que posteriormente se guardará en el disco duro de su computadora para evitar nuevas descargas y agilizar de esta forma su trabajo. EDIWIN4 JAVA Client es una solución de bajo costo y alta fiabilidad para el intercambio electrónico a través de Internet. EDIWIN4 Java Client, es un programa de software que permite intercambiar documentos electrónicos a través de Internet con la mayoría de clientes o proveedores sin tener que preocuparse por cuestiones de administración de Bases de Datos, comunicaciones, ya que solo es un cliente. EDIWIN4 Java Client se conecta al servidor EDIWIN4 Java Server mantenido por EDICOM. Este servidor es el que soporta la BD central y los procesos de creación, sellado y envío de los documentos.
- EDIWIN4 para entornos WINDOWS, es un software que permite realizar, de forma muy sencilla, comercio electrónico empresa a empresa, mediante la transmisión de documentos comerciales a través de medios telemáticos (EDI - Electronic Data Interchange), aprovechando para ello las últimas tecnologías del momento. Listo para su uso en SAP R/3, Baan, Navision, Dimoni, SIGES, SP Editores, JD Edwards, AXAPTA, etc,

⁶Electronic Data Interchange, es un software que permite la conexión a distintos sistemas empresariales como ERP o CRM.

soporte a múltiples centros servidores: SEDEB2B, TEE, IBM, AZERTIA, ATLAS400, ALLEGRO, PORTEL, X400, AS2, etc.

En particular la solución EDIWIN4 para entornos WINDOWS es la que se acomoda a las necesidades de la empresa, ya que puede integrarse al ERP o tomar un archivo de texto plano para confeccionar el archivo XML. EDIWIN facilita al usuario el manejo de los documentos enviados y recibidos vía EDI, proporcionando listados comerciales y pantallas de datos en formato legible y una serie de operaciones para la gestión de los mismos. De esta manera el usuario no debe conocer los estándares (EDIFACT, XML) en los que fueron transmitidos los mensajes.

De la misma forma también dispone de una vista avanzada en formato EDIFACT o XML que permite realizar cambios de bajo nivel sobre los documentos. Esta edición después se compila e indica si se ha cometido algún error. La impresión o previsualización también se puede hacer mediante informes específicos para ciertos mensajes de tal forma que la complejidad del código EDIFACT o XML original queda oculta al usuario. EDIWIN permite importar y exportar documentos desde diferentes formatos: EDIFACT, ODETTE, XML, X12 y específicos de usuario.

Al departamento de Tecnologías de Información, se le asignó la tarea de analizar y probar las propuestas de las diferentes empresas mencionadas (SeguriData, sign@ture, Ekomercio, Levicom y Edicom). Después de analizar y probar las propuestas y revisar los datos que contiene una factura tradicional y comparar los datos que requiere una factura electrónica en campos como la ADDENDA,⁷ la dirección administrativa de Empacadora San Marcos tomando como base la información de los distintos consultores se da cuenta que no cumple los requerimientos del SAT para generar los comprobantes fiscales digitales. Por lo tanto no es viable usar una solución genérica ya que no nos permite moldear procedimientos de facturación actual o tradicional para empatar datos, completar registros, modificar estructuras internas del software de facturación y hacer coincidir los elementos totales para generar los comprobantes fiscales digitales, aunado a la falta de herramientas y personal para el desarrollo en lenguaje RPG⁸ nativo del AS/400.

1.8. Solución propuesta

Para dar solución a la necesidad de generar Comprobantes Fiscales Digitales, propuse desarrollar un sistema que cumpla los requerimientos del SAT, para integrar a Empacadora San Marcos al Nuevo Régimen Fiscal de Comprobantes Fiscales Digitales.

⁷La ADDENDA es una etiqueta que se personaliza con los datos que cada empresa requiere para procesar una factura electrónica

⁸El lenguaje de programación RPG es un lenguaje de programación desarrollado por IBM en 1964 y diseñado para generar informes comerciales o de negocios. Sus siglas en inglés significan Report Program Generator

La propuesta es, que éste Sistema Generador de Comprobantes Fiscales Digitales se encargue de recolectar la información de ventas almacenadas en distintas tablas en el ERP, la almacena en diccionarios de datos y posteriormente se organiza en un archivo XML con todas las características oficiales que la Asociación Mexicana de Estándares para el Comercio Electrónico (AMECE) en conjunto con el Sistema de Administración Tributaria (SAT) impulsan en este nuevo esquema y para dar cumplimiento a los requerimientos establecidos.

Se planea instalar un servidor Linux de aplicaciones de la siguiente manera: se usará OpenSUSE Linux como la distribución que se ejecutará en el servidor y sobre él corren los servicios:

- Apache Web Server. Servidor Web para alojar la interfaz.
 - Mod_php_apache. Módulo para la ejecución de paginas con contenido dinámico escritas en lenguaje de programación PHP⁹
- Postfix Mail Server. Agente de correo electrónico para la entrega de los Comprobantes Fiscales Digitales.
- unixODBC. Librería que proporciona una comunicación con controladores ODBC en unix
 - iSeriesAccess. Controlador de ODBC proporcionado por la empresa IBM para servidores iSeries AS/400.
- PHP. Lenguaje de programación para las paginas Web dinámicas de la interfaz Web.
- Python. Lenguaje de programación con el cual se desarrollará el sistema Generador de Comprobantes Fiscales Digitales.
 - pyODB. Módulo de Python que comunicará con unixODBC y los controladores de ODBC instalados.

Todos estos servicios sirven como herramienta para establecer la comunicación entre el servidor AS/400 y el servidor Linux que alojará el sistema Generador de Comprobantes Fiscales Digitales.

Con el sistema Generador de Comprobantes Fiscales Digitales que se desarrollará en lenguaje de programación Python, el departamento de facturación ingresará al servidor Linux vía interfaz WEB, la cual estará desarrollada en el lenguaje de programación PHP y usará el servicio Web Apache para mostrar la interfaz del Sistema Generador de Comprobantes Fiscales Digitales.

La interfaz le pedirá al usuario que seleccione el distribuidor y que ingrese los folios de facturas electrónicas a generar pasando como argumentos un identificador de distribuidor y los

⁹PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas.

folios de las facturas a procesar, lo cual generará un comando para el bash¹⁰. Ésto, lanzará un Script¹¹ de Bash que ejecutará el programa indicado para cada distribuidor, dichos programas crearán la o las Facturas Electrónicas del distribuidor seleccionado el cual se escribirá en lenguaje de programación Python.

Lo anteriormente descrito, se muestra de manera gráfica en el siguiente esquema.

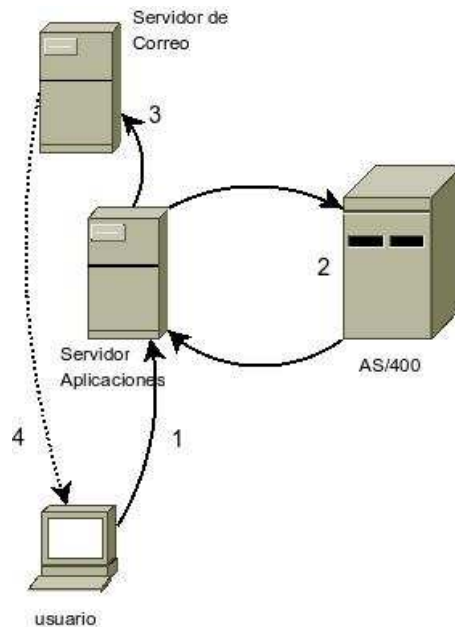


Figura 1.5: Esquema General del Sistema Generador de Comprobantes Fiscales

Breve descripción general del funcionamiento del Sistema Generador de Comprobantes Fiscales Digitales.

1. El usuario se conectará al servidor Linux de aplicaciones donde se encuentra el Sistema Generador de Comprobantes Fiscales Digitales.
2. El servidor se encargará de consultar la base de datos del servidor AS/400, al terminar el servidor de aplicaciones deberá conectarse al servidor de correo para enviar los archivos XML.
3. El servidor de correo enviará los archivos XML creados a los distintos destinatarios definidos.

En el servidor Linux de aplicaciones, se ejecutará el módulo PyODB que sirve para manipular el paquete UnixODBC que utiliza el controlador iSeriesAccess de IBM, para poder conectarse

¹⁰bash es un shell de Unix (intérprete de comandos de Unix)

¹¹un script es un guión o conjunto de instrucciones. Permiten la automatización de tareas creando pequeñas utilidades. Es muy utilizado para la administración de sistemas UNIX

al manejador de base de datos del ERP dentro del AS/400 que se llama DB2. A través de DB2 podremos acceder a la información de las tablas de la Base de Datos denominada AMFLIBT. AMFLIBT contiene los registros de COM (Comerce Object Management, Manejador de Objetos de Comercio). Dichos registros contienen datos en particular como lo son:

- Encabezado de la Factura.
- Datos del Receptor de la Factura.
- Domicilio Fiscal del Receptor.
- Líneas de Detalle por producto. (cantidad, descripción, EAN¹², costo unitario, descuentos, costo total)

Para generar los Comprobantes Fiscales Digitales se propone realizar siete consultas (QUERY¹³). En la primera consulta se recolectan los datos básicos del encabezado de la Factura con los datos que en la interfaz se ingresan y también se obtienen datos del emisor en este caso Empacadora San Marcos, la serie del documento el folio del documento y el número de aprobación del folio que se utilizarán para la siguiente consulta.

En la segunda consulta se buscará que porcentaje de descuento general tiene la factura y cual es la referencia de pedido.

En la tercera se propone utilizar la referencia de pedido para obtener datos del receptor como son lugar de entrega, número de cliente, EAN del cliente, RFC¹⁴ y los detalles de cada producto.

Para la cuarta consulta se extraerán detalles de cada producto, Código de Barras (UPC¹⁵) y descuentos de cada producto.

En la quinta, sexta y séptima consultas se extraerán datos de la ADDENDA¹⁶ que se incluirán en la Factura.

El resultado de las consultas se almacenará en distintos diccionarios de datos (estructuras de datos en memoria) denominados Emisor, Receptor, Ítems y ADDENDA.

¹²European Article Number es un sistema de códigos de barras adoptado por más de 100 países y cerca de un millón de empresas

¹³query, ingles que en español significa consulta y en base de datos es una búsqueda o pedido de datos almacenados en una base de datos. En forma genérica, query también puede tratarse de una inserción, actualización, búsqueda y/o eliminación en una base de datos.

¹⁴en México el RFC es una clave que tiene toda Persona física que realice alguna actividad licita que este obligada a pagar impuestos o toda Persona moral sin excepción. A estas personas se les llama contribuyentes

¹⁵Universal Product Code o código universal del producto

¹⁶La ADDENDA es una etiqueta que se personaliza con los datos que cada empresa requiere para procesar una factura electrónica

Para terminar de procesar la información se pasará por un módulo de normalización que consiste en quitar caracteres inválidos de codificación IMB a UTF-8¹⁷.

Se creará un archivo XML con dos llaves o palabras clave que serán reemplazadas por el sello y la cadena original dando validez al documento. Para posteriormente sellarlo con el certificado correspondiente dependiendo del Distribuidor que solicite la factura (DICASA, Monterrey, Guadalajara, San Luis, Culiacán, etc.) Este archivo XML se guardará en el servidor Linux organizado en carpetas por nombre de distribuidor y el nombre del archivo constará de una clave de distribuidor seguida del número de folio de la factura con extensión XML durante 5 años por disposición oficial. Al mismo tiempo, este archivo se envía por correo electrónico al personal de facturación de la empresa Empacadora San Marcos.

Al terminar de ejecutarse el comando lanzado por PHP, devuelve la lista de las facturas procesadas en forma de ligas de descarga, cada liga apunta al archivo XML correspondiente a cada folio.

Con todo esto se da solución a la necesidad de Empacadora San Marcos de Generar sus propios Comprobantes Fiscales Digitales.

¹⁷UTF-8 (8-bit Unicode Transformation Format) es un formato de codificación de caracteres

Capítulo 2

Conceptos Básicos

2.1. Estado de la empresa

Empacadora San Marcos, una empresa de productos alimenticios que compite por un mercado nacional e internacional es una empresa con un crecimiento notable en los últimos 7 años y a medida que la empresa crece, los recursos humanos, financieros, materiales, tecnológicos y de información son cada vez mas difíciles de administrar. Poder analizar las oportunidades de mercado es importante por lo cual la información representa una prioridad, así es como en el año 2002 se plantea la idea de cambiar su software administrativo, por uno mas robusto que ofreciera un mejor control de los productos involucrados en la elaboración de la amplia gama de productos, se analizan varias propuestas de software, los denominados Enterprise Resource Planning¹, los Planificadores de Recursos de la Empresa. son un sistema de información integral que incorpora los procesos operativos, es decir, todo lo que involucra la manufactura de un producto, por ejemplo, control de inventarios, materia prima, materiales de empaque, transporte y logística interna entre áreas, mantenimiento, etc. y la información del de negocio, es decir los socios comerciales y sus estados en cuanto a crédito y todo lo involucrado en el área financiera.

El propósito fundamental de un ERP es otorgar apoyo a los clientes del negocio, tiempos rápidos de respuesta a sus problemas así como un eficiente manejo de información que permita la toma oportuna de decisiones y disminución de los costos totales de operación

2.2. Equipo actual

En Empacadora San Marcos, el ERP esta instalado en un servidor AS/400 IBM iSeries con las siguientes características:

- 2Gb Memoria RAM.
- CPU POWER3 200 MHz.

¹ERP

- 105Gb de almacenamiento en 3 discos duros en arreglo RAID 5.
- Sistema Operativo OS/400 V5 Release 2.
- Manejador de Base de Datos DB2.

La mayoría de las computadoras en la empresa incluyendo los equipos portátiles y servidores cuentan con procesadores x86, por lo tanto usan el sistema operativo Windows de Microsoft.

2.3. Software y Hardware con el cual se cuenta

El software con el cual contamos para el desarrollo del sistema Generador de Comprobantes Fiscales Digitales (GeComFiDi) se lista a continuación:

- Sistema Operativo: Linux: SuSE Enterprise Server 10 - Kernel 2.6.16.27 x86_64.
- Paquete de UnixODBC.
- Controlador iSeries Access para UnixODBC.
- Controlador de pyODB para UnixODBC.
- Paquete de programación Python.
- jEdit, editor de textos.
- Apache Web Server.
- Paquete de programación PHP.
- Paquete de programación Python
- OpenSSL 0.9.5a

El servidor donde se instalará el sistema Generador de Comprobantes Fiscales Digitales tiene las siguientes características:

- 2Gb Memoria RAM.
- 555.10Gb de almacenamiento en 2 discos duros en arreglo RAID 1.
- CPU 2 Intel(R) Xeon(TM) CPU 3.00GHz (8 nucleos)

2.3.1. Python, Datos Generales

Python es un poderoso y dinámico lenguaje de programación y es usado en una gran variedad de aplicaciones. Python es frecuentemente comparado con Tcl, Perl, Ruby, Scheme o Java. Pero algunas de las características que sobresalen son:[2]

- Sintaxis sencilla, clara y legible.
- Fuertes capacidades de auto entendimiento
- Manejo de Orientación de Objetos más intuitiva
- Expresión natural al programar en modo procedimental
- Total Modularidad, soporte a paquetes por jerarquías
- Manejo de Error de excepciones
- Alto nivel en manejo dinámico de tipos
- Extensa colección de librerías y contribuciones de terceras personas.
- Extensiones y módulos escritos en C, C++
- Incrustable en muchas aplicaciones y usos tipo script

La Librería Estándar

Los seguidores de Python usan la frase “baterías incluidas” para describir la librería estándar. El lenguaje en si mismo es flexible y poderoso que prácticamente se puede resolver cualquier problema que se presente. Construir su propio servidor web en tan solo tres líneas de código.

Los módulos XMLrpclib y SimpleXMLRPCServer convierten en trivial la tarea de implementar llamadas a procedimientos remotos. A pesar de su denominación, no hace falta conocer ni manipular XML, el paquete email es una biblioteca para gestionar mensajes de correo electrónico, incluyendo los documentos de mensajes MIME u otros basados en la RFC 2822. A diferencia de *smtplib* y *poplib* que realmente envían y reciben los mensajes, el paquete *email* dispone de un juego de herramientas completo para construir o decodificar estructuras de mensajes complejos (incluyendo adjuntos) y para implementar codificaciones de protocolos de internet y cabeceras.[2]

Los paquetes XML.dom² y XML.sax³ proporcionan servicios robustos de análisis de este popular formato de intercambio de datos. De igual modo, el módulo *csv* permite lecturas y

²Document Object Model, o DOM, es una API del World Wide Web Consortium (W3C) para acceder o modificar documentos XML

³Simple API for XML o xml.sax paquete que contiene módulos de la API para XML

escrituras en este formato común de base de datos. Juntos, estos módulos y paquetes simplifican enormemente el intercambio de datos entre aplicaciones en Python y otras herramientas.

La internacionalización está apoyada por diversos módulos, incluyendo *gettext*, *locale* y el paquete *codecs*.

Python deja que el programador escriba el código que necesita, rápido y gracias a su altamente optimizado compilador y a las librerías, el código de Python se ejecuta más rápido que otras aplicaciones.

Python Integrable

Python puede integrarse con COM⁴, .NET⁵, y objetos CORBA⁶ Para Java, use Jython, una implementación de Python para la Máquina Virtual de Java. Python también cuenta con soporte para Internet Communications Engine (ICE) y muchas otras tecnologías de comunicación.

Si el programador encuentra algo que no pueda hacer Python, entonces podrá crear su propia extensión del módulo que necesite usando C o C++.

Python se ejecuta en cualquier lugar

Python esta disponible para la mayoría de los sistemas operativos: Windows, Linux/Unix, OS/2, Mac, Amiga y muchos mas. Versiones específicas para teléfonos móviles como Nokia Series 60. y también existe pippy, una versión de Python para PalmOS y actualmente también corre sobre los iPhones y iPod Touch de Apple.

Python es Libre

Las implementaciones de Python son liberadas bajo licencia Open Source, esto lo hace de uso y distribución libre, inclusive para el uso comercial. La licencia de Python es liberada por Python Software Foundation.

2.3.2. Filosofía

Python es un lenguaje de programación multiplataforma y multiparadigma. Esto permite varios estilos de programación: Programación orientada a objetos, programación estructurada, programación funcional y programación orientada a aspectos[2].

⁴Python Extensions for Windows

⁵librerías para ironpython

⁶arquitectura común de intermediarios en peticiones a objetos

Otros paradigmas más están soportados mediante el uso de extensiones. Python usa tipado dinámico de datos y *reference counting*⁷ para el manejo de memoria. Una característica importante de Python es la resolución dinámica de nombres, lo que enlaza un método y un nombre de variable durante la ejecución del programa.

Otro objetivo del diseño del lenguaje es la facilidad de extensión. Nuevos módulos pueden ser fácilmente escritos en C o C++. Python puede ser utilizado como un lenguaje de extensión para módulos y aplicaciones que necesitan de una interfaz programable.[2]

2.3.3. Descripción

Python es considerado como la oposición a Perl, lenguaje con el cual mantiene una rivalidad amistosa. Los usuarios de Python consideran a éste mucho más limpio y elegante para programar. Python permite dividir el programa en módulos reutilizables desde otros programas Python. Viene con una gran colección de módulos estándar que se pueden utilizar como base de los programas (o como ejemplos para empezar a aprender Python). También hay módulos incluidos que proporcionan E/S de archivos, llamadas al sistema, sockets y hasta interfaces a GUI (interfaz gráfica con el usuario) como Tk, GTK, Qt entre otros.

Python es un lenguaje interpretado, lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar. El intérprete se puede utilizar de modo interactivo, lo que facilita experimentar con características del lenguaje, escribir programas desechables o probar funciones durante el desarrollo del programa. También es una calculadora muy útil.

⁷En ciencias de la computación *reference counting* es una técnica para almacenar el número de referencia, apuntadores y manejadores a cada recurso como son bloques de memoria.

Capítulo 3

Diseño del Sistema GeComFiDi

3.1. Introducción

En este capítulo, se describe el diseño del Sistema Generador de Comprobantes Fiscales Digitales, denominado Sistema GeComFiDi, la instalación de los módulos necesarios para lograr la conexión entre el servidor AS/400 y el servidor de Aplicaciones Linux, los requerimientos legales del Sistema de Administración Tributaria y la forma en que estos integran y dan validez al Comprobante Fiscal Digital, así como el algoritmo propuesto para el funcionamiento del Sistema Generador de Comprobantes Fiscales Digitales.

3.2. Conexión a la Base de Datos

Para iniciar con la programación se necesita conectarse a la base de datos DB2, la cual esta en el servidor AS/400, el método para conectarse en plataformas Windows, es usando un controlador de base de datos que proporciona IBM y así poder configurar un Open Database Connectivity (ODBC)¹.

En plataformas Unix, existen varios administradores de ODBC, tal como JDBC, unixODBC, ADODB, en nuestro caso particular usaremos unixODBC² para lograr la conexión a la base de datos.

Para la conexión se requiere configurar unixODBC y los controladores iSeries Access³ para Linux y por último el módulo de python para la comunicación entre python y unixODBC quien a su vez usa el manejador proporcionado para iSeries Access quien realiza la consulta en el servidor AS/400.

¹estándar de acceso a bases de datos, que permite mantener independencia entre los lenguajes de programación, los sistemas de bases de datos (las bases de datos y su software gestor), y los sistemas operativos

²<http://www.unixodbc.org>

³<http://www-03.ibm.com/servers/eserver/iserier/access/linux>

A continuación se mencionarán los preparativos necesarios que el servidor Linux necesita para lograr conectarse a la base de datos DB2 almacenada en el Servidor AS/400.

3.2.1. Instalación de unixODBC

Para acceder a una base de datos usando ODBC, en Linux es necesario instalar unixODBC, en el caso de Empacadora San Marcos se usará unixODBC-2.2.11, el cual se puede descargar de: <http://unixodbc.org/>.

Una vez descargado hay que compilarlo, por lo regular esta en formato `tar.gz`, a continuación se resume el proceso de descarga, desempaquetado e instalación.

A continuación se muestran las instrucciones en la consola de comandos de Linux para descargar e instalar unixODBC.

```
bash:/usr/src # wget -c http://www.unixodbc.org/unixODBC-2.2.11.tar.gz
bash:/usr/src # tar xvfz unixODBC-2.2.11.tar.gz
bash:/usr/src # cd unixODBC-2.2.11
bash:/usr/src/unixODBC-2.2.11 # ./configure; make; make install
```

Comandos para la descarga e instalación de unixODBC.

En la primera línea se usa el comando `wget`⁴ para descargar el archivo.

En la segunda línea se utiliza el comando `tar`⁵ para extraer el contenido de archivo descargado.

En la tercera línea se ingresa al directorio donde se extrajo el contenido del archivo descargado.

En la última línea, según la documentación del paquete unixODBC se necesitan ejecutar esos tres comandos, en este caso se indican en la misma línea de comandos, es decir se ejecutará el comando `configure`, luego el comando `make` y por último el comando `make install`.

Con eso, tenemos instalado unixODBC en nuestro sistema operativo Linux.

3.2.2. Instalación del Driver de iSeriesAccess For Linux

Para conectarnos con un origen de datos, necesitamos el driver, que nos permita entablar comunicación entre nuestro servidor Linux y el servidor AS/400, este driver es desarrollado

⁴es una herramienta de Software Libre que permite la descarga de contenidos desde servidores web

⁵El programa `tar`, como se indicó en la introducción es usado para almacenar archivos y directorios en un solo archivo. Dentro de los entornos Unix

por la empresa IBM. Se puede descargar desde el sitio de la empresa (requiere registro, pero es gratis).

<http://www-1.ibm.com/servers/eserver/iserries/access/linux/>

el archivo descargado, para este caso fue:

iSeriesAccess-5.2.0-1.8.i386.rpm o iSeriesAccess-5.4.0-0.9.1.x86_64.rpm

```
bash:~ # rpm -Uvhi iSeriesAccess-5.4.0-0.9.1.x86_64.rpm
```

Instalación del controlador iSeriesAccess de IBM.

El archivo que proporciona IBM se instala de forma distinta, ya que es un archivo pre compilado en este caso y ha sido empaquetado para su distribución en entornos Linux con el método usado por la empresa RedHat, denominado RPM Package Manager o RPM, originalmente llamado Red Hat Package Manager.

Una vez concluida la instalación de unixODBC y del driver de iSeries Access, configuraremos el System DSN(Data Source Name en español, Nombre Fuente de datos o Nombre de origen de datos) en unixODBC para que las aplicaciones que se conecten a la base de datos puedan hacer uso de el.

3.2.3. Configuración de unixODBC

Para lograr conectarnos con el servidor es necesario configurar un System DSN (Nombre Fuente de datos o Nombre de origen de datos), para ello hay que entrar al subdirectorio donde se desempaqueto el archivo `unixODBC-2.2.11.tar.gz` el cual contiene la interfaz gráfica `ODBCConfig` , para ello necesitamos tener privilegios de super usuario `root`⁶, y ejecutar el archivo que esta en nuestro caso en:

`/usr/src/unixODBC-2.2.11/ODBCConfig.`

```
bash/usr/src/unixODBC-2.2.11/ODBCConfig # ODBCconfig
```

En la imagen 3.1 se muestra el cuadro de diálogo correspondiente a la interfaz gráfica de configuración de `ODBCConfig`.

La interfaz gráfica de configuración de `ODBCConfig`, muestra un cuadro de diálogo con 6 pestañas con las siguientes funciones que a continuación se describen.

⁶En sistemas operativos del tipo Unix, `root` es el nombre convencional de la cuenta de usuario que posee todos los derechos en todos los modos (mono o multi usuario). `root` es también llamado superusuario. Normalmente esta es la cuenta de administrador.

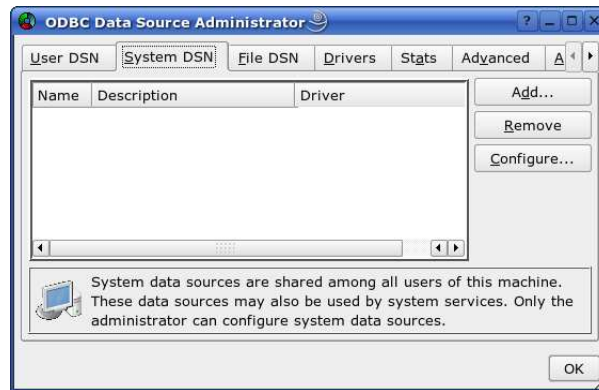


Figura 3.1: Interfaz grafica de ODBCConfig.

- *User DSN*: La configuración de un origen de datos de usuario, se almacenará en el directorio de dicho usuario, esto permite configurar un origen de datos sin tener privilegios de administrador de sistema, los demás usuarios no podrán usar este origen de datos.
- *System DSN*: Un origen de datos de Sistema, es un origen de datos que se comparte a todos los usuarios del servidor, estos orígenes de datos son usados para dar servicios a aplicaciones, solo el administrador puede configurar un origen de datos de sistema.
- *File DSN*: Archivo de origen de datos puede ser almacenado en un sistema de archivos y ser compartido a los usuarios, esto requiere que cada cliente cuente con los controladores necesarios para manipular la base de datos.
- *Drivers*: Los manejadores de bases de datos son los que facilitan la comunicación entre el manejador de datos y el servidor de datos, la mayoría de los manejadores de bases de datos se pueden descargar de Internet de la página de los proveedores de la base de datos, Normalmente solo el administrador puede agregar estos manejadores de base de datos.
- *Stats*: Muestra en número de conexiones, descriptores y estados de los objetos ODBC.
- *Advanced* En esta sección se puede configurar los registros de eventos y el trayecto donde se almacenará el archivo de registro.
- *About*: contiene la información legal e información de los desarrolladores.

En la imagen 3.2 agregamos nuestro “system dns”, seleccionando el driver correspondiente a la base de datos que deseemos usar, en el caso particular de Empacadora San Marcos, el servidor solo se conectará al Servidor iSeries, por lo cual solo se muestra ese controlador.

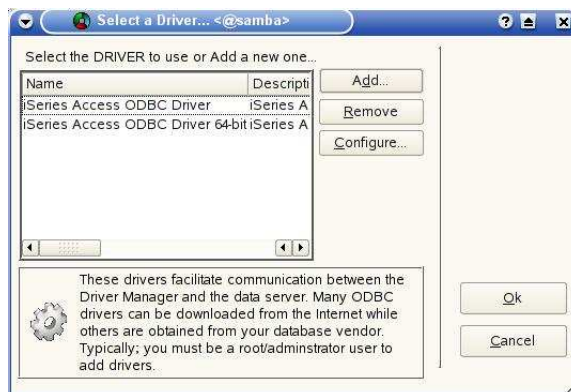


Figura 3.2: Drivers disponibles

En el caso particular de Empacadora San Marcos, se configura un origen de datos de sistema, esto para que pueda estar disponible a todos los usuarios del servidor, se selecciona la pestaña de “System DSN”, el cuadro de diálogo se muestra vacío.

Para agregar un origen de datos, se presiona el botón add y aparece un cuadro de diálogo para seleccionar el driver correspondiente al origen de datos.

Se elige el Driver iSeries Acces ODBC, el cual hay que configurar de acuerdo al entorno operativo, es decir los parámetros que permiten conectarse.

A continuación se listan los parámetros que se requieren para agregar un Origen de Datos:

- *Name*: Nombre asociado al origen de datos.
- *Description*: Identifica al driver que será utilizado.
- *Driver*: Controlador de base de datos a utilizar.
- *System*: Nombre del sistema (requiere un servidor de nombres configurado y funcionando) o IP del servidor el cual se quiere conectar.
- *UserID*: Un usuario valido dentro del servidor.
- *Password*: Contraseña del usuario que se conectará.
- *Naming*: Seleccione 0 para SQL o 1 para usar nombres de tablas del sistema AS/400.
- *Defaul Libraries*: Especifique el nombre de librerías separadas por espacios que estan disponibles en el sistema AS/400.
- *Database*: Nombre de la base de datos que se utilizará.

- *Connection Type*: tipo de acceso a la base de datos, 0 lectura y escritura, 1 lectura y llamadas, 2 solo lectura.
- *Commit Mode* Soporte a base de datos IBM.
- *Extended Dynamic*: Soporte a paquetes dinámicos de datos de IBM.
- *Default Pkg Library*: Librería por defecto que sea usada, esta depende del ERP.
- *Default Package*: Paquetes de datos y formatos propios de IBM.
- *Allow Data Compression*: Permite enviar los datos comprimidos a través de la conexión ODBC, 0 deshabilitado; 1 habilitado.
- *Library View*: Permite seleccionar que librerías serán mostradas, 0 por defecto o 1 todas las disponibles, incluye las de sistema.
- *Allow Unsuported Char*: Permite el uso de caracteres propios de IBM; 0 desactivar, 1 convertir a ASCII ignorando errores.
- *Force Translation*: Convertir los datos binarios CCSID (CCSID es una abreviación utilizado por IBM para “Coded Character Set Identifier” de IBM).
- *Trace*: Monitoreo del estado de la comunicación y memoria; 0 desactivado o 1 activado.

En la figura 3.3 se configuran los parámetros de acuerdo al ambiente de trabajo particular de Empacadora San Marcos, el Origen de Datos de Sistema se configurará de la siguiente forma:

A continuación se muestran los parámetros de configuración particulares para la operación entre servidores de la empresa Empacadora San Marcos para poder conectarse al origen de datos.

- *Name*: db2.
- *Description*: iSeries Access ODBC Driver.
- *Driver*: iSeries Access ODBC Driver.
- *System*: 10.0.2.9.
- *UserID*: LPEREZR.
- *Password*: *****.
- *Naming*: 0.
- *Defaul Libraries*: QGPL AMFLIBT AMFLIBTPC.
- *Database*: AMFLIBT.

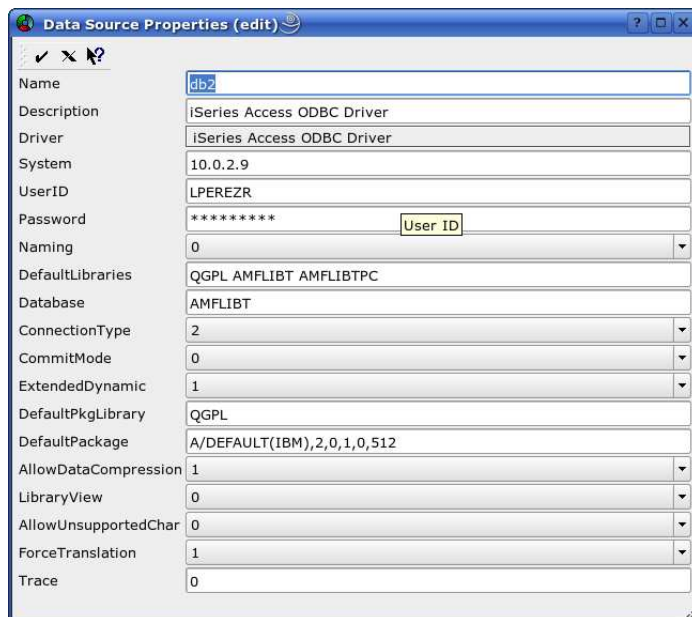


Figura 3.3: Configuración del ambiente

- *Connection Type*: 2.
- *Commit Mode*: 0.
- *Extended Dynamic*: 1.
- *Default Pkg Library*: QGPL.
- *Default Package*: A/DEFAULT(IBM),2,0,1,0,512.
- *Allow Data Compression*: 1.
- *Library View*: 0.
- *Allow Unsupported Char*: 0.
- *Force Translation*: 1.
- *Trace*: 0.

Los archivos de configuración relacionados con los controladores de los distintos orígenes de datos se encuentran dentro del directorio `/etc/unixODBC`. El archivo que contiene la configuración del origen de datos se llama `odbc.ini`, el archivo que contiene los trayectos necesarios para la poder manipular el controlador de la base de datos se llama `odbcinst.ini`.

Para las personas con experiencia administrando entornos Linux, les resulta útil el poder modificar los archivos de configuración desde una consola o terminal sin la necesidad de usar una interfaz gráfica y un editor de texto como kwrite.

El archivo `odbcinst.ini` contiene información relacionada con los trayectos a necesarios para la correcta ejecución del controlador de base de datos, por lo cual no es relevante mostrar aquí.

El archivo `odbc.ini` contiene los parámetros particulares para la operación entre servidores es decir la información que se proporcionó en el cuadro de diálogo **Origen de Datos de Sistema**

A continuación se muestra el contenido del archivo `odbc.ini` una vez finalizada la configuración usando la interfaz grafica mostrada en la figura 3.3

```
[db2]
Description = iSeries Access ODBC Driver
Driver = iSeries Access ODBC Driver
System = 10.0.2.9
UserID = xxxxxx
Password = xxxxxx
Naming = 0
DefaultLibraries = QGPL AMFLIBT AMFLIBTPC
Database = AMFLIBT
ConnectionType = 2
CommitMode = 0
ExtendedDynamic = 1
DefaultPkgLibrary = QGPL
DefaultPackage = A/DEFAULT(IBM),2,0,1,0,512
AllowDataCompression = 1
MaxFieldLength = 32
BlockFetch = 1
BlockSizeKB = 128
ExtendedColInfo = 0
LibraryView = 0
AllowUnsupportedChar = 0
ForceTranslation = 1
Trace = 0
```

Contenido del archivo `odbc.ini`

3.2.4. Instalación de pyODB.

pyODB es un módulo del Python de ODBC que da una manera fácil y rápida de trabajar con bases de datos. Proporciona un conjunto de instrucciones simplificadas para manejar los distintos drivers de unixODBC.

Se puede descargar el módulo pyODB del repositorio de software libre Source Forge⁷ en el espacio destinado al proyecto pyODB desde: <http://sourceforge.net/projects/pyodb/>, el archivo descargado contiene los archivos fuente y tendrá que ser compilado, por lo general es `tar.gz` por lo cual es necesario descomprimir el archivo, compilarlo e instalarlo.

A continuación se muestra un resumen del proceso de descarga, desempaquetado, compilación e instalación desde el intérprete de comandos de Linux.

⁷SourceForge.net es un proyecto que almacena proyectos de software libre al rededor de l mundo, SourceForge les provee almacenamiento gratis a los proyectos

```
bash:/usr/src # wget -c http://sourceforge.net/sf/pyodb/pyodb-0.6.tar.gz.  
bash:/usr/src # tar xvfz pyodb-0.6.tar.gz.  
bash:/usr/src # cd pyodb-0.6  
bash:/usr/src/pyodb-0.6 # python setup.py build  
bash:/usr/src/pyodb-0.6 # python setup.py install
```

Descarga e instalación de pyODB.

En la primera línea se usa el comando `wget`⁸ para descargar el archivo.

En la segunda línea se utiliza el comando `tar`⁹ para extraer el contenido de archivo descargado.

En la tercera línea se ingresa al directorio donde se extrajo el contenido del archivo descargado.

En la cuarta línea, según la documentación del paquete es necesario crear el archivo binario, para ello se le indica con el parámetro `build`.

En la última línea, se instala el archivo que previamente se construyó en el paso anterior, para ello el parámetro `install`.

Al terminar estos pasos, tenemos instalado en el Sistema Operativo Linux, el módulo pyODB listo para ser usado en Python y poder manipular la base de datos vía unixODBC. y el Dirver iSeries Access.

Una vez realizada la conexión al servidor ya podremos consultar la base de datos para obtener los datos de la factura tradicional y consultar las tablas donde se encuentran los datos que particularmente lleva una factura electrónica o comprobante fiscal digital.

Un Comprobante Fiscal Digital, es un archivo XML el cual cumple los requisitos legales enmarcados en el Diario Oficial de la Federación, donde se enumeran las características técnicas que debe cumplir. A continuación se mencionan los requerimientos legales.

3.3. Requerimientos legales sobre comprobantes fiscales

3.3.1. Formato electrónico único

El contribuyente que opte por emitir comprobantes fiscales digitales deberá generarlos bajo el siguiente estándar XSD base y los XSD complementarios que requiera, validando su for-

⁸es una herramienta de Software Libre que permite la descarga de contenidos desde servidores web

⁹El programa `tar`, como se indicó en la introducción es usado para almacenar archivos y directorios en un solo archivo. Dentro de los entornos Unix

ma y sintaxis en un archivo con extensión XML, siendo el único formato para representar y almacenar comprobantes de manera electrónica o digital.[3]

Para poder ser validado, el comprobante fiscal digital deberá estar referenciado al namespace o etiqueta dentro del archivo XML del comprobante fiscal digital y referenciar la validación del mismo a la ruta publicada por el SAT en donde se encuentra el esquema XSD objeto de la presente sección:

```
http://www.sat.gob.mx/sitio_internet/cfd/2/cfdv2.xsd
```

A continuación se muestra el código XML que le da validez a un comprobante fiscal digital.

```
<Comprobante
XMLns='http://www.sat.gob.mx/cfd'
XMLns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:schemaLocation=' http://www.sat.gob.mx/cfd
http://www.sat.gob.mx/sitio_internet/cfd/2/cfdv2.xsd
.....
</Comprobante>
```

Adicionalmente a las reglas de estructura planteadas dentro del presente estándar, el contribuyente que opte por este mecanismo de generación de comprobantes deberá sujetarse tanto a las disposiciones fiscales vigentes, como a los lineamientos técnicos de forma y sintaxis para la generación de archivos XML especificados por el consorcio w3, establecidos en www.w3.org.

En particular se deberá tener cuidado de aquellos casos especiales que se presenten en los valores especificados dentro de los atributos del archivo XML como aquellos que usan el carácter & , el carácter " , el carácter ' , el carácter < y el carácter > que requieren del uso de secuencias de escape.

1. En el caso del & se deberá usar la secuencia &
2. En el caso del " se deberá usar la secuencia "
3. En el caso del < se deberá usar la secuencia <
4. En el caso del > se deberá usar la secuencia >
5. En el caso del ' se deberá usar la secuencia '

Ejemplos:

Para representar

```
nombre='Juan & José & 'Niño'''
```

se usará

```
nombre='Juan & José & "Niño"'''
```

Cabe mencionar de que a pesar de que la especificación XML permite el uso de secuencias de escape para el manejo de caracteres acentuados y el carácter ñ, dichas secuencias de escape no son necesarias al expresar el documento XML bajo el estándar de codificación UTF-8 si fue creado correctamente, misma que es utilizada como único estándar por el SAT.

3.3.2. Generación de sellos digitales para comprobantes fiscales digitales

Elementos utilizados en la generación de Sellos Digitales:

- Cadena Original, el elemento a sellar, en este caso de un comprobante fiscal digital.
- Certificado de Sello Digital y su correspondiente clave privada.
- Algoritmos de criptografía de clave pública para firma electrónica avanzada.
- Especificaciones de conversión de la firma electrónica avanzada a Base 64.

Para la generación de sellos digitales se utiliza criptografía de clave pública aplicada a una cadena original.

3.3.3. Cadena Original

Se entiende como cadena original, a la secuencia de datos formada con la información contenida dentro del comprobante fiscal digital, establecida en el Diario Oficial de la Federación en el Rubro C *Estándar de comprobante fiscal digital extensible* del anexo 20 de la resolución de la miscelánea fiscal¹⁰. Siguiendo para ello las reglas y la secuencia aquí especificadas:

Reglas Generales:

1. Ninguno de los atributos que conforman al comprobante fiscal digital deberá contener el carácter | (“pipe”) debido a que este será utilizado como carácter de control en la formación de la cadena original
2. El inicio de la cadena original se encuentra marcado mediante una secuencia de caracteres || (doble “pipe”)

¹⁰http://www.amece.org.mx/amece/Legislacion_Factura_Electronica.php

3. Se expresará únicamente la información del dato sin expresar el atributo al que hace referencia. Esto es, si la serie del comprobante es la “A” solo se expresará |A| y nunca |Serie A|
4. Cada dato individual se encontrará separado de su dato subsiguiente, en caso de existir, mediante un carácter | (“pipe” sencillo)
5. Los espacios en blanco que se presenten dentro de la cadena original serán tratados de la siguiente manera:
 - a) Se deberán remplazar todos los tabuladores, retornos de carro y saltos de línea por espacios en blanco
 - b) Acto seguido se elimina cualquier carácter en blanco al principio y al final de cada separador | (“pipe” sencillo)
 - c) Finalmente, toda secuencia de caracteres en blanco se sustituye por un único carácter en blanco
6. Los datos opcionales no expresados, no aparecerán en la cadena original y no tendrán delimitador alguno
7. El final de la cadena original será expresado mediante una cadena de caracteres || (doble “pipe”)
8. Toda la cadena de original se expresará en el formato de codificación UTF-8
9. El nodo o nodos adicionales <Complemento> se integrarán al final de la cadena original respetando la secuencia de formación para cada complemento y número de orden del Complemento.

Secuencia de Formación: La secuencia de formación será siempre en el orden que se expresa a continuación, tomando en cuenta las reglas generales expresadas en el párrafo anterior.[3].

Los siguientes son los campos que se mencionan en el anexo 20 del Diario Oficial de la Federación

1. Información del nodo Comprobante
 - a) version
 - b) serie
 - c) folio
 - d) fecha
 - e) noAprobacion
 - f) anoAprobacion

- g)* tipoDeComprobante
- h)* formaDePago
- i)* condicionesDePago
- j)* subTotal
- k)* descuento
- l)* total

2. Información del nodo Emisor

- a)* rfc
- b)* nombre

3. Información del nodo DomicilioFiscal

- a)* calle
- b)* noExterior
- c)* noInterior
- d)* colonia
- e)* localidad
- f)* referencia
- g)* municipio
- h)* estado
- i)* pais
- j)* codigoPostal

4. Información del nodo ExpedidoEn

- a)* calle
- b)* noExterior
- c)* noInterior
- d)* colonia
- e)* localidad
- f)* referencia
- g)* municipio
- h)* estado
- i)* pais
- j)* codigoPostal

5. Información del nodo Receptor

- a) rfc
- b) nombre

6. Información del nodo Domicilio

- a) calle
- b) noExterior
- c) noInterior
- d) colonia
- e) localidad
- f) referencia
- g) municipio
- h) estado
- i) pais
- j) codigoPostal

7. Información de cada nodo Concepto

nota: esta secuencia deberá ser repetida por cada nodo Concepto relacionado

- a) cantidad
- b) unidad
- c) noIdentificacion
- d) descripcion
- e) valorUnitario
- f) importe

8. Información de cada nodo InformacionAduanera

nota: esta secuencia deberá ser repetida por cada nodo InformacionAduanera relacionado

- a) numero
- b) fecha
- c) aduana

9. Información del nodo CuentaPredial

- a) numero

10. Información de cada nodo Retención

nota: esta secuencia a, b, deberá ser repetida por cada nodo Retención relacionado, el total de impuestos retenidos no se repite.

- a) impuesto
- b) importe
- c) totalImpuestosRetenidos

11. Información de cada nodo Traslado

nota: esta secuencia a, b, deberá ser repetida por cada nodo Traslado relacionado, el total de impuestos trasladados no se repite.

- a) Impuesto
- b) tasa
- c) importe
- d) totalImpuestosTrasladados

3.3.4. Generación del Sello Digital

Para toda cadena original a ser sellada digitalmente, la secuencia de algoritmos a aplicar es la siguiente:[4]

1. Aplicar el método de reducción MD5¹¹ cadena original a sellar incluyendo los nodos Complementarios. Este procedimiento genera una salida de 16 bytes para todo mensaje. La posibilidad de encontrar dos mensajes distintos que produzcan una misma salida es de 1 en 2135, y por lo tanto en esta posibilidad se basa la inalterabilidad del sello, así como su no reutilización. Es de hecho una medida de la integridad del mensaje sellado, pues toda alteración del mismo provocará un resultado en la codificación hash¹² totalmente diferente, por lo que no se podrá autenticar el mensaje.

El algoritmo de reducción criptográfico MD5 no requiere del uso de secuencias de relleno (padding).

MD5 no requiere semilla alguna. El algoritmo cambia su estado de bloque en bloque de acuerdo a la entrada previa.

2. Con la clave privada correspondiente al certificado digital del emisor del mensaje y del sello digital, encriptar el resumen del mensaje obtenida en el paso 1 utilizando para ello el algoritmo de encriptado RSA.

¹¹En criptografía, MD5 acrónimo de Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5 es un algoritmo de reducción criptográfico de 128 bits ampliamente usado.

¹²En informática, Hash se refiere a una función o método para generar claves o llaves que representen de manera casi unívoca a un documento, registro, archivo, etc., resumir o identificar un dato a través de la probabilidad, utilizando una función hash o algoritmo hash. Un hash es el resultado de dicha función

El resultado será una cadena binaria que no necesariamente consta de caracteres imprimibles, por lo que deberá traducirse a una cadena que sí conste solamente de tales caracteres. Para ello se utilizará el modo de expresión de secuencias de bytes denominado *Base 64*, que consiste en la asociación de cada 6 bits de la secuencia a un elemento de un *alfabeto* que consta de 64 caracteres imprimibles. Puesto que con 6 bits se pueden expresar los números del 0 al 63, si a cada uno de estos valores se le asocia un elemento del alfabeto se garantiza que todo byte de la secuencia original puede ser mapeado a un elemento del alfabeto Base 64, y los dos bits restantes formarán parte del siguiente elemento a mapear. Este mecanismo de expresión de cadenas binarias produce un incremento de 25 % en el tamaño de las cadenas imprimibles respecto de la original.

La codificación en base 64, así como su decodificación, se hará tomando los bloques a procesar en el sentido de su lectura, es decir, de izquierda a derecha.

Por tanto, los caracteres utilizados en el alfabeto de Base 64 son:

A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, , / y en el orden descrito les corresponden los índices del 0 al 63 en un arreglo de 64 elementos. Para traducir de binario a Base 64, se examina la secuencia binaria evaluando 6 bits a la vez; si el valor de los primeros 6 bits es 0, entonces se imprime la letra A; si es 1, entonces se imprime la letra B y así sucesivamente hasta completar la evaluación de todos los bits de la secuencia binaria evaluados de 6 en 6.

3.3.5. Criptografía de la Clave Pública

La criptografía de Clave Pública se basa en la generación de una pareja de números muy grandes relacionados íntimamente entre sí, de tal manera que una operación de encriptado sobre un mensaje tomando como clave de cifrado a uno de los dos números, produce un mensaje alterado en su significado que solo puede ser devuelto a su estado original mediante la operación de desencriptado correspondiente tomando como clave de cifrado al otro número de la pareja.

Uno de estos dos números, expresado en una estructura de datos que contiene un módulo y un exponente, se conserva secreta y se le denomina *clave privada*, mientras que el otro número llamado *clave pública*, en formato binario y acompañado de información de identificación del emisor, además de una calificación de validez por parte de un tercero confiable, se incorpora a un archivo denominado *certificado de firma electrónica avanzada o certificado para sellos digitales*.

El Certificado puede distribuirse libremente para efectos de intercambio seguro de información y para ofrecer pruebas de autoría de archivos electrónicos o acuerdo con su contenido

mediante el proceso denominado *firma electrónica avanzada*, que consiste en una característica observable de un mensaje, verificable por cualquiera con acceso al certificado digital del emisor, que sirve para implementar servicios de seguridad para garantizar: La integridad (facilidad para detectar si un mensaje firmado ha sido alterado), autenticidad, certidumbre de origen (facilidad para determinar qué persona es el autor de la firma y valida el contenido del mensaje) y no repudiación del mensaje firmado (capacidad de impedir que el autor de la firma niegue haber firmado el mensaje)

Estos servicios de seguridad proporcionan las siguientes características a un mensaje con firma electrónica avanzada:

- Es infalsificable.
- La firma electrónica avanzada no es reciclable (es única por mensaje).
- Un mensaje con firma electrónica avanzada alterado, es detectable.
- Un mensaje con firma electrónica avanzada, no puede ser repudiado.

Los certificados de sello digital se generan de manera idéntica a la firma electrónica avanzada y al igual que las firmas electrónicas avanzadas el propósito del sello digital es emitir comprobantes fiscales con autenticidad, integridad, verificables y no repudiables por el emisor. Para ello bastará tener acceso al mensaje original o cadena original, al sello digital y al certificado de sello digital del emisor.

Al ser el certificado de sello digital idéntico en su generación a una firma electrónica avanzada, proporciona los mismos servicios de seguridad y hereda las características de las firmas digitales. Por consecuencia un comprobante fiscal digital sellado digitalmente por el contribuyente tiene las siguientes características:

- Es infalsificable.
- El sello digital de un comprobante fiscal digital no es reciclable (es único por documento).
- Una cadena original de un comprobante fiscal digital sellada digitalmente, que hubiese sido alterada es detectable.
- Una cadena original de un comprobante fiscal digital sellada digitalmente no puede ser repudiada.

Los algoritmos utilizados en la generación de un sello digital son los siguientes:

MD5, que es una función hash (picadillo o resumen) de un solo sentido tal que para cualquier entrada produce una salida compleja de 16 bytes denominada resumen.

RSAPrivateEncrypt, que utiliza la clave privada del emisor para encriptar la digestión del mensaje.

RSAPublicDecrypt, que utiliza la clave pública del emisor para descryptar la digestión del mensaje.

A manera de referencia y para obtener información adicional, se recomienda consultar el sitio de comprobantes fiscales digitales que se encuentra dentro del portal del SAT: www.sat.gob.mx

3.4. Algoritmo Propuesto

Una vez mencionados los detalles técnicos y procesos internos que la empresa Empacadora San Marcos debe de cumplir, en la figura 3.4, se muestra a continuación el algoritmo que da solución al problema de Generar los Comprobantes Fiscales Digitales.

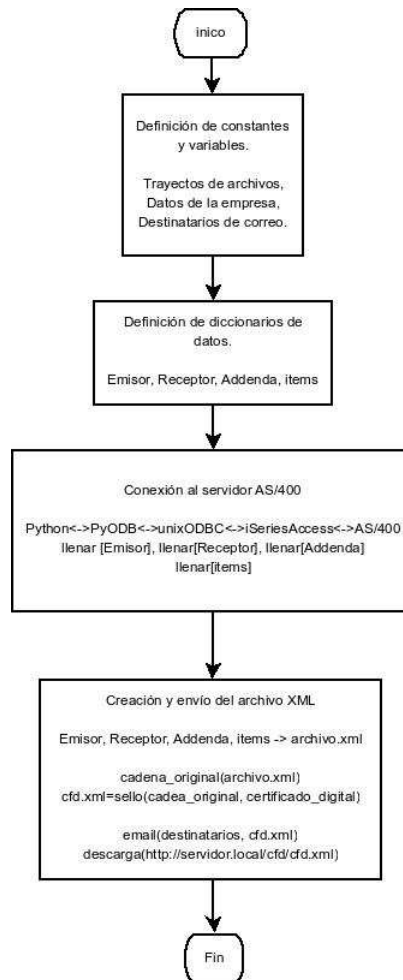


Figura 3.4: Esquema general del Algoritmo

Se enumeran los procesos de forma general, iniciando con la preparación de datos y terminando con el envío del archivo sellado.

Iniciamos estableciendo los datos de la empresa y los destinatarios de correo electrónico.

Se describe la conexión y consulta a las distintas tablas, quitar caracteres no imprimibles y de control de los datos obtenidos, se preparan los diccionarios de datos que almacenarán la información como son cliente, lugar de entrega, número de productos y su descripción.

También se describe el proceso de la creación y sellado del archivo de acuerdo a los lineamientos publicados en el Diario Oficial de la Federación y que respalda el Gobierno Mexicano en sus diferentes organismos como el Sistema de Administración Tributaria SAT y la Asociación Mexicana de Estándares para el Comercio AMECE.

1. Definir constantes y variables

- 1.1. Trayectos de archivos: Se establecen las rutas en que se guardarán los archivos XML de Facturas Electrónicas y archivos de texto plano.
- 1.2. Datos de la empresa: Se establecen los datos de la empresa Emisora de Facturas Electrónica para ingresarlos al diccionario de datos.
- 1.3. Datos para el envío del archivo por correo electrónico: Se establecen las direcciones de correo electrónico al personal de departamento de facturación a quienes se le hará llegar el comprobante Fiscal Digital.

2. Definir Diccionario donde se almacenarán los datos

- 2.1. Diccionario de datos para el emisor: En éste punto se crean los campos donde se almacenará la información obtenida de la consulta.
- 2.2. Diccionario de datos receptor: Aquí, se crean los campos que almacenarán la información del lugar de entrega, también obtenido de una consulta.
- 2.3. Diccionario vacío para datos de detalle: Se crean los campos donde se almacenarán las líneas de detalle por producto.
- 2.4. Diccionario de datos para el apartado ADDENDA: Se crean los campos que contendrán datos particulares dependiendo del Receptor, entre ellos puedes ser, número de orden de compra, fecha de orden de compra, número de aprobación, datos del domicilio del comprador, etc.

3. Conexión al Servidor AS/400

- 3.1. Se carga módulo PyODB: En esta etapa Python utiliza el módulo PyODB para llamar a UnixODBC.
- 3.2. Se utiliza driver iSeriesAccess: usando el driver iSeriesAccess instalado en el servidor Linux para lograr la conexión al Servidor AS/400 y poder utilizar las tablas del manejador de base de datos DB2.

- 3.3. Extracción de datos para el encabezado de la factura: Se toma de la base de datos los elementos correspondientes del cliente y datos relacionados al pedido.
 - 3.4. Extracción de datos básicos ADDENDA: Algunos datos no comerciales se agregan en este diccionario de datos, dependiendo del cliente, como el número regional de la tienda y el número regional del lugar de entrega.
 - 3.5. Extracción y compilación de datos: serie, naturaleza, folios y descuentos:
 - 3.6. Extracción de datos particulares ADDENDA: Datos que Empacadora Agrega como número de piezas por caja, peso, número regional de la empresa etc.
 - 3.7. Extracción de datos del cliente y lugar de entrega: Datos del lugar donde se entregará la mercancía.
 - 3.8. Extracción de datos del detalle de la factura: Es decir, cada uno de los elementos que identifican a cada producto comercializado que se mencionan en la factura, como el número del código de barras, descripción, el costo, etc.
 - 3.9. Extracción de datos finales ADDENDA Datos relacionados con la consulta anterior como el total de descuentos, el IVA, etc.
 - 3.10. Normalizar datos en diccionarios: Los datos que se obtienen de las tablas, contienen caracteres especiales y serán reemplazados por caracteres UTF-8, completar los decimales en caso de ser necesario y evitar caracteres especiales como comillas simples, apóstrofes, etc.
 - 3.11. Asignación de número y año de aprobación de los folios: Cada Factura Electrónica debe llevar el número de aprobación que el SAT asigna, de acuerdo a la serie de cada factura.
4. Creación y envío del archivo XML
 - 4.1. Creación del archivo XML con datos básicos Se crea un archivo XML con los datos de los diccionarios de datos, dejando una palabra reservada en el lugar de la “cadena original” y uno más en el lugar del “Sello Digital”.
 - 4.2. Creación Cadena Original: a partir del archivo XML previo, se ejecuta un script el cual extraerá del archivo XML su cadena original la cual reemplazará una palabra reservada.
 - 4.3. Sellado archivo XML: usando la información del paso anterior, la cadena original y el certificado digital para sellar el archivo y hacerlo valido.
 - 4.4. Envío vía correo electrónico: Se crea un correo electrónico para el personal del departamento de facturación con copia al personal de sistemas con el archivo XML adjunto.
 - 4.5. Una vez enviado el archivo XML, se respalda el archivo para su almacenamiento en una carpeta dentro del servidor, la cual se estará copiando a una unida de cinta al final del día.

- 4.6. Mes con mes se hace un respaldo acumulativo de los documentos generados en unidades de disco compacto y se entregan al departamento de facturación para su resguardo.

3.5. Diagramas del sistema GeComFiDi

A continuación se muestra el diagrama del sistema GeComFiDi. Dicho diagrama esta dividido en cuatro partes que son: definición de diccionarios y datos de la empresa, la conexión al servidor, el análisis, el tipo de cliente y por ultimo la creación y envío del archivo XML.

La primera parte consiste en la definición de constantes, definición de los datos de conexión al servidor AS/100, definición de los diccionarios de datos EMISOR, CLIENTE, ADDENDA e ITEMS, por ultimo, leer los datos de Empacadora San Marcos.

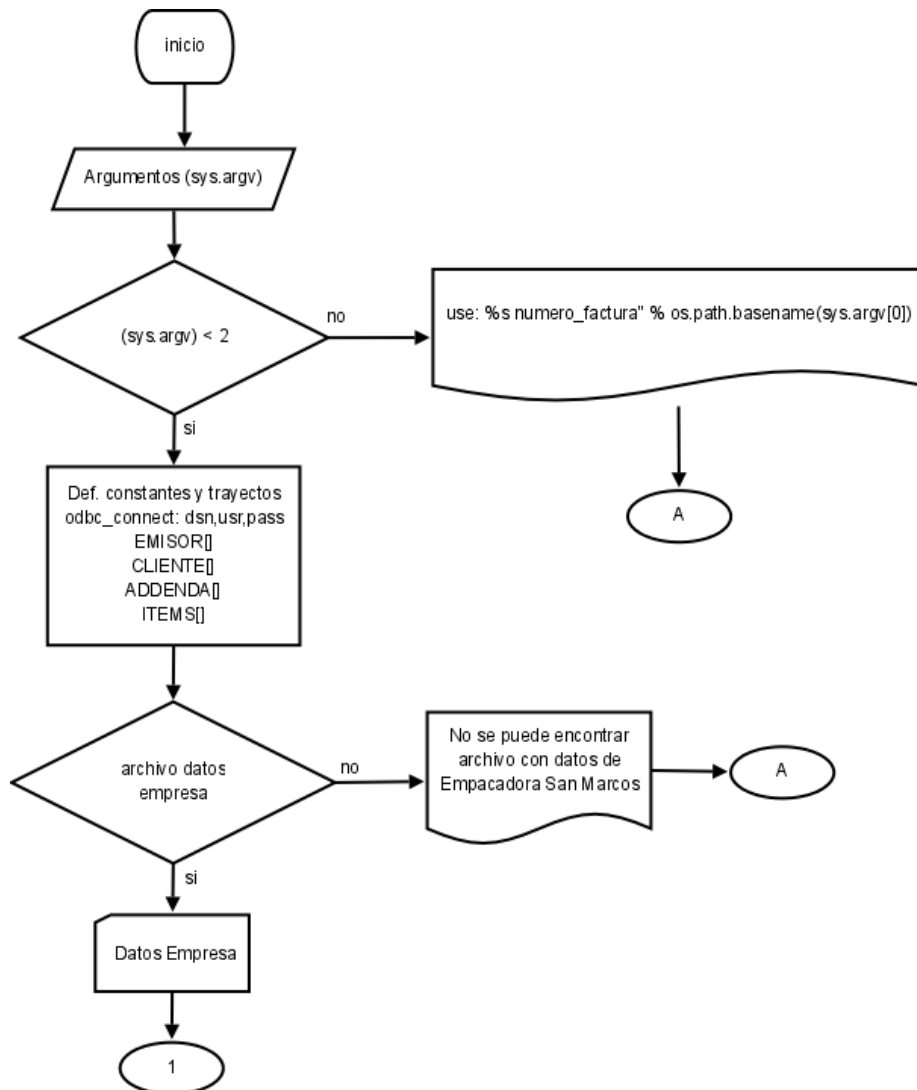


Figura 3.5: creación de constantes, diccionarios de datos y datos de la empresa

En la segunda parte del diagrama del sistema, se realiza la conexión al servidor AS/400 entre python, pyoDB y unixODBC usando los argumentos de nombre del origen de datos, usuario y contraseña, una vez conectados, se consulta la información relacionada al número de factura, consulta donde se extraen datos importantes para crear el resto de las consultas posteriores, con estos primeros datos se llena parte de los diccionarios de datos denominados EMISOR y CLIENTE.

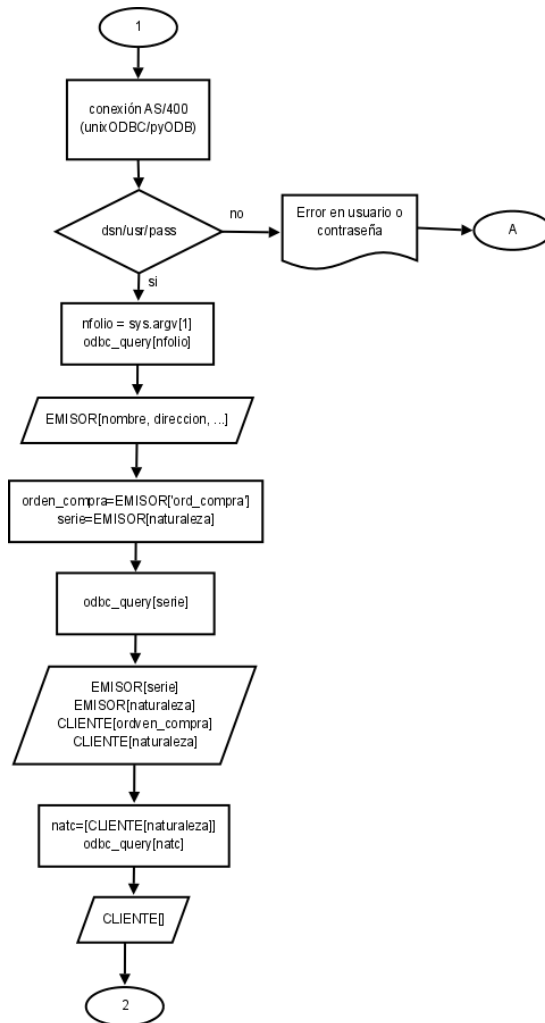


Figura 3.6: Conexión al servidor y llenado de diccionarios de datos

En la Tercera parte, se consulta los datos relacionados al tipo de cliente como lo es: número de tienda, lugar de entrega, etc. También se consulta un dato importante, el respectivo descuento, cada cliente tiene un descuento asignado, un ejemplo las categorías son: Mayoreo, Gobierno y Autoservicio.

Se continúa con el llenado de los diccionarios de datos denominados CLIENTE y ADDENDA.



Figura 3.7: Tipo de cliente, su tipo de descuento y llenado de diccionarios

En la cuarta parte, se enumeran cada una de las partidas que comprende la factura, listando cada uno de los productos que se están facturando. El siguiente paso es normalizar los datos, este proceso consiste en quitar los caracteres que no se pueden imprimir propios del sistema AS/400, por ultimo se crea el archivo XML con los datos de los diccionarios, se sella, se envía por correo electrónico y se crea un vinculo para su descarga en caso de ser necesario.

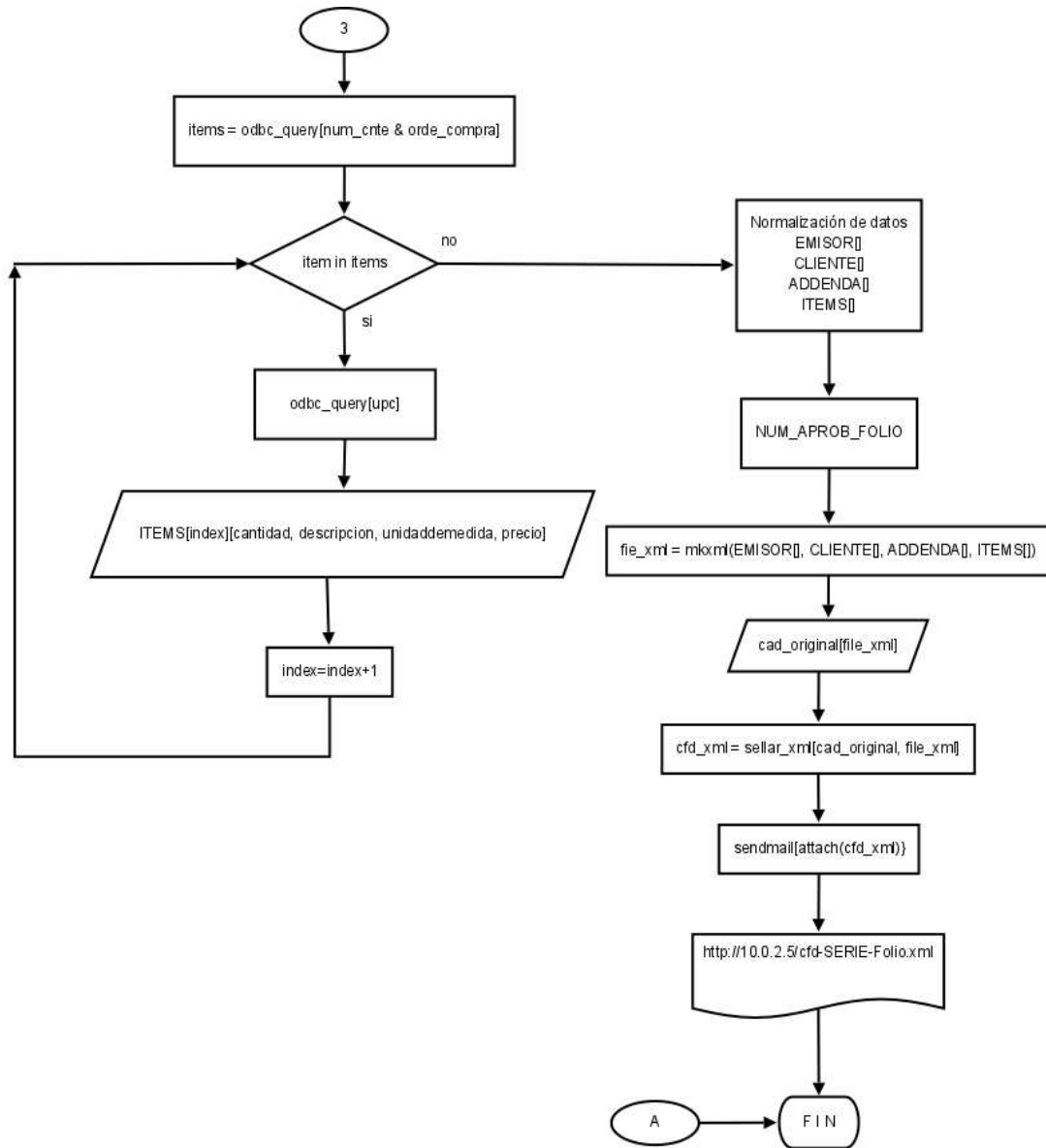


Figura 3.8: listado de productos, normalización y creación del archivo XML

3.6. Descripción funcional

A continuación se describe el funcionamiento del **Sistema Generador de Comprobantes Fiscales Digitales**, el usuario debe seguir los siguientes pasos:

1. Ingresar en un navegador Web a <http://samba.i.esanmarcos.com.mx/cfd> donde encontrarán con la pantalla en la que el personal del departamento de Contabilidad de Empacadora San Marcos debe seleccionar un distribuidor de los diez que aparecen, posteriormente en un campo de texto se deben colocar los folios de las facturas a procesar.

A continuación en la figura 3.9, se muestra un ejemplo, se procesarán las facturas 1, 2 y 3 del distribuidor de la ciudad de México FDMX

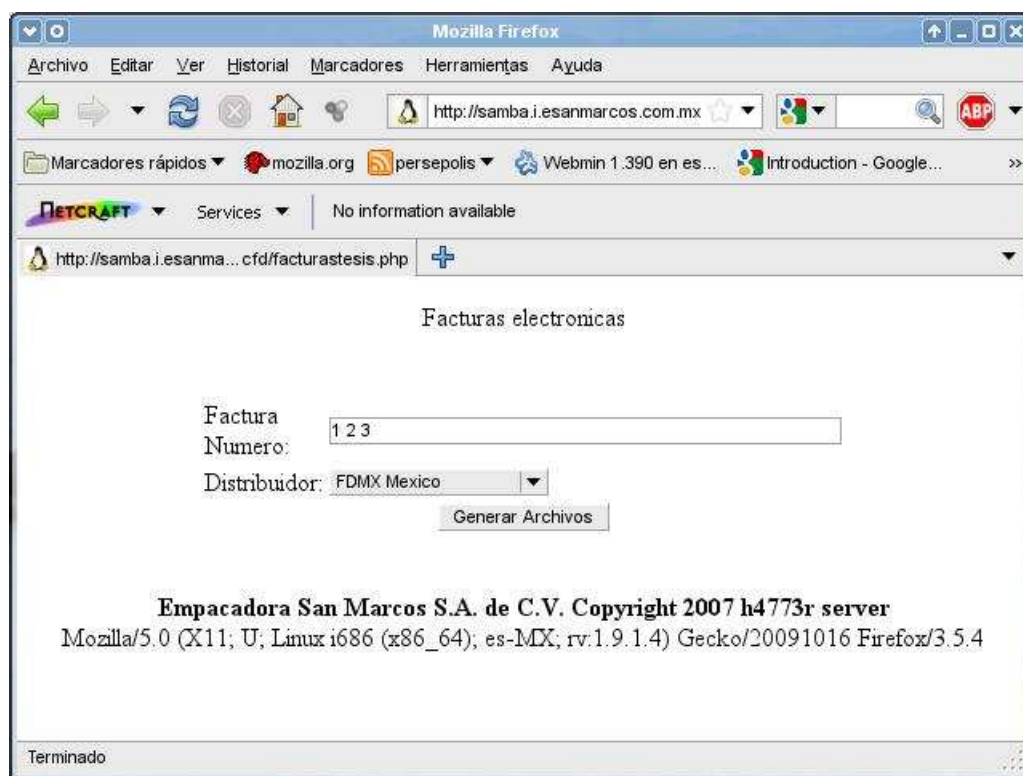


Figura 3.9: Formulario donde se ingresará el folio a procesar y el distribuidor a quien corresponde.

El formulario permite seleccionar uno de los distintos distribuidores, los cuales son:

- Aguas Calientes, FAGS
- Chihuahua, FCHI
- México, FDMX
- Guadalajara, FGDL

- Hermosillo, FHER
- Monterrey, FMTY
- Planta, FPNA
- Sinaloa, FSIN
- San Luis Potosí, FSLP
- Tijuana, FTIJ

una vez llenado el campo del folio o folios y seleccionado el distribuidor correspondiente, el usuario deberá dar clic en el botón “Generar Archivos” que se muestra en el formulario.

2. El Sistema se encarga de procesar los folios de las facturas, tarda aproximadamente 3 minutos en procesar un folio, es decir, hacer las búsquedas en las Bases de Datos y mostrar al usuario una nueva pantalla con ligas a páginas Web que posteriormente nos llevarán a visualizar el Archivo XML procesado.

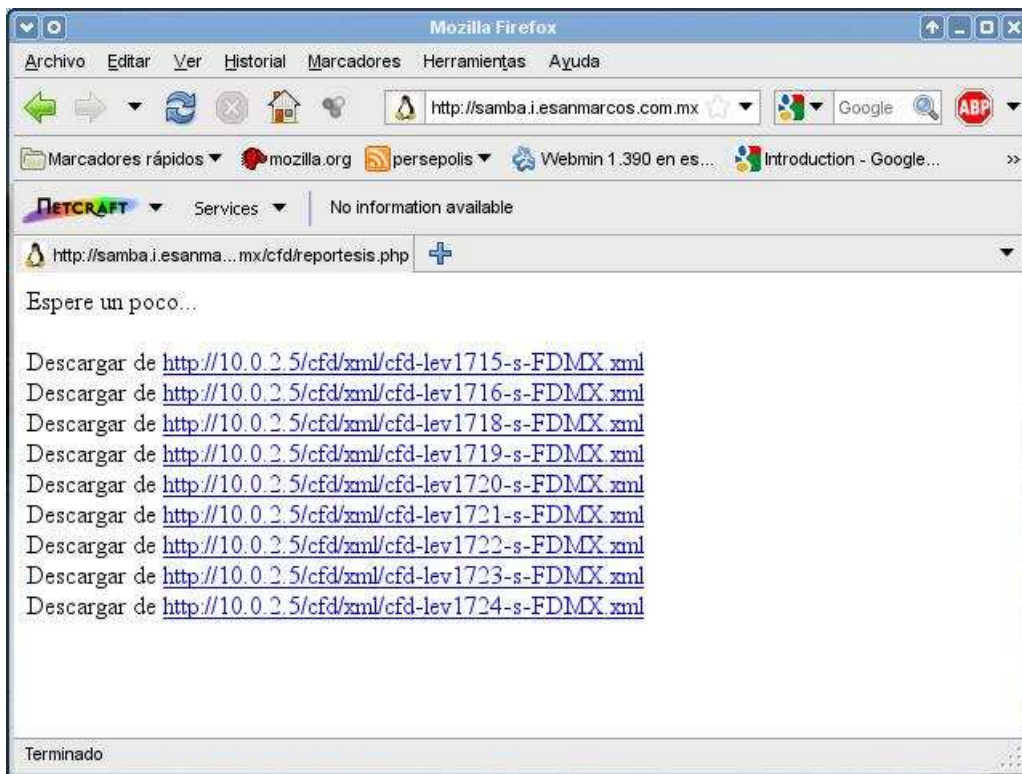


Figura 3.10: Vinculos de descarga a los archivos XML generados

- Posteriormente los archivos XML son entregados vía correo electrónico a los Departamentos de Contabilidad y Sistemas. En este punto también se archivan en el servidor en un directorio especial los cuales serán respaldados al finalizar el día, le ley indica conservar los archivos XML por espacio de 3 años.

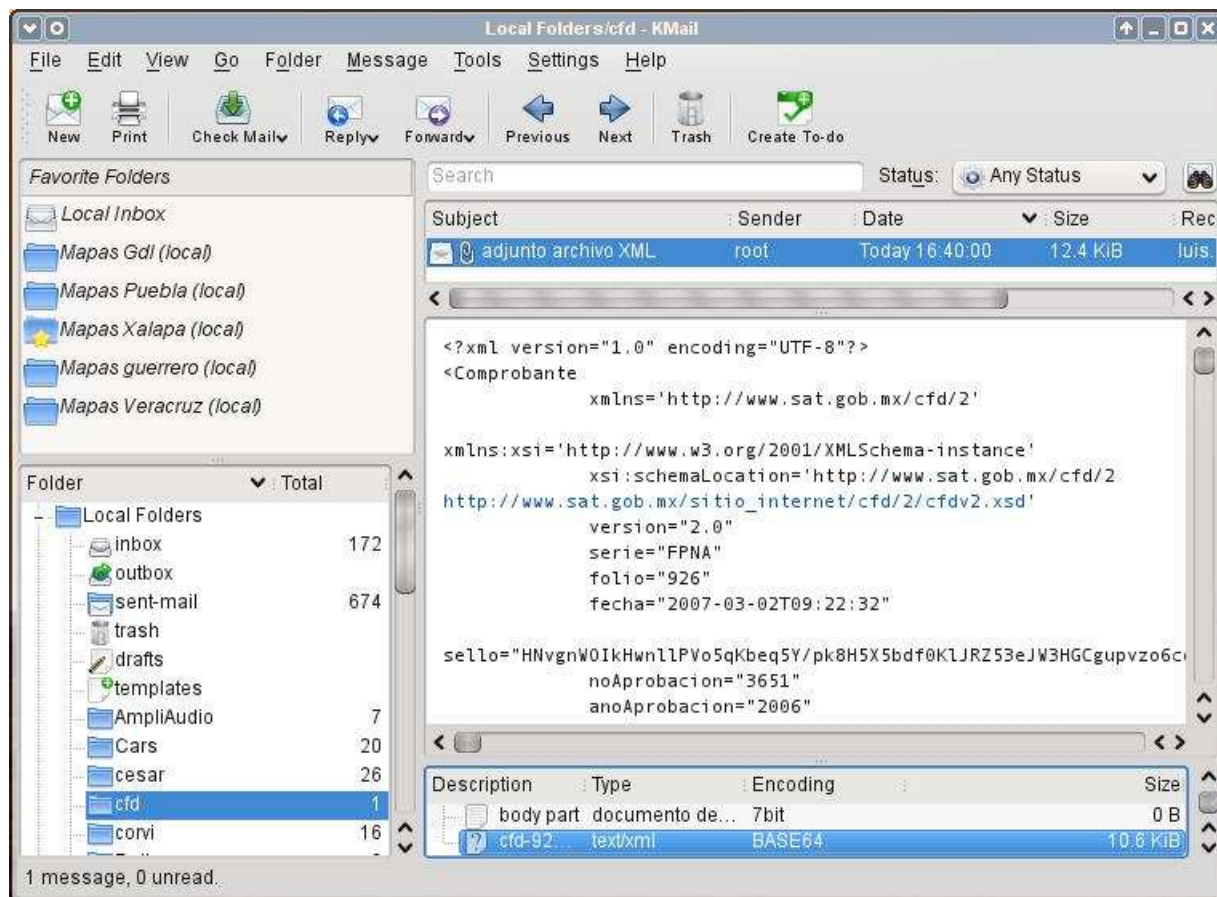


Figura 3.11: Archivo XML visualizado en un cliente de correo electrónico

las ligas a los archivos XML también sirven para descargar dichos archivos en caso de que falle la entrega vía correo electrónico.

- En casos especiales, la mercancía enviada al cliente requiere de un documento para acreditar o sustentar las mercancías o la compra con el cliente, en este caso se imprime una forma semejante a una factura tradicional, con la leyenda correspondiente **Este documento es una impresión de un Comprobante Fiscal Digital.**

A continuación se muestra la forma impresa de un Comprobante Fiscal Digital.

Capítulo 4

Implementación del Sistema GeComFiDi

4.1. Descripción general del Sistema GeComFiDi

Este trabajo de tesis, Sistema Generador de Comprobantes Fiscales Digitales está dividido en tres partes fundamentales:

- La interfaz de usuario, es simple y fácil de usar, el usuario solo tiene contacto con un formulario donde solo selecciona el distribuidor que se desea y se colocan los folios a procesar, enviando la información al script de bash¹ correspondiente al distribuidor seleccionado.
- Los scripts en bash los cuales cumplen el Linux Filesystem Standard, FSSTND² haciendo así fácil de administrar ya que los scripts en bash están en el directorio `/bin` y ellos se encargan de lanzar el programa en python.
- Los módulos y la biblioteca general forman el CorePythonCFD se encuentran almacenados en el directorio `/usr/lib64/python2.4/`, un módulo por cada distribuidor, en total 10 módulos, los cuales se encargan de extraer la información de las distintas tablas de la base de datos que se encuentra en el servidor AS/400.

La elección de usar distintos módulos facilita el poder modificar datos en un distribuidor y los demás seguir facturando.

A continuación se muestra la tabla entre los clientes, los scripts y los programas.

¹bash es un shell de Unix (intérprete de comandos de Unix)

²el cual estipula los nombres, la ubicación y la función de la mayoría de los directorios y los archivos del sistema

Distribuidor	Script bash	módulos python
Aguas Calientes	FAGS	FAGS.py
Chihuahua	FCHI	FCHI.py
México	FDMX	FDMX.py
Guadalajara	FGDL	FGDL.py
Hermosillo	FHER	FHER.py
Monterrey	FMTY	FMTY.py
Planta	FPNA	FPNA.py
Sinaloa	FSIN	FSIN.py
San Luis Potosí	FSLP	FSLP.py
Tijuana	FTIJ	FTIJ.py

4.1.1. La interfaz de usuario.

La interfaz web, fue desarrollada en PHP³ es simple y muy fácil de usar, ya que el formulario usado solo cuenta con tres elementos. El campo de texto para colocar el número o números de folios a generar, un campo de selección de distribuidor correspondiente y el botón de acción para enviar el formulario.

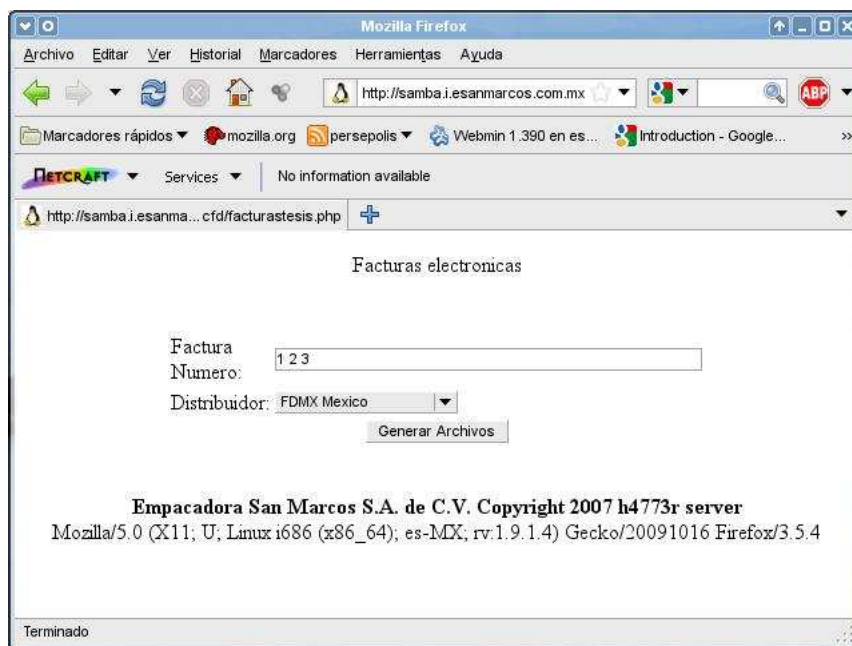


Figura 4.1: interfaz de usuario

³acrónimo de PHP Hypertext Pre-processor, es un lenguaje de programación usado normalmente para la creación de contenido para sitios web

Los dos datos enviados por el formulario servirán de parámetros para la función `shell_exec`⁴ para ejecutar el script adecuado, las variables contienen el nombre del script asociado al distribuidor y el segundo parámetro contiene una cadena de caracteres con el número o números de folios a procesar, dando como resultado la ejecución.

Al terminar la ejecución, la interfaz muestra las ligas en caso de que se quieran descargar los archivos XML generados, una liga por cada folio procesado.

4.1.2. Scripts en bash

El script en bash ejecutará la aplicación tantas veces como elementos tenga el segundo parámetro, es decir si el segundo parámetro contiene `120 121 123`, el script se ejecutará tres veces, es decir el número de argumentos que contenga la cadena separados por un espacio.

A continuación se muestra el script en bash llamando al módulo en python.

```
#!/bin/bash
cd /usr/lib64/python2.4/
for a in $*
do
    python FPNA.py $a
done
```

Los scripts en bash permiten al administrador la generación de folios desde la consola de comandos. A continuación se muestra la ejecución desde la consola de comandos, se muestra el nombre del script, seguido de los folios desde el 1715 hasta el 1723.

```
samba:~ # FDMX 1715 1716 1717 1718 1719 1720 1721 1722 1723
http://10.0.2.5/cfd/xml/cfd-1715-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1716-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1717-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1718-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1719-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1720-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1721-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1722-s-FDMX.xml
http://10.0.2.5/cfd/xml/cfd-1723-s-FDMX.xml
samba:~ #
```

El funcionamiento es similar, ya que los archivos XML son enviados por correo a los destinatarios, pero las ligas de descarga solo son anunciadas, se puede copiar la dirección y pegarla en un navegador web para ver el archivo XML.

⁴`shell_exec` Ejecutar un comando mediante el intérprete de comandos y regresa la salida completa como una cadena

4.1.3. CorePythonCFD

El sistema para la empresa Empacadora San Marcos está desarrollado en base a las necesidades internas, en particular 10 módulos semejantes y una librería general; cada módulo corresponde a un distribuidor, facilitando las modificaciones o actualizaciones especiales necesarias en un distribuidor sin dejar inactivo el resto.

Estos módulos en Python se encargan de la conexión a la base de datos y la extracción de la información, son ejecutados desde Script⁵, los módulos son:

Distribuidor	Script bash	módulos python
Aguas Calientes	FAGS	FAGS.py
Chihuahua	FCHI	FCHI.py
México	FDMX	FDMX.py
Guadalajara	FGDL	FGDL.py
Hermosillo	FHER	FHER.py
Monterrey	FMTY	FMTY.py
Planta	FPNA	FPNA.py
Sinaloa	FSIN	FSIN.py
San Luis Potosí	FSLP	FSLP.py
Tijuana	FTIJ	FTIJ.py

Estos módulos realizan los siguientes pasos:

- Cargar las bibliotecas necesarias, como `pyodb`, `glob`, `string`, `sys`, `os`, `re`, `time`, `stunts`.
- Análisis del número de argumentos mínimos necesarios.
- Lectura del archivo que contiene la información de la Empresa Empacadora San Marcos S.A. de C.V.
- Prepara la conexión al servidor AS/400 con datos como nombre del servidor, usuario y contraseña.
- Conexión con las tablas de la base de datos DB2 llamada AMFLIBT que contiene los datos necesarios, por ejemplo las tablas: MBF9REP (Encabezado de facturas), ITEMASA (Información del inventario de productos), MBCUCPS1 (Libro de precios por detalle), MBDHREP (Encabezados de datos para embarque).
- Al terminar de recolectar la información, hace un llamado a las funciones de la biblioteca general, para normalizar los datos y generar el archivo XML.

⁵Permiten la automatización de tareas creando pequeñas utilidades. Es muy utilizado para la administración de sistemas UNIX

La librería general denominada `stunts.py` es llamada desde los módulos Python y se encarga de crear los distintos diccionarios de datos, definir los trayectos de almacenamiento, destinatarios del correo electrónico, eliminación de caracteres no imprimibles, es decir los procedimientos relacionados a los datos que ya fueron extraídos de la base de datos, la librería contiene los siguientes procedimientos:

- Declaración de los diccionarios de datos EMISOR, RECEPTOR, ITEMS y ADDENDA.
- Definición del trayecto a los directorios donde se almacenarán los archivos XML.
- Definición de los destinatarios de correo electrónico y el procedimiento para enviar el correo.
- Los distintos procesos para normalizar los datos: Corregir decimales, eliminar espacios y caracteres inválidos, etc.
- El procedimiento que asigna el número de aprobación y año de aprobación a cada Folio.
- El procedimiento para crear el archivo XML con la información contenida en los diccionarios de datos.
- El proceso que hace las llamadas al sistema para extraer la Cadena Original del archivo XML y usar el certificado digital para poder sellar el archivo XML y darle la validez.
- Envía por correo electrónico el archivo XML.
- Regresa una cadena de caracteres por cada folio procesado, esta cadena es una liga para descargar dicho archivo XML en caso de ser necesario

Al terminar la ejecución del módulo y la librería general, una cadena de caracteres es retornada por cada folio procesado y utilizada en la interfaz de usuario realizada en PHP, dicha cadena de caracteres es utilizada para crear la liga de donde se podrán descargar el archivo XML, es decir, si el usuario procesa 25 folios, se crearán 25 ligas, cada una asignada a un folio procesado el cual corresponde a un archivo XML.

A continuación se muestra un ejemplo de las ligas creadas hacia los archivos XML, la cual permite visualizar dicho archivo.

```
http://10.0.2.5/cfd/xml/cfd-1722-s-FDMX.xml
```

La liga permite ver el archivo XML con folio 1722 del distribuidor de la ciudad de México.

La figura 4.2 muestra un pequeño diagrama sobre el proceso de creación de los Comprobantes Fiscales Digitales.

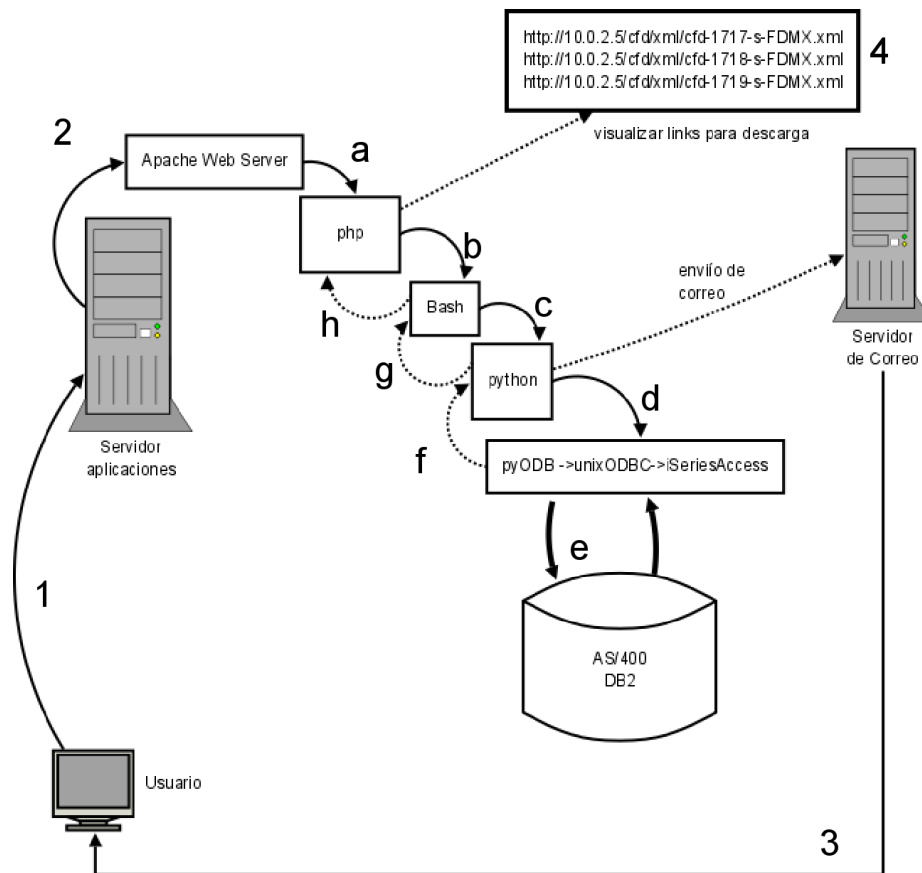


Figura 4.2: Proceso de generación de comprobantes fiscales digitales

La descripción del proceso de generación de los comprobantes fiscales es la siguiente:

1. El usuario intenta conectarse al servidor de aplicaciones.
2. El servidor responde la petición de conexión.
 - a) ApacheWebServer acepta petición y usa el lenguaje PHP para proveer una página web donde se encuentra un formulario. El usuario llena el formulario con el folio o folios y selecciona el distribuidor.
 - b) PHP ejecuta el script de Bash correspondiente a los datos llenados en el formulario.
 - c) El script de Bash ejecuta el módulo de Python con los argumentos necesarios.
 - d) Python utiliza pyODB para comunicarse con unixODBC y el driver de iSeriesAccess para consultar la base de datos DB2 en el servidor AS/400.

- e) Al terminar las consultas y tener todos los datos, se crea el archivo XML.
 - f) El archivo XML es enviado por correo electrónico.
 - g) Una cadena de texto es regresada con la ubicación del archivo XML.
 - h) PHP crea una liga para cada folio procesador de su correspondiente archivo XML.
3. el usuario recibe un correo electrónico con un archivo XML.
 4. Se muestra la o las ligas para ver o descargar los archivos XML procesados.

4.2. Diagrama de Bloques de ejecución del sistema Ge-ComFiDi

El siguiente diagrama de bloques muestra paso a paso el proceso de recolección de los datos necesarios, es decir las distintas consultas que se realizan para poder obtener los datos y colocarlos en los arreglos correspondientes, normalizar su contenido y que después serán utilizados para crear el archivo XML, se sellará y se enviará por correo electrónico

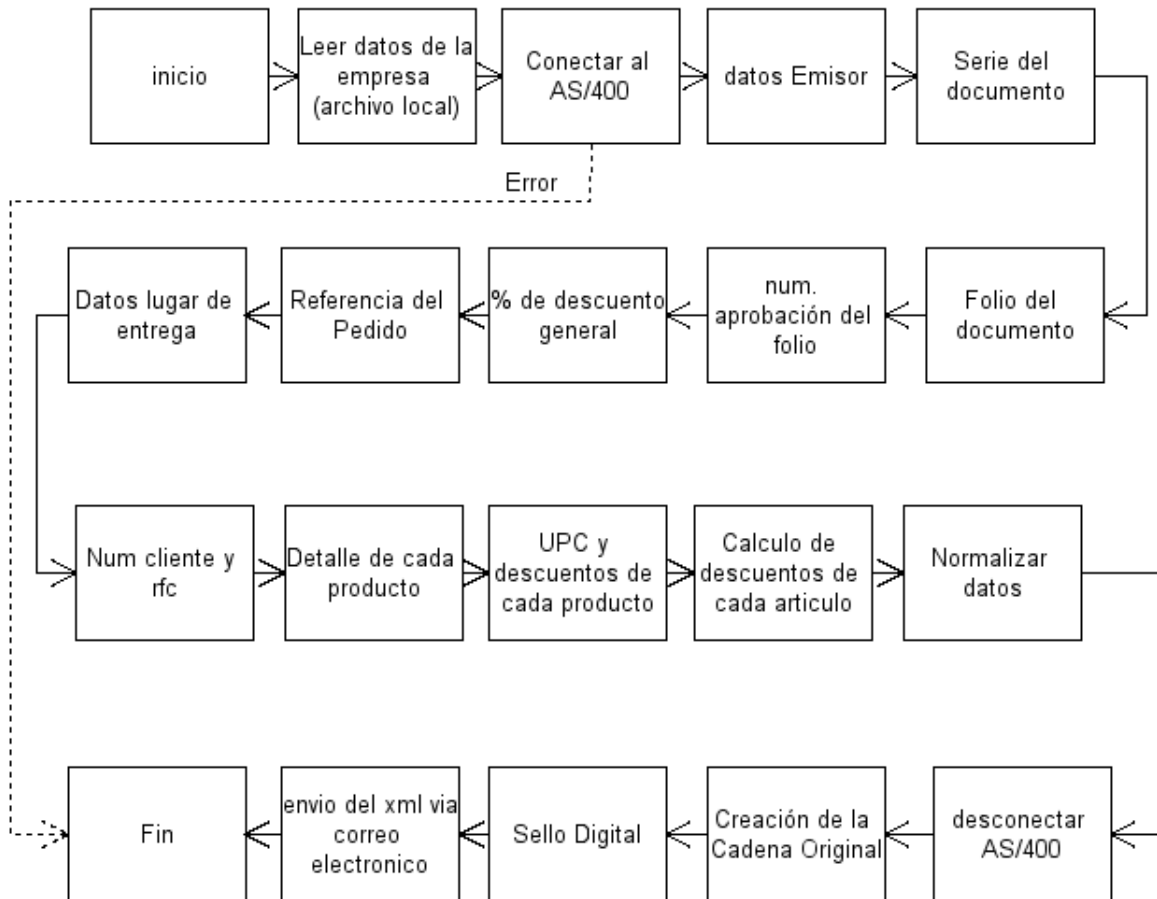


Figura 4.3: Diagrama de bloques

4.3. Descripción del código

En esta sección se describe el código en Python que crea el archivo XML usando la información contenida en los diccionarios de datos, es decir, se le agrega la información del emisor como serie, folio, fecha, número de aprobación, número de certificado, el RFC⁶ del emisor del Comprobante Fiscal Digital, el nombre del emisor, la dirección fiscal del emisor, etc. También los datos del cliente, tales como su domicilio fiscal, el lugar de entrega.

Se le agregan dos cadenas temporales las cuales serán reemplazadas en el momento de ser selladas, una tendrá el sello digital y otra tendrá su cadena original.

Para el listado de los productos, se recorre el arreglo correspondiente con los datos relacionados a dicho producto como por ejemplo: su código, precio unitario, peso, etc.

Por ultimo se le agrega el campo correspondiente a cada cliente con los datos que particularmente dicho cliente solicita, a este campo se le denomina ADDENDA y contiene datos como: tipo de empaque, número de piezas por caja, etc.

4.3.1. Creación archivo XML

El archivo XML se ve de la siguiente forma:

```
1. fxml.write('<?XML version="1.0" encoding="UTF-8"?>')
2. fxml.write('<Comprobante')
3. fxml.write('XMLns="http://www.sat.gob.mx/cfd/2')
4. fxml.write('XMLns:xsi="http://www.w3.org/2001/XMLSchema-instance')
5. fxml.write('xsi:schemaLocation="http://www.sat.gob.mx/cfd/2
    http://www.sat.gob.mx/sitio_internet/cfd/2/cfdv2.xsd')
6. fxml.write('version="2.0" ')
7. fxml.write('serie="EMISOR['serie']')
8. fxml.write('folio="EMISOR['factura']')
9. fxml.write('fecha="EMISOR['fecha']+T'+EMISOR['hora']')
10. fxml.write('sello="EMISOR['selloApro']')
11. fxml.write('noAprobacion="EMISOR['selloApro']')
12. fxml.write('noCertificado="EMISOR['certificado']+'>')
```

En la línea 1. se especifica que el archivo XML esta codificado en formato UTF-8, en la línea 2 es donde inician los campos propios del comprobante fiscal digital.

De la línea 3 a la línea 6 son las especificaciones que indican el tipo de validación utilizado. La línea que especifica XML:xsi="http://www.w3.org/2001/XMLSchema-instance" indica que se está usando validación mediante el estándar de esquema XSD. Según documentación del Diario Oficial de la Federación, anexo 20 de la miscelánea fiscal 2006.

⁶Registro Federal de Contribuyentes

De la línea 6 a la línea 12 son los nodos que contienen los datos relacionados con el emisor, es decir la serie, folio, fecha y hora de creación, el número de sello digital, el número de aprobación de dicho sello y el certificado que da validez a dicho sello.

```

13. fxml.write('<Emisor rfc='+EMISOR['rfc']')
14. fxml.write('nombre='+EMISOR['nombre']+'>')
15. fxml.write('<DomicilioFiscal calle='+EMISOR['direccion']')
16. fxml.write('municipio='+EMISOR['municipio']')
17. fxml.write('estado='+EMISOR['estado']')
18. fxml.write('pais='+EMISOR['pais']')
19. fxml.write('codigoPostal='+EMISOR['cp']/>')
20. fxml.write('</Emisor>')

```

De la línea 13 a la 20 se agregan al archivo XML los datos correspondientes al emisor, tales como el registro federal de contribuyentes (R. F. C.), nombre del emisor del comprobante fiscal digital, el domicilio fiscal, es decir la dirección postal de quien emite el comprobante, el municipio, el estado, el país y el código postal.

```

21. fxml.write('<Receptor rfc='+CLIENTE['rfc']')
22. fxml.write('nombre='+CLIENTE['nombre']>')
23. fxml.write('<Domicilio calle='+CLIENTE['direccion']')
24. fxml.write('colonia='+CLIENTE['colonia']+')
25. fxml.write('municipio='+CLIENTE['localidad']')
26. fxml.write('estado='+CLIENTE['estado']')
27. fxml.write('pais='+CLIENTE['pais']'/>')
28. fxml.write('</Receptor>')

```

De la línea 21 a la 28 corresponde a los datos relacionados al cliente, estos datos son similares a los datos requeridos para el emisor, por ejemplo el registro federal de contribuyentes (R. F. C.), el nombre, su domicilio, la colonia, el municipio, el estado y el país del receptor.

```

29. fxml.write('<Conceptos>')
30. for x in ITEMS:
31.     array = ITEMS[x]
32.     fxml.write('<Concepto cantidad='+array[0]')
33.     fxml.write('unidad='+array[2]')
34.     fxml.write('descripcion='+array[1]')
35.     fxml.write('valorUnitario='+array[4]')
36.     fxml.write('importe='+pesos(array[5])'/>')
37. fxml.write('</Conceptos>')

```

De la línea 29 a la línea 37 se agregan al archivo el detalle de los productos comercializados y sus características como lo son su descripción, el costo unitario y el importe.

```
38. fxml.write('<Impuestos>')
39.   fxml.write('<Traslados>')
40.     fxml.write('<Traslado impuesto="IVA" importe="0"/>')
41.   fxml.write('</Traslados>')
42. fxml.write('</Impuestos>')
```

En las líneas 38 a la 42 se agregan los impuestos a los productos que se mencionan en la factura, por disposición federal, los alimentos tienen tasa cero de impuesto al valor agregado.

A continuación se describe un ejemplo del contenido del campo ADDENDA, este campo es usado para información extra entre el vendedor y el cliente previamente convenida.

```
43 fxml.write('<Addenda>')
44   fxml.write('<requestForPayment
      type="SimpleInvoiceType"
      contentVersion="1.3.1"
      documentStructureVersion="AMC005"
      documentStatus="ORIGINAL">')
45.   fxml.write ('<requestForPaymentIdentification>')
46.     fxml.write ('<entityType>INVOICE</entityType>')
47.     fxml.write ('<uniqueCreatorIdentification>'
      'EMISOR['serie']+EMISOR['folio']'
      '</uniqueCreatorIdentification>')
48.   fxml.write ('</requestForPaymentIdentification>')
49.   fxml.write ('<orderIdentification>')
50.     fxml.write ('<referenceIdentification
      type="ON">'
      '+EMISOR['ordcompra']
      '+</referenceIdentification>')
51.     fxml.write ('<ReferenceDate>'
      '+VADDENDA['fecorcom']
      '+</ReferenceDate>')
52.   fxml.write ('</orderIdentification>')
   :
   :
95.   fxml.write ('</Addenda>')
96. fxml.write ('</Comprobante>')
```

Este es parte del contenido que Grupo Soriana requiere en el campo ADDENDA en los comprobantes fiscales digitales.

En la línea 43 inicia el campo denominado ADDENDA.

En la línea 44 según indicaciones de soriana se agrega el campo *requestForPayment* y

su contenido `contentVersion="1.3.1" documentStructureVersion="AMC005" documentStatus="ORIGINAL"`.

En la línea 45, 46 y 47 el campo `requestForPaymentIdentification` que identifica el tipo de comprobante fiscal digital y contiene los siguientes elementos: `<entityType>INVOICE </entityType>`, `<uniqueCreatorIdentification>' + EMISOR['serie'] + EMISOR['folio'] + '</uniqueCreatorIdentification>`

En la línea 48 se cierra el identificador de campo `</requestForPaymentIdentification>`

En la Línea 49 a la 52 mas datos particulares como lo son el número de la orden de compra, la fecha de la orden de compra y otros mas por ejemplo: el Global Location number o GLN del cliente, el GLN del vendedor, la serie y el folio del comprobante fiscal, el tipo de moneda usada en la factura, el código de barras del producto, la descripción de cada uno de los productos, etc.

La línea 95 cierra el campo de datos ADDENDA y la línea 96 cierra el identificador de todo el documento XML denominado comprobante.

4.3.2. Creación de la Cadena Original y el Sello Digital.

Una vez que se extrae la cadena original del archivo XML, esta es usada en combinación con el certificado que el Sistema de Administración Tributaria otorga a las empresas, en nuestro caso se usará el certificado de prueba.

El siguiente código toma la cadena original y usando la herramienta openssl usa el certificado para poder generar el sello en base64

```
xsltproc cadena_originalv2.xsl cfd | openssl dgst -md5 -sign
AAA010101AAA.key.pem | openssl enc -base64 -A
```

Posteriormente el sello generado se le agrega al archivo XML reemplazando la cadena `HATTER_SELLO_KEY` por el correspondiente

```
1. def sellar_XML(cfd,sello):
2.     stext='HATTER_SELLO_KEY'
3.     for line in fileinput.input(cfd,inplace=1):
4.         lineno = 0
5.         lineno = string.find(line, stext)
6.         if lineno >0:
7.             line=line.replace(stext, sello)
8.             sys.stdout.write(line)
9.     return 0
```

Finalmente se tiene el archivo XML completamente valido, es decir con todos los requerimientos legales establecidos en el Sistema de Administración Tributaria (SAT).

4.4. Desarrollo, Pruebas y Resultados

4.4.1. Instalación y configuración del servidor de aplicaciones Linux.

El departamento de tecnologías de la información se encarga de instalar el software necesario en el servidor Linux, es decir se instala ApacheWebServer, Python y su paquete de desarrollo, PHP, pyODB, unixODBC y su paquete de desarrollo y el driver de IBM iSeriesAccess.

El Sistema Generador de Comprobantes Fiscales Digitales se probó en dos partes:

- Pruebas de extracción de datos: Las pruebas iniciaron en el departamento de Tecnologías de la información, con la comunicación con la base de datos del servidor AS/400, usando números de folio de prueba y consultado las tablas que contienen la información necesaria, verificando que los datos cumplan los requisitos y que la información este completa.
- Pruebas de validación de información. Una vez que se reunió toda la información necesaria para crear el Comprobante Fiscal Digital, el departamento de Facturación y el departamento de Tecnologías de la Información, hicieron pruebas de sellado con el certificado digital de prueba proporcionado por el Sistema de Administración Tributaria. Los Comprobantes Fiscales Digitales fueron enviados a la Asociación Mexicana de Estándares para el Comercio Electrónico y los clientes para validar el contenido mínimo requerido.

El Sistema desarrollado denominado Sistema Generador de Comprobantes Fiscales Digitales, se instaló en la empresa Empacadora San Marcos donde actualmente se esta usando. El sistema GeComFiDi genera alrededor de 50 facturas por día, esto da como resultado un aproximado de 15,000 facturas generadas desde la implementación de este sistema.

4.4.2. Actualizaciones de software en el servidor de aplicaciones Linux.

El mantenimiento al Sistema Generador de Comprobantes Fiscales Digitales consta principalmente de actualizaciones de las versiones del lenguaje de programación y a los módulos como pyODB e iSeriesAccess, a los módulos en Python se les hacen cambios cada que los requisitos de un cliente cambian, ya sea para agregar campos al diccionario Addenda o cuando lo indica la Asociación Mexicana de Estándares para el Comercio Electrónico de acuerdo al la miscelánea fiscal publicada en el Diario Oficial de la Federación.

4.4.3. Instalación del Sistema Generador de Comprobantes Fiscales Digitales.

El sistema Generador de Comprobantes Fiscales Digitales se empaqueta en un archivo con formato `tar.gz` posteriormente se copia el archivo comprimido denominado "gecomfidi.tar.gz"

que contiene los módulos, la librería general, los datos de la empresa Empacadora San Marcos y un script para su instalación, al desempaquetar el archivo usando el comando `tar -xvfz gecomfidi.tar.gz`, tendremos una carpeta llamada `gecomfidi`, al ingresar en ella solo ejecutaremos como super usuario, el comando `sh install.sh` para que se haga una pequeña prueba de dependencias y se instalen los componentes necesarios dentro de las carpetas adecuadas.

4.4.4. Estado Actual del Trabajo de Tesis.

Actualmente el Sistema Generador de Comprobantes Fiscales Digitales se encuentra trabajando y dando servicio a los distintos distribuidores como lo son Aguascalientes, Chihuahua, México DF, Guadalajara, Hermosillo, Monterrey, Sinaloa, San Luis Potosí, Tijuana y manteniendo el comercio con los distintos clientes como lo son, Chedraui, Comercial Mexicana, Grupo Sahuayo/Corvi, HEB.

Capítulo 5

Conclusiones y Perspectivas

Tomando como base los resultados obtenidos en el desarrollo de este trabajo de tesis, se puede decir que satisfactoriamente se realizó un Sistema Generador de Comprobantes Fiscales Digitales y que se han cumplido el objetivo general y los particulares anteriormente planteados.

- Se instaló OpenSuSE Linux como plataforma para el desarrollo el Sistema Generador de Comprobantes Fiscales Digitales (GeComFiDi).
- Se instaló y configuró Apache Web Server en el Servidor de Aplicaciones Linux como servidor Web.
- Se instaló el lenguaje de programación Python y sus complementos de desarrollo.
- Se instaló y configuró unixODBC y complementos de desarrollo, para la conexión entre el servidor Linux y la base de datos DB2 alojada en el servidor AS/400.
- Se instaló y configuró pyoDB, módulo de unixODBC para establecer conexiones en aplicaciones desarrolladas en Python.
- Se instaló y configuró iSeriesAccess, módulo de unixODBC que permitió establecer la conexión al servidor AS/400.
- Se configuró y verificó la conexión del servidor de aplicaciones Linux con el servidor AS/400
- Se desarrolló la interfaz Web para el Sistema Generador de Comprobantes Fiscales Digitales.
- Se Programó un script en Python que obtiene información de las tablas de la Base de Datos ubicada en el servidor AS/400.
- Se creó un script que normaliza los datos que son necesarios para llenar los campos del archivo XML.

- Se creó un script que genere el archivo XML listo para sellarse y ser valido como Comprobante Fiscal Digital.
- Se creó un script que usa el certificado digital del SAT para sellar el archivo XML y le dá validez.
- Se creó un script que envía por correo electrónico el archivo XML.
- Se creó un script para el respaldo mensual de los archivos XML.
- El sistema logra la integración funcional entre servidores AS/400 y Servidores Linux.

- El desarrollo fue hecho con lenguaje de programación Python que permite desarrollar software con más velocidad, ofrece más apoyo e infraestructura para programas grandes que el intérprete de órdenes, tiene incluidos tipos de datos de alto nivel, como matrices flexibles y diccionarios, que llevarían días de programación en C, permite dividir su programa en módulos reutilizables desde otros programas, Python viene con una gran colección de módulos estándar que es posible utilizar como base de programas.

- Al lograr conectarse al servidor AS/400 y poder acceder a la base de datos, se da la posibilidad de seguir desarrollando más soluciones con software libre.

Esto es en la parte del área de Tecnológicas de la Información, en el área Fiscal, se cumplieron los siguientes aspectos.

- El sistema desarrollado cumple con los requerimientos establecidos por el Sistema de Administración Tributaria (SAT) y la Asociación Mexicana de Estándares para el Comercio Electrónico (AMECE).

- El sistema se puede ir actualizado, de acuerdo a los cambios requeridos por el SAT o en requerimientos de acuerdo a cada distribuidor, también actualizar de acuerdo a las nuevas versiones del lenguaje de programación y módulos como pyODB o el controlador iSeriesAccess.

- El sistema cumple con la necesidad de Empacadora San Marcos al generar las Facturas Electrónicas; día a día las necesidades crecen al irse agregando más socios comerciales al nuevo esquema.

- La creación de este sistema tiene un impacto de ahorro económico en la empresa, ya que se evitó comprar un sistema con un costo aproximado de \$ 50, 000.00 pesos, más costos de licencias e instalación.

En general, este sistema cuenta con una interfaz amigable para el usuario y sobre todo funcional. Y como conclusión final, tenemos que el sistema GeComFiD, se encuentra actualmente funcionando al 100 % y brindando solución al requerimiento de ingresar al nuevo esquema de comprobantes fiscales digitales.

El impacto de este trabajo de tesis en las labores diarias del departamento de Facturación y de sistemas es importante, ya que ha permitido a la empresa posicionarse entre aquellas que actualmente cuenta con el proceso de Facturación Electrónica, cumpliendo con los estándares marcados por el SAT y los departamentos contables involucrados. Si bien es cierto, son muchos los beneficios con que se cuenta actualmente en Empacadora San Marcos, también podemos hablar de nueva funcionalidades con las que el Sistema GeComFiDi puede contar y mejorar el proceso antes mencionado.

Perspectivas:

- Mostrar indicadores del número de facturas generadas a la semana y mes.
- Agregar un control de acceso y estadísticas de uso que permita generar informes para los encargados de sistemas y de facturación.
- Desarrollar una extensión al sistema que permita la búsqueda de documentos y reimpresión en papel.
- Desarrollar una extensión al sistema que permita mandar alertas vía correo electrónico cuando se presente un error y no se pueda procesar un folio de alguna factura.
- Integrar una interfaz móvil WAP¹
- Número de facturas emitidas hasta el momento.

¹Wireless Application Protocol o WAP (protocolo de aplicaciones inalámbricas) es un estándar abierto internacional para aplicaciones que utilizan las comunicaciones inalámbricas, p.ej. acceso a servicios de Internet desde un teléfono móvil.

Apéndice A

Instalación básica de paquetes en GNU/Linux

En este apéndice, mostraremos las formas mas comunes de instalar paquetes en Linux, tomaremos cuatro tipos de instalaciones Paquetes tipo `tar.gz`, paquetes tipo `rpm`, paquetes tipo `bin` y los paquetes tipo `gz/deb`.

Instalar un nuevo paquete o programa en GNU/Linux no es tan fácil como en sistemas operativos Windows, para instalar un paquete en GNU/Linux se requiere saber que tipo de empaquetado tiene y ser o tener los privilegios de root o administrador.

A.1. Los paquetes rpm

rpm Package manager originalmente llamado **Red Hat Package Manager**, es una herramienta de administración de paquetes pensada básicamente para Linux. Es capaz de instalar, actualizar, des-instalar, verificar y solicitar programas. Originalmente desarrollado por Red Hat para Red Hat Linux, en la actualidad muchas distribuciones Linux lo usan dentro de los cuales las más destacadas son Fedora Linux, MandrivaLinux, SuSE Linux y Conectiva Linux.

El proceso básico para instalar un nuevo paquete desde la consola de comandos se requiere tener privilegios de root y usar el siguiente comando:

```
rpm -Uvhi paquete1.rpm paquete2.rpm ... paquete-n.rpm
```

donde después del comando rpm se listan las opciones:

- U** Esto actualiza o instala el paquete, es similar a instalarlo con `-i` pero si existe elimina la versión anterior e instala la nueva.
- v** Muestra información del progreso de la instalación, generalmente el progreso de la instalación de los paquetes y la suma de comprobación hash.

-h Imprime 50 marcas hash en el paquete desempaquetado, use `-v` para una mejor visualización de la comprobación hash

-i Esto instala un nuevo paquete

por último el paquete o paquetes a instalar y listo, el nuevo paquete esta instalado.

A.2. Paquetes bin

Los paquetes tipo bin, son archivos también llamados paquetes auto extraibles, generalmente para distribuir software precompilado o módulos para dar soporte a hardware, por ejemplo el controlador de la tarjeta nVidia®. o la distribución de java®.

A continuación un ejemplo de como instalar el entorno de ejecución de java®.

1. En el terminal, escriba:`su`.
2. Escriba la contraseña de usuario `root`.
3. Cambie al directorio en el que desee efectuar la instalación. Escriba: `cd`
4. Por ejemplo, para instalar el software en el directorio `/usr/java/`, escriba: `cd /usr/java`
Nota acerca del acceso de usuario root: Para instalar el JRE en una ubicación accesible desde todo el sistema, como `/usr/local`, deberá iniciar la sesión como usuario root para obtener los permisos necesarios. Si no tiene acceso de usuario root, instale el JRE en su directorio de inicio o en un subdirectorío para el que disponga de permiso de escritura.
5. Cambie los permisos del archivo descargado para hacerlo ejecutable. Escriba:
`chmod a+x jre-1_5_0_02-linux-amd64.bin`
6. Inicie el proceso de instalación. Escriba:
`./jre-1_5_0_02-linux-amd64.bin`

siga las instrucciones y datos requeridos como el contrato de licencia, etc, el paquete quedará instalado.

A.3. Paquetes tgz y deb

Algunas distribuciones de Linux tienen su propia forma de empaquetar programas, en el caso de Slackware¹ y Debian², cada uno utiliza un método distinto, para Slackware el empaquetamiento es en formato `.tgz`, una variante de archivos precompilados y empaquetamiento tar y compresión zip, mientras que Debian usa uno semejante pero con extensión `.deb`

Para instalar un paquete en Slackware se siguen los siguientes pasos:

¹es una distribución de Linux. Es una de las más antiguas distribuciones y la más antigua de las activamente mantenidas en la actualidad.

²Debian Project mantiene una distribución de Linux y es una comunidad conformada por desarrolladores y usuarios, que pretende crear y mantener un sistema operativo GNU basado en software libre

1. En el terminal, escriba: `su`.
2. Escriba la contraseña de usuario `root`.
3. Cambie al directorio donde se encuentra en programa a instalar, por ejemplo `/usr/src/software`
4. Por ejemplo, para instalar el software `wine`³, entrar en el directorio donde se descargó por ejemplo `/usr/src/software`
5. Inicie el proceso de instalación. Escriba:
`installpkg wine-0.9.49-i486-1kjz.tgz`

Al final del proceso el paquete `wine` quedará instalado.

En Debian, el proceso es similar

1. En el terminal, escriba: `su`.
2. Escriba la contraseña de usuario `root`.
3. Cambie al directorio donde se encuentra en programa a instalar, por ejemplo `/usr/src/software`
4. Por ejemplo, para instalar el software `wine`, entrar en el directorio donde se descargó
5. Inicie el proceso de instalación. Escriba:
`dpkg -i wine-0.9.49.x86.deb`

Al final del proceso el paquete `wine` quedará instalado.

A.4. Paquetes `tar.gz` o `tar.bz2`

Los paquetes distribuidos en formato `tar.gz` o `tar.bz2` regularmente contienen archivos fuente, es decir son los programas sin compilar por lo tanto el usuario deberá compilarlos y después instalarlos, este tipo de paquetes se pueden instalar en cualquier distribución de GNU/Linux. Esta forma de empaquetar archivos usa dos herramientas, la herramienta `tar` para crear un solo archivo a partir de uno o varios y la herramienta `gzip` o `bzip2` que nos sirve para comprimir el archivo.

`tar` es usado para almacenar archivos y directorios en un solo archivo. Dentro de los entornos Unix `tar` aparece como una orden que puede ser ejecutada desde la línea de órdenes de una consola de texto o desde un simple terminal, Su nombre proviene de `tape archive` y

³acrónimo recursivo que en inglés significa Wine Is Not an Emulator - Wine no es un emulador, es una implementación de la API de Win16 y Win32 para sistemas operativos basados en Unix bajo plataformas Intel

sirve para almacenar varios archivos o directorios dentro de un solo archivo ya sea dentro de una cinta o en un disco.

Para **tar**, sus opciones comunes son:

- x extrae archivos contenidos en el archivo tar
- v Muestra en pantalla el progreso
- f utiliza un archivo o el dispositivo ARCHIVO
- z filtra el archivo a través de gzip

Los archivos **.gz** o **bz2** son archivos comprimidos cada uno tiene su propio método, pero el resultado es reducir el tamaño del archivo inicial, en este caso, se intenta reducir el tamaño del archivo **tar**.

Para **gz** o **gzip** la herramienta de compresión de datos, sus opciones comunes son:

- d Descomprime
- v Muestra en pantalla el progreso

Otra herramienta de compresión de datos es la herramienta **bzip2**, sus opciones comunes son:

- d Descomprime
- v Muestra en pantalla el progreso

Ahora, para extraer el contenido de un archivo **tar.gz** se hará de la siguiente forma:

1. En el terminal, escriba:**su**.
2. Escriba la contraseña de usuario **root**.
3. Cambie al directorio donde se encuentra en programa a instalar, por ejemplo **/usr/src/software**
4. Por ejemplo, para instalar el software **gtk-gnutella**, entrar en el directorio donde se descargó, por ejemplo **/usr/src/software**
5. Inicie el proceso de extracción. Escriba:
tar xvfz gtk-gnutella-snapshot.tar.gz.
Nota: la herramienta **tar** usando la opción **z** extrae automáticamente todo el contenido.
6. El contenido estará en el subdirectorio homónimo, por ejemplo **/usr/src/software/gtk-gnutella-snapshot/**.

Para extraer el contenido de un archivo **tar.bz2** se hará de la siguiente forma:

1. En el terminal, escriba: `su`.
2. Escriba la contraseña de usuario `root`.
3. Cambie al directorio donde se encuentra en programa a instalar, por ejemplo `/usr/src/software`
4. Por ejemplo, para instalar el software `gtk-gnutella`, entrar en el directorio donde se descargó, por ejemplo `/usr/src/software`
5. Inicie el proceso de descompresión. Escriba:
`bzip2 -dv gtk-gnutella-snapshot.tar.bz2.`
lo cual nos dejará un archivo llamado `gtk-gnutella-snapshot.tar`
6. Ahora a extraer el contenido del archivo `tar`. Escriba:
`tar xvf gtk-gnutella-snapshot.tar.`
7. El contenido estará en el subdirectorio homónimo, por ejemplo: `/usr/src/software/gtk-gnutella-snapshot/`.

Estos son algunos métodos para instalar paquetes en GNU/Linux, para mas información, leer la documentación de cada distribución de Linux.

Apéndice B

Python

Python es un lenguaje de programación fácil de aprender y potente. Dispone de eficaces estructuras de datos de alto nivel y una solución de programación orientada a objetos simple pero eficaz. La elegante sintaxis de Python, su gestión de tipos dinámica y su naturaleza interpretada hacen de él el lenguaje ideal para guiones (scripts) y desarrollo rápido de aplicaciones en muchas áreas y en la mayoría de las plataformas.

El intérprete de Python y su extensa biblioteca estándar están disponibles libremente, en forma de fuentes o ejecutables, para las plataformas más importantes, en la sede web de Python, <http://www.python.org>, y se pueden distribuir libremente. La misma sede contiene también distribuciones y direcciones de muchos módulos, programas y herramientas Python de terceras partes, además de documentación adicional.

Python es un lenguaje interpretado, lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar. El intérprete se puede utilizar de modo interactivo, lo que facilita experimentar con características del lenguaje, escribir programas desechables o probar funciones durante el desarrollo del programa de la base hacia arriba. También es una calculadora muy útil.

Python permite escribir programas muy compactos y legibles. Los programas escritos en Python son normalmente mucho más cortos que sus equivalentes en C o C++, por varios motivos:

- Los tipos de datos de alto nivel permiten expresar operaciones complejas en una sola sentencia.
- El agrupamiento de sentencias se realiza mediante sangrado (indentación) en lugar de begin/end o llaves.
- No es necesario declarar los argumentos ni las variables.

B.1. La instalación

La instalación se realizara sobre el sistema operativo Linux usando el archivo. `Python-2.5.2.tar.bz2` por lo tanto desde la consola de comandos haremos la instalación.

Lo descargamos desde el sitio oficial:

```
wget -c http://python.org/ftp/python/2.5.2/Python-2.5.2.tar.bz2
```

Al terminar la descarga, haremos la descompresión del archivo:

```
bzip2 -dv Python-2.5.2.tar.bz2
```

Lo cual nos dejara un archivo empaquetado llamado `Python-2.5.2.tar` y desempaquetaremos usando el siguiente comando:

```
tar xvf Python-2.5.2.tar
```

Quedará una carpeta llamada `Python-2.5.2` donde al ingresar a la carpeta encontraremos los archivos necesarios para instalar Python, una vez dentro de la capeta procedemos a configurar el entorno ejecutando: `./configure`, al terminar el proceso ejecutaremos `make` para compilar los archivos binarios, posteriormente podemos ejecutar una prueba para verificar que la compilación se realizo correctamente `make test`, si las pruebas fueron satisfactorias, procedemos a instalarlo tecleando con la cuenta root `make install`.

Al teminar el priceso, podemos teclear `python` desde la consola y veremos el intérprete de ordenes.

B.2. El intérprete

En Unix, el intérprete de Python se suele instalar como `/usr/local/bin/python` en aquellas máquinas donde esté disponible. En Windows, se instala en el directorio **Archivos de programa**. Poner este directorio en la ruta de ejecutables hace posible arrancarlo tecleando en el intérprete de órdenes la orden:

```
python
```

Como la elección del directorio donde reside el intérprete es una opción de instalación, es posible que se halle en otros lugares.

Teclear un carácter fin de archivo (Control-D en Unix, Control-Z en DOS o Windows) en

el intérprete causa la salida del intérprete con un estado cero. Si eso no funciona, se puede salir del intérprete tecleando las siguientes órdenes: `import sys; sys.exit()`.

Cuando se leen órdenes desde una tty, se dice que el intérprete está en modo interactivo o línea a línea. En este modo, espera a la siguiente orden con el indicador principal, que suele ser tres signos 'mayor' ("`>>>`"). Para las líneas adicionales, se utiliza el indicador secundario, por omisión tres puntos ("`...` ").

El intérprete muestra un mensaje de bienvenida con su número de versión e información de derechos de copia, antes de mostrar el primer indicador:

```
persa@darkside:~> python
Python 2.5.2 (r252:60911, Dec 1 2008, 18:10:01)
[GCC 4.3.1 20080507 [gcc-4_3-branch 135036]] on linux2
Type help, copyright, credits or license for more information.
>>>
```

En sistemas Unix tipo BSD, los guiones Python se pueden hacer ejecutables directamente, como guiones de línea de órdenes, poniendo la siguiente línea.

```
#!/usr/bin/env python
```

A esta línea se le conoce en el mundo Unix como shebang, hashbang o sharpbang. El par de caracteres `#!` indica al sistema operativo que dicho script se debe ejecutar utilizando el intérprete especificado a continuación.

B.3. Hola Mundo

El primer programa que vamos a escribir en Python es el clásico Hola Mundo, y en este lenguaje es tan simple como:

```
print 'Hola Mundo'
```

Vamos ahora a crear un archivo de texto con el código anterior, de forma que pudiéramos distribuir nuestro pequeño gran programa entre nuestros amigos. Abre tu editor de texto preferido y copia la línea anterior. Guárdalo como `hola.py`, por ejemplo.

Ejecutar este programa es tan sencillo como indicarle el nombre del archivo a ejecutar al intérprete de Python.

```
python hola.py
```

para ejecutar nuestro pequeño programa sin llamar al interprete, debemos agregar la línea shebang `#!/usr/bin/env python` mencionada anteriormente, el archivo se vería de la siguiente manera.

```
#!/usr/bin/env python
print 'Hola Mundo'
```

Por supuesto además de añadir el shebang, tendremos que dar permisos de ejecución al programa.

```
chmod +x hola.py
```

Para correr el programa desde la consola como si tratara de un ejecutable cualquiera:

```
./hola.py
```

B.4. Tipos de Datos Basicos

B.4.1. Cadenas de Caracteres

Son literalmente cualquier cadena o secuencia de caracteres que puedan imprimirse en la pantalla. (De hecho pueden ser también ciertos caracteres de control que no son imprimibles).

En Python las cadenas pueden representarse de varias formas:

Con comillas simples:

```
'Esta es una cadena'
```

Con comillas dobles:

```
“Esta otra es una cadena muy similar”
```

Con tres comillas dobles:

```
“ “ “ Esta es una cadena muy larga que puede
ocupar varias líneas si así lo deseamos y
Python la mantendrá del mismo modo en que
nosotros la escribimos...” ” ”
```

Un uso especial de esta última forma puede verse en la generación de la documentación de las funciones de Python.

Podemos acceder a los caracteres individuales de una cadena al tratarla como una matriz de caracteres. Hay varias operaciones que el lenguaje de programación provee para ayudarnos a manipular las cadenas, tales como buscar una subcadena dentro de otra, unir dos cadenas, copiar una cadena en otra, etc.

B.4.2. Números Enteros

Los números enteros son aquellos que no tienen decimales, tanto positivos como negativos (además del cero). En Python se pueden representar mediante el tipo `int` (de integer, entero) o el tipo `long` (largo). La única diferencia es que el tipo `long` permite almacenar números más grandes. Es aconsejable no utilizar el tipo `long` a menos que sea necesario, para no malgastar memoria.

El tipo `int` de Python se implementa a bajo nivel mediante un tipo `long` de C. Y dado que Python utiliza C por debajo, como C, y a diferencia de Java, el rango de los valores que puede representar depende de la plataforma. En la mayor parte de las máquinas el `long` de C se almacena utilizando 32 bits, es decir, mediante el uso de una variable de tipo `int` de Python podemos almacenar números de -2^{31} a $2^{31} - 1$, o lo que es lo mismo, de -2.147.483.648 a 2.147.483.647. En plataformas de 64 bits, el rango es de -9.223.372.036.854.775.808 hasta 9.223.372.036.854.775.807.

El tipo `long` de Python permite almacenar números de cualquier precisión, limitado por la memoria disponible en la máquina.

B.4.3. Números Reales

Los números reales son los que tienen decimales. En Python se expresan mediante el tipo `float`. En otros lenguajes de programación, como C, tenemos también el tipo `double`, similar a `float` pero de mayor precisión (`double` = doble precisión). Python, sin embargo, implementa su tipo `float` a bajo nivel mediante una variable de tipo `double` de C, es decir, utilizando 64 bits, luego en Python siempre se utiliza doble precisión, y en concreto se sigue el estándar IEEE 754: 1 bit para el signo, 11 para el exponente, y 52 para la mantisa. Esto significa que los valores que podemos representar van desde $\pm 2,2250738585072020 \times 10^{-308}$ hasta $\pm 1,7976931348623157 \times 10^{308}$.

B.5. Operadores Básicos

Veamos qué podemos hacer con nuestros números usando los operadores por defecto. Para operaciones más complejas podemos recurrir al módulo `math`.

B.5.1. Operadores aritméticos

Operador	Descripción	Ejemplo
+	Suma	$r = 3 + 2$ es 5
-	Resta	$r = 4 - 7$ es -3
-	Negación	$r = - 7$ es -7
*	Multiplicación	$r = 2 * 3$ es 6
**	Exponente	$r = 2 ** 6$ es 64
/	División	$r = 3.5 / 2$ es 1.75
//	División entera	$r = 3.5 // 2$ es 1.0
%	Módulo	$r = 7 \% 2$ es 1

El operador de módulo devuelve el resto de la división entre los dos operandos. En el ejemplo, $7 / 2$ sería 3, con 1 de resto, esto es 1.

La diferencia entre división y división entera no es otra que la que indica su nombre. En la división el resultado que se devuelve es un número real, mientras que en la división entera el resultado que se devuelve es solo la parte entera.

Bibliografía

- [1] Unixsupport *Avenida Urdaneta, Esquina Animas, Edificio IBERIA, P.H., La Candelaria. CARACAS - VENEZUELA* tel:(58) (0212) - 5631503, 5631295
<http://www.unixsup.com>
- [2] Guido van Rossum, python - Manual de referencia y Guía de aprendizaje de Python.
<http://pyspanishdoc.sourceforge.net/tut/tut.html> Python Software Foundation
- [3] AMECE - Asociación Mexicana de Estándares para el Comercio Electronico.
<http://www.amece.org.mx/>
Marco Legal.
http://www.amece.org.mx/amece/Legislacion_Factura_Electronica.php
- [4] Sistema de Administración Tributaria. <http://www.sat.gob.mx/>
Marco Legal.
http://www.sat.gob.mx/sitio_internet/e_sat/comprobantes_fiscales/15_6544.html
- [5] South Consulting Signature <http://www.southconsulting.cl/>
empresa consultora líder en proveer soluciones de Facturación Electrónica.
<http://www.southconsulting.cl/Signature-Facturacion.shtml>
MÉXICO, Bosque de Duraznos 127, Piso 2, Col. Bosques de Las Lomas, C.P. 11700,
México D.F. Tel.: (52-55) 5148 3417
- [6] The open ssl proyect - <http://www.openssl.org/docs/>
- [7] pyODB proyect - <http://sourceforge.net/projects/pyodb/>
- [8] Programar en Bash - <http://es.tldp.org>
- [9] United Nations Economic Commission for Europe *Information Service* Palais des Nations CH - 1211 Geneva 10, Switzerland <http://www.unece.org/>
- [10] Seguridata <http://www.seguridata.com/>
empresa de tecnología mexicana, se dedica al desarrollo de productos y servicios.
- [11] ASPEL *Aspel de México, S.A. de C.V.* <http://www.aspel.com.mx/>

- [12] Ekocommerce. *Empresa de Grupo Ekonom, un pionero en el sector Mexicano de Telecomunicaciones y Comercio Electrónico*. <http://www.ekocommerce.com/>
- [13] Edicom. *Edicom compañía dedicada al desarrollo e implantación de sistemas transaccionales de alto rendimiento B2B*. <http://www.edicomgroup.com/mx/>
- [14] Levicom. *Levicom tiene una trayectoria de más de 20 años en el desarrollo de sistemas de cómputo*. <http://www.levicom.com.mx/>
- [15] MAPICS ERP <http://www.infor.com/> <http://www.cimatic.com.mx/>
- [16] Linux Filesystem Hierarchy *Howto File System have a Hierarchy* <http://tldp.org/LDP/Linux-Filesystem-Hierarchy/html/index.html>
- [17] Steve Shah *Administración de Linux*, ISBN: 97010588285, 4ta Edición McGraw-Hill Interamericana.
- [18] Sebastian Sanchez Prieto *Unix- Linux Guia Practica*, ISBN: 9701510917, 3ra Edición AlfaOmega Editores.
- [19] Peter Kuo *SuSE Linux Enterprice Server Administrator's Handbook*, ISBN: 067232735X, 1ra Edición Novell Press.
- [20] Arnold Robbins. *Programación en Linux*, ISBN: 8441518394, 1ra Edición Anaya Multimedia - Anaya editores.
- [21] Guido van Rossum. *An Introduction to Python*, Release 2.5, Published by Network Theory Limited, 15 Royal Park, Bristol BS8 3AL United Kingdom
- [22] Wesley J. Chun. *Core Python Programming* Prentice Hall PTR, Pearson Education
- [23] Alex Martelli *Python: Guia de referencia*, ISBN: 9788441523173, 1ra Edición Anaya Multimedia
- [24] Python recipes Activestate <http://code.activestate.com/recipes/langs/python/>
- [25] articles on Python and related technologies <http://effbot.org/>
- [26] Johann-Christian Hanke *PHP 5: Sitios Web Dinamicos*, ISBN: 9782915605310, 1ra Edición Pearson Educación
- [27] Steven Holzner *PHP : Manual de Referencia*, ISBN: 9789701067574, 1ra Edición McGraw-Hill Interamericana
- [28] Justine Middleton, Tony Barker, Torsten Hartig, Jim Scott *The AS/400 NetServer Advantage* IBM Corporation, International Technical Support Organization Red-Books, 1ra Edición IBM RedBooks - SG24-5196-00

- [29] Yessong Johng, Takaki Kimachi, Wei Wang. *DB2 UDB for Linux on iSeries: Implementation Guide* IBM Corporation, International Technical Support Organization RedBooks, 4ra Edición IBM RedBooks - SG24-6081-00 This edition applies to DB2 UDB for Linux on iSeries V8.1.2 and OS/400 V5R2.
- [30] Whei-Jen Chen, Rav Ahuja, Louisa Ford, Paul Jud, Kang Yong Ying. *Up and Running with DB2 for Linux* IBM Corporation, International Technical Support Organization RedBooks, 1ra Edición IBM RedBooks - SG24-6899-00 This edition applies to DB2 Universal Database Version 8.1 for use with SuSE Linux Version 8.1 and Red Hat Linux Version 8.0 operating systems.