



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE INGENIERÍA

**SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE
PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN
AL AMBIENTE.**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN**

**PRESENTA:
LENIN GUEVARA LÓPEZ**



**DIRECTOR DE TESIS:
M. EN E. ROSALBA RODRÍGUEZ CHÁVEZ**

México, D.F. Noviembre de 2014



AGRADECIMIENTOS

Para todas las personas especiales en mi vida que han dejado huella, que han creído en mí y me han alentado a continuar y siempre han estado cuando más los he necesitado, este logro es de todos, porque todos son una parte importante para mí.

A mis padres

Rubén Guevara y Evelia López a quienes agradezco infinitamente por poder hacer esto posible, por estar siempre conmigo a mi lado y hacer de mí quien soy, gracias por todo su amor, comprensión y apoyo al guiarme a través de este largo y arduo camino de mi carrera profesional, siempre los llevo en mi corazón.

A mi hermano

Rubén Guevara por ser un ejemplo a seguir y siempre apoyarme, por su tiempo, compañía y consejo, mostrándome que con esfuerzo y dedicación se puede alcanzar las metas que uno se fija.

A mis Familiares

A mis abuelos, tíos, tías y primos de la familia Guevara y López que me impulsaron a terminar este ciclo en mi vida, gracias por todo su apoyo.

A Rosalba Rodríguez Chávez

Por su tiempo, su conocimiento, dedicación y alentarme en este trabajo, siempre apoyándome para concluir esta etapa en mi carrera.

Enrique Peñaloza, Nancy Domínguez y Erik de la Cruz

Por ser unos grandes jefes, compañeros de trabajo y maravillosos amigos, quienes me dieron la oportunidad de crecer no solo profesionalmente sino como persona con su tiempo, su conocimiento, las risas, consejos y su forma de ser.

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Janet Belinda Barragán, Andy Acosta, Ramón Aguirre, Janneth Baltazar, Francisco Martínez, Enrique Servín y Claudia Navarrete

Quienes me apoyaron en mi servicio social y primer acercamiento al ámbito laboral, de quienes aprendí mucho y no solo como compañeros, también como amigos, gracias por esa etapa tan importante en mi vida, el equipo completo de “La pecera” que fue cambiando pero que siempre conservamos esa esencia de amistad.

Luis Enrique Paredes, César García, Alberto Perrusquia

Compañeros de trabajo, de escuela y excelentes amigos que me han ayudado a culminar esta parte de mi carrera, que me han animado, aconsejado, apoyado y guiado, por la calidad humana y profesional de quienes he aprendido tanto y que aun continuo aprendiendo, por las risas, por los momentos difíciles y por tantas experiencias vividas que hemos compartido y las que aún nos quedan por delante.

A Víctor Hugo Roa

Por el apoyo, paciencia, dedicación, palabras de aliento, consejos y risas brindados en la realización de este trabajo.

A los integrantes de “la zona”

Con los que compartí unos años maravillosos en la universidad y que muy alegremente puedo decir que son muchos como para enlistarlos en tan corto espacio pero que saben quiénes son y que nuestras vivencias fueron muchas, buenas, malas pero sobre todo siempre unidos amigos míos, muchas gracias.

A mis amigos

A todas esas personas que aun que no puedo enlistar para no olvidar ningún nombre, pero saben que han llegado a mi vida y han llegado para quedarse, los aprecio muchísimo y agradezco todo el apoyo brindado, por estar al pendiente y ser parte de este largo trayecto.

“2014, Año de Octavio Paz”



Procuraduría Federal de Protección al Ambiente
Dirección General de Administración
Dirección General Adjunta de Evaluación e Informática
Dirección de Desarrollo Tecnológico
Jefatura de Departamento Webmaster

OFICIO No. PFPA/6.4/8C.27/253/14
EXP.: PFPA/6.4/8C.27/00001-14

Lic. Angélica Gutiérrez Vázquez

Coordinadora de Servicio Social de la División de Ingeniería Eléctrica

P R E S E N T E

México, D. F. a 22 de octubre del 2014.

Por este medio menciono a usted, que no existe inconveniente alguno, de que la información del **“Sistema de Línea de Mando para la Organización de personal de la Procuraduría Federal de Protección al Ambiente”**, se publique o forme parte de la tesis del alumno Lenin Guevara López, con número de cuenta 301189013, quien participo en el desarrollo de dicho Sistema.

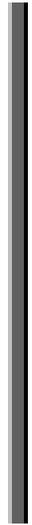
Sin más por el momento, aprovecho para enviarle un saludo y ponerme a sus órdenes para cualquier aclaración.

ATENTAMENTE
JEFE DE DEPARTAMENTO WEBMASTER

Ing. Luis Enrique Paredes López

LEPL/mtag

1/1



ÍNDICE

Contenido

Introducción.....	3
1.- Problemática.....	7
2.- Objetivo	7
3.- Justificación	7
4.- Marco teórico	11
4.1.-La computación como apoyo en el proceso administrativo.	11
4.1.1.- Proceso administrativo.....	12
4.2.- Ingeniería de software.....	13
4.2.1 Requerimientos del software	16
4.2.2 Flujo de los procesos	17
4.2.3 Ciclo de desarrollo de sistemas	18
4.2.4 Metodologías de software.....	22
4.2.4.1.- Desarrollo de software adaptable (ASD, adaptive software development) ..	22
4.2.4.2.- Proceso de desarrollo unificado (RUP).....	23
4.3 Bases de datos	25
4.3.1-Características	26
4.3.1.1.-Integridad	26
4.3.1.2.-Consistencia.....	27
4.3.1.3.-Redundancia.....	28
4.3.2.-Administrador de la base de datos (DBA)	28
4.3.3.-Diccionario de datos.....	29
4.3.4.-Modelos de datos.....	29
4.3.4.1.-Entidad-Relación	30
4.3.4.2.-Modelo relacional	31
4.3.5.- Álgebra relacional.....	32
4.3.6.- Cálculo relacional	33
4.3.7.- Normalización.....	34
4.3.7.1.-Primera forma normal (1FN).....	34
4.3.7.2.-Segunda forma normal (2FN).....	35
4.3.7.3.-Tercera forma normal (3FN).....	36

**SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA
PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.**

4.3.8.-Objetos de la base de datos	36
4.3.9.-DBMS (database management system), RDBMS (relational database management system).....	37
4.3.10. - SQL (structured query language).....	38
4.3.10.1.-Lenguaje de definición de datos (DDL)	39
4.3.10.2.-Lenguaje de manipulación de datos (DML)	39
4.3.10.3.-Lenguaje de control de datos (DCL)	40
4.3.11.-PL/SQL y TransactSQL	40
4.3.11.1.-Vistas	41
4.3.11.2.-Disparadores (Trigger).....	42
4.3.11.3.-Procedimientos	42
4.3.11.4.-Cursores.....	44
4.3.11.5.-Comparación entre SQL Server, Postgresql, Oracle y MySQL.....	44
4.4 Redes de datos y seguridad	46
4.4.1.- Redes y subredes.....	48
4.4.2-Confidencialidad.....	50
4.4.3-Disponibilidad.....	50
4.4.4.- Protocolo TCP/IP.	51
4.4.5.-HTTP	53
4.4.5.1.-Metodo de petición GET	54
4.4.5.2.-Metodo de petición POST	54
5.- Desarrollo	59
5.1.- Descripción general del sistema.....	59
5.2.- Metodología	59
5.2.1.- Requerimientos	59
5.2.2.- Casos de uso	61
5.2.2.1.- Escenario de ingreso	62
5.2.2.2.- Escenario de alta de subordinado	63
5.2.2.3.- Escenario de alta de tercer evaluador.....	63
5.3.- Diagrama Entidad-Relación	64
5.3.1.- Diccionario de datos.....	64
5.4.- Herramientas.....	68

5.4.1.- Eclipse	71
5.4.2.- Java	71
5.4.3.- Apache Tomcat 6	72
5.4.4.- SQL Server 2008	73
5.5.- Interfaz de usuarios	73
5.5.1.- Diseño final del sistema.....	74
5.6.- Desarrollo de la capa de vista.....	83
5.6.1.- Codificación	83
5.7.- Desarrollo de la capa de controlador	83
5.7.1.- Codificación	83
5.8.- Desarrollo de la capa de modelo.....	83
5.8.1.- Codificación	84
6.- Pruebas, implementación y mantenimiento	87
6.1.- Pruebas de unidad.....	87
6.2.- Pruebas de integración.....	87
6.3.- Pruebas de validación.....	87
6.4.- Pruebas del sistema.....	88
6.5.- Implantación.....	88
6.6.- Requerimientos de instalación	88
6.7.- Mantenimiento.....	89
Conclusiones	93
Glosario.....	97
Bibliografía	103
Mesografía y hemerografía	111



INTRODUCCIÓN

Introducción

¿Qué es la PROFEPA?

La Procuraduría Federal de Protección al Ambiente (PROFEPA), es un órgano administrativo desconcentrado de la Secretaría de Medio Ambiente y Recursos Naturales (SEMARNAT) con autonomía técnica y operativa.

Su nacimiento data del 4 de junio de 1992, fecha en la que el diario oficial de la federación publicó el reglamento Interior de la secretaría de desarrollo social que la crea.

La PROFEPA, tiene como tarea principal incrementar los niveles de observancia de la normatividad ambiental, a fin de contribuir al desarrollo sustentable y hacer cumplir las leyes en materia ambiental.

La PROFEPA, está Integrada por:

- Procurador
- Subprocuraduría de Auditoría Ambiental.
 - Dirección general de planeación y promoción de auditorías.
 - Dirección general de operación de auditorías.
- Subprocuraduría de inspección industrial.
 - Dirección general de asistencia técnica industrial.
 - Dirección general de inspección de fuentes de contaminación.
 - Dirección general de inspección ambiental en puertos, aeropuertos y fronteras.
- Subprocuraduría de recursos naturales.
 - Dirección general de impacto ambiental y zona federal marítimo-terrestre.
 - Dirección general de inspección y vigilancia forestal.
 - Dirección general de inspección y vigilancia de vida silvestre, recursos marinos y ecosistemas costeros.
- Subprocuraduría jurídica.
 - Dirección general de delitos federales contra el ambiente y litigio.
 - Dirección general de control de procedimientos administrativos y consulta.
 - Dirección general de denuncias ambientales, quejas y participación social.
- Dirección general de administración.
- Dirección general de coordinación de delegaciones.
 - Delegaciones en las 32 Entidades Federativas.

La problemática que se abordará más adelante se centra más en la cantidad de tiempo y

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

recursos humanos con lo que se cuenta para poder mantener la relación de empleados y poder realizar sus evaluaciones con el mínimo de error, un claro ejemplo es ser evaluado dos veces por empleados diferentes al no saber a quién le corresponde la evaluación, este sistema busca apoyar en estas tareas siendo una herramienta para llevar un control sobre dicha información.

La implementación del sistema profesional de carrera, precisa la realización de evaluaciones al personal, que tienen como objeto medir el desempeño de los servidores públicos. Estas evaluaciones y sus resultados muestran información sobre la productividad de los empleados.

Los métodos de aplicación se basan principalmente en líneas de mando (relaciones de subordinación jerárquica) que se modifican con el tiempo, además debido a que la periodicidad de la aplicación es anual se deben actualizar las metas de desempeño cada año y evidentemente su evaluación.

Cada superior jerárquico tiene la obligación de capturar dichas metas, y de realizar la evaluación del personal que tiene a su cargo, el desconocimiento de realizar de manera adecuada este registro, ha sido un gran obstáculo, debido a que en muchos casos no se conoce esta información oportunamente; la cual tiene que estar centralizada, ya que está generada por cada delegación y enviada a las oficinas centrales para su captura.

La naturaleza de estas líneas de mando es dinámica, pues deben ajustarse ya sea por una baja o por una alta en la plantilla del personal.

En los periodos de aplicación de la evaluación, la información sobre los cambios en las líneas de mando es recopilada en un espacio de tiempo muy breve e inclusive extemporáneo, pues ésta es capturada incluso cuando la evaluación ya está llevándose a cabo, por lo que es necesario repetir evaluaciones de algunas líneas jerárquicas junto con los reportes impresos asociados y esto genera un costo administrativo considerable.

En este trabajo de tesis se plantea y desarrolla un sistema de línea de mando, para realizar este trámite más fácil y transparente, además de tener la información oportunamente y que ésta pueda ser explotada por los sistemas de evaluación de la institución, siendo así un bloque constitutivo de ellos.

En el capítulo uno, se planteará detalladamente la problemática a resolver que nos lleva a desarrollar el marco teórico y el sistema.

En el capítulo dos, se describe el objetivo y se abundará en la justificación del desarrollo del sistema planteado con lo que se desarrolla este sistema.

En el capítulo tres, se plantea la optimización del proceso de movimientos jerárquicos en la línea de mando.

En el capítulo cuarto, se define el marco teórico con el que se desarrolla el sistema con temas de ingeniería del software, bases de datos y redes de datos.

En el capítulo cinco, se define la metodología a utilizar, casos de uso, escenarios, diseño y las herramientas que se utilizarán para el desarrollo del sistema.

En el capítulo seis se dan las pruebas, implementación y mantenimiento aplicado al sistema.

Finalmente, se emiten las conclusiones a este proyecto de tesis.

1.- Problemática

La “Ley del Servicio Profesional de Carrera”, es la legislación que define los mecanismos que garantizan la calidad profesional de todos los funcionarios de la Administración Pública Federal, su cumplimiento es de carácter obligatorio para todas las dependencias del Estado Mexicano. Una parte fundamental de estos mecanismos son las evaluaciones periódicas de los funcionarios públicos. La PROFEPA, ha automatizado estos procesos a través de sistemas institucionales de evaluación, los cuales requieren de una línea de mando actualizada, debido a que los servidores públicos son evaluados por sus superiores inmediatos y por un tercer evaluador, el cual puede ser el superior jerárquico del superior inmediato o cualquier otro servidor público que sea designado para la evaluación.

El proceso de cambio es lento y complicado ya que no se cuenta con suficiente personal para atender todas las solicitudes de cambio en la jerarquía de los trabajadores que se originan en las oficinas centrales y en las 32 Delegaciones de la PROFEPA en la República Mexicana.

En el periodo de evaluación se realizan muchos cambios los cuales deben estar capturados lo antes posible para notificar a los jefes sobre la gente que tiene que evaluar.

2.- Objetivo

Diseñar e implementar un sistema que agilice el registro y la modificación de la línea de mando en la PROFEPA para mantener actualizados y organizados los registros de la relación jerárquica. Mediante este sistema, un servidor público autorizado puede modificar la relación de subordinados a su cargo y asignarles, actualizar o eliminar a un tercer evaluador para cada uno ellos.

3.- Justificación

Anteriormente sin un sistema, esta labor se llevaba a cabo por los empleados del área de personal en documentos de Excel, en los que se debían hacerse llamadas o peticiones a los usuarios para que actualizaran su línea de mando, así en los demás sistemas de personal, en el momento de requerirlo, se les indicaba a que personas les tocaba hacer trámites, haciendo otras tareas más lentas y en ocasiones no estaba actualizada la lista de quién debían realizar su evaluación o captura de información.

Con este sistema las evaluaciones se optimizan en tiempo y forma.



MARCO TEÓRICO

4.- Marco teórico

En este capítulo, se desarrollarán los temas para comprender el desarrollo del sistema propuesto.

4.1.-La computación como apoyo en el proceso administrativo.

Para abordar este capítulo es indispensable identificar los términos de computación, tecnología, Internet y proceso administrativo, así como su evolución.

Según (hypatia, mastermagazine, 2010) se parte del concepto de *cómputo* entendiéndose como una cuenta o cálculo y a la computación como una serie de cálculos aplicados para un fin utilizando una computadora, ahora bien, antes del siglo XIX la computación era desarrollada por las propias personas ya que se realizaban cálculos para llegar a un resultado, con el paso del tiempo se buscaron nuevas formas para facilitar este tipo de cálculo, así como herramientas, teniendo esto en cuenta se diseñaron máquinas para poder realizar operaciones aritméticas que hacían los trabajos más rápidos, lamentablemente solo se limitaban a operaciones aritméticas básicas, sin posibilidad de ser programadas para ejecutar operaciones automáticas y/o más complejas sin necesidad de que intervinieran usuarios.

Según (diccionario de la lengua española, 2010: s/p) Tecnología, se puede entender como las teorías o técnicas con las que se puede aprovechar de una mejor manera los conocimientos científicos.

(Diccionario de la lengua española, 2010) con el avance tecnológico en el área computacional, se realizan procesos mucho más rápidos y automatizados, pues anteriormente debían hacerse y ser supervisados personalmente.

Según (Informática aplicada a la gestión de empresas, pág. 12) así pues nace la informática del acrónimo INFORmación autoMÁTICA, que así como la tecnología es la aplicación de conocimientos científicos, ésta lo hace a través de las computadoras, permitiendo crear sistemas rápidos, confiables y automatizados que no requieren de una gran supervisión por parte de personas, reduciendo tiempos en actividades.

A partir de la creación de la computadora, muchas de las tareas que se hacían complicadas o engorrosas se facilitaron, haciendo más ágiles los trámites, o actividades, optimizando el tiempo y usando menos recursos humanos, teniendo en cuenta que bien utilizada y bien programado el sistema de cómputo dentro de la misma los errores pueden reducirse considerablemente, esto no quiere decir que esté libre de tener errores un sistema, debe ser supervisado para asegurarse de que el sistema está funcionando de una manera

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

adecuada, y si fuera necesario hacer correcciones en la realización de las tareas que le fueron asignadas.

Según (Red Científica, 2010, pág. 10) los adelantos tecnológicos, la evolución de las computadoras, se deben a que el ser humano intenta hacer que las actividades que realiza sean ejecutadas por un máquina, en esta evolución se ha tratado de emular las funciones de seres humanos, en principio básico como impulsos eléctricos en el cerebro funciona así una computadora.

Actualmente Internet, es una herramienta informática y de comunicación muy poderosa, que envía grandes cantidades de datos a diferentes partes del mundo y que sean procesados para obtener diferente información para cada usuario que desea consultarla, además es un medio por el cual se puede difundir, la cual en muchos de los casos es gratuita o que ofrece un servicio con un costo por darla a conocer.

En particular en este proyecto, Internet ha sido usado para difundir un mensaje y llamar la atención hacia un programa, en el que también ofrece el registro vía web desde cualquier parte de la zona que se tiene contemplada que en este caso es a nivel nacional.

Por lo anterior, la Computación en nuestros días hace referencia tanto a tecnología, computadoras, informática e internet, ya que es una ciencia en la cual se estudia el desarrollo de sistemas para el manejo y almacenamiento de la información usando herramientas destinadas a para este fin.

Información, es el conjunto de datos que han sido tratados o que se les aplica un proceso en el cual se acomodan de un modo lógico, coherente y comprensible para los usuarios finales.

En la actualidad, se almacena, maneja y explota, la información para poder tomar decisiones dentro de la empresa o ámbito en donde se desarrolla el sistema.

En este caso lo que se almacenará son datos personales y relaciones entre los usuarios de este sistema.

Según (Informática aplicada a la gestión de empresas, pág. 14) entendiéndolo por dato un elemento que carece de significado ni coherencia por sí mismo que pueden ser un valor numérico o caracteres, todo esto sin procesar.

En las organizaciones es muy importante la información, ya que es utilizada para cumplir los objetivos y metas propuestos. A continuación, se describen las fases del proceso administrativo en una organización.

4.1.1.- Proceso administrativo

Según (Administración, un enfoque global, 1998, pág. 35-37) el proceso administrativo ayuda a mantener un control y planeación de un proyecto a través de un flujo y retroalimentación entre cada una de sus fases las cuales son:

- Planeación.

- Organización.
- Dirección.
- Control.

Iniciando con la planeación, se plantearán las actividades a realizar, así como la forma más práctica y óptima de seguirlas para llegar a los objetivos que se quieren alcanzar.

En la organización, se definen los recursos humanos y materiales necesarios, así como las tareas asignadas a cada uno de ellos, se establece una estructura y desarrollo óptimo para la asignación de las actividades a realizar para cada uno de estos recursos, la interacción entre ellos y sus tareas tanto individuales como grupales.

En la dirección, se definen líderes que influyen en la guía y realización de las actividades para llevar a cabo las metas que son planeadas y asignadas en la organización.

En el control, se observa cómo se realizan las actividades, buscando que éstas se desarrollen de una manera eficaz, siguiendo las normas y metas a las que se desean llegar para comprobar si se cumplen los objetivos, o se tienen que hacer las correcciones a los errores individuales y organizacionales, haciendo así que los resultados se apeguen más a los planes hechos.

4.2.- Ingeniería de software

Para comprender la definición de ingeniería de software, se definirá que significa la palabra “software”, por lo que a continuación se describirán de forma breve algunas de definiciones, así como su clasificación y requerimientos:

Según (León, 1996, pág. 14) es considerado un productor de servicios para satisfacer las necesidades de los usuarios.

Según (Diccionario De La Lengua Española, 2010, s/p) lo considera como: “Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora”.

Según (Pressman, 2005. pág. 1) es considerado como un producto para las computadoras de cualquier tamaño o arquitectura, desarrollado por ingenieros en software.

Según (Campderrich, 2003, pág. 16) el software es un producto de consumo utilitario y masivo, un medio auxiliar que interviene de manera más o menos indirecta, pero a menudo imprescindible en su gestión y cada vez más en su proceso productivo, también existe un consumo privado de software , se puede considerar como un producto industrial.

En general a cada producto de tipo industrial corresponde un tipo de ingeniería que se entiende como un conjunto de métodos, técnicas y herramientas, que se utilizan tanto para el desarrollo o prototipo del producto como para fabricarlo, ya que el software no es una excepción, por lo tanto hay una ingeniería para el mismo.

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Según (Sommerville, 2005, pág.5) son los programas y la documentación asociada a éstos, para su correcto funcionamiento, que puede estar compuesto por más de un programa independiente, y pueden desarrollarse tanto para usuarios en general como particulares. Por lo anterior, el software es un producto que se desarrolla para ser utilizado en diversos dispositivos electrónicos como: computadoras, ipads, celulares, etc., para la satisfacción de las necesidades de los usuarios.

A continuación se describe la clasificación del software.

Según (León, 1996, pág. 29) el software se clasifica en tres tipos:

- Software base, es aquel en el que otros programas pueden usar para ejecutarse (como son los compiladores o un sistema operativo).
- Software de aplicación, es aquel en el que los usuarios hacen uso para desarrollar una tarea y así obtener un resultado de acuerdo al contexto en el que lo vayan a utilizar, esta clasificación a su vez se divide en:
 - Horizontal, que es el que se dirige al público en general como pueden ser los procesadores de texto.
 - Vertical, que se dirige a un público más restringido (como pueden ser un software local para alguna empresa).
- Software empotrado, es el que se ejecuta en una computadora, únicamente sobre un hardware y el usuario no tiene interacción con él, por ejemplo un driver de computadora.

Según (Sommerville, 2005, pág.6) el software puede clasificarse en dos tipos:

- Genéricos, son aquellos que son fabricados para un público en general, las empresas los producen con fines lucrativos y su venta incluye generalmente una licencia, Por ejemplo procesadores de texto, editores de imágenes, paquetería de oficina, etc.
- Personalizados, son aquellos que son diseñados para un cliente en específico y hardware específico, por ejemplo: sistemas para control estadístico de un área o tareas específicas de un cliente.

Según (Pressman, 2005. Pág. 8) el software se divide en siete en 7 categorías:

- Software de sistemas, son pequeños programas por separado que hacen una tarea específica y son utilizados para ensamblar un software más grande.
- Software de aplicación, son sistemas en los que cada uno hacen actividades independientes y en tiempo real de un trabajo específico.
- Software científico y de ingeniería, son sistemas destinados a procesos numéricos

para el desarrollo científico, únicamente para hacer los análisis numéricos.

- Software empotrado, es aquel que se encuentra dentro de la memoria para que sea ejecutado internamente, mediante funciones que utilizará el usuario final o el sistema con funciones limitadas o de control.
- Software de Línea de Productos, este tipo de software está diseñado con un propósito en específico pero dirigido a varios tipos de clientes, como son los sistemas de registro.
- Aplicaciones basadas en web, ya que son en un aspecto amplio aplicaciones, pueden considerarse como un conjunto de archivos de hipertexto con la finalidad de mostrarle al cliente información, ya sea en forma escrita o gráfica.
- Software de Inteligencia Artificial, resolviendo problemas muy complejos que no se pueden resolver de una manera directa, aplica algoritmos de aprendizaje, reconocimiento de patrones, robótica, etc.

A continuación se describirá la clasificación del software de acuerdo a su licencia.

Según (PC actual, 2010, pág. 4) la Licencia en el software es un acuerdo que se tiene entre él o los que desarrollan el programa y el usuario que hará uso del mismo, donde el usuario se compromete a seguir ciertas condiciones de uso al aceptar la licencia.

Las licencias, se dividen en dos grandes grupos, las de licencia de propietario y las de licencia libre.

Software propietario, hace una división más en el software de propietario, el shareware y el freeware.

El shareware, es un tipo de software que tienen cierta clase de permisos de uso, pueden ser programas de prueba o programas en los que su funcionamiento es limitado hasta adquirir la licencia completa.

Los de freeware, son un tipo de software en los que el usuario puede disponer de todas las funciones que ofrece, pero no puede modificar el código fuente del mismo, no puede venderlo o utilizarlo de un modo empresarial, en ocasiones estos software suelen ser versiones viejas, este software limita su función tanto en aplicaciones, uso y modificaciones. Cabe resaltar que el software que se encuentra disponible en el mercado es generalmente de propietario, Por ejemplo: Microsoft office, Sistema operativo Windows y photoshop, entre otros.

Software libre u open source, no es necesariamente gratuito y si no lo es, su venta es bajo ciertas condiciones, la idea es que el usuario recibe el código fuente junto con los permisos de modificarlo, según sea necesario, éste se puede distribuir respetando la licencia del software libre.

Según (Portalprogramas.com, 2010, Pág. 5) el software libre no tiene que ver con aspectos económicos, puede distribuirse y/o modificarse con sus restricciones, sin fines comerciales y puede ser para usos personales o educacionales, por ejemplo: OpenOffice, Gimp e InfraRecorder, entre otros.

Según (Blanchard, 1995, Pág. 18) la ingeniería del software es una aplicación de

conocimientos científicos y de ingeniería, para poder satisfacer una necesidad utilizando retroalimentación y control en sus métodos, integrando factores técnicos y tomando en cuenta el ciclo de vida, la utilidad del sistema y relaciones funcionales, así como sus actividades a desarrollar, dentro de los conocimientos científicos encontramos a la observación, identificación, descripción, investigación experimental y explicación del caso a tratar.

4.2.1 Requerimientos del software

Según (Pressman, 2005, Pág. 155) son una tarea donde se identifican las necesidades, características y funciones del sistema o software que requieren los clientes. Es de fundamental importancia recalcar, que los requerimientos pueden cambiar durante el desarrollo del software, debido a que los clientes identifican nuevos factores que podrían mejorar el funcionamiento.

Según (León, 1996, Pág. 42) son una especificación que proporciona el cliente respecto del problema que se va a resolver, por lo que se plantea una idea, es deseable que ésta sea clara y cubra las necesidades, para que así se puedan describir las funciones que contendrá el sistema, aquí es donde se pueden obtener los requerimientos, las necesidades y las funciones, una vez obtenidas se deben plasmar en un documento con la finalidad de no perder datos y así poder satisfacer las necesidades de los usuarios.

En los requerimientos del software se desarrolla un modelo lógico a grandes rasgos de las funciones del sistema, tomando en cuenta el caso ideal donde no se tienen limitaciones, estas funciones están dirigidas a cubrir todas las expectativas del usuario, posteriormente se ligan las limitantes entre ambos esquemas para tener el diseño.

Según (Sommerville, 2005, pág. 25) se explica el desempeño ideal del sistema, sus funciones y propiedades más básicas, esenciales y deseables que debe de tener el sistema, estos requerimientos deben de ser consultados con el cliente y con los usuarios finales, dividido en tres tipos esenciales de requerimientos:

- Requerimientos funcionales abstractos, en donde se plasman las funciones básicas a grandes rasgos de una forma abstracta, si se desea hacerse con más detalle, se explicará en los subsistemas.
- Propiedades del sistema, éstas no tienen que ver con el desarrollo del sistemas, más bien las propiedades intervienen en aspectos como son la seguridad con la que contará, el desempeño que tendrá ante algunas situaciones y la disponibilidad en éstas.
- Características que no debe mostrar el sistema, así como nos especifican que es lo que se desea que muestre el sistema, pues habrá cosas que se solicitan que se tengan, pero que no deben de ser mostradas, como las fechas de registro ya que éstas son usadas en estadísticas, que no tienen por qué ser mostradas en la

información del registro.

En esta etapa, se colocan en un documento los requerimientos y funcionalidades que ofrecerá el sistema que se va a implementar, contestando así las preguntas “¿Qué?” y “¿cómo?”.

Según (Campderrich, 2003, pág. 109) el primer proceso en el desarrollo de un software es la recolección de requisitos, que es la base para los otros componentes del proceso.

La recogida de requisitos, busca obtener información sobre dos aspectos esenciales de software que se desarrolla:

- Los procesos que debe realizar sobre los datos.
- La manera en que debe pedir a los usuarios los datos de entrada y qué función del software quieren utilizar en cada momento y presentarles los resultados, qué es lo que se identifica como “interfaz de usuario” del futuro software.

Aunque se trata de dos aspectos del software bastante diferentes, puesto que las fuentes de información son las mismas, conviene recoger estos dos tipos de requisitos al mismo tiempo.

Fases de recolección de datos:

- En la fase de inicio se identifican casi todos los casos de uso, para delimitar el alcance del proyecto, pero sólo se describen detalladamente los más críticos.
- Durante la fase de elaboración se describen la gran mayoría de los casos de uso, aunque sólo se implementa una pequeña parte de los mismos.
- En la fase de construcción se describen e implementan el resto de los requisitos.
- En la fase de transición sólo intervienen los requisitos eventuales, nuevos o modificados.

4.2.2 Flujo de los procesos

Según (Sommerville, 2005, pág. 13) en el flujo de procesos se plasman las ideas, así como conocimientos que se tienen sobre el software, en este punto convergen tanto los usuarios, como el desarrollador usando como medio de comunicación las herramientas de desarrollo, tecnologías o software a utilizar.

Según (Pressman, 2005. Pág. 28) son metodologías que serán utilizadas para el desarrollo del software, en el cual se especifican las tareas a realizar, según la dificultad o complejidad se pueden ir dividiendo para llevar a cabo un proceso que define quién está haciendo qué, cuándo y cómo lograr cierta meta.

Algunas de las tareas más utilizadas a grandes rasgos en un proceso de un sistema pequeño son:

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

- Comunicación, es una parte fundamental en el que se tiene que consultar al cliente, para tener todos los requerimientos del sistema lo mejor explicados posible.
- Planeación, es donde se planean las tareas que van a ser desempeñadas, así como cuáles serán los riesgos y los resultados esperados de las mismas.
- Modelado, es donde se crean bosquejos o modelos de lo que será el sistema según lo requerido, para ser explicados al usuario con facilidad y que pueda entender si es así como desea que el funcionamiento del sistema se lleve a cabo.
- Construcción, es donde el desarrollo del código del sistema se lleva a cabo y se hacen las pruebas de funcionamiento necesarias para encontrar errores o ajustes en el sistema.
- Despliegue, es donde se da una muestra del sistema en su totalidad, o un adelanto al cliente para que analice y evalúe si es correcto el funcionamiento y que realice las correcciones necesarias.

Estas etapas pueden variar en tanto más grande es el proyecto pueden hacerse más y mucho más complejas.

Según (Cantone, 2006, Pág.17) en las metodologías los flujos que se siguen son diversos, pero se puede generalizar en tres grandes grupos, los cuales son:

- Planificación, plantea el proyecto tanto en los ámbitos económicos como de tiempo de desarrollo.
- Implementación, es donde se dividen las actividades a desempeñar para la realización óptima del proyecto.
- Puesta en producción, es donde se pide al cliente que abale el proyecto, el cual debe estar ya en funcionamiento, para verificar si cumple con sus expectativas y satisface todos sus requerimientos.

Los flujos de los procesos nos describen como es que la información se mueve a través del sistema, es importante que en el desarrollo de software se trabaje en los ciclos de desarrollo.

4.2.3 Ciclo de desarrollo de sistemas

Según (León, 1996, Pág. 41) en el ciclo de desarrollo de software se selecciona un tipo de metodología que cumpla con las necesidades a desarrollar del sistema, ya que no es única una metodología, se trabaja con la que cubra los requerimientos y la organización que se lleva.

Modelo de cascada: En este modelo, cada una de las etapas que lo conforman llevan una secuencia de tiempo, esto es que se comienza con una inmediatamente después de que se

concluye la anterior.

- Definición de requisitos, en esta etapa se definen los requisitos y requerimientos del sistema software a partir de consultas con los clientes y los usuarios del futuro sistema software, de esta etapa surge el documento de especificación de requisitos que contiene toda la descripción del sistema sin entrar en detalles de diseño.
- Diseño, en esta etapa se dividen los requerimientos en subsistemas, y describen las relaciones fundamentales del sistema de software, de esta etapa surge el documento de diseño del software que contiene toda la descripción del sistema desde el punto de vista del diseño.
- Implementación, en esta etapa el diseño del software se lleva a cabo implementándolo en un lenguaje de programación, aquí se implementa el código fuente, se crean las bibliotecas y se reutilizan los componentes.
- Transferencia del producto, en esta etapa los programas se integran y se prueban como un sistema completo para asegurar que se cumplen los requerimientos del software, después de las pruebas el sistema se entrega al cliente.
- Evolución, es la etapa más larga de todos los procesos de desarrollo, el sistema se instala y se pone en funcionamiento corrigiendo todos los errores no descubiertos en las etapas anteriores, también se mejora la implementación añadiendo nuevos requerimientos siempre que el usuario los necesite.

Como ventajas del modelo se tienen:

- Es muy utilizado y casi todo desarrollador identifica la secuencia.
- Una vez definidos concretamente los requisitos, es muy rápido llegar a la fase de desarrollo.
- Una mejor gestión dado que se lleva una documentación en cada fase.

Según (Sommerville, 2005, Pág. 8) dentro de los ciclos de desarrollo de los procesos del software se encuentran los siguientes:

Modelo de cascada, en el que se dividen las tareas, tomándolas por separado y se basan en las anteriores actividades realizadas, éstas tienen su propia fase en la que se definen los requerimientos, se desarrolla y se prueba de una manera aislada, se firma esa fase del proyecto y se continúa con la siguiente (Fig. 1).

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

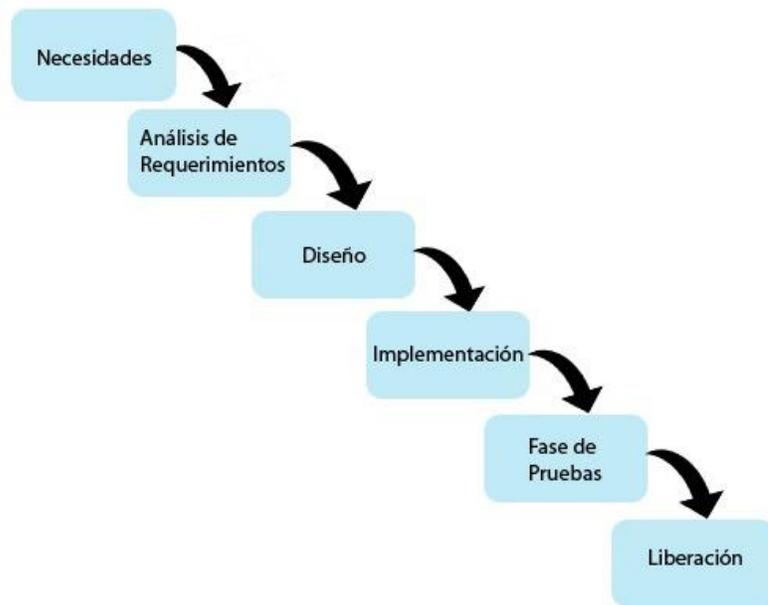


Fig. 1 Ciclo de desarrollo en cascada

Modelo iterativo, se crea un sistema muy rápido y general en el que solo se usan las ideas más abstractas, se muestra al cliente y el sistema se “mejora” según las especificaciones que vaya dando hasta producir el sistema de la forma más detallada que se pide (Fig. 2).

Según (León, 1996, Pág. 41) a diferencia del modelo de cascada en este modelo se van proponiendo “prototipos” al usuario, esto ayuda mucho a que vaya validando el sistema antes de la implementación final. Estos “prototipos” se pueden agrupar según la función y finalidad que se le den en dos grupos básicos:

- Modelo basado en prototipos desechables.
- Modelo de prototipado incremental.

Algunas características son:

- Se tiene una mayor visibilidad en el desarrollo del sistema dado que se experimenta con prototipos de prueba en ejecución antes del sistema completo.
- Muestra de modo gráfico para mayor comprensión tanto las interfaces como los comportamientos que tendrá el sistema.
- La combinación de este modo gráfico y de la documentación que se va llevando a través del desarrollo facilita la aprobación de estos “prototipos”.
- Los modelos que son desarrollados no se desperdician ya que sobre ellos se hace la mejora o refinamiento para el siguiente.

Debido a la evaluación de requisitos, y los “prototipos” desechables, puede no mantenerse la arquitectura en el desarrollo incremental.

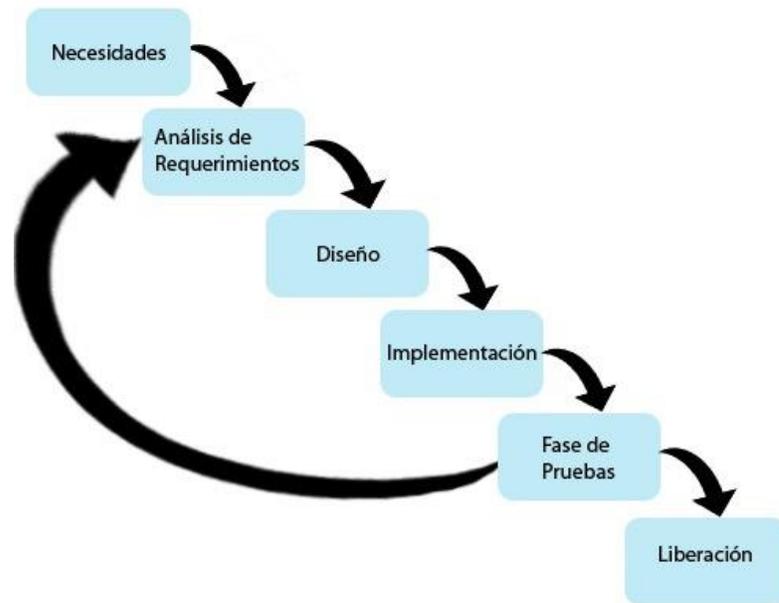


Fig. 2 Ciclo de desarrollo iterativo

Según (Sommerville, 2005, Pág. 25) existe un modelo denominado espiral (Fig. 3), este modelo va “evolucionando” de un sistema básico al principio que van sacando versiones cada vez más complejas a través de los siguientes pasos:

- Se entabla una conversación entre el usuario y el desarrollador sobre lo que debe hacer el sistema.
- La planificación, es en la que se definen los tiempos y tareas que se desarrollarán en el proyecto.
- El análisis de riesgo, es en donde se gestionan y evalúan los riesgos en el desempeño de cada una de las tareas.
- La ingeniería, es en la que se llevan a cabo las tareas para los desarrollos de las aplicaciones para el sistema.
- En la construcción y acción, se ponen en marcha las tareas planteadas para crear las aplicaciones probando, ejecutando y proporcionándole las documentaciones necesarias al usuario para su uso.
- La evaluación del cliente, es dar los pasos a seguir para que el sistema haga lo que el usuario requirió en la etapa de ingeniería.

Y esto se hace en cada etapa del espiral hasta conseguir un sistema complejo y las necesidades del cliente cubiertas.

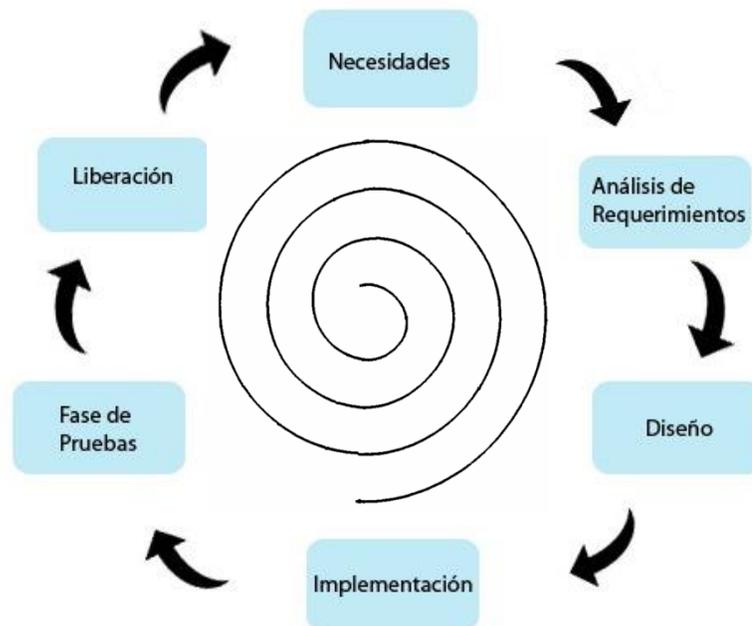


Fig. 3 Ciclo de desarrollo espiral

4.2.4 Metodologías de software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de aplicaciones, se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

La creación de estas metodologías es principalmente para:

- Reducir el tiempo del desarrollo.
- Mejorar la calidad.
- Reducir costos.

A continuación se mencionan algunas de ellas.

4.2.4.1.- Desarrollo de software adaptable (ASD, adaptive software development)

Según (Highsmith, 2002, pág.309) es una metodología de desarrollo de software, que surgió de una metodología de desarrollo rápido para aplicaciones. ASD, incorpora el principio de la adaptación continua, es decir que su principio es adaptarse al cambio en lugar de luchar contra él.

ASD, da forma a las fases básicas de la gestión ágil en:

- Especulación, compuesta por los siguientes pasos:
 - Inicio, para determinar la misión del proyecto.
 - Determinación del marco temporal del proyecto.
 - Determinación del número de iteraciones y la duración de cada una.
 - Determinación del objetivo de cada una.
 - Asignación de funcionalidad a cada iteración.

- Colaboración, es el desarrollo en equipo del trabajo de construcción y gestión del producto.

- Aprendizaje, donde en cada iteración se revisa:
 - Calidad, con criterios de cliente se utiliza para poder aprender de los errores e iniciar nuevamente el ciclo de desarrollo.
 - Calidad, con criterios técnicos.
 - Funcionalidad desarrollada.
 - Estado del proyecto.

Las características básicas de ASD son:

- Iterativo.
- Orientado a los componentes software más que a las tareas.
- Tolerante a los cambios.

Según (Toro, 2013, Pág. 27) las principales características de este método son que es iterativo, orientado a componentes del software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje.

4.2.4.2.- Proceso de desarrollo unificado (RUP)

Según (utvm.edu.mx, 2014) el proceso unificado conocido como RUP, es un modelo que permite el desarrollo de software a gran escala, mediante pruebas continuas y retroalimentación, garantizando el cumplimiento de ciertos estándares de calidad.

Los pasos para el método son:

- Modelo de negocios: Regla de negocios.
- Requerimientos del sistema.
- Análisis y diseño: Se analizan los requerimientos y se desarrolla el sistema.
- Implementación: Se pone en funcionamiento el sistema.

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

- Pruebas: Se realizan pruebas para asegurar que aplica las reglas de negocio.
- Configuración y cambios: se configuran los sistemas donde correrá el sistema.
- Administración del proyecto: se prepara el sistema en los equipos.
- Implementación: el sistema se pone en funcionamiento.

La estructura del ciclo de vida del proceso de desarrollo unificado es la siguiente y los pasos anteriores cruzan por cada fase.

- Fase de concepción, tiene como propósito definir y acordar el alcance del proyecto con los patrocinadores, identificar los riesgos potenciales asociados al mismo, proponer una visión muy general de la arquitectura de software y producir el plan de las fases y el de iteraciones.
- Fase de elaboración, se seleccionan los casos de uso que permiten definir la arquitectura base del sistema, se realiza la especificación y el primer análisis del dominio del problema, se diseña la solución preliminar.
- Fase de construcción, se completa la funcionalidad del sistema, para ello se deben clarificar los requerimientos pendientes, administrar los cambios de acuerdo a las evaluaciones realizados por los usuarios y se efectúan las mejoras para el proyecto.
- Fase de transición, asegura que el software esté disponible para los usuarios finales, ajusta los errores y defectos encontrados en las pruebas de aceptación, capacita a los usuarios y proveer el soporte técnico necesario.

Según (Toro, 2013, Pág. 28) se divide en siete etapas para el proceso de desarrollo de software

- Modelo de negocio: Se definen de manera clara las reglas de negocio relativas al manejo de la información.
- Requisitos funcionales: Se determinan los requisitos de funcionamiento y de operación.
- Análisis y diseño: Se diseñan los programas, módulos, rutinas, y demás componentes del sistema, buscando una arquitectura óptima del mismo.
- Implementación: Se instalan los bancos de datos y se montan igualmente las facilidades de comunicación de los programas e interfaces previstas.
- Pruebas: Se realizan todas las pruebas tanto a nivel de los módulos independientes como las resultantes de la integración de estos.
- Configuración y administración de cambios: Se llevan a cabo los cambios tanto en la configuración de equipos, servidores y programas así como en las diferentes interfaces.
- Instalación: Se pone en funcionamiento el producto del proyecto.

En forma general, sus características son:

- Es el proceso de desarrollo más general de los existentes actualmente.

- Es una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace, qué, cuándo y cómo).

En la tabla 1 observamos una comparativa entre ambas metodologías

Se eligió para el sistema de mando de PROFEPA, el modelo RUP por estar centrado en la arquitectura y guiado por los casos de uso. El ciclo de vida de RUP, es una implementación del Desarrollo en Espiral, fue creado ensamblando los elementos en secuencias semi-ordenadas, al definir se desarrolla una solución preliminar que puede ir variando en cada entrega de la aplicación dejando así una versión disponible para los usuarios finales y así ajustar errores.

Tabla 1. Comparativa de metodologías RUP y ASD

Metodología	Estabilidad	Flexibilidad	Rendimiento	Diseño	Implementación	Pruebas
RUP	Se basa en la utilización de casos de uso, según sea la arquitectura del software.	Se adapta al espacio, tiempo, necesidad y recursos disponibles	Se sub divide el trabajo por áreas y se asignan responsabilidades. Incremental	Sistemas orientado a objetos, iterativo e incremental	Adaptable a cualquier grado de complejidad del proyecto	Verificación y Adaptabilidad , Aplicando cada ciclo según sus funciones y aislando las fallas
ASD	Se basa en la adaptación continua a circunstancias cambiantes	Su funcionamiento es cíclico y reconoce que en cada iteración se producirán cambios e incluso errores.	No hay un ciclo de planificación-diseño-construcción del software, sino un ciclo especular colaborar-aprender	Iterativo, Orientado a los componentes de software , Tolerante a los cambios	Se realizan estimaciones de tiempo sabiendo que pueden sufrir desviaciones	Su funcionamiento es cíclico y reconoce que en cada iteración se producirán cambios e incluso errores.

En los requerimientos de los sistemas, en el caso de que sean de un gran tamaño o que manipulan información valiosa que puede usarse para análisis estadístico o toma de decisiones, es importante contar con un banco de información en donde almacenar todos los datos recopilados, para esto se cuenta con gestores de bases de datos que nos apoyan a almacenar y consultar.

En el siguiente tema se mencionará que son las bases de datos, los lenguajes con los que se trabajan las bases de datos, así como analizarán que son los gestores de bases de datos.

4.3 Bases de datos

Como una introducción a este tema, se mencionan algunas definiciones de base de datos como:

Según (Silberschatz, 2002, pág.1) una base de datos es una colección de datos relevantes

para una empresa.

Según (Pérez, 2003, pág. 25) el término base de datos “es un conjunto de datos relacionados entre sí, que están estructurados de forma que puede accederse a ellos automáticamente e independientemente de los programas que gestionan esos datos. Esta independencia se refiere a la posibilidad de modificar la estructura de los datos sin necesidad de modificar los programas que los manipulan, evitando con ellos los problemas de la actualización previamente existentes”.

Según (Date, 2001, pág.10) una base de datos es un conjunto de datos perdurables que son utilizados por los sistemas de aplicación de alguna empresa dada.

Según (Ramez, 2000, pág.2) es una abstracción del mundo real que almacena registros o datos, estos están relacionados entre sí y agrupados lógicamente para obtener información coherente.

Por lo anterior, una base de datos es una colección de datos relacionados entre sí que tienen ciertas características.

A continuación se describen algunas de las características de la base de datos.

4.3.1-Características

Muchos autores definen sus propias características al diseñar una base de datos, sin embargo, concuerdan que toda base de datos, debe tener integridad, consistencia y redundancia.

4.3.1.1.-Integridad

Según (Silberschatz, 2002, pág.3) los valores que serán ingresados a la base de datos deben cumplir con ciertas restricciones como por ejemplo que todos los valores sean numéricos.

Según (Date, 2001, pág.18) se asegura de que al introducir los valores, estos sean correctos por medio de restricciones, tomando así a la integridad como la exactitud o corrección en los datos ingresados.

Según (Camps, 2005, pág.18) la corrección o consistencia de los datos por medio de reglas de integridad que se pueden dar muchas razones además de una mala inserción de los datos; como fallas en el sistema o en el software.

Según (Mora, 2003, pág. 6) existen una serie de condiciones que se deben cumplir cuando se crea y mantiene una base de datos, algunas de estas condiciones pueden ser:

- Unicidad: Registros únicos dentro de una entidad, para no repetir información.
- Referencia entre tablas: Una relación entre tablas (ejemplo: antes de insertar o actualizar en una tabla A relacionada con una tabla B, debe estar el registro en B para poder haber una relación de existencia entre ambas).
- Condiciones de inserción: Un atributo está diseñado para almacenar un solo tipo de valores en específico de un universo (ejemplo: atributo sexo, acepta un solo carácter que puede ser una letra, del universo del alfabeto únicamente puede aceptar "F" o "M").

Cuando las condiciones que se especifican al principio se cumplen se dice que las condiciones de integridad se cumplen.

Según (Cobo, 2007, pág. 10) cuando la información que se guarda en la base de datos es utilizada por algún sistema es real y confiable, es decir cuando la información es correcta, entonces la información es íntegra, las razones por las que no se tendría integridad en la información pueden ser:

- No se tiene consistencia en la información.
- Almacenamiento de información falsa o que no sea posible la existencia de esa información (ejemplo que introduzcan nombres falsos, o fechas de nacimiento que no son posibles para una persona en el rango de tiempo).

La integridad referida a la consistencia de la información está en función del cumplimiento de las reglas que se imponen.

4.3.1.2.-Consistencia

Según (Cobo, 2007, pág. 9) la consistencia se refiere a que no exista información contradictoria o incongruente en la base de datos, esta puede darse cuando existen copias de la misma en diferentes directorios y al hacerse la modificación de uno de estos, los demás permanecen sin modificar.

Según (Silberschatz, 2002, pág.2) almacenando una gran cantidad de copias de la información, al ser modificadas algunas de estas copias pueden no coincidir, al evitar esta incompatibilidad se mantendría la consistencia.

Según (Mora, 2003, pág. 6) el término consistencia va muy ligado al término de integridad, y se relaciona como la congruencia en la base de datos, como todo sistema puede darse el caso de una falla cuando se está realizando una modificación en dos tablas, si se realiza un cambio en una y en otra no, es posible, que la información no sea congruente si los datos están relacionados, a esto se llamaría una inconsistencia en la información.

La consistencia que el manejador de base de datos se debe proporcionar va ligada al diseño de la misma para evitar en lo posible la duplicidad de información y si esta existe,

garantizar que al hacer la inserción o actualización de uno de los registros se modifiquen los valores repetidos también en otras direcciones.

4.3.1.3.-Redundancia

Según (Cobo, 2007, pág. 9) es almacenar múltiples copias de la misma información siendo cada una para un sistema diferente, no toda la redundancia es mala, siendo controlada es buena, ya que permite hacer uso de esta para realizar búsquedas más rápidas.

Según (León, 2008, pág. 36) se manifiesta en copias de la misma información, esta redundancia no controlada genera un gran consumo de espacio en los medios de almacenamiento, haciendo que el tiempo de acceso a consultar sea mayor.

Según (Silberschatz, 2002, pág. 256) cuando hay redundancia se almacena información secundaria o adicional que habitualmente no se necesita pero que puede utilizarse en una situación de fallo de alguna unidad de almacenamiento para reconstruir la información perdida.

La redundancia a gran escala produce que los tiempos para almacenar y consultar información sean más altos, en un nivel controlado puede ser de utilidad, tanto en velocidad de acceso, consultas o respaldos.

4.3.2.-Administrador de la base de datos (DBA)

Según (Date, 2001, pág.16) es un profesional en tecnologías de la información. El trabajo del DBA radica en crear la base de datos real y realizar los controles técnicos que se necesitan llevar a cabo las disposiciones de las políticas hechas por el administrador de datos. El DBA además es responsable de asegurar que el sistema esté funcionando con el rendimiento adecuado y de aportar una variedad de otros servicios técnicos.

Según (Cobo, 2007, pág. 8) es la persona o personas encargadas del manejo del sistema con funciones como:

- Establecer y alterar el esquema de la base de datos así como las restricciones de integridad.
- Implantar y cambiar los métodos de acceso a la información y las estructuras en las que se almacena la misma.
- Dar permisos de acceso a los usuarios de la base de datos.
- Avalar el funcionamiento correcto del sistema, dando soporte técnico y solucionando los problemas de intento de acceso ilegal a la base de datos, además de los casos se hace lento.
- Generar los respaldos de seguridad de las bases de datos conocidos como "backups".

Según (León, 2008, pág. 20) es encargado principal de un DBMS con las principales funciones:

- Definir el diseño: Es el que crea en lenguaje de definición de datos (Data Definition Language, DDL por sus siglas en inglés) el diseño original de la base de datos.
- Organización física y la modificación del diseño.
- Gestionar la seguridad: Garantiza la persistencia de los datos, con copias de seguridad y dar permisos a los usuarios.
- Colocar las restricciones de consistencia.

Según (Gómez, 2008, pág. 129) es el encargado de la administración de los recursos del DBMS y de la base de datos. Administra, coordina y vigila a quien otorga permisos.

4.3.3.-Diccionario de datos

Según (León, 2008, pág. 224) contiene las descripciones y definiciones de los elementos, la interrelación que existe entre ellos en la base de datos, así como de su estructura y características con la que fue creada.

Según (Date, 2001, pág.46) es un diccionario diseñado por usuarios para usuarios, en donde se guardan los datos o descripciones de los datos de la base de datos, además muestra la información como objetos, diagramas, restricciones; entre más detallado más extenso.

Según (Cobo, 2007, pág. 15) depósito de información de una base de datos:

- Descripciones internas:
 - Reglas de correspondencia, restricciones, esquemas internos, etc.
- Descripciones externas:
 - Diagramas, esquemas externos, usuarios y sus permisos, etc.
- Información de seguridad como son las claves de acceso.

El diccionario de datos es la recopilación de la información sobre la base de datos, proporciona la descripción de sus objetos, sus atributos, restricciones, relaciones, funciones y diagramas.

4.3.4.-Modelos de datos

Según (Date, 2001, pág.14) en los modelos de datos se define la forma lógica en la que se comportaran los objetos de la base de datos, el modelo es lo que deben identificar y comprender los usuarios.

Según (Piattini, 2000, pág.23) es la aplicación de reglas y conceptos para traducir

manipular y describir lo más cercano posible al comportamiento del mundo real en una abstracción para poder almacenarla en una base de datos.

Según (Silberschatz, 2002, pág.17) es la descripción de los datos, su relación y su consistencia del mundo real por medio de herramientas conceptuales.

Los modelos de datos apoyan a identificar y comprender de una manera más simple los objetos del mundo real y los cuales se van a almacenar en la base de datos.

4.3.4.1.-Entidad-Relación

Según (Date, 2011, pág.425) una entidad es algo que podemos identificar y definir en el mundo real (por ejemplo una computadora, un árbol, una persona, etc.) para clasificarlas posteriormente en normales y débiles que son aquellas que no pueden existir sin la existencia de otra (un puesto en una empresa, si no existe la entidad empleado y la entidad departamento, para poder crear empleados en el Departamento debe de haber un jefe de Departamento), cada entidad posee sus propiedades que poseen todas en común.

Una relación, es un vínculo que existe entre las entidades (por ejemplo la relación entre una empresa y un empleado es un contrato, la relación entre un alumno, un profesor y un salón de clases y una materia es un grupo que las contiene con un horario de uso).

El diagrama entidad-relación es la vinculación entre entidades, existen tres tipos de estas relaciones: “uno a uno”, “uno a muchos” y “muchos a muchos”

Según (Piattini, 2000, pág. 49) es un modelo para poder tener una vista de las relaciones que existen entre los datos de una manera abstracta.

Una entidad puede ser cualquier objeto de la realidad que se quiere almacenar en una base de datos. (Ejemplo: Real: estudiante, profesor, salón; abstracto: curso.).

Cada una posee atributos, propiedades o características que tiene un tipo las hace homogéneas entre las mismas.

La relación, es una asociación entre las entidades (por ejemplo la relación que existe entre un curso y un profesor es que el profesor imparte el curso, la relación que existe entre una empresa y un empleado es que la empresa contrata un empleado) pueden existir más de una relación entre dos entidades (el curso recibe a un profesor, un empleado trabaja para una empresa).

Según (Silberschatz, 2002, pág. 18) consta de las entidades que son objetos básicos abstractos basados en los que existen en el mundo real y de las relaciones que existen entre éstos, expresando así el mundo real en un esquema conceptual.

Por lo que se define a una entidad como una cosa u objeto del mundo real que se distingue de todos los demás, los conjuntos del mismo tipo de entidades comparten los mismos atributos o propiedades (por ejemplo una persona es una entidad, todas las personas tienen un nombre, una raza, una nacionalidad, un color de ojos, un año de nacimiento) los atributos describen las propiedades de cada miembro del conjunto, al

almacenar en la base de datos es información que contienen todos los miembros de este conjunto de entidades y cada atributo contiene un valor para cada entidad.

Las entidades son aquellos objetos en las que se desea guardar en la base de datos cada una de estas entidades poseen propiedades generales para definir a esa entidad (una CPU contiene un número de serie, un modelo, especificaciones técnicas como disco duro, memoria RAM que contiene, etc.) al colocar valores en estos datos generales que todas las entidades de este mismo tipo contiene las hace únicas para definirla y diferenciarla de otras del mismo tipo.

4.3.4.2.-Modelo relacional

Según (Date, 2001, pág. 109) el modelo relacional se ocupa de tres aspectos de la información: “la estructura de datos”, “la manipulación de datos” y “la integridad de los datos”.

Dentro de la manipulación se tienen dos aspectos importantes que son el “álgebra relacional” y el “cálculo relacional”.

En la estructura de los datos se tienen los tipos de datos y los dominios de éstos, teniendo también la relación entre las entidades, tupla, cardinalidad, atributo, grado, dominio y clave primaria.

Anteriormente se definió lo que es una relación, una clave primaria es un atributo único, una tupla corresponde a una fila y un atributo a una columna, cardinalidad es el número de tuplas que se tienen, el grado es el número de atributos, el dominio es el número de valores.

Según (Piattini, 2000, pág. 93) es un modelo que estructura de forma lógica las relaciones de los datos en tablas, haciendo más sencillo y uniforme al hacer las colecciones de entidades como tablas haciéndolo más uniforme, representa tanto las entidades como las relaciones que representan sus asociaciones con tablas.

Según (Silberschatz, 2002,págs. 53, 160) este modelo permite generar esquemas relacionales para almacenar información sin redundancia innecesaria, así como recuperarla de una forma sencilla al diseñar, a partir de este una base de datos relacional, creando un conjunto de tablas con un nombre exclusivo

Cada una de éstas almacena un conjunto de entidades, cada renglón o tupla es una entidad y cada fila representa un atributo o relación común entre éstas, para cada atributo se tienen un conjunto de valores al que se le da el nombre de dominio

Para todas las relaciones que se creen, sus dominios deben de ser atómicos, se consideran atómicos cuando se llega a una unidad indivisible (por ejemplo el nombre de una persona consta de apellido paterno, materno y nombre, para que sea atómico el nombre debe dividirse en estas tres unidades), si el valor es nulo indica que el valor es desconocido o que no existe.

Anteriormente se definió lo que es una “relación”, una llave primaria la cual es un identificador único, que diferencia a un conjunto de entidades una de la otra (por ejemplo número de trabajador, todos los trabajadores pueden tener valores de los

atributos en común como el sueldo, pero el número de trabajador diferencia al trabajador 1 del trabajador 2), el dominio de una tabla es el universo de valores que puede tomar ese atributo (por ejemplo atributo "sexo" es masculino y femenino), el grado es el número de atributos tiene una entidad (ejemplo: tabla "libro" tiene un título, autor, editorial, año; su grado es 4), cardinalidad es el número de tuplas (ejemplo: en una biblioteca se tienen 2000 libros registrados diferentes, la cardinalidad de la tabla libro es 2000), atributo es una propiedad en común de una entidad, en una tabla es una columna (ejemplo: nombre, año, autor, etc.).

4.3.5.- Álgebra relacional

Según (Date, 2001, pág. 150) consta de ocho operadores: unión, intersección, diferencia, producto cartesiano, seleccionar, proyectar, juntar y dividir.

A continuación se describe lo que regresa cada uno de estos operadores.

- Selección: Una relación que contiene todos los renglones de una relación especificada que satisfacen una condición especificada.
- Proyectar: Una relación que contiene todas las tuplas o subtuplas que quedan en la relación especificada después de quitar los atributos especificados.
- Producto: Una relación que contiene todas las tuplas posibles que son una combinación de dos tuplas, una de cada una de dos relaciones especificadas.
- Unión: Una relación que contiene todas las tuplas que aparecen en una o en las dos relaciones.
- Intersección: Una relación que contiene todas las tuplas que aparecen en las dos relaciones especificadas (en ambas, no en una u otra).
- Diferencia: Una relación que contiene todas las tuplas que aparece en la primera pero no en la segunda de las dos relaciones especificadas.
- Juntar: Una relación que contiene todas las tuplas posibles que son una combinación de tuplas de cada una de dos relaciones especificadas, tales que las dos tuplas que contribuyen a cualquier combinación dada tengan un valor común para los atributos comunes de las dos relaciones (y ese valor común aparece sólo una vez, y no dos, en la tupla resultante).
- Dividir: Toma de dos relaciones unarias y una relación binaria y regresa una relación que contiene todas las tuplas de una relación unaria que aparecen en la relación binaria y que a la vez coincide con todas las tuplas de la otra relación unaria.

Según (Silberschatz, 2002, págs.50-80) son un conjunto de operaciones que se le aplica a las tablas, para que el usuario pueda efectuar consultas, las operaciones fundamentales son: selección y proyección conocidas como unarias por que se hacen sobre una sola relación.

- Selección: En la selección se muestran las tuplas que cumplen con el predicado aplicado, simbolizado por la letra sigma minúscula (σ).
- Proyección: Devuelven una relación de argumentos, eliminando los argumentos no deseados, eliminando las filas no deseadas y dejando únicamente las que se quiere que se muestren, simbolizada por la griega mayúscula pi (Π).
- Unión: Está representada por \cup es la unión de conjuntos en los que no aparecen los valores repetidos, considere r y s como los conjuntos, en el cual valor de resultado debe ser igual en el número de atributos, y de estos atributos el i -ésimo de r y el i -ésimo de s deben ser iguales para todo i .
- Diferencia: Está representado por $-$ devuelve de las relaciones los valores que estén en una pero no en la otra.
- Producto cartesiano: Permite combinar la información de dos relaciones, denotado por \times .
- División: Ena expresión adecuada cuando desean saber “para todos” cuando desean todos valores de una relación (r_1) y con los valores seleccionados de otra relación (r_2).
- Intersección: Valores en los que coinciden ambos conjuntos.

4.3.6.- Cálculo relacional

Según (Silberschatz, 2002, pág. 75) sin especificar cómo es que se obtendrá la información, describe la información que se desea obtener, es una consulta no procedimental:

$$\{t | P(t)\}$$

Esto es:

Siendo P el predicado a cumplir, el conjunto de tuplas para los que P es cierto para t .

Según (Date, 2001, pág. 198) el cálculo es una enunciación descriptiva, a diferencia del álgebra que es prescriptiva, en la que se dan operadores claros como son la unión o la proyección, que dice como construir la relación, el cálculo nos da una definición de la relación.

Según (León, 2008, pág. 55) se describe la forma de conseguir la información que se desea de una forma organizada y formal pero sin definir el modo de cómo obtenerla.

En el cálculo las tuplas están formadas de átomos, o sea, que únicamente contiene un único valor.

Las descripciones que se dan para poder obtener la información del enunciado que se desea se puede simplificar a cálculo relacional en donde no se da de forma detallada la solución pero se da de una manera concreta lo que se busca.

4.3.7.- Normalización

Según (Silberschatz, 2002, pág. 525) las normas son reglas que definen la semántica y la sintaxis de los lenguajes de programación o las funciones a realizar en una aplicación. Generalmente se aplican en las bases de datos complejas, por ejemplo una base de datos creada por partes independientes entre sí o que tienen varias bases de datos heterogéneas en una misma empresa y necesitan estar interconectadas para intercambiar información es importante contar con estas normas.

Estas normas conocidas como “normas formales” son desarrolladas por una organización de normalización y aplicadas de manera general sin un procedimiento formal de reconocimiento.

Según (Date, 2001, pág.348) evitando los malos diseños en las bases de datos en las que existe redundancia, se ordena la información de tal manera que quede coherente, esto se logra siempre y cuando los valores válidos estén normalizados.

Según (Kroenke, 2003, pág. 127) en esencia la normalización se realiza cuando se tiene una entidad que contiene en un atributo más de un tema central en específico, este atributo deberá dividirse en uno o más de tal manera que cada atributo sea simple.

Según (León, 2008, pág. 36) es un conjunto de reglas que implica el estudio y conversión de estructuras de datos relacionales con propiedades de consistencia y mínima redundancia siendo así estable.

En la normalización se pueden eliminar las anomalías y recursividades que se puedan encontrar en la base de datos, aplicando modelos establecidos conocidos como formas normales.

4.3.7.1.-Primera forma normal (1FN)

Según (Silberschatz, 2002, pág. 161) en un conjunto de elementos si se considera que cada uno de sus componentes son indivisibles, en este caso se considera que están en primera forma normal.

Según (Kroenke, 2003, pág. 128) en una tabla cada una de sus celdas no puede contener un grupo de valores o arreglo de valores, debe contener un valor único; así como total de entradas que se haga a una columna debe ser del mismo tipo, ni puede haber nombres repetidos en las celdas.

Según (Covadonga, 1987, pág.49) menciona que un conjunto no contiene grupos repetidos cuando está compuesto por entradas unitarias o simples.

En la primera forma normal (1FN) se debe encontrar un único valor en cada atributo, no puede contener más de uno referenciado a un tema, en dado caso que esto suceda, además deberá contener un atributo por cada tema (por ejemplo un puesto y la persona

que ocupa el puesto en el mismo atributo llamado usuario, en este caso debe dividirse en puesto y usuario, el usuario a su vez en nombre, apellido paterno y apellido materno). En la tabla 2 podemos observar la comparativa de una tabla sin normalizar a la izquierda y la misma tabla normalizada a la derecha.

Tabla 2.- Primera forma normal

Tabla sin normalizar			Tabla normalizada en 1FN			
Nombre	área	Ext.	nombre	área	Ext.1	Ext. 2
Juan	Informática	15155,15156	Juan	Informática	15155	15156
María	RH	15233,15234	María	RH	15233	15234
Francisco	Derecho	15158	Francisco	Derecho	15156	

4.3.7.2.-Segunda forma normal (2FN)

Según (Kroenke, 2003, pág.129) menciona que se utiliza la 2FN cuando la dependencia de atributos que no son una llave primaria están ligados a una llave primaria; esto es, que si una entidad tiene una llave primaria y los demás atributos son dependientes de ésta, además se toma en cuenta que una entidad con una llave primaria compuesta no puede estar en segunda forma normal.

Según (Covadonga, 1987, pág. 49) teniendo la entidad en primera forma normal, todos sus atributos que no sea llave primaria tengan una dependencia directa de la llave primaria.

En la segunda forma normal la suma los atributos (tupla) son identificados por la llave primaria (por ejemplo la RFC de una persona es una llave primaria y tanto sus atributos de la tupla como son el nombre, apellidos, fecha de nacimiento y lugar de nacimiento dependen directamente de la llave primaria lo que hace que cada tupla o registro sean diferentes).

En la tabla 3 podemos observar que el nombre del área depende de la clave del área pero no así los datos del usuario que dependen de la clave del empleado.

Tabla 3.- Segunda forma normal

Tabla sin normalizar					Tablas normalizadas en 2FN				
Id_emp	nombre	edad	área	Nom_área	Id_emp	nombre	edad	área	Nom_área
1	Juan	20	A1	Informática	1	Juan	20	A1	Informática
2	María	25	A2	RH	2	María	25	A2	RH
3	Pedro	35	A2	RH	3	Pedro	35		
								Id_emp	área
								1	A1
								2	A2
								3	A2

4.3.7.3.-Tercera forma normal (3FN)

Según (Silberschatz, 2002, pág. 177) en la tercera forma normal en las entidades debe existir una relación común (Relación A y relación B), una de ellas sea una superclave de otra distinta (A superclave de C), los atributos de B-A debe estar contenido en alguna clave candidata de C.

Según (Kroenke, 2003, pág.129) con la existencia de una dependencia funcional entre entidades como puede ser A con B y B con C y se puede determinar atributos de C mediante A, se le conoce como dependencia transitiva.

Una vez que se tiene una relación entre dos entidades y esta está relacionada con otra (estando en la segunda forma normal) y a su vez una está relacionada con una llave candidato de una tercera entidad.

En la tabla 4 podemos observar una tabla sin normalizar y otra en 3FN.

Id_emp → (nombre, edad, dir_area , nom_area)
 dir_area → (nom_area)

Tabla 4.- Tercera forma normal

Tabla sin normalizar					Tablas normalizadas en 3FN			
Id_emp	nombre	edad	dir_area	Nom_area	Id_emp	nombre	Edad	dir_area
1	Juan	20	A1	Informática	1	Juan	20	A1
2	María	25	A2	RH	2	María	25	A2
3	Pedro	35	A3	Derecho	3	Pedro	35	A3
					dir_area	Nom_area		
					A1	Informática		
					A2	RH		
					A3	Derecho		

4.3.8.-Objetos de la base de datos

Según (Taboada, 2005, pág.33) los objetos de la base de datos son elementos que componen la base de datos que pueden ser manipulados con las sentencias “SELECT” (selección), “DROP” (eliminación) y “ALTER” (modificación), algunos de estos elementos son:

- Base de datos.
- Tablas.
- Vistas.
- Índices.
- Procedimientos.
- Disparadores.

- Reglas.

Según (Gabillaud, 2003, pág.22) Se pueden agrupar los objetos en tres grandes categorías:

- Gestión y almacenamiento de datos: tablas, tipo de datos, restricciones de integridad, valor predeterminado, reglas e índices.
- Acceso a los datos: vistas y procedimientos almacenados.
- Gestión de integridad compleja: Trigger.

4.3.9.-DBMS (database management system), RDBMS (relational database management system)

Según (Cobo, 2007, pág. 7) el sistema de gestión de base de datos (SGBD, o DBMS por sus siglas en inglés) es el software o concentrado de software con los que se tiene la capacidad de crear y mantener una base de datos, funciona como interfaz entre el usuario y el sistema operativo, facilita un medio eficaz cuando se requiere guardar o consultar información de la base de datos.

Según (Date, 2001, pág.5) software de computadora que permite almacenar, actualizar y recuperar información; todas las solicitudes de acceso hechas son administradas por el manejador de base de datos.

Según (León, 2008, pág. 13) es un grupo de elementos y conjunto de programas que permiten a los usuarios tener acceso a la información ya sea para consultarlos o actualizarlos; suministra un enfoque abstracto de la información, el sistema esconde ciertos detalles referentes a la forma como los datos se almacenan.

Según (Nevado, 2010, pág. 25) es el software que ayuda a los usuarios a procesar, describir, gestionar y acceder los datos almacenados en una base de datos; facilitan un grupo ordenado de programas, procedimientos y lenguajes que permiten a los diferentes usuarios cumplir con sus tareas comunes con los datos. Garantizando a los demás la seguridad de los mismos; el éxito reside en mantener la seguridad e integridad de los datos, lógicamente tiene que proporcionar herramientas a los distintos usuarios.

Según (Abajo, 1997, pág. 86) en un RDBMS, es capaz de llevar a cabo un trabajo con varias tablas a la vez y mostrar información de dos tablas que compartan un campo en común, acepta entradas como es la captura y salidas de información, como cualquier sistema de gestión de bases de datos, procura cumplir con tareas como:

- Insertar Información.
- Editar los registros.
- Borrar información.

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

- Buscar y mostrar datos.
- Organizar y mostrar información de la BD en diferentes consultas.
- Crear, implementar o imprimir reportes que listen, agrupen o extraigan la información.

Según (Gabillaud, 2003, pág. 13) los RDBMS administran de forma independiente el nivel lógico (objetos o entidades) y el nivel físico (archivos), proporcionando así una gran sencillez a la administración de los datos cuando no se cuenta con un alto nivel de conocimientos en informática.

Algo muy importante para usar un RDBMS es planear el diseño de la base de datos, para poder usar las herramientas y que facilite la obtención de información.

Las inserciones a la base de datos pueden ser por ejemplo, insertar los nombres de los profesores, los salones, horarios y alumnos, así como la forma de relacionarlos, por ejemplo se puede tener un campo en común que los relaciona; la información de consulta podría ser, identificar cuántos alumnos están inscritos en cada grupo en cada horario, cuántos profesores dan la misma clase, cuántas clases se dan de una misma materia.

A continuación se describirá de manera sucinta el lenguaje de bases de datos (SQL) y más adelante se compararán los manejadores de bases de datos.

4.3.10. - SQL (structured query language).

Según (Martín, 2009, pág. 5) es el lenguaje estándar ANSI/ISO para definir, manipular y controlar las bases de datos relacionales. Es un lenguaje declarativo en el cual se señala que se desea hacer, por otro lado, los lenguajes procedimentales necesitan que se les especifique cómo se debe realizar cualquier sentencia en la base de datos.

Según (Groussard, 2009, pág. 358) el lenguaje brinda la posibilidad de conversar con la base de datos, se cuenta con varias sentencias del lenguaje SQL, dependiendo de la base de datos manejada, sin embargo SQL cuenta con una sintaxis estándar, normalizada, independiente que puede ser utilizada en cualquier base de datos.

Según (Taboada, 2005, pág. 33) uno de los lenguajes que más se utilizan en bases de datos relacionales es el SQL. Se le considera un lenguaje de alto nivel estándar de para poder interactuar con bases de datos. Se trata de una combinación de lenguaje de definición de datos y lenguaje de manipulación de datos, así como sentencias que permiten tener control sobre el acceso a los datos de la base y de las operaciones sobre ella y que constituyen el lenguaje de control de datos.

Según (León, 2008, pág. 91) el lenguaje de consulta estructurado es un estándar informático utilizado para definir, modificar, gestionar datos y controlar como se realizan

cambios en las bases de datos utilizando tablas, índices, claves, registros y campos para almacenar la información. Las instrucciones del SQL están divididas en tres categorías principales: DDL, DML y DCL, de acuerdo con su funcionalidad.

Es el lenguaje que tiene mayor uso a la hora de hacer peticiones a la base de datos, ya que, aun que cada manejador puede tener sentencias diferentes de funciones, el SQL es un lenguaje estándar para las consultas básicas, nos permite crear, manipular y controlar objetos, información e incluso la misma base de datos.

4.3.10.1-Lenguaje de definición de datos (DDL)

Según (Nevado, 2010, pág. 32) un DDL le da la posibilidad al diseñador de la base de datos el definir los elementos que la componen, su estructura y las relaciones entre ellos, los criterios de integridad y restricciones, etc. Al igual que las características físicas y lógicas de los usuarios.

Según (Moliner, 2005, pág. 183) es aquel define los datos, nos permite crear, modificar o borrar objetos y así como las relaciones entre ellos en la base de datos.

Según (León, 2008, pág. 92) es la parte del SQL que se usa para describir datos y objetos de una base de datos, al crearlos, modificarlos o eliminarlos.

El lenguaje de definición de datos como su nombre lo indica se definen los elementos u objetos y estructura de la base de datos.

4.3.10.2.-Lenguaje de manipulación de datos (DML)

Según (Nevado, 2010, pág. 33) una vez descritos los datos es preciso almacenarlos en las estructuras creadas con anterioridad, para poder utilizarlos una vez que se hayan capturado. Los usuarios podrán consultar la información o modificarla. La consulta podrá ser de los tipos:

- Selectiva.- devuelve los registros que cumplen con alguna condición, de un determinado criterio de selección.
- Totalidad de los datos.- devuelve todos los datos de la base de datos o aquellos que pertenecen a un determinado tipo.

La modificación puede realizarse por medio de tres operaciones diferentes:

- Borrar.
- Modificar.
- Insertar.

Según (Moliner, 2005, pág. 184) la tarea de este lenguaje es la de realizar consultas sobre

la base de datos y para insertar, modificar o eliminar la información almacenada.

Según (León, 2008, pág. 91) es la parte del SQL que se usa para la recuperación o manipulación de los datos. Son las instrucciones responsable de las consultas y las modificación de los contenidos de las tablas, estas instrucciones son: Select (recuperar información), Insert (agrega un nuevo registro), Update (modifica la información contenida), Delete (eliminar registros), commit (guarda las actualizaciones o cambios que se hayan introducido), rollback (elimina los cambios que se hayan producido después del último commit).

4.3.10.3.-Lenguaje de control de datos (DCL)

Según (Nevado, 2010, pág. 33) es el lenguaje para facilitar la tarea del administrador a la hora de dar seguridad y mantenimiento a la base de datos. Otorga funciones de servicio como:

- Modificar el tamaño de los ficheros
- Estadísticas de uso y tareas de seguridad como:
 - Respaldos de seguridad.
 - Reinicio del sistema.
 - Defensa contra accesos no autorizados.

Según (Moliner, 2005, pág. 184) la función de este lenguaje es mantener la seguridad de la base de datos.

Según (León, 2008, pág. 110) se utiliza para asignar permisos a los objetos de las bases de datos, tanto a objetos como a funciones algunos como son los permisos de lectura y escritura de información o de modificación a la estructura.

4.3.11.-PL/SQL y TransactSQL

Según (Silberschatz, 2002, pág. 220) SQL: 1999 permite la definición de funciones, procedimientos y métodos mediante el componente procedimental. Algunos sistemas de bases de datos soportan sus propios lenguajes procedimentales, tales como PL/SQL en Oracle y TransactSQL en SQL Server de Microsoft.

Oracle tiene dos lenguajes procedimentales principales PL/SQL y java. PL/SQL fue el lenguaje original de Oracle para procedimientos almacenados.

Según (msdn.microsoft.com, 2014, s/p) Transact-SQL es fundamental para trabajar con SQL Server. Todas las aplicaciones que se comunican con SQL Server lo hacen enviando instrucciones Transact-SQL al servidor, independientemente de la interfaz de usuario de la aplicación.

A continuación se proporciona una lista de las aplicaciones que pueden generar Transact-SQL:

- Aplicaciones generales de productividad en oficinas.
- Aplicaciones que utilizan una interfaz gráfica de usuario (GUI) para permitir al usuario seleccionar las tablas y columnas cuyos datos desea ver.
- Aplicaciones que utilizan instrucciones del lenguaje general para determinar los datos que el usuario desea ver.
- Aplicaciones de la línea de negocios que almacenan sus datos en bases de datos SQL Server. Estas aplicaciones pueden incluir aplicaciones de otros proveedores o escritas internamente.
- Scripts Transact-SQL que se ejecutan mediante utilidades como **sqlcmd**.
- Aplicaciones creadas con sistemas de desarrollo tales como Microsoft Visual C++, Microsoft Visual Basic o Microsoft Visual J++, y que usan API de base de datos tales como ADO, OLE DB y ODBC.
- Páginas web que extraen datos de bases de datos SQL Server.
- Sistemas de bases de datos distribuidos desde los que se replican datos SQL Server en varias bases de datos o se ejecutan consultas distribuidas.
- Almacenamientos de datos en los que los datos se extraen de los sistemas de procesamiento de transacciones en línea (OLTP) y se resumen para el análisis dirigido a la toma de decisiones.

4.3.11.1.-Vistas

Según (Gabillaud, 2003, pág.) son tablas virtuales “que contienen” el resultado de una instrucción **select**.

Una de las ventajas principales que se notan al utilizar vistas, es que no almacena datos, sino que hace referencia a una o varias tablas de origen mediante una consulta **select**, consulta que se ejecuta cada vez que se hace referencia a la vista, de este modo cualquier modificación que se realice sobre los datos de origen es inmediatamente visible en la vista, cuando esta vuelve a ejecutarse.

Según (Kroenke, 2003, pág. 654) es una relación que se construye a partir de una sola instrucción SQL **select**. Las vistas SQL tienen cuando mucho una trayectoria.

Las vistas son utilizadas más en casos como:

- Ocultar a los usuarios determinadas columnas o filas proporcionando así un nivel de seguridad adicional.
- Simplificar el uso de tablas que contengan muchas columnas, muchas filas o nombres muy complejos.

4.3.11.2.-Disparadores (Trigger)

Según (Gabillaud, 2003, pág.22) procedimiento almacenado que se ejecuta automáticamente cuando se ejecuta una orden SQL que modifica el contenido de una tabla: **insert**, **update** y **delete**. El desencadenador se asocia siempre a una tabla y a una instrucción SQL. Permite implementar reglas de integridad entre varias tablas o mantener datos no normalizados.

Según (Silberschatz, 2002, pág. 613) se tienen dos tipos de disparadores por fila (row) y por instrucciones (statement). Por fila se pueden ejecutar una vez por fila que se vea afectada con instrucciones como **insert**, **update** y **delete** por instrucciones (statement) se ejecutará una sola vez por instrucción.

Según (msdn.microsoft.com, 2014) un desencadenador es una clase especial de procedimiento almacenado que se ejecuta automáticamente cuando se produce un evento en el servidor de bases de datos. Los desencadenadores DML se ejecutan cuando un usuario intenta modificar datos mediante un evento de DML.

Los eventos DML son instrucciones **insert**, **update** o **delete** de una tabla o vista. Estos desencadenadores se activan cuando se desencadena cualquier evento válido, con independencia de que las filas de la tabla se vean o no afectadas.

Los desencadenadores DDL se ejecutan como respuesta a diversos eventos, corresponden principalmente a instrucciones **create**, **alter** y **drop** de Transact-SQL, y a determinados procedimientos almacenados del sistema que ejecutan operaciones de tipo DDL.

Los desencadenadores logon se activan en respuesta al evento **logon** que se genera cuando se establece la sesión de un usuario. Los desencadenadores pueden crearse directamente a partir de instrucciones de Transact-SQL o de métodos de ensamblados creados en Common Language Runtime (CLR) de Microsoft .NET Framework y cargados en una instancia de SQL Server. SQL Server permite crear varios desencadenadores para cualquier instrucción específica.

En cada caso un disparador se puede definir como un disparador “before” o “after” dependiendo si se va a invocar antes o después de llevar a cabo la operación.

4.3.11.3.-Procedimientos

Según (Osorio, 2008, Pág. 135) un procedimiento almacenado (Stored Procedure en inglés) es un programa que se almacena adjunto a una base de datos. Generalmente se escribe en un lenguaje de base de datos como T-SQL para SQL Server, PL/SQL para Oracle database o PL/PgSQL para Postgres SQL.

Los procedimientos almacenados se utilizan para varias aplicaciones generalmente para encapsular un proceso grande y complejo que se ejecutará más rápido como procedimiento almacenado que siendo implementado en un programa corriendo en un

sistema cliente y comunicándose con la base de datos mediante el envío de consultas SQL y recibiendo sus resultados.

Los procedimientos tienen entre muchas ventajas, simplificar la creación y, particularmente, el mantenimiento de los programas involucrados con una base de datos (aplicaciones cliente) ya que se centralizan las operaciones comunes, evitando redundancias.

Según (aulaclic.es, 2014, pág. 1) un procedimiento almacenado está formado por un conjunto de instrucciones Transact-SQL que definen un determinado proceso, puede aceptar parámetros de entrada y devolver un valor o conjunto de resultados. Este se guarda en el servidor y puede ser ejecutado en cualquier momento.

Los procedimientos almacenados se diferencian de las instrucciones SQL ordinarias y de los lotes de instrucciones SQL en que están precompilados. La primera vez que se ejecuta un procedimiento, el procesador de consultas de SQL Server lo analiza y prepara un plan de ejecución que se almacena en una tabla del sistema. Posteriormente, el procedimiento se ejecuta según el plan almacenado.

Puesto que ya se ha realizado la mayor parte del trabajo de procesamiento de consultas, los procedimientos almacenados se ejecutan casi de forma instantánea por lo que su uso mejora notablemente la potencia y eficacia del SQL.

Según (Gabillaud, 2009, Pág. 225) Los procedimientos almacenados son objetos correspondientes a un conjunto de instrucciones del DML que pueden ejecutarse por medio de una llamada a su nombre o por instrucción **execute**. Son auténticos programas que pueden recibir parámetros, devolver valores, se guardan en el cache de memoria en forma compilada tras su primera ejecución, lo que aumenta el rendimiento, pueden ser temporales, es decir, creados para una (local) o más sesiones (global) del user.

Un procedimiento almacenado puede definirse como una serie de instrucciones transact SQL, almacenadas en la base de datos y perfectamente identificadas por su nombre. Para permitir que esta serie de instrucciones se adapte al mayor número de casos, algunos valores de las secuencias de comandos son parametrizables en el momento de la llamada al procedimiento.

Los distintos usos típicos de procedimientos almacenados son los siguientes:

- Encadenamiento de instrucciones.
- Mejora del rendimiento.
- Seguridad de ejecución.
- Manipulación de los datos del sistema.
- Implementación de las reglas de negocio.
- Tratamientos en cascada.

4.3.11.4.-Cursores

Según (sqltransact2.blogspot.mx, 2014, pág. 1) un cursor es una variable que nos permite recorrer con un conjunto de resultados obtenido a través de una sentencia **select** fila a fila.

Cuando trabajemos con cursores debemos seguir los siguientes pasos.

- Declarar el cursor, utilizando **declare**.
- Abrir el cursor, utilizando **open**.
- Leer los datos del cursor, utilizando **fetch... into**.
- Cerrar el cursor, utilizando **close**.
- Liberar el cursor, utilizando **deallocate**.

Según (Escofet, 2009, pág. 1) un cursor es una estructura de datos creada en memoria RAM producto de una sentencia **select** y que permite navegar dentro de las filas para obtener la información.

Según (Gabillaud, 2001, Pág. 2004) se utilizan los cursores cuando es necesario tratar las filas individualmente en un conjunto o cuando SQL no pueda actuar únicamente sobre las filas afectadas.

Suelen consumir mucha memoria y tiempo de tratamiento.

4.3.11.5.-Comparación entre SQL Server, Postgresql, Oracle y MySQL

A continuación en la tabla 5 podemos observar una comparativa entre los manejadores de bases de datos SQL Server, PostgreSQL y Oracle.

Tabla 5.- comparativa entre SQL Server, PostgreSQL y Oracle

Nombre	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
Descripción	Microsofts DBMS relacional	RDBMS código abierto ampliamente utilizado	RDBMS ampliamente utilizado	DBMS Postgres basado en objetos relacionales
Sitio Web	www.microsoft.com/sqlserver	www.mysql.com	www.oracle.com/us/products/-database	www.postgresql.org
Documentación Técnica	www.microsoft.com/sqlserver/en-us/default.aspx	dev.mysql.com/doc	www.oracle.com/technetwork/-indexes/-documentation/	www.postgresql.org/docs/-manuals

			index.html	
Desarrollador	Microsoft	Oracle	Oracle	PostgreSQL Global Development Group
Primera versión	1989	1995	1980	1989
Licencia	Comercial	Código abierto	Comercial	Código abierto
Lenguaje en el que ha sido desarrollado	C++	C and C++	C and C++	C
Sistemas operativos sobre los cuales funciona	Windows	Linux OS Solaris Windows	Linux OS Solaris Windows	Linux OS Solaris Windows
Modelo de Base de Datos	Base de datos relacional DBMS	Base de datos relacional DBMS	Base de datos relacional DBMS	Base de datos relacional DBMS
Esquema de datos	Sí	Sí	Sí	Sí
Índices secundarios	Sí	Sí	Sí	Sí
SQL	Sí	Sí	Sí	Sí
APIs y otros métodos de acceso	OLE DB Tabular Data Stream (TDS)	ADO.NET JDBC	ODP.NET Oracle Call Interface (OCI)	native C library streaming API for large objects
Lenguajes de Programación Soportados	ADO.NET JDBC ODBC .Net Java PHP Python Ruby Visual Basic	ODBC Ada C C# C++ D Eiffel Erlang Haskell Java Objective-C OCaml Perl PHP Python Ruby Scheme Tcl	JDBC ODBC C C# C++ Clojure Cobol Eiffel Erlang Fortran Groovy Haskell Java JavaScript Lisp Objective-C OCaml Perl PHP Python R Ruby	ADO.NET JDBC ODBC .Net C C++ Java Perl Python Tcl

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

			Scala Tcl Visual Basic	
Scripts	Lenguajes Transact-SQL y .NET	Sí	PL/SQL	funciones definidas por el usuario
Triggers	Sí	Sí	sí	sí
Métodos de particionamiento	Tablas pueden ser distribuidos a través de varios archivos (de particionamiento horizontal), pero no sharding (El Sharding es una técnica que consiste en particionar los datos de tu base de datos horizontalmente agrupándolos de algún modo que tenga sentido y que permita un direccionamiento más rápido.)	partición horizontal en MySQL Cluster	partición horizontal	no, pero se puede realizar utilizando la herencia de tablas
Llaves foráneas	Sí	Sí	Sí	Sí
Conceptos de transacción	ACID (Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad)	ACID (Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad)	ACID (Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad)	ACID (Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad)
Concurrencia	Sí	Sí	Sí	Sí

4.4 Redes de datos y seguridad

Según (Barceló, 2004, pág. 29) el envío de datos no necesariamente debe llevarse a cabo en tiempo real, por tanto, no es preciso establecer el camino entre los dos puntos antes de empezar la transmisión y mantenerlo mientras dura el intercambio de datos. En lugar de ello, se empaquetan los bits que deben transmitirse y se dan a la central más próxima para que los envíe cuando pueda a la siguiente, y así sucesivamente hasta que lleguen al

destino. Si cuando un paquete llega a una central todos los enlaces con la siguiente están ocupados, no pasa nada, lo hace esperar poniéndolo en una cola para enviarlo cuando haya un enlace disponible.

Según (Valdivia, 2014, pág. 112) Las redes de datos surgen ante la necesidad de comunicar y compartir información entre usuarios. Una red de datos informática mantiene los beneficios de un equipo de cómputo personal pero añadiendo la posibilidad de compartir los recursos de software y hardware con otros usuarios lo que aporta unas enormes ventajas:

- Recursos compartidos: Los equipos conectados a una red pueden compartir determinados componentes.
- Dispositivos de protección de datos: se pueden programar la realización de copias de seguridad.
- Posibilidad de acceder a un soporte de datos compartidos: los usuarios pueden disponer de una base de datos común.
- Procesamiento distribuido: se pueden repartir una tarea entre varios equipos, obteniendo una mayor potencia.
- Alternativas de comunicación: los usuarios pueden comunicarse con otros a través de aplicaciones informáticas específicas.
- Capacidad de ampliación: Las redes de datos permiten rápida expansión y es muy fácil añadir elementos nuevos.

Según (microalcarria.com, 2014, pág. 8) la seguridad tiene su nacimiento con la aparición de los ataques a la información por parte de intrusos interesados en el contenido de ésta. El objetivo de la seguridad de la información es:

- Mantener el secreto, evitando los accesos no autorizados.
- Mantener la autenticidad, evitando modificaciones no autorizadas.

Dentro del concepto de seguridad debemos distinguir la seguridad física de la seguridad lógica, y para tener un concepto más claro, detallaremos a continuación cada una de ellas.

La seguridad física comprende el aspecto del hardware, la manipulación del mismo, así como también el ambiente en el cual se van a instalar los equipos. Para garantizar la seguridad de los mismos se pueden considerar los siguientes criterios:

- Uso del equipo por personal autorizado.
- Sólo podrá tener acceso al equipo aquella personal que cuente con conocimientos mínimos sobre computación.
- Tener más de un servidor de base de datos, lo cual asegurará la integridad total de la información.
- Ubicación de las instalaciones, la cual debe cumplir las normas internacionales de

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

calidad (ISO 9000).

- Control de alarma la cual notifique en todo momento sobre la integridad física del sistema.

La seguridad lógica comprende el aspecto de los sistemas, tanto operativos como de información. Dentro de las medidas a tomar para garantizar la seguridad de los mismos se recomienda las siguientes:

- Construcción de contraseñas en diversos niveles del sistema, donde permita solo el acceso con base a niveles de seguridad de usuarios con permiso.
- Con base al sistema operativo que use como plataforma, utilizar algoritmos que generen claves para poder encriptar los archivos de contraseñas dentro del sistema, me permita mayor seguridad en un entorno de red.
- Generar un módulo del sistema para la emisión de reportes para el administrador del sistema, en donde se muestre tablas de uso del sistema, usuarios y los niveles de acceso por parte de los tales para poder determinar el uso y acceso al sistema.
- Es necesario contar con el diseño de módulos que ejecuten un Control de alarma la cual notifique en todo momento sobre la integridad de la información del sistema.

4.4.1.- Redes y subredes.

Según (profesores.fi-b.unam.mx, 2014, Pág. 8) las redes de computadoras son un conjunto de dispositivos conectados entre sí mediante uno o más medios de transmisión a fin de llevar a cabo la transferencia eficiente y confiable de información. Uno de los principales beneficios de las redes locales es precisamente compartir recursos y dispositivos.

Cuando se comparten recursos se hace que los programas, el equipo y en particular los datos estén disponibles para todos los que se conecten a la red local.

En lugar de tener todos los hosts conectados en cualquier parte a una vasta red global, es más práctico y manejable agrupar los hosts en redes específicas. Históricamente, las redes basadas en IP tienen su raíz como una red grande. Como esta red creció, también lo hicieron los temas relacionados con su crecimiento. Para solucionar estos problemas, la red grande fue separada en redes más pequeñas que fueron interconectadas. Estas últimas generalmente se llaman subredes.

Red y subred son términos utilizados indistintamente para referirse a cualquier sistema de red hecho posible por los protocolos de comunicación comunes compartidos del modelo TCP/IP.

Según (dynamics.unam.edu, 2014, pág.1) una red tiene como objetivo principal, compartir recursos físicos (equipos y sus periféricos) y recursos lógicos (archivos de datos y programas), actualizándolos, organizándolos y explotándolos.

Redes LAN (Local Area Network – Red de Área Local.) van de los 10 m a menos de 10 Km de distancia, por lo general están ubicados los nodos de la red en la misma habitación o en el mismo piso de un edificio.

Redes MAN (Metropolitan Area Network – Red de Área Metropolitana) van de los 10 Km a menos de 1000 Km esto puede representar desde una ciudad hasta un país entero y está formada por la conexión de varias redes LAN.

Redes WAN (Wide Area Network – Red de Área Extendida) abarcan de los 1000 km hasta los 10000 km lo cual puede representarse con un continente o el planeta entero está formado por varias redes MAN y el ejemplo más común y más conocido es Internet.

Las subredes se aplican para las redes de tipo MAN ya que en una red LAN sólo hay un host.

Existen dos tipos de diseño para las subredes de comunicación:

- Canales punto a punto
- Canales de difusión

Sistemas o canales punto a punto

En este caso los Host contiene varios cables o líneas telefónicas conectados entre sí, en las que se puede compartir la información a través de estos cables, en el caso que 2 host se quieran comunicar y no tengan información en común utilizan de manera indirecta otros que se denominan intermedios.

En estos casos la información se almacena en los host intermedios hasta que el medio de comunicación este libre para enviar la información.

Para este tipo de diseño es necesario tomar en cuenta la topología de la red que puede ser: Estrella, Anillo, Árbol. Por lo general se utiliza en redes WAN.

Sistemas o canales de difusión

Este tipo de sistemas se utiliza en las redes LAN y en algunas redes WAN, contienen un sólo canal de comunicación que conecta a toda la red, la información que se envía a una máquina le llega a todas las demás, pero sólo es leída por la computadora destino, este tipo de sistema permite enviar un solo mensaje a todas las computadoras de la red, sin necesidad de repetir el mensaje. Las topologías adecuadas para este sistemas son: Bus, Satélite y Anillo.

Según (uca.edu.sv, 2014, pág. 1) el protocolo de IP usa direcciones de IP para identificar los host y encaminar los datos hacia ellos.

Todos los host o nodos deben tener una dirección IP única para poder ser identificados en la red. El nombre de host se traduce a su dirección de IP consultando el nombre en una

base de datos de pares nombre – dirección.

Una dirección de IP tiene un formato de dos partes que son la dirección de red y la dirección local.

La dirección de red identifica la red a la que está conectado el nodo. La dirección local identifica a un nodo particular dentro de la red de una organización. Toda organización que planea una red LAN basada en protocolo IP o conectarse a la Internet debe conseguir un bloque de direcciones de IP únicas.

Durante muchos años, sólo había tres tamaños de bloques de direcciones, grande, mediano y pequeño. Existían tres formatos diferentes de direcciones de red para cada uno de los tamaños de bloques. Los formatos de direcciones eran:

- Clase A para redes muy grandes.
- Clase B para redes de tamaño medio.
- Clase C para redes pequeñas.

4.4.2-Confidencialidad

Según (Atelin, 2006, Pág. 43) la confidencialidad es a menudo necesaria durante la transmisión de datos en distintos y variados soportes, para convertir la información en confidencial se cifra por codificación, se somete al cifrado, de esta manera sólo el emisor y el receptor pueden leerlos

Según (Tanenbaum, 2003, Pág. 722) la confidencialidad consiste en mantener la información fuera de las manos de usuarios no autorizados, se logra usando correo certificado, poniendo documentos bajo llave o cifrando la información que se envía.

Según (Purser, 1990, Pág. 259) son los servicios que aseguran que las conversaciones de los usuarios no puedan interpretarse por terceros, esto se implanta normalmente cifrando o transformando la información inteligible de las conversaciones en forma oculta, normalmente por medio de algoritmos controlados por una o más claves, el proceso inverso a este se conoce como descifrado.

Según (Informática aplicada a la gestión de empresas, 2000, pág.135) conseguir la privacidad de los datos impidiendo que terceros puedan entender la información que se envía, se consigue mediante sistemas de cifrado o encriptación de la información.

4.4.3-Disponibilidad

Según (Dordoigne, 2001, pág. 64) la disponibilidad permite garantizar el servicio en cualquier circunstancia y para ello se requiere la implementación de soluciones que hagan más fiables los servicios así como los medios de almacenamiento.

Según (Quesnel, 2012, pág. 147) es la aptitud de un servicio de Tecnología de la

Información (TI) o un componente para cumplir la función esperada en un momento dado o durante un periodo de tiempo definido, la disponibilidad también se refleja como los usuarios acceden a los sistemas de información y lo que contienen.

Según (Colobran; 2008; pág. 78) es la capacidad de mantener operativas las aplicaciones de las organizaciones, eliminando las paradas de los sistemas de información. Los sistemas informáticos se tienen que haber configurado para reducir al mínimo porcentaje el tiempo de inactividad o de falta de disponibilidad para conseguir la máxima cota de utilidad.

4.4.4.- Protocolo TCP/IP.

Según (Herrera, 2003, Pág.259) es un grupo de protocolos diseñado para la comunicación entre computadoras suministrando, a su vez, servicio de red como: registro de entrada remoto, transferencia remota de archivos, correo electrónico, etc. Un protocolo de comunicación debe manejar los errores en la transmisión, administrar el enrutamiento y entregar los datos, así como controlar la transmisión real mediante el uso de señales de estado predeterminadas.

Es uno de los protocolos de comunicación más antiguos en los estándares de redes internas. Se basa en el concepto clientes/servidores: cualquier dispositivo que inicia una comunicación se llama cliente y el dispositivo que responde, servidor.

Está compuesto por dos de las partes del software de internet particularmente importantes e innovadoras. El software de protocolo de internet (IP) proporciona la comunicación básica.

Según (Atelin, 2007, Pág. 17) El transmission Control Protocol (TCP) es un protocolo de enrutamiento que garantiza un servicio fiable, orientado a la conexión para un grupo importante de octetos.

El Internet Protocol (IP) proporciona un sistema de entrega de paquetes, sin conexión y no fiable. Administra las direcciones lógicas, que dividen el identificador del nodo en un número de red lógico y un número de periférico sobre 4 octetos (en IP versión 4). Ver la tabla 6.

Tabla 6.- Capas del modelo TCP/IP

Aplicación	Telnet	ftp	dns	Snmp
	Windows sockets			
Transporte	TCP		UDP	
Internet	ARP	IP		ICMP
Interfaz de red	Ethernet		Token Ring	Otros

Según (Mathon, 2001, Pág. 18) el usuario percibe TCP/IP como un conjunto de programas que ofrecen servicios utilizando la red con diferentes finalidades. Los servicios principales de aplicación pueden ser del tipo de transferencia de archivos, mensajería, web, conexiones remotas u otros.

IP es lo suficientemente universal para funcionar sobre diferentes tecnologías de interfaces de red, ya se trate de soportes punto a punto o de soportes multipunto, en red

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

local o para redes de área extensa.

Un servicio de transporte fiable es necesario en las aplicaciones de red para una comunicación sin errores, ello se consigue con reenvíos automáticos en caso de que se produzca un error. Permite a una aplicación establecer una conexión de un punto a otro con una aplicación que funciona en otra máquina, de la misma forma que si se tratase de una conexión directa y permanente, el servicio de TCP (Transmission Control Protocol) es el encargado de este servicio de transporte fiable, algunas ventajas son:

- **Independencia de la topología de red:** Es independiente de cualquier hardware o fabricante. Puede funcionar sobre una gran variedad de topologías utilizando una unidad de transmisión llamada datagrama que especifica la forma de transmitir información en una topología de red determinada.
- **Conectividad universal y heterogénea:** Cada máquina conectada vía TCP/IP recibe una dirección única y cualquier par de máquinas en una red podrá establecer una conexión. Los nodos intermedios utilizan las direcciones contenidas en los datagramas para tomar las decisiones de direccionamiento de paquetes.
- **Acuse de recibo entre puntos:** TCP/IP asegura un acuse de recibo directo entre las máquinas de origen y destino. Esto funciona exactamente igual aun suponiendo que las máquinas no se encuentran en la misma red física.

Según (Miñana, 2010, Pág. 36) TCP/IP es el sucesor de un proyecto de investigación realizado por la agencia de proyectos de investigación avanzados (ARPA), cuyo objetivo principal fue la interconexión de redes distintas, las cuales debían mantener las comunicaciones establecidas entre dos equipos. En la tabla 7 se realiza una tabla de Equivalencia OSI-TCP/IP.

Tabla 7.- Equivalencia OSI TCP/IP

ARQUITECTURA OSI	MODELO DE CAPAS TCP/IP
Capa de aplicación	Capa de aplicación
Capa de presentación	
Capa de sesión	
Capa de transporte	Capa de transporte
Capa de red	Capa de interred
Capa de enlace	Capa host a red
Capa física	

- **Capa de aplicación:** Se encuentra la comprensión de información, gestión de sesiones y seguridad, define las funciones que van a tener los programas.
- **Capa de transporte:** Función que determina el tipo de conexión ya sea orientado a conexión o no orientado a conexión. También se encarga de la división de paquetes y la asignación de una cabecera con el fin de poder reensamblarlos en el destino.
- **Capa de interred:** Una de sus funciones es el direccionamiento, esto es, establecer direcciones IP para localizar al destinatario de la información e identificar el origen.

- Capa Red o Subred: Se utiliza algún protocolo que conecte el equipo a la red para el envío de información por ella.

4.4.5.-HTTP

Según (Dewit, 2010, Pág. 10) HyperText Transfer Protocol (HTTP) previsto inicialmente para la transferencia de archivos HTML, su uso ha evolucionado hacia la activación en el servidor de procesos que generan automáticamente contenido HTML.

El desarrollo de una aplicación internet debe tener presente dos características importantes del protocolo HTTP:

- HTTP es un protocolo desconectado.

Una petición HTTP es emitida, procesada en un servidor, y es seguida de la emisión de una respuesta. El servidor web no conserva con posterioridad ninguna información sobre la respuesta procesada. Si el mismo cliente hace una nueva petición, nada le indica al servidor que se trata del mismo cliente de la respuesta anterior.

- Es un protocolo texto.

Las peticiones y las respuestas HTTP son codificadas en ASCII 7 bits. Los caracteres acentuados, o multi-octeto (UNICODE) y los datos binarios deben ser codificados en la emisión y decodificados en la recepción. El encabezado de la petición permite, pues, especificar la norma de codificación utilizada, así como la que se espera en respuesta.

HTTP no es especialmente potente para la transferencia de datos. Sin embargo sus puntos fuertes son su robustez y su universalidad, que al asociarlo con XML (eXtensible Markup Language), ofrece una solución eficaz para la interoperabilidad entre sistemas, base sobre la que reposa la noción de servicio web.

Según (Groussard, 2010, pág. 27) su funcionamiento se basa “pregunta/respuesta”. El cliente genera una pregunta con la forma de una petición HTTP. Esta petición contiene por lo menos los datos que permiten identificar el recurso solicitado por el cliente. La petición HTTP es un bloque de texto que transita del cliente al servidor. Este bloque de texto debe tener un formato concreto para que sea reconocido por el servidor se compone de dos partes, la cabecera de la petición HTTP que es obligatoria y el cuerpo de la petición que es opcional.

Una respuesta HTTP solamente puede existir si se ha enviado previamente una petición HTTP al servidor. El servidor nunca toma la iniciativa de enviar datos a un servidor si éste no lo ha solicitado.

Según (Quero, 2007, pág. 103) se trata de un protocolo de intercambio de información entre los clientes web y los servidores HTTP, es decir, es un protocolo de comunicación

cliente-servidor.

Son operaciones de solicitud y respuesta. El cliente envía un mensaje de datos al servidor con una solicitud y servidor responde con otro mensaje que incluye el posible resultado. Sus principales características son:

- La comunicación se realiza a partir de caracteres de 8 bits.
- Permite la transferencia de objetos multimedia.
- El cliente puede utilizar comandos para dialogar con el servidor.
- Cada operación HTTP implica una conexión nueva con el servidor y en cada una de ellas no se tiene en cuenta las anteriores.

4.4.5.1.-Metodo de petición GET

Según (Groussard, 2010, pág. 27) la petición GET permite obtener un recurso disponible en el servidor. Este tipo de petición no contiene ningún tipo de dato en el cuerpo de la petición. Si tienen que enviarse parámetros al servidor, estos se añaden a continuación de la URL.

Según (Acofi, 2010, pág. 8) envía al servidor una cadena de caracteres que incluye la dirección del cliente en un formato Uniform Resource Locator (URL), seguida por el carácter “?” y por una cadena que contiene todos los datos en la forma (nombre y valores de los elementos de entrada).

Cuando el servidor recibe esta cadena, separa la subcadena que esta luego del carácter “?” y se la asigna a una variable de ambiente llamada QUERY_STRING.

Según (Payne, 2002, pág. 543) es un protocolo estándar que permite que los clientes se comuniquen con los servidores utilizando HTTP, la información enviada como parte de la URL, esto es, cualquier parámetro necesario para procesar la petición se adjunta mediante una cadena de consulta con la estructura nombre/valor.

4.4.5.2.-Metodo de petición POST

Según (Groussard, 2010, pág. 28) la petición POST se destina más bien al envío de datos al servidor recompilados en un formulario HTML. Aunque también se puede realizar una petición GET, el uso de una petición POST presenta varias ventajas:

- No hay ningún límite teórico a la cantidad de información que se envía hacia el navegador porque esta información se inserta en el cuerpo de la petición HTTP. Con el método GET, la información que se envía se añade mediante parámetros en la URL, con el riesgo de sobrepasar el límite de tamaño máximo de ésta.
- La petición HEAD tiene un funcionamiento similar al de la petición GET excepto en que el servidor no devuelve nada en el cuerpo de la respuesta. Los navegadores

utilizan este tipo de peticiones para la gestión del almacenamiento en cache de datos.

- Los datos se transmiten por el cuerpo de la petición HTTP, no se pueden ver en la barra de dirección del navegador, como sucede en el caso de una petición HTTP GET.

Según (Acofi, 2010, Pág. 8) El servidor envía una cadena de datos a través de la entrada estándar del sistema operativo, que es luego leída por la aplicación, es preferible utilizar el método POST con un gran volumen de datos.

Los valores se envían como parte de la consulta HTTP.

Por lo tanto las entradas no aparecen incorporadas a la URL, haciendo más práctico el mantener variables ocultas como contraseñas.

Según (Payne, 2002, Pág. 543) es muy semejante al método GET, pero en lugar de adjuntar los parámetros en la URL, los coloca directamente en el mensaje de petición HTTP, son transferidos a través de una cabecera http independiente y no son visibles para el usuario, este método es más utilizado en formularios, en donde el volumen de información que se envía al servidor es muy grande.

En resumen se puede decir que para los formularios es preferible utilizar el método POST ya que si se introduce una contraseña, esta podría ser fácilmente leída si se utiliza el método GET simplemente recorriendo la cadena enviada.

En el caso del método GET la cantidad de información a transmitir está condicionada por la longitud máxima de los URL que pueden ser procesados por el servidor y el cliente web.

No se conserva la privacidad de los datos transferidos ya que aparecen en la URL y quedan almacenados en el historial del navegador.

A continuación se describirá la metodología aplicada para el desarrollo del sistema.



DESARROLLO

5.- Desarrollo

En este capítulo se describe la metodología utilizada, el diseño, la implementación, pruebas y las mejoras del sistema.

5.1.- Descripción general del sistema

- Diseñar un sistema al que puedan ingresar los usuarios, modificar a su superior y a al personal que tienen a su cargo.
- Definición de roles de usuario y administrador para acceso al sistema.
- Con este sistema se tendrá un control de quién debe llevar a cabo las evaluaciones en la institución, así como quién será el segundo evaluador y la asignación de tercer evaluador.
- Reducción de tiempos en los trámites, así como un control en los de registros evitando el retraso en los mismos.
- El sistema tendrá un historial de actualizaciones y quién las realizó.

5.2.- Metodología

Es de fundamental importancia contar con una metodología para el desarrollo de software.

En el capítulo anterior se abordaron las metodologías ASD y RUP.

La Metodología utilizada fue RUP (Rational Unified Process- Proceso Unificado Racional)

A continuación se aplica la metodología en el desarrollo del sistema.

5.2.1.- Requerimientos

Los requerimientos funcionales de un sistema describen los que debe hacer, sus entradas, salidas, y excepciones.

Partiendo de la información que debe contener el sistema, se determinó la estructura general que debe de tener. Para acceder al sistema éste solicitará un usuario y una contraseña. Existirán los siguientes niveles de usuarios:

- Administrador: Tiene acceso a los módulos de validación, administrador y usuario.
- Usuario: Tiene acceso al módulo de validación y usuario.

Los módulos contarán con los siguientes apartados:

Módulo de logueo

1.- Se ingresa al sistema introduciendo el RFC.

1.1.-El sistema valida la estructura del RFC, si se introduce una cadena con

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

estructura errónea el sistema pedirá que ingrese un RFC válido.

1.2.- Si el RFC no está registrado en el sistema se mandará un mensaje de alerta que indica que es incorrecto o no se encuentra registrado en el sistema.

1.3.- Si el RFC es correcto nos direcciona a introducir la contraseña.

2.- Muestra pantalla con datos de RFC y nombre de usuario registrado en el sistema, así como el campo para introducir su contraseña.

2.1.- Si la contraseña es incorrecta mandará un mensaje de contraseña incorrecta.

2.2.- Si la validación de contraseña es correcta, se verificarán los permisos del usuario, tiene permiso nivel 1 será redireccionado al módulo de administrador, si tiene un nivel 2 será redireccionado al módulo de usuario.

Módulo de Administración

1.- Se desplegará un buscador de empleados en dónde se puede buscar usuarios para modificar su línea de mando, el administrador puede entrar como el usuario mismo a su línea de mando y modificar sus registros.

1.1.- Los filtros del buscador son: nombre, apellido paterno, apellido materno, RFC, área de adscripción, número de nómina y si ya cuenta con superior jerárquico. Se pueden hacer combinaciones de los filtros.

2.- Se mostrará una lista de empleados a los que se puede acceder a su registro presionando un hipervínculo sobre su nombre.

3.- Se mostrarán los datos del empleado en dónde solo puede ser modificada su área de adscripción, los datos personales son cargados directamente de la nómina, los campos de superior jerárquico, RFC de superior jerárquico, tercer evaluador y RFC de tercer evaluador se cargan una vez que el jefe superior lo ha colocado en su lista de subordinados y ha designado a su tercer evaluador.

4.- Para agregar un subordinado se presionará el botón “agregar subordinado” en dónde se pedirá que se introduzca el RFC de la persona que se desea agregar.

El botón de “buscar” mostrará el nombre del empleado y si ya cuenta con un jefe, al agregarlo si tenía un jefe con anterioridad notificará en un mensaje de alerta que el usuario que está por agregar pertenecía a alguien más y se modificará esta línea de mando.

Los datos de movimientos son guardados en la bitácora, tanto el jefe nuevo, jefe anterior y quién realizó el cambio y cuándo.

5.- Una vez que se cuente con la lista de subordinados se podrá dar de baja, delante del nombre del subordinado se encuentra el botón “baja” con el que se eliminará de su línea de mando.

6.- Permitirá que a los subordinados se les pueda agregar un tercer evaluador (indispensable para otros sistemas institucionales) o modificar si fuera el caso, en el botón de “agregar” en la lista de tercer evaluador, pedirá introducir el RFC del tercer evaluador, y se mostrarán sus datos y se agregarán.

7.- El administrador también es capaz de modificar toda una línea de mando al mismo tiempo en el apartado “Modificar Línea de Mando Completa” se introduce el RFC del

nuevo jefe que se desea para toda la línea de mando y se agregarán todos al mismo tiempo.

Módulo de usuario

1.- Se mostrarán los datos del empleado en donde no se puede modificada la información; su área de adscripción solo puede ser modificada por el administrador y los datos personales son cargados directamente de la nómina, el nombre y RFC tanto del superior jerárquico como del tercer evaluador se cargan una vez que el jefe superior lo ha colocado en su lista de subordinados y ha designado a su tercer evaluador.

2.- Para agregar un subordinado se presiona el botón “agregar subordinado” en dónde se pedirá que se introduzca el RFC de la persona que se desea agregar; el botón de “buscar” mostrará el nombre del empleado y si ya cuenta con un jefe, al agregarlo si tenía uno con anterioridad y se notificará en un mensaje de alerta que el usuario que esta por agregar pertenecía a alguien más y se modificará esta línea de mando.

Los datos de movimientos son guardados en la bitácora, tanto el jefe nuevo, jefe anterior y quién realizó el cambio y cuándo.

3.- Una vez que se cuente con la lista de subordinados se podrá dar de baja, delante del nombre del subordinado se encuentra el botón “baja” con el que se eliminará de su línea de mando.

4.- Permitirá que a los subordinados se les pueda agregar un tercer evaluador (indispensable para otros sistemas institucionales) o modificar si fuera el caso, en el botón de “Modificar” en la lista de tercer evaluador, pedirá introducir el RFC del tercer evaluador, mostrará sus datos y lo agregará.

Los requerimientos no funcionales no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste.

- La fiabilidad o buen funcionamiento en el sistema.
- La capacidad de almacenamiento, que se sugiere en el servidor donde se instala se cuente con suficiente espacio para la base de datos
- La cantidad de usuarios que acceden concurrentemente, si todos los usuarios con permiso accedan al sistema se tendrían 400 usuarios.

Los requerimientos de dominio, se derivan del dominio del sistema más que de las necesidades específicas de los usuarios. Pueden ser requerimientos funcionales nuevos, restringir los existentes o establecer cómo se deben ejecutar cálculos particulares, como pueden ser algunas restricciones de derechos de autor.

5.2.2.- Casos de uso

Con base la especificación de requerimientos realizado anteriormente, se creó un diagrama de casos de uso donde se muestran los actores que interactúan con el sistema, las operaciones y los flujos de eventos (Fig. 4).

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

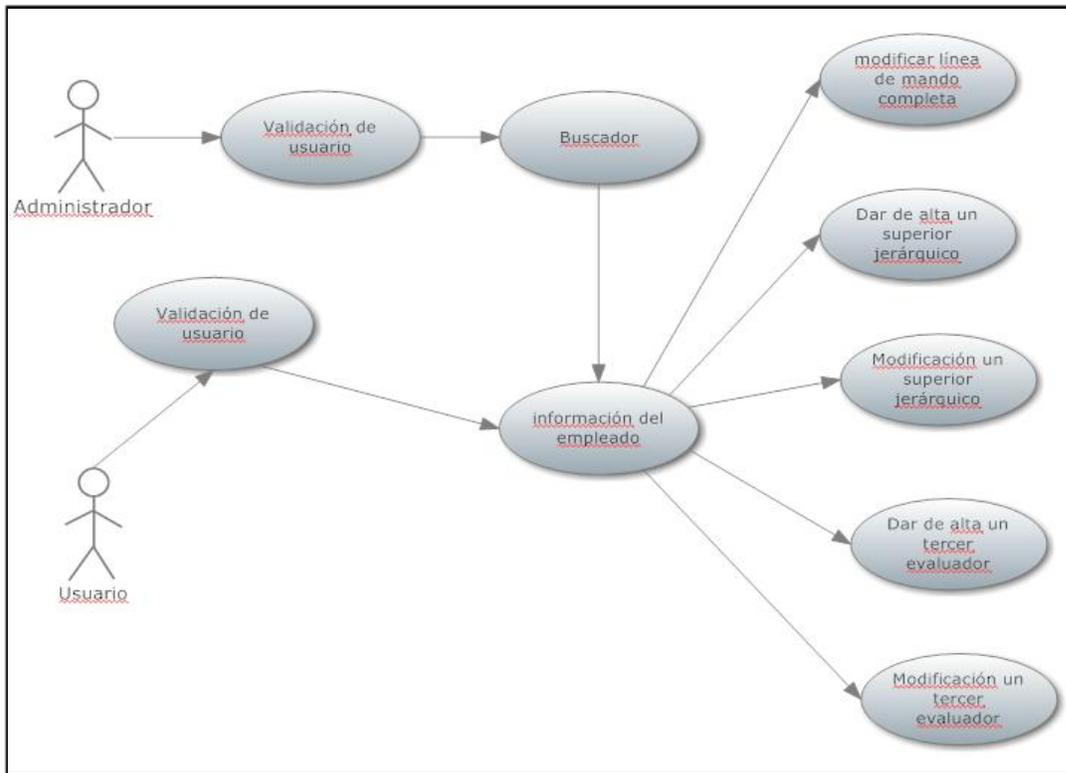


Fig. 4 Diagrama de casos de uso

5.2.2.1.- Escenario de ingreso

1. El usuario o administrador ingresa usuario y contraseña.
2. El sistema valida los datos ingresados.
3. Si datos correctos, ingresa al menú del sistema.
4. Si datos incorrectos, se ingresa datos nuevamente.

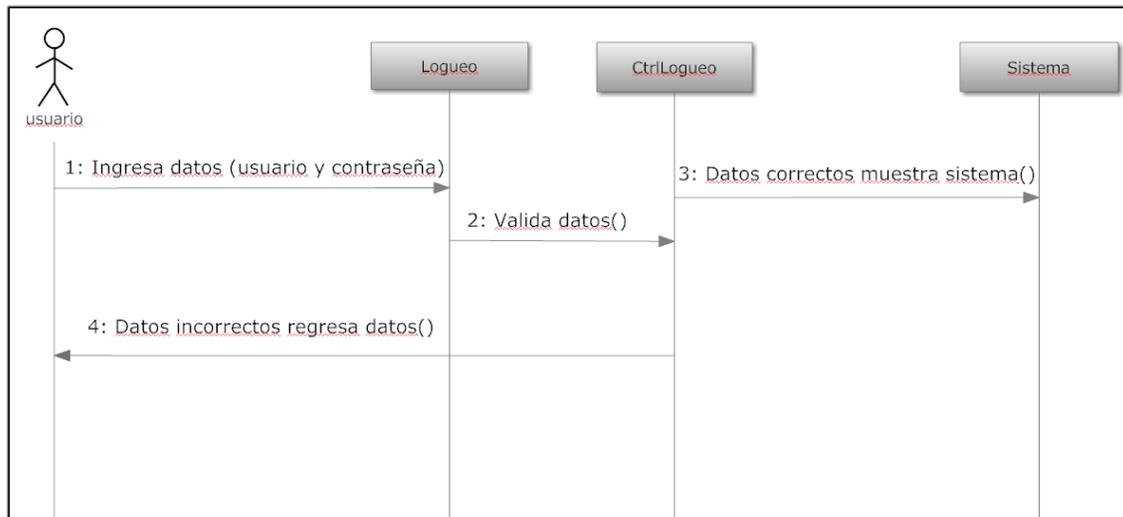


Fig. 5 Escenario de ingreso

5.2.2.2.- Escenario de alta de subordinado

1. El usuario o administrador da clic en agregar subordinado.
2. El usuario o administrador ingresa datos (RFC y/o número de empleado).
3. Si se encuentran coincidencias muestra datos de empleado.
4. Si no se encuentran coincidencias, vuelve a ingresar datos.
5. El usuario o administrador da clic en agregar.

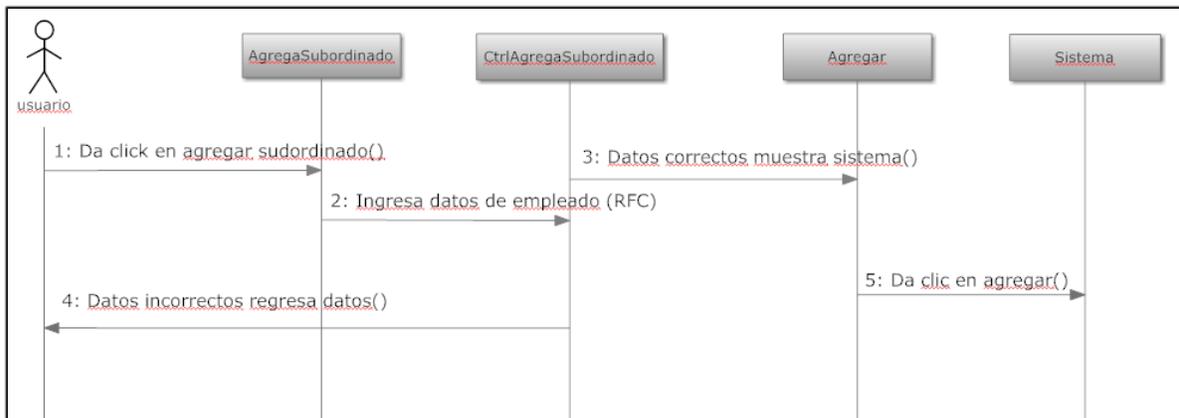


Fig. 6 Escenario de alta de subordinado

5.2.2.3.- Escenario de alta de tercer evaluador

1. El usuario o administrador da clic en modificar.
2. El usuario o administrador ingresa datos (RFC y/o número de empleado).
3. Si se encuentran coincidencias muestra datos de empleado.
4. Si no se encuentran coincidencias, vuelve a ingresar datos.
5. El usuario o administrador da clic en agregar.

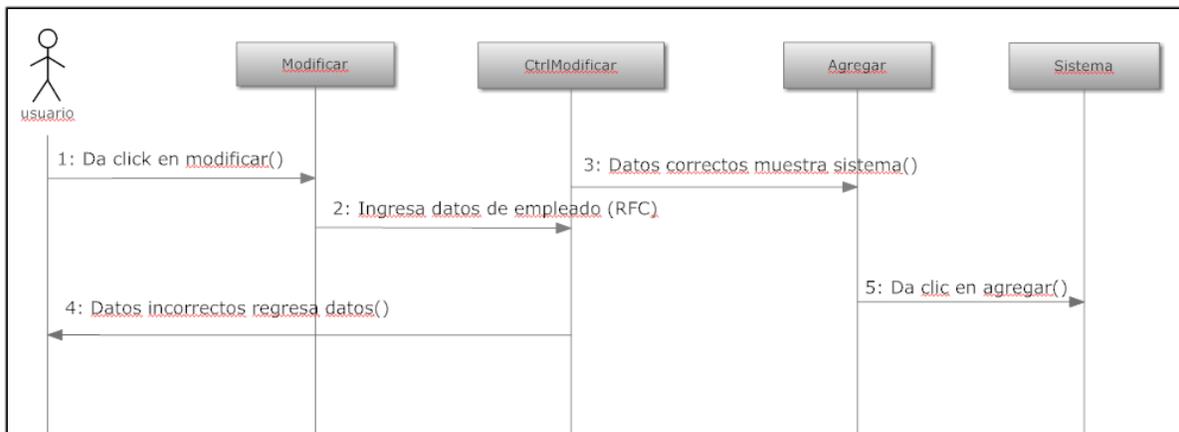


Fig. 7 Escenario de alta de tercer evaluador

5.3.- Diagrama Entidad-Relación

Aplicando las tres formas normales previamente mencionadas se obtuvo el siguiente diagrama entidad-relación del sistema (Fig. 8).

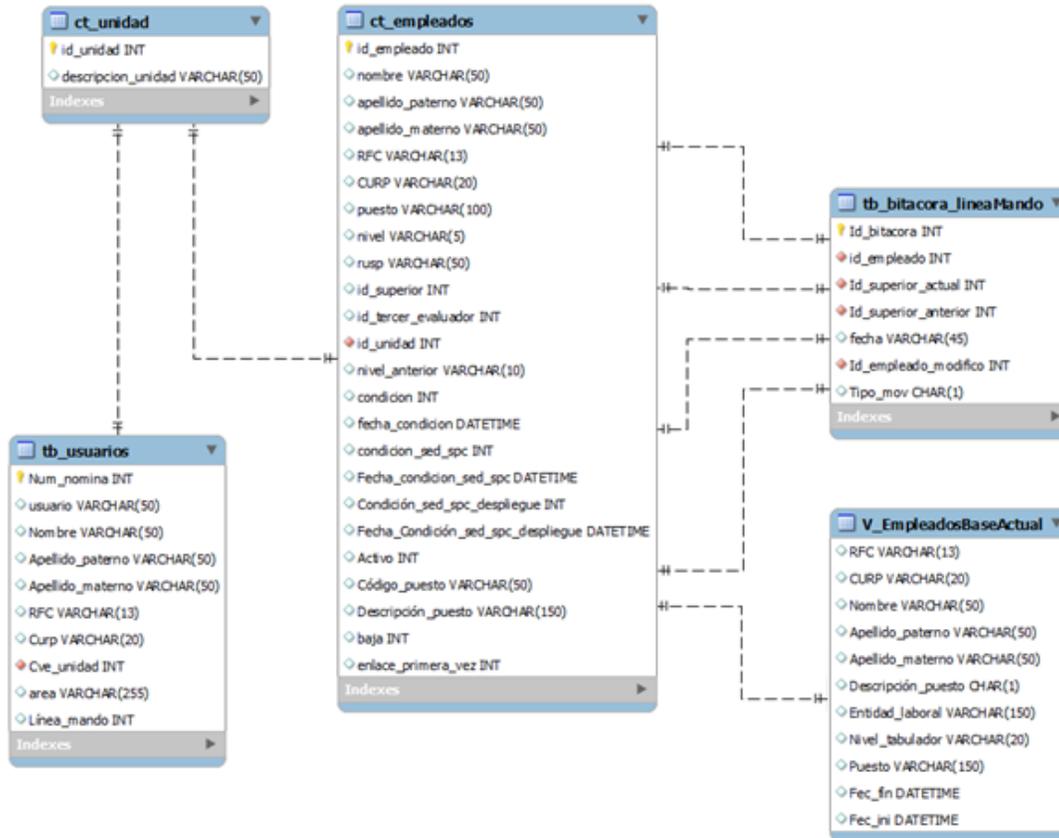


Fig. 8 Diagrama Entidad-Relación

5.3.1.- Diccionario de datos

A continuación se describen los campos que integrarán el diccionario de datos.

Tabla ct_empleados

Información de empleados que pueden ser modificados y consultados en su línea de mando

Tabla 8.- Estructura de la tabla empleados que se almacenan en el sistema de línea de mando

NOMBRE	TIPO	LONGITUD	DESCRIPCIÓN	KEY
Id_empleado	Int	[0-9]	Número de empleado	PK
Nombre	Varchar	50 caracteres	Nombre del empleado	--
Apellido_paterno	Varchar	50 caracteres	Apellido paterno del	--

			empleado	
Apellido_materno	Varchar	50 caracteres	Apellido materno del empleado	--
Rfc	Varchar	13 caracteres	RFC del empleado	--
Curp	Varchar	20 caracteres	CURP del empleado	--
Puesto	Varchar	100 caracteres	Puesto del empleado	--
Nivel	Varchar	5 caracteres	Nivel del puesto del empleado	--
Rusp	Varchar	50 caracteres	Registro Único de Servidores Públicos	--
Id_superior	Int	[0-9]	Número de empleado del jefe inmediato	--
Id_tercer_evaluador	Int	[0-9]	Número de empleado del tercer evaluador asignado	--
Id_unidad	Int	[0-9]	Identificador único de unidad a la que pertenece	FK
Nivel_anterior	Varchar	10 caracteres	Nivel anterior que tenía el empleado	--
Condición	Int	[0-9]	Condición para modificar línea de mando con lo que implica acceder al sistema	--
Fecha_condicion	Smalldate time	aaaa-mm-dd	Fecha en la que se aceptan los términos para modificar la línea de mando	--
Condición_sed_spc	Int	[0-9]	Condición de términos para el "sistema de evaluación de desempeño del servicio profesional de carrera"	--
Fecha_condicion_sed_spc	Smalldate time	aaaa-mm-dd	Fecha en la que se aceptan los términos del sistema "sed spc"	--
Condición_sed_spc_despliegue	Int	[0-9]	Condición de términos para el despliegue de metas del "sistema de evaluación de desempeño del servicio profesional de carrera"	--
Fecha_Condicion_sed_spc_despliegue	Smalldate time	aaaa-mm-dd	Fecha en la que se aceptan los términos para el despliegue de metas del sistema "sed spc"	--
Activo	Int	[0-9]	Si el empleado se encuentra actualmente activo en la dependencia	--
Codigo_puesto	Varchar	50 caracteres	Código de puesto en el que se desempeña el empleado	--
Descripción_puesto	Varchar	150 caracteres	Descripción del puesto que desarrolla el empleado	--
Baja	Int	[0-1]	0-usuarios dados de alta 1- Usuarios dados de	--

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

			baja	
enlace_primera_vez	Int	[0-1]	0-usuarios que ya tienen tiempo en la dependencia 1-Usuarios que acaban de ingresar	--

Tabla tb_bitacora_lineaMando

Historial de movimientos realizados en la línea de mando

Tabla 9.- Historial de movimientos realizados en la línea de mando

NOMBRE	TIPO	LONGITUD	DESCRIPCIÓN	KEY
Id_bitacora	Int	[0-9]	Identificador único de la tabla	PK
Id_empleado	Int	[0-9]	Número de empleado modificado	FK
Id_superior_actual	Int	[0-9]	Número de empleado del jefe que se asigno	FK
Id_superior_anterior	Int	[0-9]	Número de empleado del jefe anterior	FK
Fecha	Smalldate time	aaaa-mm-dd	Fecha en que se realizó el cambio	--
Id_empleado_modifico	int	[0-9]	Número de empleado que modificó el registro de la línea de mando	FK
Tipo_mov	char	("A","B","C")	Tipo de movimiento realizado: A : Alta B : Baja C : Cambio o actualización	--

Tabla ct_unidad

Catálogo de unidades administrativas

Tabla 10.- Historial de movimientos realizados en la línea de mando

NOMBRE	TIPO	LONGITUD	DESCRIPCIÓN	KEY
id_unidad	Int	[0-9]	Identificador único de la tabla	PK
descripcion_unidad	varchar	50 caracteres	Nombre de la unidad	--

Tabla V_EmpleadosBaseActual

Lista de empleados que están actualmente activos en la dependencia

Tabla 11.- muestra a los trabajadores activos

NOMBRE	TIPO	LONGITUD	DESCRIPCIÓN	KEY
RFC	varchar	13 caracteres	Rfc del empleado	--
CURP	varchar	20 caracteres	Curp del empleado	--
Nombre	varchar	50 caracteres	Nombre del empleado	--
Apellido_paterno	varchar	50 caracteres	Apellido paterno del empleado	--
Apellido_materno	varchar	50 caracteres	Apellido materno del empleado	--
Descripción_puesto	varchar	50 caracteres	Descripción de puesto del empleado	--
Sexo	char	"F", "M"	Sexo del empleado	--
Código_puesto	varchar	50 caracteres	Código de puesto del empleado	--
Id_empleado	int	[0-9]	Número de trabajador del empleado	--
Entidad_laboral	varchar	150 caracteres	Entidad laboral del empleado	--
Nivel_tabulador	varchar	20 caracteres	Nivel de puesto del empleado	--
Puesto	varchar	150 caracteres	Puesto que tiene el empleado	--
Fec_fin	Smalldate time	aaaa-mm-dd	Fecha de inicio en la plaza actual	--
Fec_ini	Smalldate time	aaaa-mm-dd	Fecha en la que concluye el contrato	--

Tabla tb_usuarios

Historial de usuarios en la dependencia

Tabla 12.- Historial de usuarios en la dependencia

NOMBRE	TIPO	LONGITUD	DESCRIPCIÓN	KEY
Num_nomina	int	[0-9]	Número de empleado	PK
Usuario	varchar	50 caracteres	Nombre de usuario único por empleado	--
Nombre	varchar	50 caracteres	Nombre del empleado	--
Apellido_paterno	Varchar	50 caracteres	Apellido paterno del empleado	--
Apellido_materno	Varchar	50 caracteres	Apellido materno del empleado	--
RFC	Varchar	13 caracteres	RFC del empleado	--
Curp	Varchar	20 caracteres	CURP del empleado	--
Cve_unidad	int	[0-9]	Unidad a la que pertenecen	FK
Área	Varchar	255	Área a la que pertenece en la	--

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

		caracteres	dependencia	
Línea_mando	int	[0-9]	1- Administrador 2.- Usuario	--

5.4.- Herramientas

En este tema se detallarán las herramientas que se utilizaron para el desarrollo y funcionamiento del sistema.

A continuación se muestra una comparativa de IDE de desarrollo de java (tabla 13).

Tabla 13.- Comparativa de Eclipse y Netbeans

Nombre	Definición	Características	Ventajas
Eclipse	Herramienta de programación de código abierto multiplataforma , usado para desarrollar entornos de desarrollo integrados	<ul style="list-style-type: none"> • Dispone de un editor de texto con resaltado de sintaxis. • La compilación es en tiempo real. • Tiene pruebas unitarias JUnit, control de versiones con CVS, integración con Ant, asistentes (wisards) para creación de proyectos, clases, test, etc., y refactorización. • Soporta JavaScript 	<ul style="list-style-type: none"> • Entorno de desarrollo integrado (IDE) de Eclipse. • Emplea modulos en ingles (plug-in). • Herramienta universal de código abierto y extensible para todo. • Plataforma ligera para componentes de software
Netbeans	Entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación java	<ul style="list-style-type: none"> • Es un producto libre y gratuito sin restricciones de uso. • Sintaxis resaltada. • Soporta JavaScript. • Completado de de código. • Refactorización. • Menor consumo de memoria. 	<ul style="list-style-type: none"> • La plataforma de Netbeans puede ser usada para desarrollar cualquier tipo de aplicación. • Instalación y actualización simple. • Incluye templates y wizards.

A continuación se hace una comparación de dos servidores de aplicaciones (tabla 14).

Tabla 14.- Comparativa de Tomcat y Glassfish

Característica	Tomcat	Glassfish
Escalable	si	sí
Autenticación JDBC	sí	sí
Transacciones de contenedores gestionados	no	sí
Soporte para hosting virtual	sí	sí
Cada host tiene su propio dominio de autenticación	sí	no
Cada huésped puede escuchar en un puerto diferente	si	sí

A continuación se hace una comparación de lenguajes de programación (tabla 15).

Tabla 15.- Comparativa de lenguajes de programación

Concepto	ASP .NET	PHP	Java	Python	Ruby
Costo de servidor	Alto	Gratuito	Gratuito	Gratuito	Gratuito
Sintaxis de lenguaje base	VB y C#	C / C++	C/ C++	C/ C++	Perl, Smalltalk, Eiffel, Ada, y Lisp
Orientado a objetos	Sí	No completamente	Sí	Sí	Sí
Sistemas operativos	Windows o Linux pero usando el proyecto Mono	Linux o Windows	Linux o Windows	Linux o Windows	Linux o Windows
Servidor	IIS o Mono	Apache, compilador propio	Apache, Tomcat y Glassfish	Apache, compilador propio	Apache, compilador propio
Empresa	Microsoft y Xamarin (para Mono)	The PHP Group (open source)	Oracle (open source)	Python software foundation (open source)	Grupo Ruby (open source)
Base de datos	MsSQLServer	Mysql	Oracle, mysql	Mysql y PostgreSQL	Mysql y PostgreSQL

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

(principalmente)					
Rapidez de ejecución Generación de página web	3er lugar	4to lugar	Último lugar	1er lugar	2do lugar
Propósito	Generar dinámicamente páginas web	Generar dinámicamente páginas web	Generar dinámicamente páginas web	Enfatiza la productividad y la lectura fácil del código	Código “divertido” y fácil de modificar por parte del desarrollador.
Apoyo de aprendizaje	Sitio web, foros, documentos proporcionados por Microsoft. En general buen soporte. Muy centralizada	Mucha, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda	Mucha, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda	Mucha, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda	Menos, pero descentralizada. No hay una entidad que de forma oficial centralice la ayuda
Soporte a móviles (todos por medio de un browser)	Native: Windows phone		Native: android		
Ambiente de desarrollo	Ms Visual Studio a costo Y herramientas open source	Eclipse y otras herramientas open source	Eclipse, netbeans y otras herramientas open source	Eclipse, netbeans y otras herramientas open source	Eclipse, netbeans y otras herramientas open source

Con las comparaciones anteriores se describen las herramientas estándar que se utilizan

en la dependencia

5.4.1.- Eclipse

Según (Sznajdleder, 2013, Pág. 3) Una IDE (Integrated Development Enviroment) es una herramienta que permite editar programas, compilarlos, depurarlos, documentarlos y ejecutarlos, etc.

Para trabajar con Java existen en el mercado diferentes IDE. Algunas son de código abierto (open source) como Eclipse y NetBeans y otras son comerciales e impulsadas por las empresas de tecnología como JBuilder (de Borland), JDeveloper (de Oracle), WebPhere (de IBM), etcétera.

Según (Eclipse.org, 2014) eclipse es una comunidad de personas y organizaciones que deseen colaborar en el comercio de usar software de código abierto. Sus proyectos se centran en la construcción de una plataforma de desarrollo abierta formada por marcos extensibles, herramientas y tiempos de ejecución para la construcción, despliegue y gestión de software a través del ciclo de vida.

La fundación Eclipse es un miembro sin fines de lucro que alberga los proyectos de Eclipse y ayuda a cultivar tanto una comunidad de código abierto y un ecosistema de productos y servicios complementarios.

El proyecto Eclipse fue creado originalmente por IBM en noviembre de 2001 y con el apoyo de un consorcio de proveedores de software.

Según (wiki.eclipse.org, 2014) Eclipse está abierto ya que su diseño permite una fácil extensión por parte de terceros. Se trata de un entorno de desarrollo integrado (IDE), ya que proporciona herramientas para gestionar espacios de trabajo; para construir, lanzar y depurar aplicaciones; compartir objetos con un equipo y en el código de la versión; y para personalizar fácilmente la experiencia en programación. Eclipse es una plataforma que está diseñado para ser extendido indefinidamente con más y más sofisticadas herramientas.

5.4.2.- Java

Según (java.com/es/download/faq/whatis_java.xml, 2014, s/p) Java es una tecnología que se usa para el desarrollo de aplicaciones que convierten a la Web en un elemento más interesante y útil. Java no es lo mismo que javascript, ya que se trata de una tecnología sencilla que se usa para crear páginas web y solamente se ejecuta en el explorador.

Java le permite jugar, cargar fotografías, chatear en línea, realizar visitas virtuales y utilizar servicios como, por ejemplo, cursos en línea, servicios bancarios en línea y mapas interactivos.

Según (java.com/es/download/whatis_java.jsp, 2014, s/p) Java es un lenguaje de programación y la primera plataforma informática creada por Sun Microsystems en 1995.

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios. Java se ejecuta en más de 850 millones de ordenadores personales de todo el mundo y en miles de millones de dispositivos, como dispositivos móviles y aparatos de televisión.

Existe un gran número de aplicaciones y sitios Web que no funcionan a menos que Java esté instalado, y muchas más que se crean a diario, es rápido, seguro y fiable. De portátiles a centros de datos, de consolas de juegos a súper equipos científicos, de teléfonos móviles a Internet, Java está en todas partes.

Java ha sido probado, ajustado, ampliado y probado por toda una comunidad de desarrolladores, arquitectos de aplicaciones y entusiastas de Java. Está diseñado para permitir el desarrollo de aplicaciones portátiles de elevado rendimiento para el más amplio rango de plataformas informáticas posible. Al poner a disposición de todo el mundo aplicaciones en entornos heterogéneos, las empresas pueden proporcionar más servicios y mejorar la productividad, las comunicaciones y colaboración del usuario final y reducir drásticamente el costo de propiedad tanto para aplicaciones de usuario como de empresa, Java nos permite:

- Escribir software en una plataforma y ejecutarla virtualmente en otra
- Crear programas que se puedan ejecutar en un explorador y acceder a servicios Web disponibles
- Desarrollar aplicaciones de servidor para foros en línea, almacenes, encuestas, procesamiento de formularios HTML y mucho más
- Combinar aplicaciones o servicios que utilizan el lenguaje Java para crear aplicaciones o servicios con un gran nivel de personalización
- Escribir aplicaciones potentes y eficaces para teléfonos móviles, procesadores remotos, microcontroladores, módulos inalámbricos, sensores, gateways, productos de consumo y prácticamente cualquier otro dispositivo electrónico.

5.4.3.- Apache Tomcat 6

Según (Tomcat.apache.org, 2014, s/p) Apache Tomcat es una implementación de código abierto de software de las tecnologías Java Servlet y JavaServer Pages. Las especificaciones Java Servlet y JavaServer Pages son desarrollados bajo la “Java Community Process”.

Apache Tomcat se desarrolla en un entorno abierto y participativo y publicado bajo la licencia Apache versión 2.

Según (httpd.apache.org, 2014) el Proyecto Apache HTTP Server es un esfuerzo de desarrollo de software de colaboración cuyo objetivo es crear un grado comercial robusto, con muchas características y de libre disposición, implementación del código fuente de un servidor (web) HTTP.

Según (Maciá, Pág. 114, 2010) Tomcat es un contenedor de Servlets creado por el grupo apache que implementa las especificaciones de Java Server Pages y Java Servlet definidas

por SUN mediante el comité JCP (Java Community Process). Un contenedor de servlets es un marco de ejecución que maneja en invoca Servlets por cuenta del usuario, Tomcat es conocido también como servidor de aplicaciones.

5.4.4.- SQL Server 2008

Según (Pérez, 2011, Pág.1) SQL Server es el sistema de base de datos profesional de Microsoft. Contiene una variedad de características y herramientas que se pueden utilizar para desarrollar y administrar bases de datos, así como soluciones de todo tipo relacionadas a ellas.

El motor de base de datos: Es el servicio principal para almacenar, procesar y proteger los datos, proporciona acceso controlado y procesamiento rápido de transacciones.

Según (Gabillaud, 2010 Pág. 17) SQL Server es un SGBDR (Sistema Gestor de Base de Datos Relacional) completamente integrado en Windows, permite gestionar los datos conservando su integridad y su coherencia se encarga de:

- Almacenar datos.
- Verificar las restricciones de integridad definidas.
- Garantizar la coherencia en los datos que almacena.
- Asegura las relaciones entre los datos definidos por el usuario.

Según (Osorio, 2008, Pág. 77) SQL Server es una plataforma para base de datos que se utiliza en el procesamiento transaccional en línea (OLTP) a gran escala, en las bodegas de datos y las aplicaciones de comercio electrónico así como también es una plataforma de inteligencia de negocios para soluciones de integración, análisis y creación de informes de datos.

5.5.- Interfaz de usuarios

Dado que los usuarios finales interactúan sólo con la interfaz de usuario del sistema, muchas veces esta es considerada como todo el sistema en sí, debido a que ésta permanece como una representación del mismo, por tal motivo el diseño de la interfaz de usuario es de gran importancia para la aceptación del sistema.

Los objetivos que se deben perseguir durante el diseño de dicha interfaz son los siguientes:

- Eficacia: Se debe lograr que el usuario tenga acceso al sistema, de manera que sea congruente con sus necesidades particulares.
- Eficiencia: El sistema debe guiar y facilitar la captura, además de reducir los errores en ésta.
- Retroalimentación del sistema: Al proporcionar al usuario elementos de ayuda que le permitan ubicarse dentro del sistema y guiarlo dentro de él.

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Para el diseño de ésta se utilizaron elementos que permitieran una fácil interacción con los usuarios; debido a que el sistema se convertirá en una herramienta de utilización para el personal de la PROFEPA, esta debe ser amigable e intuitiva, además debe contar con mensajes que proporcionen ayuda para el usuario, contribuyendo así a facilitar sus labores.

A cada una de las secciones del sistema se tendrá acceso mediante botones, los cuales contendrán etiquetas relacionadas con el tema o la acción a realizar.

En el sistema se encontrarán una serie de advertencia o mensajes como por ejemplo, el sistema indicará que algún dato falta de ingresar o que la sesión ha terminado.

5.5.1.- Diseño final del sistema

A continuación se muestra el diseño final de la interfaz del sistema.

Módulo de logueo

Se introduce el RFC para validar a un usuario. (Fig.9)



The image shows a web-based login interface. At the top left is the PROFEPA logo with the text 'PROFEP A' and 'PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE'. To the right, the main title reads 'SISTEMA DE LÍNEA DE MANDO' and 'SISTEMA AUTOMATIZADO DE EVALUACIÓN DE DESEMPEÑO'. The central area is titled 'Ingreso' and contains the instruction: 'Por favor ingrese su RFC con Homoclave, sin guiones o espacios:'. Below this is a single-line text input field. At the bottom of the input area is a button labeled 'Buscar Usuario'.

Fig. 9. Introducción de RFC

Al introducir un RFC no válido, el sistema envía un mensaje de error (Fig.10)



The screenshot shows the login interface of the 'SISTEMA DE LÍNEA DE MANDO' (Automated Performance Evaluation System) for PROFEPA. The header includes the PROFEPA logo and the system name. The main content area is titled 'Ingreso' and contains a text prompt: 'Por favor ingrese su RFC con Homoclave, sin guiones o espacios:'. Below this is a single text input field and a 'Buscar Usuario' button. A 'Mensaje de página web' (Web page message) dialog box is open on the right, displaying a yellow warning icon and the text 'Debe introducir un RFC válido.' (You must enter a valid RFC), with an 'Aceptar' (Accept) button.

Fig. 10. Se introduce erróneamente un RFC

Al introducir un RFC válido y con permisos nos muestra la pantalla para introducir la contraseña (Fig. 11)



The screenshot shows the password entry screen of the 'SISTEMA DE LÍNEA DE MANDO'. The header is identical to the previous screen. The main content area is titled 'Ingreso' and contains three labeled input fields: 'RFC:', 'Usuario:', and 'Contraseña:'. Below these fields are two buttons: 'Regresar' (Return) and 'Ingresar' (Login).

Fig. 11. Introducción de contraseña

Introducción de contraseña incorrecta (Fig. 12)

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.



Fig. 12. Introducción de contraseña erróneamente

Módulo de administrador

Buscador de usuario a nivel de Administrador (Fig. 13)



Fig. 13. Buscador

Utilizando cualquier filtro despliega la lista de resultados (Fig. 14)

The screenshot shows the 'SISTEMA DE LÍNEA DE MANDO' interface. At the top left is the logo for PROFEPA (PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE). The title 'SISTEMA DE LÍNEA DE MANDO' is centered, with the subtitle 'SISTEMA AUTOMATIZADO DE EVALUACIÓN DE DESEMPEÑO' below it. The main section is titled 'Buscador de Empleados' and contains a form with the following fields: 'Nombre(s):', 'Apellido Paterno:', 'Apellido Materno:', 'RFC:', 'Area de Adscripción:' (a dropdown menu), 'Número de Nómina:', and 'Con superior jerárquico:' (a dropdown menu with 'SI' selected). A 'Buscar Empleado' button is located below the form. Below the search form is a section titled 'Resultados de la Búsqueda de Empleados.' which displays a table with two columns: 'Nombre del Evaluado' and 'RFC'. The table contains one entry: 'GUEVARA LÓPEZ LENIN'. A 'Salir' button is located at the bottom of the results section.

Fig. 14. Lista de Resultados

Módulo de usuario

Seleccionando un usuario despliega sus datos, pudiéndose elegir el área de adscripción (Fig. 15)

The screenshot shows the 'Datos del Empleado' form. It contains the following fields: 'Nombre:' (GUEVARA LÓPEZ LENIN), 'RFC:', 'Area de Adscripción:' (DIRECCION GENERAL DE ADMINISTRACIÓN), 'Nivel:', 'Superior Jerárquico:', 'RFC Superior Jerárquico:', 'Tercer Evaluador:', and 'RFC Tercer Evaluador:'. An 'Actualiza Adscripción' button is located at the bottom of the form.

Fig. 15 Información del Usuario

Mensaje al ser actualizada la área de adscripción (Fig. 16)

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

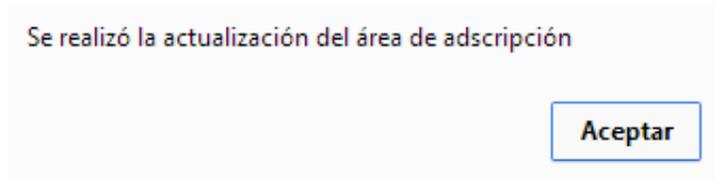


Fig. 16 Modificación exitosa del área de adscripción.

Se muestra la lista de subordinados que se tienen actualmente (Fig. 17)

Línea de Mando del Superior Jerárquico: CASTELLANOS ALCARAZ RAÚL HERBERTO

Nombre del Evaluado	RFC	Nivel	Baja	Nombre Tercer Evaluador	Tercer Evaluador
DOMÍNGUEZ RAMOS NANCY			<input type="button" value="Baja"/>	ENRIQUE PEÑALOZA RODRÍGUEZ	<input type="button" value="Modificar"/>
GUEVARA LÓPEZ LENIN			<input type="button" value="Baja"/>	ENRIQUE PEÑALOZA RODRÍGUEZ	<input type="button" value="Modificar"/>
PAREDES LÓPEZ LUIS ENRIQUE			<input type="button" value="Baja"/>	ENRIQUE PEÑALOZA RODRÍGUEZ	<input type="button" value="Modificar"/>
PEÑALOZA RODRÍGUEZ ENRIQUE			<input type="button" value="Baja"/>	NANCY DOMÍNGUEZ RAMOS	<input type="button" value="Modificar"/>

En caso de que en lista haga falta un empleado que este a su cargo:

Fig. 17 Lista de subordinados.

Se muestra el mensaje al intentar introducirse a sí mismo como subordinado (Fig. 18)

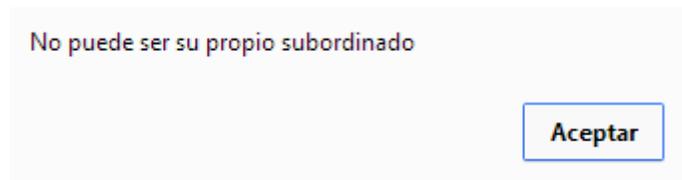


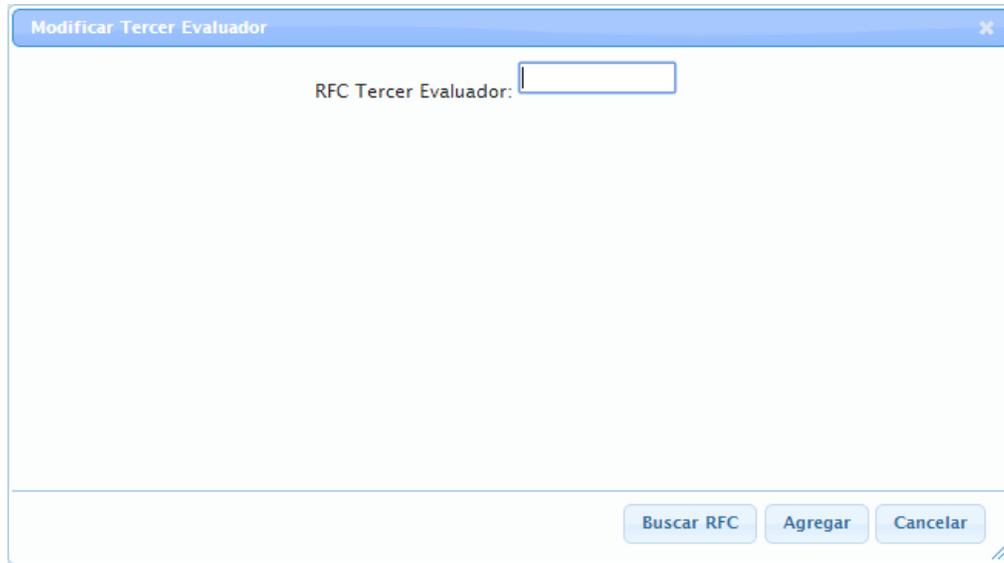
Fig.18 Mensaje de error al intentar ser subordinado propio

Se muestra el mensaje al dar de baja un subordinado (Fig. 19)



Fig. 19. Baja de subordinado

Aparece una ventana para asignar al tercer evaluador y a un subordinado (Fig. 20)



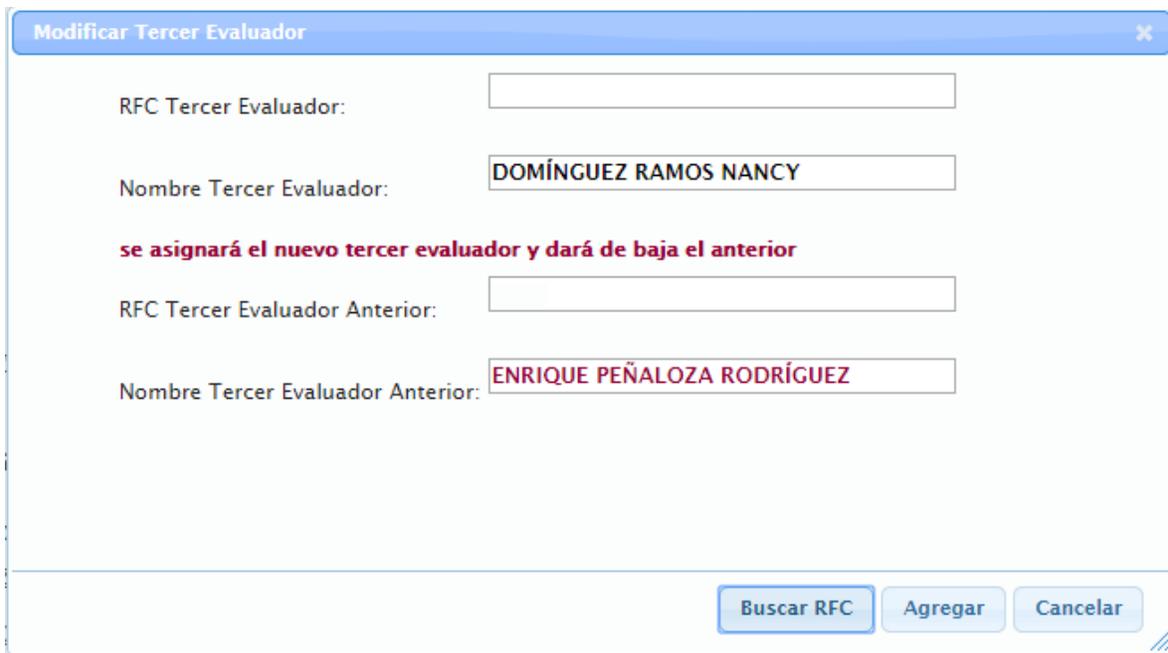
Modificar Tercer Evaluador

RFC Tercer Evaluador:

Buscar RFC Agregar Cancelar

Fig. 20 Asignación de tercer evaluador

En caso de que ya exista un tercer evaluador (Fig. 21)



Modificar Tercer Evaluador

RFC Tercer Evaluador:

Nombre Tercer Evaluador:

se asignará el nuevo tercer evaluador y dará de baja el anterior

RFC Tercer Evaluador Anterior:

Nombre Tercer Evaluador Anterior:

Buscar RFC Agregar Cancelar

Fig. 21. Cambio de tercer evaluador

Para la modificación de la línea de mando de un subordinado (Fig. 22)

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

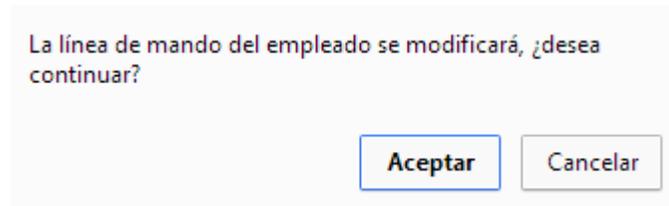


Fig. 22. Cambio de línea de mando

Para cambiar a sí mismo como jefe y tercer evaluador arroja el siguiente mensaje de error (Fig. 23)

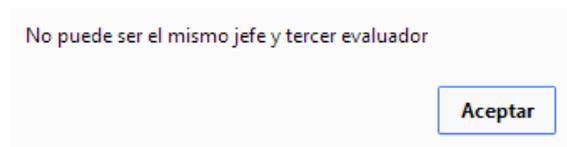


Fig. 23. Error al querer ser el mismo jefe

Muestra la Línea de mando de un jefe sin subordinados asignados (Fig. 24)

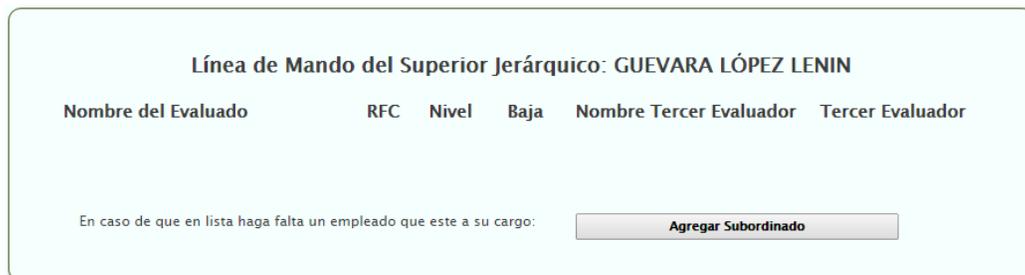


Fig. 24. línea de mando vacía

Para agregar a un subordinado (Fig. 25)



Fig. 25. Asignación de un subordinado

Para realizar el cambio de un jefe a otro cuando ya pertenece a otra línea de mando (Fig. 26)

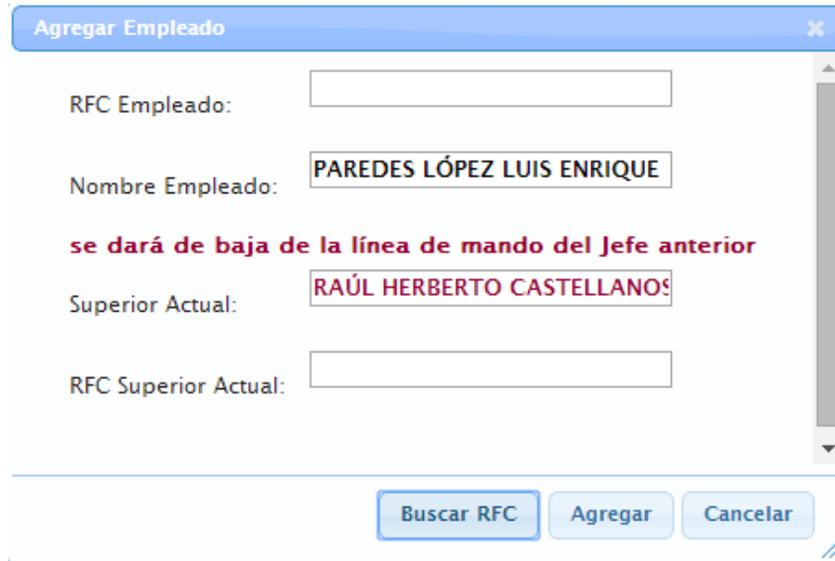


Fig.26. Cambio de línea de mando

Se muestra un mensaje de alerta para modificar una línea de mando de un empleado (Fig. 27)

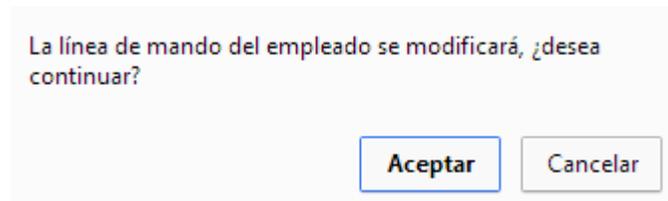


Fig. 27. Cambio de línea de mando

Para realizar la modificación de una línea de mando completa (Fig. 28)

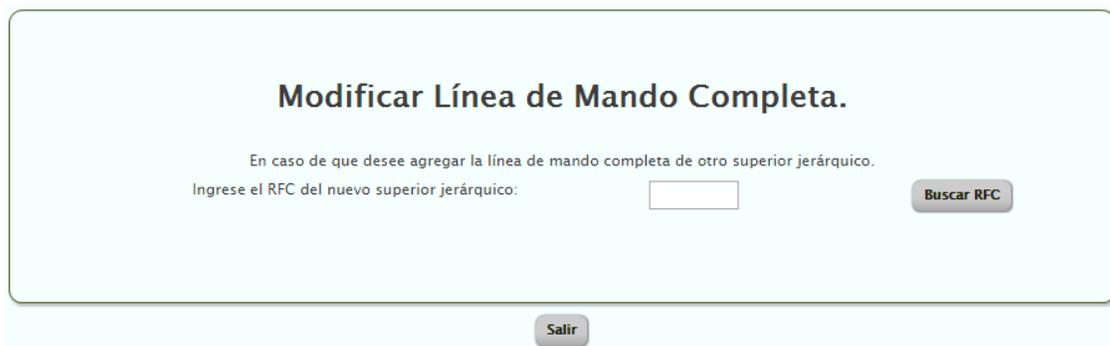


Fig. 28. Modificación completa

Para mostrar la lista de subordinados a cambiar completamente (Fig. 29)

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Modificar Línea de Mando Completa.

En caso de que desee agregar la línea de mando completa de otro superior jerárquico.

Ingrese el RFC del nuevo superior jerárquico:

Línea de Mando del Superior Jerárquico CASTELLANOS ALCARAZ RAÚL HERBERTO

Nombre del Evaluado	RFC
GUEVARA LÓPEZ LENIN	GULL860204HXA
PAREDES LÓPEZ LUIS ENRIQUE	PALL840718538

Fig. 29. Modificación completa

Se muestra un mensaje para confirmar que desea cambiar toda esta línea de mando a otro jefe (Fig. 30)

Se modificará la línea de mando completa del superior jerárquico, ¿desea continuar?

Fig. 30. Mensaje de confirmación de cambio completo de línea de mando

Se muestra un mensaje que arroja el sistema cuando no se encontraron subordinados en el RFC que se introdujo para cambiar su línea de mando (Fig. 31)

No se encontraron empleados que tengan como superior jerárquico el RFC ingresado

Fig. 31. Empleados sin subordinados para cambiar de línea de mando completa

5.6.- Desarrollo de la capa de vista

Para desarrollar la capa de vista se utilizó paginas HTML, hojas de estilo, JavaScript y AJAX con los cuales se pueden crear aplicaciones web con interfaz gráfica de forma rápida y sencilla, además de que son las herramientas estándar con las que cuenta la Institución. Las pantallas de captura identificadas para cada proceso se crearon de acuerdo con los estándares de diseño establecidos con anterioridad, utilizando todos los elementos necesarios para facilitarle al usuario la manipulación de los datos capturados.

5.6.1.- Codificación

Para la codificación de la capa de vista se usaron funciones en JavaScript para realizar distintas acciones como:

- Validar campos.
 - Validación de campos obligatorios.
 - Validación de tipos de datos permitidos para cada campo.
 - Longitud de campos.
- Mensajes de aviso, advertencia o error.
- Comunicación con la capa de controlador.
 - Tanto para enviar la información como para recibirla de la capa de controlador la tecnología AJAX.

5.7.- Desarrollo de la capa de controlador

Para la capa de controlador se hizo uso del lenguaje de programación JAVA, el cual mediante clases que reciben las acciones indicadas por la capa de vista, lleva a cabo estas.

5.7.1.- Codificación

Para la codificación del lado de la capa de controlador se utilizó una clase en particular de JAVA llamada ManejaDatos, el cual sirve de intermediaria entra la capa de vista y de modelo.

5.8.- Desarrollo de la capa de modelo

Para poder leer y escribir se hizo uso de querys en SQL.

5.8.1.- Codificación

Para poder realizar los queries mencionados se necesita conexión con la base de datos. Existen librerías creadas en JAVA que mediante un driver permiten la conexión con Microsoft SQL Server.



PRUEBAS,

IMPLEMENTACIÓN Y

MANTENIMIENTO

6.- Pruebas, implementación y mantenimiento

El plan de pruebas realizadas al sistema desarrollado incluye pruebas de bajo nivel mediante las cuales se verificó la implementación correcta de cada uno de los segmentos.

Una vez validado el software, éste se combina con otros elementos del sistema (otro software, hardware, gente, bases de datos etc.) La *prueba del sistema* verifica que cada elemento encaja de forma adecuada y que se alcanza la funcionalidad y el rendimiento total del sistema.

6.1.- Pruebas de unidad

En esta prueba se realizó la verificación de cada uno de los módulos, probándose la información que fluye hacia y desde el módulo, todos los caminos de la estructura de control, verificando los límites para asegurar que los datos que mantienen temporalmente conservan su integridad. Finalmente se probaron los caminos de manejo de errores.

Casos de prueba:

- Se les dio un valor vacío (`new String()`) a cada uno de los parámetros lo que originó problemas al momento de generar el query de consulta, así como cuando se realizó la comparación de valores para verificar los datos obtenidos.
- Al momento de darles un valor *null* a los parámetros, estos originaron excepciones en el código (**`NullPointerException`**) al momento de querer hacer uso de ellos.

Finalmente dándoles valores aleatorios del tipo String, no se produjo error en el código y tampoco de lógica.

6.2.- Pruebas de integración

En estas pruebas se observa la interacción entre los módulos los cuales se unen formando la estructura dictada por el diseño. La integración de los módulos se realizó de manera incremental para facilitar la detección y corrección de errores.

6.3.- Pruebas de validación

Durante esta prueba se valida el funcionamiento específico del sistema de acuerdo con los requerimientos determinados. En este caso el procedimiento de prueba para demostrar la conformidad con los requisitos, consistió en el alta y modificación de información por parte de un usuario en el entorno de desarrollo y bajo supervisión, la segunda parte de la prueba la realizó el usuario sin supervisión y él cual nos notificó los problemas.

6.4.- Pruebas del sistema

Antes de implementar el sistema en el servidor de producción se realizará una prueba piloto en un servidor de capacitación durante una semana, con el objetivo de observar la operación del sistema en un escenario completamente real donde se combina el sistema ya validado con el hardware, el sistema operativo, la base de datos y los usuarios reales, para probar su desempeño tanto en cuestiones técnicas como en su funcionamiento con datos reales, ya que como es de esperarse las pruebas realizadas previas a la liberación no detectan todos los errores latentes en el sistema los cuales surgen con la operación real.

Otra cosa que es importante detectar es que el sistema permita registrar todos los datos introducidos y estos sean validados antes de que sean almacenados en la base de datos, con esto se garantiza que la información registrada será real y no tenga basura que afecte en el rendimiento de la aplicación. Una vez que se depuren los errores reportados en esta etapa, el sistema estará listo para ser instalado en el servidor de producción.

6.5.- Implantación

La implantación es el proceso que asegura la operatividad del sistema de información, es aquí en donde se pueden encontrar algunas imperfecciones o defectos que no se detectaron en la etapa de pruebas y estos deberán ser corregidos.

Para la implantación del sistema se instaló el sistema en un servidor de capacitación, el cual es idéntico al servidor de producción, y como se mencionó, como prueba piloto los usuarios probaran el sistema.

Al ser un sistema web, una vez que se instale en el servidor de producción, cualquier usuario podrá tener acceso a él ingresando a la dirección en la que se encuentra.

Se cuenta con un servidor espejo tanto para la base de datos, como para la aplicación.

Se realizarán respaldos semanales de las bases datos.

6.6.- Requerimientos de instalación

Las características del equipo donde se va a llevar a cabo la instalación son:

- Procesador:
 - Mínimo 1 GHz.
 - Recomendado 2 GHz.
 - Óptimo 3 Ghz.

- Memoria:
 - Mínimo 512 MB de RAM.
 - Recomendado 1 GB de RAM.
 - Óptimo 2 GB de RAM.

- Espacio en Disco:
 - Mínimo 100 GB
 - Recomendado 200 GB
 - Óptimo 500 GB
- Sistema Operativo: Microsoft Windows Server 2008
- Software:
 - Microsoft SQL Server 2008
 - Apache Tomcat 6.0

6.7.- Mantenimiento

El mantenimiento es inherente a todo sistema de información ya que existen diversas causas por las que es necesario realizarlo.

En el mantenimiento correctivo del sistema, se corregirán todos aquellos problemas que surjan durante la operación del sistema y que no fue posible detectar durante las pruebas.

Se deberá realizar mantenimiento adaptativo al sistema cuando se presenten cambios en los procedimientos en los que se basa el sistema o bien, cuando el software cambia de versión como lo puede ser el sistema operativo, en estos casos será necesario hacer pruebas para determinar si hay que hacer algún ajuste para que el sistema opere sin problemas de compatibilidad.

Además se realizará el mantenimiento perfectivo, cuando sea necesario hacer mejoras para ayudar a realizar las actividades y manejo del sistema más sencillo y fácil de entender o surjan nuevos requerimientos para el manejo de la información.

Por último el mantenimiento Preventivo que el mismo usuario puede realizar ya que para que el sistema siga operando adecuadamente en ocasiones es necesario realizar una depuración del equipo y de la base de datos, con ello evitar problemas que se puedan generar por la falta del mismo.

De lo anterior se deduce que no existirá sólo un periodo para realizar el mantenimiento al sistema ya que éste puede ser muy largo o muy corto dependiendo de las circunstancias que se mencionaron y que se puedan presentar.



CONCLUSIONES

Conclusiones

Con el apoyo de las nuevas tecnologías de la información y la comunicación, los lenguajes de programación, software de programación, metodologías de programación, realización de pruebas, entre otros, se pudo realizar un sistema de Línea de Mando que agilizará las evaluaciones del personal de la PROFEPA y el cual presenta los siguientes beneficios:

- Se mejoró la calidad en las evaluaciones por medio de este sistema, eliminando intermediarios y pudiendo consultar la información en tiempo real y hacer las modificaciones del mismo modo, la construcción de esta aplicación está facilitando el trabajo de recolección de información y entrega de reportes de las relaciones entre los trabajadores y sus jefes.
- Al llevar un control en la bitácora del sistema se puede verificar las modificaciones que realizan las personas autorizadas para efectuar altas, bajas o cambios. Así como si procedieron dichas modificaciones.
- El utilizar la metodología RUP en este sistema se permite que el desarrollo de software se lleve a cabo a gran escala, usando un proceso continuo de pruebas y retroalimentación así como el manejador de base de datos SQL Server 2008.
- Al realizar las pruebas y depuración del sistema en tiempo real, se observó que los trámites que se agilizaron y ahorraron tiempo a la hora de llevar a cabo las evaluaciones, así como la obtención de información y las estadísticas solicitadas.
- Al estar en red proporciona una accesibilidad a los datos tanto para la consulta como para la actualización de una manera transparente, rápida y sencilla con una interfaz amigable.
- Se hizo la observación a los responsables que pueden modificar los registros que con cada versión que se libera deben capacitarse en el uso del mismo.
- El control y actualización de la información en tiempo y forma se lleva por medio de reportes en tiempo real de una manera dinámica y eficaz para los usuarios que requieren tener la información a la mano.

En lo personal, las habilidades que he adquirido durante mi estancia en la dependencia me han ayudado para complementar la parte práctica de mi carrera, aplicando los conocimientos aprendidos durante mi carrera, a la vez que comprendí cuál es el papel que juegan las evaluaciones realizadas al personal de la organización y la línea de mando, cómo funcionan y para quién aplican. El análisis, diseño y desarrolló de los prototipos del sistema desde un inicio me ayudaron a encontrar una metodología, a crear las reglas en el código, basándome en las reglas de negocio solicitadas, a buscar un software que se adaptará a los requerimientos de la organización, etc. Y finalmente a crear un sistema que está funcionando dentro de la organización. Además amplió mi panorama de creatividad al hacer los diseños y prototipos, el apoyo del personal del área donde laboro es fundamental para llevar a cabo este sistema y dar seguimiento en el análisis y puesta en marcha de dicho sistema.



GLOSARIO

Glosario

A

ASD: Adaptive Software Development - Desarrollo de Software Adaptable, es una metodología que tiene una adaptación continua al trabajo.

B

Base de Datos: Conjunto o colección de datos relevantes relacionados entre sí.

C

Cursor: variable que nos permite recorrer con un conjunto de resultados obtenido a través de una sentencia SELECT fila a fila.

D

Dato: es un elemento aislado, recabado para un cierto fin, pero que no ha pasado por un proceso que lo interrelacione con otros de manera funcional para el fin previsto.

DDL: (Data Definition Language) es aquel define los datos, nos permite crear, modificar o borrar objetos y así como las relaciones entre ellos en la base de datos.

DML: (Data Manipulation Languaje) este lenguaje realiza consultas sobre la base de datos y para insertar, modificar o eliminar la información almacenada.

DCL: (Data Control Language) se utiliza para asignar permisos a los objetos de las bases de datos.

G

GET: Es un protocolo estándar que permite que los clientes se comuniquen con los servidores utilizando HTTP incluye la dirección del cliente en un formato Uniform Resource Locator (URL), seguida por el carácter “?” y por una cadena que contiene todos los datos en la forma.

H

HTTP: protocolo de intercambio de información entre los clientes web y los servidores HTTP, es decir, es un protocolo de comunicación cliente-servidor.

I

Independencia lógica de los datos: permite cambiar la representación lógica del problema sin afectar a los programas de aplicación que los manipulan.

Independencia física de los datos: permite que la distribución en las unidades de almacenamiento y la estructura física de la información sea independiente de los cambios de la estructura lógica, un ejemplo de esto sería, al mover la base de datos de forma física (servidores, discos duros, unidades de almacenamiento) no debe verse afectada la información contenida.

Información: Es el conjunto de datos que han sido tratados o que se les aplica un proceso en el cual se acomodan de un modo lógico, coherente y comprensible para los usuarios finales.

Integridad: Es la veracidad de los datos almacenados respecto a la información existente en el dominio del problema que trata la base de datos, los datos no deben ser modificados o destruidos de forma anómala.

N

Normalización: Reglas que definen la semántica y la sintaxis de los lenguajes de programación o las funciones a realizar en una aplicación.

O

OSI: El modelo de referencia OSI (*open system interconnection*, interconexión de sistemas abiertos) conjunto de reglas y procedimientos que permite a los sistemas el intercambio de información.

P

PL/SQL: permite la definición de funciones, procedimientos y métodos mediante el componente procedimental, PL/SQL en Oracle y TransactSQL en SQL Server de Microsoft.

Privacidad: es evitar que personas no autorizadas puedan acceder a la información de la base de datos.

POST: Es un protocolo que coloca los parámetros directamente en el mensaje de petición HTTP y son transferidos a través de una cabecera http independiente y no son visibles para el usuario.

R

RUP: Rational Unified Process - Proceso Unificado Racional, es un proceso continuo de pruebas y retroalimentación para desarrollo de software a gran escala.

Redundancia mínima: La redundancia provoca problemas de inconsistencia de la información y necesidades de almacenamiento mayores, las bases de datos sugieren evitar la redundancia, pero siempre debe existir algo de redundancia a la que se denomina redundancia mínima.

S

Simplicidad: Las bases de datos deben estar basadas en representaciones lógicas simples, para permitir la verificación en la representación del problema que representan.

Seguridad: permite proteger los datos frente a la pérdida total o parcial de la información, por fallos del sistema o por accesos accidentales o intencionados de los mismos.

Sistema gestor de bases de datos (SGBD): es el software que ayuda a los usuarios a procesar, describir, gestionar y acceder los datos almacenados en una base de datos.

Software: es una secuencia de instrucciones que son interpretadas y/o ejecutadas para la gestión, redireccionamiento o modificación de un dato/información o suceso, funciona como intermediario entre el usuario y la información almacenada.

SQL: (Structured Query Language) el lenguaje de consulta estructurado es un estándar informático utilizado para definir, modificar, gestionar datos y controlar como se realizan cambios en las bases de datos.

T

Tabla: Unidad básica de almacenamiento a nivel lógico y se compone de filas y columnas, las tablas y columnas se identifican por su nombre, pero las filas se identifican por su contenido.

TCP/IP: grupo de protocolos diseñado para la comunicación entre computadoras suministrando, a su vez, servicio de red como: registro de entrada remoto, transferencia remota de archivos, correo electrónico, etc.

V

Vista: Es una representación de datos provenientes de una o más tablas, Consiste en asignar un nombre a la salida de una consulta y utilizarla como si de una tabla almacenada se tratase.



BIBLIOGRAFÍA

Bibliografía

A

Acofi, (2000), (Asociación Colombiana de Facultades de Ingeniería), “Cds/isis Para Windows”, Editorial: Univ. Nacional de Colombia, ISBN: 978-84-0000-0288

Atelin, P., (2007), “TCP/IP y protocolos de Internet”, Primera edición, Editorial: ENI, ISBM: 978-2-7460-3599-7

B

Barceló O. & José M. & Íñigo G., (2004), “Redes de computadores”, Primera edición, Editorial UOC, ISBN: 84-97-88-117-6

C

Cantone, D., (2006), “La biblia del programador, implementación y debugging”, Editorial: Gradi S.A., 1a., Buenos Aires, ISBN: 978-987-22995-7-6

Camps, R., (2005), “Introducción a las bases de datos”, Editorial: OUC, Primera edición, ISBN: 84-8429-989-9

Campderrich, F. B., (2003), “Ingeniería del Software”, Primera edición, Editorial UOC, ISBN: 84-8318-997-6

Coronel, C. & Rob, P., (2004), “Sistemas de bases de datos: diseño, implementación y administración”, 5ª. Edición, Thomson Editoriales S.A de C.V.

Cobo Y. A., (2007), “Diseño y programación de bases de datos”, Editorial: Visión Libros, ISBN: 978-84-9821-459-8

Colobran, H. M., (2008), “Administración de sistemas operativos en red”, Editorial: UOC, ISBN: 978-84-9788-760-1

Covadonga F. B., (1987), “El Modelo relacional de datos, de los fundamentos a los modelos deductivos”, Ediciones Díaz de Santos, S.A., ISBN: 8-4862-5161-3

D

Date, C.J., (2001), “Introducción a los sistemas de bases de datos”, 7ª. Edición: Pearson Educación de México, S.A. de C.V., ISBN 968-444-419-2

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Dordoigne, J., (2011), "Redes Informáticas: Nociones fundamentales", Editorial: ENI, ISBN: 978-2-7460-6748-6

Dewit, O., (2010), "ASP.NET: programación web con Visual Studio y Web Matrix", Editorial: ENI, ISBN: 2-7460-1978-7

E

Escofet C.M., (2009), "El lenguaje SQL", editorial UOC, ISBN 8-4842-9991-0.

G

Gabillaud, J., (2003), "Oracle 9i SQL, PL/SQL, SQL*Plus", Ediciones ENI, 2.edición, ISBN: 2-7460-1976-0

Gabillaud J., (2009), "SQL Server 2008: SQL, transact SQL, diseño y creación de una base de datos", Ediciones ENI, 1.edición, ISBN: 978-2-7460-4198-1

Gabillaud J., (2001) "SQL Server 2000: SQL, Implementación", Ediciones ENI, 1.edición, ISBN: 9782746013889

Gabillaud, J. "SQL Server 2008", (2010), Ediciones ENI, primera edición, ISBN: 978-2-7460-4704-4

Gómez de Silva G. A. & Ania B, A., (2008) "Introducción a la Computación", Coordinadores Editoriales, ISBN-13:978-970-686-768-1

Groussard, T., (2009), "Visual Basic.NET(VB.NET) Programe con Visual Studio 2008", Ediciones: ENI; ISBN: 978-2-7460-4505-7

Groussard, T., (2010), "Java Enterprise Edition: Desarrollo de aplicaciones web con JEE 6", Editorial: ENI; ISBN: 978-2-7460-5413-4

H

Highsmith, J., (2002), "Software Development Ecosystems", first edition, Pearson Education, Inc., ISBN: 0-201-76043-6

Herrera, P., (2003), "Tecnologías y redes de transmisión de datos", Editorial: LIMUSA, ISBN: 968-18-6383-6

K

Kroenke D. M., (2003) Procesamiento de bases de datos: Fundamentos, diseño e implementación; Editorial Pearson Prentice Hall, México; ISBN: 970-26-0325-0

Konntz, H., (1998), "Administración, un enfoque global", editorial: Mc-GRAW HILL, ISBN: 970-10-2036-7.

L

León, S. G., (1996), "Ingeniería de Sistemas del Software", editorial: ISDEFE, ISBN: 84-89338-10-8

S. Blanchard, B., (1995), "Ingeniería de sistemas", editorial: ISDEFE, ISBN: 84-68338-00-0.

López H. & Montero, A. & Martín, R. & De Pablos C. & Izquierdo V. , Nájera, J., (2000), "Informática aplicada a la gestión de empresas", Editorial: ESIC, ISBN: 84-7356-232-1

M

Mathon, P., (2001), "TCP/IP entorno Windows 2000", Editorial: ENI, ISBN: 2-7460-1263-4

Miñana, C. , (2010), "Servicios de red, una visión practica", segunda edición, Editorial: Lulu.com, ISBN: 978-1-4452-7134-7

Mora M. & Zorrilla P. & Joaquín D., (2003), "Iniciación a las Bases de Datos con Access 2002", Ediciones Díaz de Santos S.A., Madrid, ISBN: 84-7978-592-6

Martínez, N., Gómez G. A., (1997) Los sistemas de información en la empresa; Editorial: Servicio de publicaciones de la universidad de Oviedo; ISBN: 84-8317-036-1

Moliner F., (2005), Informáticos Generalitat Valenciana. Grupos A Y B Temario Bloque Específico; Editorial: MAD, S.L.; ISBN: 84-665-1829-0

Maciá P., (2010), "Administración de servicios de internet", editorial: Textos Docentes, primera edición, ISBN: 978-84-7908-989-4

N

Nevado C. M.V., (2010), Introducción a Las Bases de Datos Relacionales; Editorial: Visión Libros; ISBN: 978-84-9886-809-8

O

Osorio R., (2008), "Bases de datos relacionales teoría y práctica, primera edición", editorial: ITM, ISBN: 978-958-8351-42-1

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

Osorio R., (2008), "Base de datos relacionales", fondo editorial Instituto Tecnológico Metropolitano ITM, ISBN: 978-958-8351-42-1

Osorio R., (2008), "Bases de Datos Relacionales teoría y práctica", Editorial: Instituto Tecnológico Metropolitano, primera edición, ISBN: 978-958-8351-42-1

P

Pressman, R., (2005), "Ingeniería del software un enfoque práctico", 6ª Editorial: Mc Graw Hill, ISBN: 970-10-5473-3

Pérez C. R., (2003), Oracle 9i Administración y Análisis de Bases de Datos; 1ª. Edición; Alfaomega Grupo Editor S.A. de C.V; ISBN: 970-15-0845-9.

Paré R., C & Casillas S & Costal. L & Costa D. & Ginestà M. & ,Escofet C., Pérez M. , (2005) "Software libre: Bases de Datos", 1ª. Edición, ISBN: 84-9788-269-5

Piattini V. M., Castaño A, Marcos M. E., (2000), "Diseño de bases de datos relacionales"; 1ª. Edición Alfaomega Grupo Editor S.A. de C.V.

Purser, M., (1990), "Redes de telecomunicación y ordenadores", Editorial: Díaz de Santos, ISBN: 84-87189-46-6

Payne, C., (2002), "Aprendiendo ASP.NET en 21 lecciones avanzadas", Editorial: Pearson Educación; ISBN: 978-97-0260-340-5

Pérez M., M., (2011), "SQL Server 2008 R2 Motor de base de datos y administración", primera edición; ISBN: 978-84-938312-3-3

Q

Quesnel, J., (2012), "Entender ITIL 2011", Editorial: ENI, ISBN: 978-2-7460-7618-1

Quero C. E.: García R. A., (2007), "Mantenimiento de portales de la Información: explotación de sistemas informáticos", Primera Edición, Editorial: Thomson Paraninfo, ISBN: 978-84-9732-504-2

R

Ramez E. & Shamkant B. N., (2000), "Sistemas de Bases de Datos - Conceptos Fundamental", 2a. edición; Addison-Wesley Iberoamericana; ISBN: 0-201-65370-2

Rigaux P. & Scholl M. & Voisard A., (2002), "Spatial Databases with application to gis", Morgan kaufmann publishers, ISBN: 1-55860-588-6

S

Sommerville, I., (2005), "Ingeniería de Software", 7a. edición, editorial: Pearson, ISBN: 84-7829-074-5

Silberschatz A., Korth, H. F., Sudarshan S., Henry F., (2002), "Fundamentos de bases de datos.", 4ª. Edición, McGRAW-HILL/ Interamericana de España, S. A. U., ISBN: 0-07-228363-7

Sznajdleder, P. A., (2013), "Java a fondo: estudio del lenguaje y desarrollo de aplicaciones", Segunda edición, ISBN 978-987-1609-36-9

T

Taboada G. J.A & Cotos Y, J.M. (2005), "Sistemas de Información Medioambiental", 1ª. Edición; Editorial: Getbiblo, S.L., ISBN: 84-9745-056-6

Toro López, Francisco J., (2013), "Administración de proyectos de informática", primera edición, editorial ECOE, ISBN: 978-958-648-816-7

Tanenbaum, A. S., (2003), "Redes de computadoras", cuarta edición; Editorial: Prentice Hall, ISBN: 0-13-2066102-3

V

Valdivia Miranda, Carlos, (2014), "Sistemas informáticos y redes locales", primera Edición; Editorial Paraninfo, ISBN: 978-84-9732-449-6.



MESOGRAFÍA

Y

HEMEROGRAFÍA

Mesografía y hemerografía

HYPATIA, Revista de divulgación científico-tecnológica del gobierno de Morelos. Número 29, (2010), (s/p) Consultada el día domingo 26 de septiembre de 2010 [En Línea] Disponible en:

http://hypatia.morelos.gob.mx/index.php?option=com_content&task=view&id=508&Itemid=438

MASTERMAGAZINE, (2010), (s/p) Consultada el domingo 26 de septiembre de 2010 [En Línea] Disponible en:

<http://www.mastermagazine.info/termino/4382.php>

DICCIONARIO DE LA LENGUA ESPAÑOLA, (2010), (s/p) Consultada el martes 28 de septiembre de 2010 [En Línea] Disponible en:

<http://buscon.rae.es/drae/>

RED CIENTÍFICA, (2010), (s/p) Consultada el domingo 3 de octubre de 2010 [En Línea] Disponible en:

<http://www.redcientifica.com/doc/doc200204150002.html>

RED CIENTÍFICA, (2010), (s/p) Consultada el domingo 3 de octubre de 2010 [En Línea] Disponible en:

<http://www.redcientifica.com/doc/doc199904020005.html>

[Revista] PC Actual, Guía Práctica número 210, editorial: RBA EDIPRESSE

Portalprogramas.com, (2010), (s/p) Valor del software libre en la sociedad, consultado: 1 de junio de 2011 [En Línea] Disponible en:

<http://www.portalprogramas.com/software-libre/informe>

[Revista] PC Actual, Guía Práctica número 210, editorial: RBA EDIPRESSE

Portalprogramas.com, (2010), (s/p) Valor del software libre en la sociedad, consultado: 1 de junio de 2011 [En Línea] Disponible en:

<http://www.portalprogramas.com/ayuda/c16/software-libre>

utvm.edu.mx, (2014), (s/p), consultado: 16 de marzo de 2014 [En Línea] Disponible en:

<http://www.utvm.edu.mx/OrganoInformativo/orgJul07/RUP.htm>

msdn.microsoft.com, (2012), (s/p), Información general acerca del trabajo con objetos de base de datos, consultado: 11 de septiembre de 2012 [En Línea] Disponible en:

[http://msdn.microsoft.com/es-es/library/aa833299\(v=vs.80\)](http://msdn.microsoft.com/es-es/library/aa833299(v=vs.80))

msdn.microsoft.com, (2012), (s/p), Información general acerca del trabajo con objetos de

SISTEMA DE LÍNEA DE MANDO PARA LA ORGANIZACIÓN DE PERSONAL DE LA PROCURADURÍA FEDERAL DE PROTECCIÓN AL AMBIENTE.

base de datos, consultado: 11 de septiembre de 2012 [En Línea] Disponible en:
<http://msdn.microsoft.com/es-es/library/ms176130.aspx>

msdn.microsoft.com, (2012), (s/p), Información general acerca del trabajo con objetos de base de datos, consultado: 11 de septiembre de 2012 [En Línea] Disponible en:
[http://msdn.microsoft.com/es-es/library/aa290273\(v=VS.71\).aspx](http://msdn.microsoft.com/es-es/library/aa290273(v=VS.71).aspx)

msdn.microsoft.com, (2012), (s/p), Información general acerca del trabajo con objetos de base de datos, Consultado: 20 de enero de 2014:
<http://msdn.microsoft.com/es-es/library/bb510741.aspx>

msdn.microsoft.com, (2012), (s/p), Información general acerca del trabajo con objetos de base de datos, Consultado: 20 de enero de 2014:
[http://msdn.microsoft.com/es-es/library/ms189799\(v=sql.120\).aspx](http://msdn.microsoft.com/es-es/library/ms189799(v=sql.120).aspx)

aulacli.es, (2014), (s/p), consultado: 5 de mayo [En Línea] Disponible en:
http://www.aulacli.es/sqlserver/t_9_1.htm#ap_09_02

sqltransact2.blogspot.mx/, (2014), (s/p), consultado: 5 de mayo [En Línea] Disponible en:
<http://sqltransact2.blogspot.mx/2013/12/cursores-en-transact-sql.html>

microalcarria.com, (2014), (s/p), consultado: 16 abril en la página [En Línea] Disponible en:
http://www.microalcarria.com/descargas/documentos/Linux/seguridad/seguridad_redes_datos.pdf

dynamics.unam.edu, (2014), (s/p), consultado: 20 de mayo [En Línea] Disponible en:
https://www.google.com.mx/url?sa=t&rct=j&q=&esrc=s&source=web&cd=5&ved=0CE0QFjAE&url=http%3A%2F%2Fwww.dynamics.unam.edu%2Fusers%2Fcperez%2FREDES%2520DE%2520COMPUTADORAS%2F1.%2520Introduccion.doc&ei=ATt9U5HqNsHd8AGHjYGICQ&usg=AFQjCNHVpWbSb4LwWongUF2I2WB_kCQhcA&sig2=FQBK3b764tR5cQctJCgqhg&bvm=bv.67229260,d.b2U&cad=rja

uca.edu.sv, (2014), (s/p), consultado: 20 de mayo [En Línea] Disponible en:
<http://www.uca.edu.sv/facultad/clases/ing/m190068/>

profesores.fi-b.unam.mx, (2014), (s/p), consultado: 20 de mayo, [En Línea] Disponible en:
[http://profesores.fi-b.unam.mx/victor/CCNA/Productos/Notas%20de%20Curso/Manual%20de%20la%20Asignatura%20de%20Redes%20de%20Datos%20I%20y%20II%20%20\(avance%2050%25\).pdf](http://profesores.fi-b.unam.mx/victor/CCNA/Productos/Notas%20de%20Curso/Manual%20de%20la%20Asignatura%20de%20Redes%20de%20Datos%20I%20y%20II%20%20(avance%2050%25).pdf)

Portalprogramas.com, (2010), Valor del software libre en la sociedad, consultado: 1 de junio de 2011 [En Línea] Disponible en:
<http://www.portalprogramas.com/software-libre/informe>

Eclipse.org, (2014), ECLIPSE, (s/p) consultado: 20 de marzo de 2014, [En Línea] Disponible en:

<https://www.eclipse.org/org/>

Tomcat.apache.org, (2014), (s/p), APACHE TOMCAT consultado: 20 de marzo de 2014 [En Línea] Disponible en:

<https://tomcat.apache.org/index.html>

httpd.apache.org, (2014) APACHE (s/p), consultado: 20 de marzo de 2014, [En Línea] Disponible en:

http://httpd.apache.org/ABOUT_APACHE.html

<http://wiki.eclipse.org>, (2014) ECLIPSE, (s/p) consultado 03 de abril de 2014, [En Línea] Disponible en:

http://wiki.eclipse.org/FAQ_What_is_Eclipse%3F

java.com, (2014) JAVA (s/p) consultado: 20 de marzo de 2014, [En Línea] Disponible en:

http://www.java.com/es/download/faq/whatis_java.xml

java.com, (2014) JAVA (s/p) consultado: 20 de marzo de 2014, [En Línea] Disponible en:

http://www.java.com/es/download/whatis_java.jsp