

**REPORTE DE RESIDENCIAS PROFESIONALES**

PORTAL DE PROVEEDORES Y COMPROBACION DE GASTOS CON VALIDACIÓN DE CFDI

**EMPRESA**

PINTURA ESTAMPADO Y MONTAJE SAPI DE CV

**PERIODO:**

11 DE JULIO DEL 2016 AL 15 DE ENERO DEL 2017

**PRESENTA:**

GUILLERMO ORTIZ REBOLLEDO

NO. DE CONTROL: 12030811

INGENIERÍA EN SISTEMAS COMPUTACIONALES

**ASESOR INTERNO:**

ING. RUBÉN TORRES FRÍAS

**ASESOR EXTERNO:**

YANET RUIZ BARCENAS

**CELAYA, GUANAJUATO A 10 DE AGOSTO DEL 2016**

ÍNDICE

[1 INTRODUCCIÓN 2](#_Toc471974752)

[2 JUSTIFICACIÓN 2](#_Toc471974753)

[3 OBJETIVOS 3](#_Toc471974754)

[3.1 OBJETIVO GENERAL 3](#_Toc471974755)

[3.2 OBJETIVOS ESPECÍFICOS 3](#_Toc471974756)

[4 CARACTERIZACIÓN DEL ÁREA 3](#_Toc471974757)

[5 PROBLEMAS A RESOLVER 3](#_Toc471974758)

[5.1 ESTABLECIMIENTO DE PRIORIDADES 4](#_Toc471974759)

# INTRODUCCIÓN

En los años recientes, la facturación electrónica ha cobrado importancia debido a su simplicidad y a las ventajas que tiene sobre las facturas en papel, además de que una factura electrónica cuenta con la misma información que una factura normal en papel.

En la empresa PEMSA es un requisito realizar una comprobación de gastos cada que algún empleado requiere hacer un viaje relacionado con el trabajo, ya sea visita a otra planta del corporativo, diplomados o cursos, visitas a proveedores, entre otros. Para realizar dicha comprobación de gastos es necesario ingresar en el sistema para registrar los datos del viaje, los gastos realizados y sus respectivos detalles, así como los comprobantes expedidos por los contribuyentes, es decir, las facturas electrónicas o también llamadas CFDI (Comprobante Fiscal Digital).

Es aquí donde surge la importancia de llevar a cabo una validación de tales comprobantes. El SAT (Servicio de Administración Tributaria) establece todos los campos necesarios que los comprobantes deben incluir y los criterios para establecer si este es válido o inválido.

Por esta misma razón, en PEMSA y las demás plantas de CIE Automotive es necesario que las facturas subidas al sistema de comprobación sean válidas ante el SAT.

En este reporte se describirá a detalle el proceso de realización de un módulo nuevo para el sistema de comprobación de gastos el cual se encarga de validar todas las facturas subidas.

# JUSTIFICACIÓN

Como se menciona en el marco teórico anexo a este documento, una factura es un documento que avala la compra-venta de un bien o el préstamo de un servicio y además clave en la declaración de impuestos ya que a través de ella el SAT puede reembolsar los impuestos correspondientes a los contribuyentes.

El uso de comprobantes electrónicos o CFDIs trae varios beneficios enlistados a continuación:

* La reducción del costo de los gastos de facturación. Se eliminan los gastos de impresión, mensajería y almacenamiento y esto ha implicado una reducción de costos importante para las empresas y los usuarios.
* La seguridad y rapidez en la emisión de comprobantes. Las facturas electrónicas se deben de realizar por medio de un PAC (Proveedor Autorizado para la Certificación de CFDI). Esto significa que este organismo cuenta con todos los requisitos impuestos por el SAT en materia de seguridad para timbrar documentos fiscales. Esto a su vez optimiza y facilita la carga de comprobantes en el sistema de la empresa.
* Existen menos errores y contratiempos a la hora de realizar una comprobación de gastos en el sistema de la empresa cuando un usuario sale de viaje y/o realiza una compra, ya que se valida el comprobante desde antes de ser subido al repositorio de la empresa.
* Mayor control de los documentos y los registros de gastos en el sistema.

A partir del año 2014, conforme al artículo 29 del Código Fiscal de la Federación, todos los contribuyentes que tienen ganancias superiores a cuatro millones de pesos anuales están obligados a emitir Comprobantes Fiscales Digitales a través de internet.

Por esta misma razón es de suma importancia para PEMSA y todas las demás plantas del corporativo en México poder registrar y sobre todo validar estas facturas electrónicas en el sistema local de comprobación de gastos.

# OBJETIVOS

## OBJETIVO GENERAL

Desarrollar un módulo para el sistema de comprobación de gastos que se encargue de capturar todos los datos requeridos para que el usuario pueda registrar los gastos que realizó en algún viaje relacionado con el trabajo. Además, validar los comprobantes fiscales cargados por el mismo usuario y notificarle a este en caso de que este comprobante no cuente con la estructura y datos requeridos por el Servicio de Administración Tributaria. También es necesario implementar un módulo de consulta para que el usuario pueda ver todos los conceptos registrados en sus comprobaciones, así como los certificados de validez generados después de llevar a cabo la validación de los comprobantes fiscales digitales. Todo esto aplica para los diferentes casos que se pueden presentar: reembolsos, modificaciones y anticipos.

## OBJETIVOS ESPECÍFICOS

* Desarrollar una interfaz dentro del sistema de comprobación de gastos donde los usuarios puedan registrar los datos de la comprobación, así como la carga de los archivos PDF y XML (comprobante fiscal digital o factura electrónica).
* Desarrollar un módulo para validar los comprobantes cargados por los usuarios en base a los requerimientos del SAT.
* Desarrollar un módulo que permita a los usuarios consultar las facturas guardadas en el repositorio del sistema.
* Hacer las modificaciones necesarias en la base de datos dedicada al sistema de comprobación de gastos para guardar las validaciones hechas a los comprobantes.
* Desarrollar un programa o script que permita hacer la descarga de los archivos necesarios que publica el SAT en un servidor FTP para tener los requerimientos más recientes para las validaciones.
* Incorporar la validación de los comprobantes a los módulos de reembolso, modificación y anticipo dentro del sistema de comprobación de gastos.

# CARACTERIZACIÓN DEL ÁREA

# PROBLEMAS A RESOLVER

Se necesita implementar un módulo en el sistema de comprobación de gastos en el cual se puedan registrar los detalles de los gastos realizados y subir las facturas que les corresponden.

Los comprobantes o facturas subidas vienen tanto en formato XML como en PDF. El sistema se encargará de validar el archivo XML el cual contiene en su estructura toda la información que el SAT establece como necesaria.

Éste módulo deberá notificar al usuario en caso de que el comprobante subido sea inválido después de pasar por el proceso de validación, para que el usuario pueda subir otro comprobante o ponerse en contacto con la persona que expide la factura. Esto con el fin de tener en el sistema solamente las facturas o comprobantes válidos.

## ESTABLECIMIENTO DE PRIORIDADES

Las prioridades que se tienen para llevar a cabo este proyecto se enlistan a continuación. Se tiene en cuenta desde que inician el proyecto hasta el día que terminan las residencias en la empresa.

1. Ponerse en contacto con los empleados del área de compras y de contabilidad y finanzas de todas las plantas del corporativo en México para reunir todos los requerimientos del proyecto.
2. Realizar investigación acerca de los requisitos que establece el SAT para llevar a cabo las validaciones de un comprobante.
3. Establecer una metodología, así como el lenguaje y entorno de desarrollo que la empresa establece para realizar el proyecto.
4. Llevar a cabo del desarrollo del proyecto.
5. Programar una junta con los empleados de los departamentos involucrados para informar sobre el proyecto y dar indicaciones de cómo usar el sistema.
6. Publicar la nueva versión del sistema para que las personas de compras y contabilidad puedan hacer las pruebas necesarias.
7. Atender cualquier caso de errores o bugs que se puedan presentar.

# ALCANCES Y LIMITACIONES

# FUNDAMENTO TEÓRICO

Para poder llevar a cabo una comprobación en el sistema de gastos de le empresa, en la gran mayoría de los casos, se tiene que contar con una factura o comprobante electrónico. La parte fundamental en este proyecto es la validación de dichas facturas, es por esto que es necesario comprender en su totalidad el concepto de factura y la importancia de su validación, así como los organismos que las regulan.

## FACTURAS

Una **factura** es un documento mercantil que refleja toda la información de una operación de compraventa. La información fundamental que aparece en una factura debe reflejar la entrega de un producto o la provisión de un servicio, junto con toda la información de la operación como se enlista a continuación:

* Nombre del comprador o vendedor, ya sea una empresa o un particular.
* Identificación del producto o del servicio, como el nombre o un código.
* El precio por unidad y el precio total de todas las piezas que se han adquirido.
* Si existe algún descuento.
* Impuestos como el Impuesto sobre el Valor Añadido.

### Timbre Fiscal Digital

Una de las partes o nodo principales en una factura es el **timbre fiscal digital**. Este complemento garantiza la autenticidad al documento y resguarda la información necesaria correspondiente al timbre. Sin este complemento un CFDI no tiene validez alguna. Dentro del complemento él se encuentra el **UUID**, conocido como folio fiscal del SAT, el cual identifica al CFDI de forma única entre todos los que se han emitido. Se genera de forma automática por parte del **PAC** (Proveedor Autorizado para la Certificación de CFDI) que lo timbra.

### PAC

Un **PAC** o **Proveedor Autorizado para la Certificación de CFDI** es una empresa autorizada por el SAT (Servicio de Administración Tributaria) para la realización de un proceso denominado timbrado. Para ello esta empresa debe Certificarse de forma de obtener el permiso del SAT.

### Contenido del Timbre Fiscal Digital de una Factura

El complemento del Timbrado que deben llevar todas las facturas electrónicas debe contar con los siguientes elementos:

* **UUID**. - Conocido como folio fiscal del SAT, el cual identifica al CFDI de forma única entre todos los que se han emitido.
* **Fecha de timbrado**. - Expresa la fecha en que fue timbrado el CFDI, esta fecha no puede ser mayor a 72 horas de la expedición del CFDI.
* **Sello del CFD**. - Es el sello del CFDI antes de ser timbrado, con esto se asegura la integridad del CFDI una vez que este ha sido timbrado por un PAC.
* **Número de certificado SAT**. - Corresponde al número de serie de certificado del PAC que timbro el CFDI.
* **Sello SAT**. - Sello digital del Timbre Fiscal Digital, al que hacen referencia las reglas de resolución miscelánea aplicable. Sirve para asegurar la integridad del CFDI ya que ha sido timbrado.

### Requisitos que deben cumplir las Facturas Electrónicas

Para ver un desglose más detallado de cada complemento y cada nodo del archivo XML que debe tener, se puede consultar el Anexo 20 de la Resolución Miscelánea Fiscal emitida por el SAT. De manera general, una factura electrónica debe contar con los siguientes puntos:

* Clave del Registro Federal de Contribuyentes de quien los expida.
* Régimen Fiscal en que tributen conforme a la Ley del ISR (Consulte el procedimiento para obtener el régimen fiscal).
* Sí se tiene más de un local o establecimiento, se deberá señalar el domicilio del local o establecimiento en el que se expidan las Facturas Electrónicas.
* Contener el número de folio asignado por el SAT y el sello digital del SAT.
* Sello digital del contribuyente que lo expide.
* Lugar y fecha de expedición.
* Clave del Registro Federal de Contribuyentes de la persona a favor de quien se expida.
* Cantidad, unidad de medida y clase de los bienes, mercancías o descripción del servicio o del uso o goce que amparen.
* Valor unitario consignado en número.
* Importe total señalado en número o en letra,
* Señalamiento expreso cuando la prestación se pague en una sola exhibición o en parcialidades.
* Cuando proceda, se indicará el monto de los impuestos trasladados, desglosados por tasa de impuesto y, en su caso, el monto de los impuestos retenidos.
* Forma en que se realizó el pago (efectivo, transferencia electrónica de fondos, cheque nominativos o tarjeta de débito, de crédito, de servicio o la denominada monedero electrónico que autorice el Servicio de Administración Tributaria).
* Número y fecha del documento aduanero, tratándose de ventas de primera mano de mercancías de importación.

Además, debe contener los siguientes datos:

* Fecha y hora de certificación.
* Número de serie del certificado digital del SAT con el que se realizó el sellado.

Las facturas electrónicas (CFDI) cuentan con un elemento opcional llamado "Addenda", que permite integrar información de tipo no fiscal o mercantil, en caso de requerirse. Esta “addenda” debe incorporarse una vez que la factura haya sido validada por el SAT o el Proveedor de Certificación Autorizado (PAC) y se le hubiera asignado el folio.

## Formato de Archivos

Para facturación electrónica se utilizan diferentes formatos de archivos. El archivo XML se utiliza para poder tener una estructura y un “diccionario” de palabras definido y así todas las facturas pueden llevar un mismo formato.

### XML

XML es un meta-lenguaje que permite definir lenguajes de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible. Proviene del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML) para estructurar documentos grandes. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones deben comunicarse entre sí o integrar información.

* Es extensible: Después de diseñado y puesto en producción, es posible extender XML con la adición de nuevas etiquetas, de modo que se pueda continuar utilizando sin complicación alguna.
* El analizador es un componente estándar, no es necesario crear un analizador específico para cada versión de lenguaje XML. Esto posibilita el empleo de cualquiera de los analizadores disponibles. De esta manera se evitan bugs y se acelera el desarrollo de aplicaciones.
* Transformamos datos en información, pues se les añade un significado concreto y los asociamos a un contexto, con lo cual tenemos flexibilidad para estructurar documentos.

### PDF

Por lo general, a los archivos XML se les acompaña con un archivo en formato PDF con el fin de proporcionar a las personas una manera más sencilla de leer o interpretar los datos de la factura.

**PDF** (Portable Document Format) es un formato de almacenamiento para documentos digitales independiente de plataformas de software o hardware. Este formato es de tipo compuesto (imagen vectorial, mapa de bits y texto).

## Directorio Activo

Es el término que utiliza Microsoft para referirse a su implementación de servicio de directorio en una red distribuida de computadores. Utiliza distintos protocolos, principalmente LDAP, DNS, DHCP y Kerberos.

De forma sencilla se puede decir que es un servicio establecido en uno o varios servidores en donde se crean objetos tales como usuarios, equipos o grupos, con el objetivo de administrar los inicios de sesión en los equipos conectados a la red, así como también la administración de políticas en toda la red.

Su estructura jerárquica permite mantener una serie de objetos relacionados con componentes de una red, como usuarios, grupos de usuarios, permisos y asignación de recursos y políticas de acceso.

El **directorio activo** permite a los administradores establecer políticas a nivel de empresa, desplegar programas en muchos ordenadores y aplicar actualizaciones críticas a una organización entera. Un Active Directory almacena información de una organización en una base de datos central, organizada y accesible. Pueden encontrarse desde directorios con cientos de objetos para una red pequeña hasta directorios con millones de objetos.

Es por estas razones por las cuales a la empresa se le facilita el manejo de equipos y usuarios. En todos los sistemas, la **autenticación** se hace por medio de este medio. Cada que una persona es contratada o ingresa a la empresa, es inscrita en el directorio activo.

## Scrum

Se utilizaron las bases de **Scrum** para llevar un método de **desarrollo ágil** en el periodo de residencias, además de que promueve una mejor **comunicación** entre el equipo y el ‘cliente’ o dueño del proyecto. Algunas de las características de esta metodología de desarrollo son:

* Adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto.
* Basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto organizados, que en la calidad de los procesos empleados.
* Solapamiento de las diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o en cascada.

Scrum permite la creación de equipos **auto organizados** impulsando la co-localización de todos los miembros del equipo, y la comunicación verbal entre todos los miembros y disciplinas involucrados en el proyecto.

Un principio clave de Scrum es el reconocimiento de que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan y que los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada. Por lo tanto, Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes.

Existen varias implementaciones de sistemas para gestionar el proceso de Scrum, que van desde notas amarillas "post-it" y pizarras hasta paquetes de software. Una de las mayores ventajas de Scrum es que es muy fácil de aprender, y requiere muy poco esfuerzo para comenzarse a utilizar. Así, si se utiliza una pizarra con notas autoadhesivas cualquier miembro del equipo podrá ver tres columnas: trabajo pendiente ("backlog"), tareas en proceso ("in progress") y hecho ("done"). De un solo vistazo, una persona puede ver en qué están trabajando los demás en un momento determinado.

**Product Owner**

* El Product Owner representa la voz del cliente. Se asegura de que el equipo Scrum trabaje de forma adecuada desde la perspectiva del negocio.

**ScrumMaster (o Facilitador)**

* El Scrum es facilitado por un ScrumMaster, cuyo trabajo primario es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El ScrumMaster no es el líder del equipo (porque ellos se auto-organizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga.

**Equipo de desarrollo**

* El equipo tiene la responsabilidad de entregar el producto. Es recomendable un pequeño equipo de 3 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc).

En este periodo de residencias, se utilizaron pizarras y archivos en Excel para indicar el proceso del proyecto, además de juntas para la mejor difusión del progreso y objetivos del proyecto. En estas juntas también se acostumbró a dar retroalimentación entre el ScrumMaster (Yanet Ruiz) y el equipo de desarrollo (Guillermo Ortiz).

## Programación Orientada a Objetos

La programación orientada a objetos es un paradigma de programación que viene a innovar la forma de obtener resultados. Los objetos manipulan los datos de entrada para la obtención de datos de salida específicos, donde cada objeto ofrece una funcionalidad especial.

### Introducción a la POO

Los **objetos** son entidades que tienen un determinado "**estado**", "**comportamiento** (método)" e "**identidad**":

* La **identidad** es una propiedad de un objeto que lo diferencia del resto; dicho con otras palabras, es su identificador (concepto análogo al de identificador de una variable o una constante).

Un objeto contiene toda la información que permite definirlo e identificarlo frente a otros objetos pertenecientes a otras clases e incluso frente a objetos de una misma clase, al poder tener valores bien diferenciados en sus atributos. A su vez, los objetos disponen de mecanismos de interacción llamados métodos, que favorecen la comunicación entre ellos. Esta comunicación favorece a su vez el cambio de estado en los propios objetos. Esta característica lleva a tratarlos como unidades indivisibles, en las que no se separa el estado y el comportamiento.

Los **métodos** (comportamiento) y **atributos** (estado) están estrechamente relacionados por la propiedad de conjunto. Esta propiedad destaca que una clase requiere de métodos para poder tratar los atributos con los que cuenta. El programador debe pensar indistintamente en ambos conceptos, sin separar ni darle mayor importancia a alguno de ellos.

### Conceptos Fundamentales

La **POO** es una forma de programar que trata de encontrar una solución a estos problemas. Introduce nuevos conceptos, que superan y amplían conceptos antiguos ya conocidos. Entre ellos destacan los siguientes:

**Clase**

* Definiciones de las propiedades y comportamiento de un tipo de objeto concreto. La instanciación es la lectura de estas definiciones y la creación de un objeto a partir de ella.

**Herencia**

* Por ejemplo, herencia de la clase C a la clase D, es la facilidad mediante la cual la clase D hereda en ella cada uno de los atributos y operaciones de C, como si esos atributos y operaciones hubiesen sido definidos por la misma D. Por lo tanto, puede usar los mismos métodos y variables públicas declaradas en C. Los componentes registrados como "privados" (private) también se heredan, pero como no pertenecen a la clase, se mantienen escondidos al programador y sólo pueden ser accedidos a través de otros métodos públicos. En el caso de los componentes registrados como "protegidos" (protected) también se heredan, pero solo para esa clase, no para futuras clases heredadas. Esto es así para mantener hegemónico el ideal de POO.

**Objeto**

* Instancia de una clase. Entidad provista de un conjunto de propiedades o atributos (datos) y de comportamiento o funcionalidad (métodos), los mismos que consecuentemente reaccionan a eventos. Se corresponden con los objetos reales del mundo que nos rodea, o con objetos internos del sistema (del programa).

**Método**

* Algoritmo asociado a un objeto (o a una clase de objetos), cuya ejecución se desencadena tras la recepción de un "mensaje". Desde el punto de vista del comportamiento, es lo que el objeto puede hacer. Un método puede producir un cambio en las propiedades del objeto, o la generación de un "evento" con un nuevo mensaje para otro objeto del sistema.

**Evento**

* Es un suceso en el sistema (tal como una interacción del usuario con la máquina, o un mensaje enviado por un objeto). El sistema maneja el evento enviando el mensaje adecuado al objeto pertinente. También se puede definir como evento la reacción que puede desencadenar un objeto; es decir, la acción que genera.

**Atributos**

* Características que tiene la clase.

**Mensaje**

* Una comunicación dirigida a un objeto, que le ordena que ejecute uno de sus métodos con ciertos parámetros asociados al evento que lo generó.

**Propiedad o atributo**

* Contenedor de un tipo de datos asociados a un objeto (o a una clase de objetos), que hace los datos visibles desde fuera del objeto y esto se define como sus características predeterminadas, y cuyo valor puede ser alterado por la ejecución de algún método.

**Estado interno**

* Es una variable que se declara privada, que puede ser únicamente accedida y alterada por un método del objeto, y que se utiliza para indicar distintas situaciones posibles para el objeto (o clase de objetos). No es visible al programador que maneja una instancia de la clase.

**Componentes de un objeto**

* Atributos, identidad, relaciones y métodos.

## Bases de Datos Relacionales

Una base de datos relacional es una colección de elementos de datos organizados en un conjunto de tablas formalmente descritas desde la que se puede acceder a los datos o volver a montarlos de muchas maneras diferentes sin tener que reorganizar las tablas de la base.

Además de ser relativamente fáciles de crear y acceder, una base de datos relacional tiene la importante ventaja de ser fácil de extender. Después de la creación original de una base de datos, una nueva categoría de datos se puede añadir sin necesidad de que todas las aplicaciones existentes sean modificadas.

### Características

* Una base de datos se compone de varias tablas o relaciones.
* No pueden existir dos tablas con el mismo nombre ni registro.
* Cada tabla es a su vez un conjunto de campos (columnas) y registros (filas).
* La relación entre una tabla padre y un hijo se lleva a cabo por medio de las claves primarias y claves foráneas (o ajenas).
* Las claves primarias son la clave principal de un registro dentro de una tabla y estas deben cumplir con la integridad de datos.
* Las claves ajenas se colocan en la tabla hija, contienen el mismo valor que la clave primaria del registro padre; por medio de estas se hacen las formas relacionales.

### Conceptos Fundamentales

**Relaciones**

* En una BDR, todos los datos se almacenan y se accede a ellos por medio de relaciones previamente establecidas.

**Relaciones base**

* Las relaciones que almacenan datos son llamadas relaciones base y su implementación es llamada "tabla".

**Relaciones derivadas**

* Otras relaciones no almacenan datos, pero son calculadas al aplicar operaciones relacionales. Estas relaciones son llamadas relaciones derivadas y su implementación es llamada "vista" o "consulta". Las relaciones derivadas son convenientes ya que expresan información de varias relaciones actuando como si fuera una sola tabla.

**Restricciones**

* Una restricción es una limitación que obliga el cumplimiento de ciertas condiciones en la BD. Algunas no son determinadas por los usuarios, sino que son inherentemente definidas por el simple hecho de que la BD sea relacional.
* Algunas otras restricciones las puede definir el usuario, por ejemplo, usar un campo con valores enteros entre 1 y 10. Las restricciones proveen un método de implementar "reglas" en la base de datos.
* Las restricciones limitan los datos que pueden ser almacenados en las tablas.
* Usualmente se definen usando expresiones que dan como resultado un valor booleano, indicando si los datos satisfacen la restricción o no.
* Las restricciones no son parte formal del modelo relacional, pero son incluidas porque juegan el rol de organizar mejor los datos. Las restricciones son muy discutidas junto con los conceptos relacionales.

**Dominios**

* Un dominio describe un conjunto de posibles valores para cierto atributo. Como un dominio restringe los valores del atributo, puede ser considerado como una restricción. Matemáticamente, atribuir un dominio a un atributo significa "cualquier valor de este atributo debe ser elemento del conjunto especificado".
* Distintos tipos de dominios son: enteros, cadenas de texto, fecha, no procedurales, etc.
* Cada tabla puede tener uno o más campos cuyos valores identifican de forma única cada registro de dicha tabla, es decir, no pueden existir dos o más registros diferentes cuyos valores en dichos campos sean idénticos. Este conjunto de campos se llama clave única. Pueden existir varias claves únicas en una determinada tabla, y a cada una de éstas suele llamársele candidata a clave primaria.

**Clave primaria**

* Una clave primaria es una clave única (puede estar conformada por uno o más campos de la tabla) elegida entre todas las candidatas que define unívocamente a todos los demás atributos de la tabla para especificar los datos que serán relacionados con las demás tablas. La forma de hacer esto (relación entre tablas) es por medio de claves foráneas.

**Clave foránea**

* Una clave foránea es una referencia a una clave en otra tabla, determina la relación existente en dos tablas. Las claves foráneas no necesitan ser claves únicas en la tabla donde están y sí a donde están referenciadas.

**Clave índice**

* Las claves índice surgen con la necesidad de tener un acceso más rápido a los datos. Los índices pueden ser creados con cualquier combinación de campos de una tabla. Las consultas que filtran registros por medio de estos campos, pueden encontrar los registros de forma no secuencial usando la clave índice.

**Procedimientos almacenados**

* Un procedimiento almacenado es código ejecutable que se asocia y se almacena con la base de datos. Los procedimientos almacenados usualmente recogen y personalizan operaciones comunes, como insertar un registro dentro de una tabla, recopilar información estadística, o encapsular cálculos complejos. Son frecuentemente usados por un API por seguridad o simplicidad. Los procedimientos almacenados no son parte del modelo relacional, pero todas las implementaciones comerciales los incluyen.

# PROCEDIMIENTO Y DESCRIPCIÓN DE ACTIVIDADES

Para el proceso de desarrollo se utilizó el método ágil de desarrollo ‘Scrum’. Scrum maneja tres roles principales: Scrum Master, Product Owner y el Team (equipo). La distribución de los roles para nuestro caso se ilustra a continuación:

* Scrum master: Yanet Ruiz Bárcenas (Jefa inmediata).
  + Elimina cualquier obstáculo que pueda existir que impidan al equipo lograr algún objetivo.
  + Se asegura de que se lleve a cabo el proceso de forma correcta y que se cumplan las reglas.
* Product Owner: Annel Morales Marrón (Departamento de Compras).
  + Se asegura de que el equipo trabaje de manera adecuada desde la perspectiva del negocio.
* Equipo de desarrollo: Guillermo Ortiz Rebolledo (Desarrollador).
  + En este caso en particular, el equipo solamente se conforma por una persona.
  + Se dedica a analizar, diseñar, desarrollar, probar y documentar durante el proceso de desarrollo del sistema que se tiene como objetivo.

Esta metodología de desarrollo ha tenido un auge durante los últimos años en el sector del desarrollo de software ya que trae muchos beneficios a los equipos que lo practican, en este caso, por ejemplo:

* Existe mayor flexibilidad a cambios con base en las necesidades del cliente, en este caso siendo el departamento de compras y el de finanzas. El marco de trabajo puede adaptarse a sus necesidades sin necesidad de hacer cambios drásticos o perder tiempo o progreso.
* La calidad del producto, en este caso los módulos que serán agregados al sistema, serán de alta calidad ya que se pueden agregar y pulir las características de estos en cada iteración.
* Existe más productividad ya que el equipo de antemano sabe sus responsabilidades y trabaja de manera autónoma.
* Se reducen riesgos ya que se desarrollan primero las funcionalidades de mayor valor y se conoce la velocidad con la que el equipo avanza en el proyecto.

Las reuniones en Scrum se realizan de manera diaria. En este caso la gran mayoría de reuniones se llevaron a cabo diariamente a excepción de los días donde Yanet Ruiz, el Scrum master, por motivos de la empresa tenía que salir de planta. Las reuniones no duraban más de quince minutos en el escritorio del Scrum master y se tocan temas acerca del progreso del proyecto, qué se hizo el día anterior, que se hará ese día y si existen problemas u obstáculos que nos impidan llegar al objetivo.

Existen también las reuniones de planificación las cuales se realizan cada mes o cada quince días. En estas discutimos lo que se planeaba hacer y planeaba el tiempo que nos llevaría hacer el trabajo. Después de cada una de estas reuniones, tanto las reuniones Scrum como las de planificación tienen una parte de revisión, donde se presenta el trabajo completado y la retrospectiva donde los miembros del equipo dan sus impresiones y opiniones sobre las reuniones que se han realizado. Esto se hace con el objetivo de tener una mejora continua.