



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

*SISTEMA DE CONTROL DE SERVICIOS DE CÓMPUTO PARA EL
INSTITUTO DE INGENIERÍA*

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A N:

APRESA MONZÓN ROSAURA
GONZÁLEZ JIMÉNEZ LILIANA
RAMÍREZ Y RAMÍREZ RAÚL TRINIDAD

DIRECTOR DE TESIS:
ING. MARCO AMBRIZ MAGUEY



MÉXICO D.F., NOVIEMBRE 2011



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Gracias

A mi mamá Carolina Monzón Torres. Porque has estado a mi lado en todo momento, brindándome consejos, apoyo e infinito amor en las buenas y en las malas. Porque comprendiste mi ausencia y malos ratos que tuve en muchas ocasiones. Pero sobre todo porque me has dado una gran lección de vida. Te admiro por ser una gran madre y una gran mujer.

A mi papá Isaias Apresa González. Porque me diste la oportunidad de tener una carrera. Tus ganas de ser alguien en la vida, luchar siempre por crecer y lograr lo que te propones, han sido mi ejemplo para llegar hasta aquí y cumplir esta meta.

A mis hermanos. Caro. Por ser mi gran amiga y confidente. Porque has sabido sobrellevar mis cambios de ánimo de la mejor manera. En los momentos de alegría y cuando existieron lagrimas, tu compañía me dio fuerza para seguir. Gil. Tú motivación ha sido fundamental en mi vida, con tus atinados consejos para no ahogarme en un vaso de agua aprendí que las decisiones, algunas veces buenas y otras malas, siempre son experiencias. Jaime. Porque me has enseñado a disfrutar lo mejor de la vida, seguir creciendo y concluir cada ciclo a lo largo de mi camino. A los tres gracias por su motivación y apoyo que me brindan de manera incondicional.

A mis tíos. Herminio y María Elena. Por motivarme a seguir adelante y brindar apoyo a toda la familia en tantas etapas difíciles que hemos pasado.

A mis padrinos Lourdes y Adrian. Por estar al pendiente a lo largo de mi vida, en mis logros y mis fracasos, siempre recibiendo motivación para salir adelante con la familia.

A Raúl Trinidad Ramírez y Ramírez. Por ser siempre mi amigo en cualquier situación. Por entregarte en todo momento para concluir de manera exitosa esta tesis. Por entenderme y aceptarme con mis errores y defectos. Por apoyarme, guiarme y aconsejarme para continuar y seguir con mi camino. Por dejarme ser parte de tu vida.

A Liliana. Por tu compañía, amistad, entrega y esfuerzo realizado a lo largo del desarrollo de este proyecto.

A mis mejores amigas. Marcela, Isabel, Elda, Alejandra, Daniela y Griselda. Por su amistad sincera. Por darme el ejemplo y el apoyo para culminar este proyecto. Por dejarme compartir con ustedes grandes alegrías y también tristezas. Por escucharme y orientarme en muchas ocasiones que me sentía derrotada. Siempre las llevo en mi corazón.

A Elizabeth Hernández Moreno. Porque a pesar de no conocerme, confiaste en mí y compartiste una gran experiencia, que me dio la oportunidad de abrir los ojos, valorar la vida, ser feliz y luchar por concluir esta etapa.

Al Ing. Marco Ambríz Maguey. Por la paciencia, consejos, apoyo y dirección en el desarrollo de la tesis.

A Jonathan Velázquez, Mauricio Velázquez, Genaro González y Julio de León. Por la confianza, apoyo y conocimientos compartidos a lo largo de la estancia en el Instituto de Ingeniería y durante el desarrollo de la tesis.

A la Universidad Nacional Autónoma de México, Facultad de Ingeniería, Instituto de Ingeniería y profesores. Por darme la oportunidad de culminar una carrera profesional. Por transmitirme sus conocimientos y enseñanzas que se ven reflejados en la entrega de este proyecto de tesis y en mi desempeño como profesionista.

Finalmente, gracias a aquellos familiares, amigos, amigas y compañeros de trabajo, que no es posible mencionarlos a todos, pero también contribuyeron con su granito de arena, estando siempre pendiente de la realización de la tesis.

*“Por mi raza hablará el espíritu”
Rosaura Apresa Monzón*

AGRADEZCO

A Dios por haberme dado el don de la vida, así como la bendición de mi familia y porque a lo largo de mi andar me ha demostrado que está junto a mí, iluminando mi camino tanto en las penas como en las alegrías y sobre todo por darme la oportunidad de llegar a concluir este proyecto fundamental en mi vida.

A mis padres Trino y Carmelita, que gracias a su esfuerzo y dedicación, lograron sacar adelante a mi familia y lo siguen haciendo a pesar de las dificultades que se presenten, por su guía, su amor y amistad es por lo que estoy terminando esta etapa importante en mi vida y con su ejemplo y compañía muchas más.

A mi hermana Mary Carmen y su novio Luis, que me apoyan incondicionalmente y que siempre están para cuando se les necesitan, a ti Carmelita por estar siempre a mi lado y pendiente de mí toda la vida y aunque no pensamos igual somos lo mismo porque venimos del mismo lugar, a Luis por estar con nosotros en las buenas y en la malas y pararse junto a nosotros en los momentos más difíciles.

A mis padrinos Silverio y Estela, y a su familia, por estar al pendiente de ésta y todas las aspiraciones en mí vida y cuyos consejos me dieron la fuerza para seguir adelante en todo momento y por estar a mi lado cuando más lo necesitamos.

A mi tío Juan y a toda su familia, por el apoyo e interés recibido en la realización de esta tesis, por sus consejos, por su compañía en los buenos y malos momentos.

A Rosaura Apresa, por todo lo que hemos vivido, por su amistad, sus consejos, su paciencia y su apoyo en todos los proyectos que hemos compartido, que además de ser compañera es amiga y pieza fundamental para la realización de este objetivo, esperando poder llegar a compartir muchos más. Agradezco a dios el haberme dado la oportunidad de tener un camino en común hasta este punto y le agradeceré infinitamente si me lo sigue permitiendo. Rosy, siempre estás en mi corazón.

A mi hermano del alma, Anyhelo Moreno Castro, que por su gran esfuerzo, compañía y apoyo, he podido sacar adelante este trabajo, gracias a su participación activa dentro de mi familia y el apoyo que le brinda a mi mamá. Nunca podremos terminar de agradeceréte.

A las dos familias Ramírez, por el apoyo recibido su cariño, amor y compañía.

A la Facultad de Ingeniería y sus profesores, por haberme formado profesionalmente y haberme transmitido sus conocimientos, para poder ser un miembro útil en la sociedad mexicana, así como las ganas de seguir adelante en todo momento.

Al Instituto de Ingeniería, por habernos brindado sus instalaciones y la oportunidad de realizar este proyecto que nos abren muchas posibilidades en la vida laboral.

Al Ing. Marco Ambríz Maguey, por el apoyo brindado, la paciencia y la confianza en nosotros para la realización de este trabajo.

A mis compañeros y amigos de la Coordinación de Sistemas de Cómputo, por habernos apoyado tanto en la realización de este compromiso como en nuestro paso por el Instituto, su paciencia y su entrega cuando se les requería.

A Jonathan Velázquez, Mauricio Velázquez, Genaro González y Julio de León, por su apoyo incondicional en la realización de esta tesis, por su ayuda que fue fundamental y sobre todo la amistad que nos brindaron al ser parte de un mismo grupo de trabajo.

A mis compañeras de proyecto Rosaura y Liliana, que dieron su mejor esfuerzo para poder complementarnos y así lograr un objetivo compartido que se cumplió, espero nos sirva en todos nuestros demás objetivos y podamos compartir más proyectos.

También quiero agradecer este trabajo a todas las personas involucradas en él, que de alguna manera su apoyo, compañía o paciencia, dieron como fruto la culminación de este trabajo.

Finalmente quiero dedicar el esfuerzo realizado y el proyecto en general a una persona que no solo compartió los lazos que nos une como familia, como un padre y un hijo, si no como un verdadero amigo que en las buenas y en las malas estuvo y estará a mi lado, que lo recordare toda la vida, porque esa vida se la debo en todos los sentidos, a él, mi padre, José Trinidad Ramírez Godínez, mi mejor amigo, mi maestro mi compañero de aventuras y cómplice en muchas ocasiones, una persona que amó a su familia en todo momento y de la que agradezco mucho haber sido parte, una vida ejemplar. Donde quiera que estés, esto es por ti, para ti y de ti.

Ramírez y Ramírez Raúl Trinidad.
“Por mi raza hablará el espíritu”

La presente Tesis es resultado de la participación de varias personas que leyendo, opinando teniendo paciencia, dando ánimo, acompañando en momentos de crisis y en los momentos de felicidad me apoyaron y brindaron confianza de forma incondicional y a quienes quiero expresar mi gratitud.

En primer lugar quiero agradecer a la Universidad Nacional Autónoma de México y al Instituto de Ingeniería porque gracias a éstas grandes instituciones me he podido desarrollar profesionalmente.

Un sincero agradecimiento al Ing. Marco Ambríz Maguey por todo el tiempo, apoyo, ánimo, paciencia, sugerencias e ideas en la dirección de este trabajo.

Gracias a Rosaura y a Raúl que me permitieron trabajar con ustedes en este proyecto, al apoyo personal y profesional que me han ofrecido, gracias.

A mis compañeros y amigos con los cuales compartí no sólo horas de trabajo y de los cuales aprendí mucho; Jonathán, Mauricio, Genaro, Julio, gracias por su amistad y ayuda que no tiene precio.

Todo ésto nunca hubiera sido posible sin el apoyo y esfuerzo de mis padres Otilia Jiménez Nava y Pedro González Flores, gracias por su cariño y comprensión sin condiciones ni medida. Gracias por brindarme una familia hermosa, por la educación que con su ejemplo me brindaron, por ser luchadores y sacarnos adelante a mí y a mis hermanos Fabiola, Pedro, Gaby y en especial a Lalito. Gracias a mis hermanos por el apoyo y por ser en gran medida mi motor para ser mejor, porque todos siempre me han demostrado que no hay barreras que no podamos vencer en lo que nos proponamos. Dios los bendiga, les de salud y mucha vida para poder retribuirles un poco de lo que me ha dado.

Gracias a ti Jonathan que desde un principio hasta el día de hoy sigues dándome ánimo para emprender y terminar cada una de las metas que me propongo, por tus comentarios, sugerencias y opiniones, por ser mi mejor amigo y enseñarme que el verdadero amor es el deseo de crecer juntos, porque tus triunfos y derrotas también son las mías, es para ti este triunfo. Te amo.

Liliana González Jiménez

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1. ANTECEDENTES	3
1.1 INSTITUTO DE INGENIERÍA	3
1.1.1 Misión y visión.....	5
1.2 SISTEMAS DE CÓMPUTO	6
1.2.1 Bases de datos.....	6
1.2.2 Capacitación informática.....	7
1.2.3 Cómputo personal y periféricos.....	7
1.2.4 Portales y páginas web	7
1.2.5 Redes y telecomunicaciones.....	8
1.2.6 Sistemas Linux-Unix	8
1.2.7 Sistemas Microsoft Windows.....	9
CAPÍTULO 2. ANÁLISIS DEL SISTEMA	11
2.1 DESCRIPCIÓN DEL PROBLEMA.....	11
2.2 REQUERIMIENTOS	11
2.2.1 Módulo para la recepción, consulta y confirmación de solicitudes recibidas	12
2.2.2 Módulo de administración.....	12
2.2.3 Módulo de estadísticas y reportes.....	14
2.2.4 Módulo de registro de personal y equipo externo	14
2.3 PROPUESTA DE SOLUCIÓN	14
2.4 METODOLOGÍA USADA	15
2.5 PLANEACIÓN Y ESTIMACIÓN DE TIEMPOS Y COSTOS.....	16
2.6 CALIDAD DE SOFTWARE.....	19
CAPÍTULO 3. ANÁLISIS DE RECURSOS DISPONIBLES.....	25
3.1 RECURSOS DE SOFTWARE.....	25
3.1.1 Manejador de base de datos - Microsoft SQL Server.....	25
3.1.2 Servidor – IIS	28
3.1.3 Lenguaje de programación - Plataforma .NET.....	29
3.2 RECURSOS DE HARDWARE	30
3.2.1 Hardware de desarrollo.....	31
3.2.2 Hardware de producción.....	31
3.2.3 Hardware y software recomendado para los usuarios	31
3.3 RECURSOS HUMANOS.....	32

CAPÍTULO 4. DISEÑO DEL SISTEMA.....	33
4.1 DISEÑO DE LA BASE DE DATOS	33
4.1.1 Modelo entidad relación	38
4.2 DISEÑO DE CASOS DE USO	50
4.2.1 Casos de uso para administrador 1	51
4.2.2 Casos de uso para administrador 2	54
4.2.3 Casos de uso para usuarios	57
4.2.4 Diseño de la interfaz	61
CAPÍTULO 5. DESARROLLO DEL SISTEMA	65
5.1 IMPLEMENTACIÓN DE BASE DE DATOS	65
5.2 PROGRAMACIÓN E INTERFAZ	68
5.2.1 Recepción, consulta y confirmación de solicitudes recibidas	72
5.2.2 Administración	73
5.2.3 Estadísticas y reportes	76
5.2.4 Registro de personal y equipo externo	76
CAPÍTULO 6.PRUEBAS Y LIBERACIÓN DEL SISTEMA	79
6.1 TIPOS DE PRUEBAS	79
6.2 DOCUMENTACIÓN DE PRUEBAS.....	80
6.3 NAVEGADORES MÁS UTILIZADOS	88
6.4 LIBERACIÓN	89
CONCLUSIONES.....	95
BIBLIOGRAFÍA.....	97
ANEXO A	99
ANEXO B	113
GLOSARIO.....	117

INTRODUCCIÓN

El objetivo de esta tesis es describir los diversos pasos que se llevaron a cabo para el desarrollo del Sistema de Control de Servicios de Cómputo para el Instituto de Ingeniería(II), el cual tiene la finalidad de ofrecer a su comunidad el registro, administración y control de solicitudes de servicios brindados por el personal de las distintas áreas de la Coordinación de Sistemas de Cómputo(CSC), tales como son: bases de datos, cómputo personal y periféricos, redes y telecomunicaciones, portales y páginas Web, Windows, Unix y capacitación informática. Así como, brindar estadísticas y reportes para realizar análisis de servicios e informes institucionales.

Para brindar un servicio de calidad ha sido preocupación del II y de la CSC en particular, utilizar las tecnologías web y herramientas de cómputo de la más avanzada tecnología en beneficio de la institución.

El II es el centro de investigación en ingeniería más importante del país, su personal trabaja en proyectos de investigación vinculados con los sectores público y privado, se forman recursos humanos del más alto nivel y se difunden los resultados de las investigaciones.

La CSC está ubicada en el edificio 12 del II, ésta lleva a cabo la planeación, operación y mantenimiento de la infraestructura de cómputo y telecomunicaciones, la cual consta de computadoras personales, estaciones de trabajo, servidores, equipos de red de datos, teléfonos, equipo de videoconferencia, sistemas de información y bases de datos, entre otros. A través de esta infraestructura y mediante cuentas de usuario a la comunidad académica, administrativa y estudiantes del II; la CSC brinda servicios de soporte técnico, conexión remota SSH, correo electrónico, divulgación de información a través de páginas web, desarrollo de sistemas, entre otros.

En el capítulo 1, se hace una breve reseña del II y su organización. Así como, una descripción de las diferentes áreas de la CSC y sus correspondientes actividades que deben ser controladas y administradas por el sistema.

En el capítulo 2 se describe la forma de llevar a cabo el análisis del sistema, fundamental para su desarrollo. Iniciando con la descripción del problema y requerimientos del sistema para proponer una solución bajo un método de desarrollo y calidad adecuado. Con ello poder desarrollar el Sistema de Control de Servicios, siempre tomando en cuenta los posibles costos que el desarrollo de software implica.

En el capítulo 3 nos dedicamos a recabar y analizar la información de los diferentes recursos con los que cuenta el II para el desarrollo de software, en específico en la CSC donde se genera la mayoría del software para el Instituto. Estos recursos los clasificamos en: software, hardware, humanos y otros.

En el capítulo 4 abordamos el diseño de la base de datos mostrando el diagrama entidad-relación, así como la documentación de los casos de uso que fueron necesarios para identificar el comportamiento del sistema y el diseño de la interfaz por tipo de usuario.

En el capítulo 5 explicamos la forma de llevar a cabo la implementación del sistema, que abarca, la creación de la base de datos, la programación e interfaz de los módulos que se describieron, analizaron y plantearon en los capítulos anteriores, además de mostrar algunas de las pantallas principales para su ejemplificación.

En el capítulo 6 describimos la manera en que se establecieron y realizaron las pruebas del sistema. Se documentaron algunos casos, para ejemplificar las diferentes pruebas y garantizar el buen funcionamiento y respuesta dada la demanda de los usuarios. Así mismo, se describen los pasos necesarios para la puesta en marcha del sistema, tales como: configuración del servidor, permisos de acceso, alojamiento, etc.

CAPÍTULO 1

ANTECEDENTES

1.1 INSTITUTO DE INGENIERÍA

El Instituto de Ingeniería de la Universidad Nacional Autónoma de México (UNAM) es una comunidad integrada por 94 investigadores, 101 académicos, una población oscilante de más de 500 becarios que realizan trabajos de tesis de licenciatura, maestría y doctorado y 140 personas del área administrativa, por lo que su población actual es de alrededor de 835 personas. Sus instalaciones ocupan 15 edificios en la zona de Ciudad Universitaria (CU), en la Ciudad de México, con una extensión de 26,020 metros cuadrados construidos entre laboratorios, cubículos, áreas comunes y un auditorio. Además posee 2 unidades foráneas en Juriquilla, Querétaro y Sisal, Yucatán.

Tradicionalmente, en el II se han desarrollado actividades de investigación básica y aplicada; estudios de problemas específicos y asesorías; y servicios tecnológicos especializados en diversas áreas de la ingeniería. En particular, han sobresalido las actividades realizadas en ingeniería civil y en ingeniería ambiental; pero también se tienen avances importantes en algunas otras como: estructuras, hidráulica, sismos, geotecnia, tratamiento de aguas y tratamiento de suelos. Actualmente se han abordado nuevas áreas como infraestructura, vivienda sustentable, hospitales, energía, tecnologías de la información y transporte, que son vitales para México.

El II es una institución dinámica que ha cubierto con gran éxito las áreas de la ingeniería en las que ha participado y realizado aportaciones significativas en las actividades de desarrollo experimental e innovación tecnológica. Éstas han contribuido a solucionar problemas nacionales importantes con amplio beneficio social y frecuentemente, de interés más allá del de nuestro país. Así, los resultados de las actividades de desarrollo tecnológico e investigación del II tienen gran impacto en los sectores productivo, social y gubernamental.

Como consecuencia, algunos proyectos son financiados con recursos que la UNAM otorga y la mayor parte, mediante contratos de investigación con empresas o corporaciones solicitantes. Por ello, el prestigio del Instituto es ampliamente reconocido.

El II está organizado como se muestra en el organigrama (Figura 1.1), en el que podemos ver los cuatro niveles siguientes:

- Primer Nivel: Dirección, Consejo Interno, Comisión Dictaminadora, Comité Página Web, Comité de Administración y Planeación, Subcomité de Adquisiciones y Comité de Publicaciones.
- Segundo Nivel: Secretaría de Planeación y Desarrollo Académico, Secretaría Académica, Subdirección de Hidráulica y Ambiental, Subdirección de Estructuras y Geotecnia, Subdirección de Electromecánica, Secretaría Técnica, Unidad de Gestión de Convenios y Contratos y Secretaría Administrativa.
- Tercer Nivel: 14 Coordinaciones, Unidad de Servicios de Información, Unidad de Docencia y Formación de Recursos Humanos, Unidad de Patentes y Transferencia Tecnológica, Unidad de Promoción y Comunicación, Unidad de Informática y Control Estadístico, Unidad de Apoyo a Cuerpos Colegiados, Unidad de Vinculación, Unidad Académica Juriquilla, Unidad

Académica SISAL, Departamento de Contabilidad, Departamento de Bienes y Suministros, Departamento de Personal y Departamento de Presupuestos.

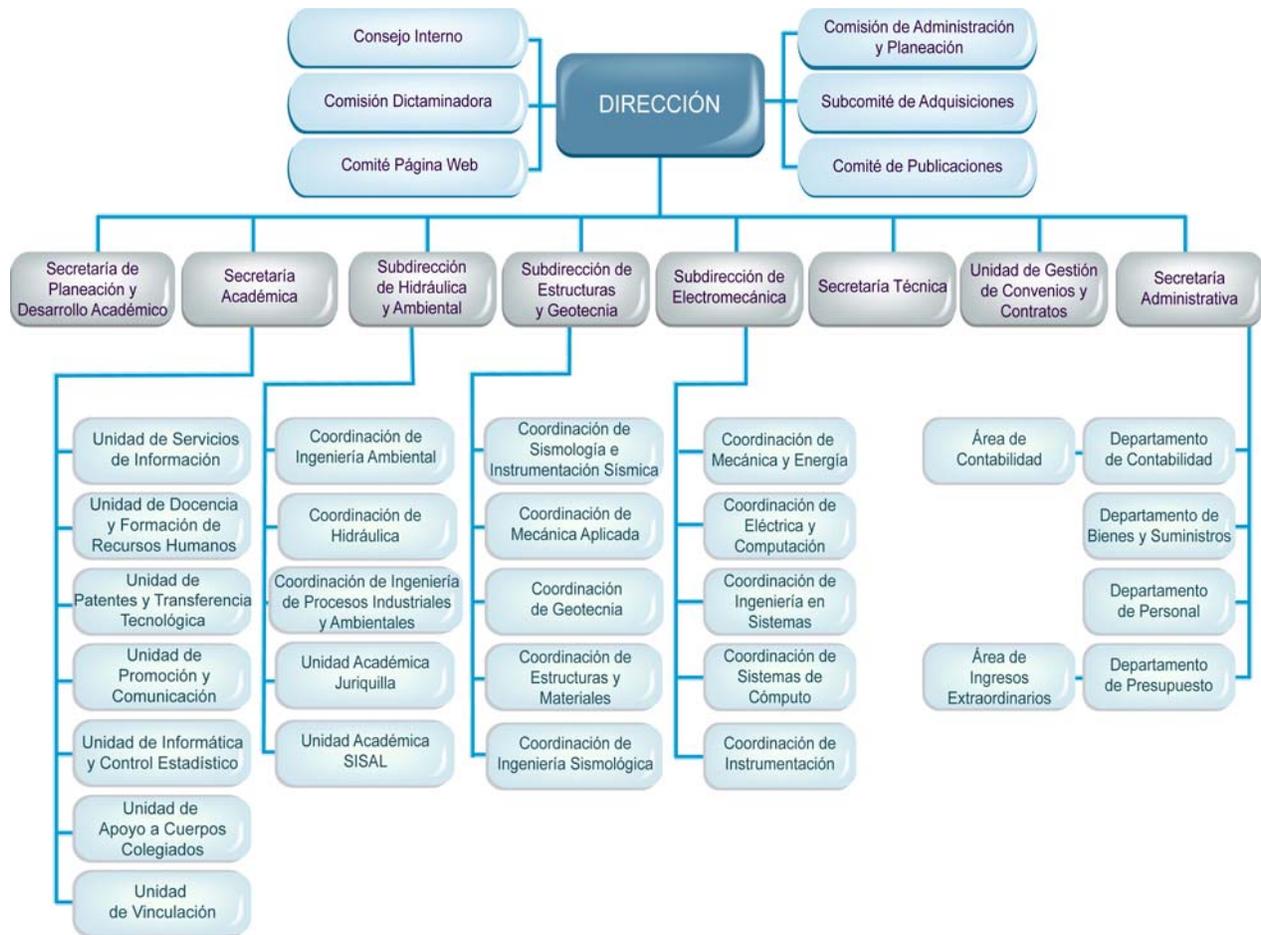


Figura 1.1 Organigrama

1.1.1 Misión y visión

La misión del Instituto de Ingeniería es contribuir al desarrollo del país y al bienestar de la sociedad a través de la investigación en ingeniería; la formación de recursos humanos y la vinculación con la sociedad. Para cumplir su misión, el II tiene las siguientes funciones:

- Realizar investigación fundamental y aplicada, preferentemente dirigida a la solución de problemas de interés nacional.
- Formar investigadores y personal especializado mediante el ejercicio de la investigación.
- Participar en las labores docentes de la UNAM y coadyuvar en la formación de profesores.
- Colaborar con otras dependencias de la UNAM.
- Difundir los resultados de las investigaciones.
- Transferir los resultados de la investigación mediante la vinculación con la sociedad.

En el desempeño de estas funciones el II puede colaborar con dependencias afines, culturales o científicas del país y del extranjero.

La visión del Instituto de Ingeniería es:

- Convertirse en el centro de investigación en ingeniería más relevante, de mayor calidad y pertinencia en México y América Latina.
- Ser generador de conocimientos de vanguardia útil para resolver los grandes problemas nacionales, tales como recursos hídricos, energía, transporte, comunicaciones y vivienda.
- Ser formador de nuevas generaciones de ingenieros de alta calidad y comprometidos con la ingeniería mexicana y el país.

1.2 SISTEMAS DE CÓMPUTO

La Coordinación de Sistemas de Cómputo se dedica a trabajar en los temas de cómputo, informática y telecomunicaciones para apoyar al personal académico en sus proyectos de investigación y propiciar que la convergencia y el uso de las tecnologías se asimilen y aprovechen como una herramienta de cambio y modernización; un agente de innovación y un ingrediente de productividad.

Se han realizado diferentes investigaciones relacionadas con los últimos avances en materia de desarrollo de sistemas informáticos basados en tecnologías de páginas web y motores de bases de datos SQL para ofrecer al personal los más avanzados sistemas de información académica y administrativa. Además de investigación sobre nuevos sistemas de seguridad informática para proteger la infraestructura actual del Instituto.

Adicionalmente, se realizan actividades de actualización y mantenimiento para conservar en óptimas condiciones la operación del hardware, software e infraestructura de cómputo a través de la contratación de servicios de mantenimiento externo con compañías privadas.

También la CSC ofrece asesoría y apoyo técnico en materia de cómputo al personal del II y formación de recursos humanos en el conocimiento y manejo de nuevas tecnologías.

1.2.1 Bases de datos

En el área de Bases de datos se analizan propuestas para la realización de nuevos sistemas de apoyo a la comunidad del II, que permitan realizar o complementar las tareas de investigación, administración o manejo de información de forma automática y reduciendo tiempos y costos. Encargándose de la planeación, recopilación de requerimientos, propuesta de diseño, desarrollo del sistema, pruebas y una vez que se entrega el sistema solicitado y los usuarios están de acuerdo, se realiza la liberación.

Así mismo se encarga de dar mantenimiento a los sistemas de información y soporte técnico a los usuarios. Algunos sistemas desarrollados en el área son los siguientes:

- SBD AII (Sistema de Base de Datos Académica).
- SICOE (Sistema de Control de Estudiantes).
- Sistema de Avisos.
- SREII (Sistema de Reservación de Espacios del Instituto de Ingeniería).
- Sistema de Publicaciones.
- Sistema de Base Mexicana de Sismos Fuertes.
- SRETI (Sistema de Reservación de Espacios de la Torre de Ingeniería).
- SAHII (Sistema de Archivos Históricos del Instituto de Ingeniería).
- Sistema Coordinador de Eventos.
- Directorio II.
- Sistema IIidentifica.
- Sistema de Administración de Publicaciones ShakeMaps.

Y actualmente se encuentran en desarrollo:

- Sistema de Registro de Solicitudes (Proyecto de ésta Tesis).
- Bono de Ingresos Extraordinarios.
- Aplicaciones Móviles.
- Consulta de Proyectos en el SIRF.

- Sistema de Control de Pantallas.
- Páginas para Congresos del II.

1.2.2 Capacitación informática

El objetivo fundamental es promover y difundir la cultura informática por medios electrónicos e impresos a todos los niveles jerárquicos del II, a través de capacitación asesoría y apoyo técnico a su personal.

Algunas funciones básicas del área son:

- Promover, coordinar y elaborar material didáctico para los diversos cursos de capacitación en informática.
- Fungir como vínculo con aquellas entidades externas que brinden cursos de capacitación en cómputo.
- Impartir cursos de computación a niveles básico, intermedio y avanzado.
- Brindar apoyo logístico durante la preparación y el desarrollo de cursos.
- Brindar asesorías relacionadas con la cultura informática.
- Detectar necesidades de capacitación informática.
- Formar a futuros instructores en cómputo.

1.2.3 Cómputo personal y periféricos

Es el área donde se atienden servicios a petición de los usuarios acerca de actualización y mantenimiento del hardware y software disponible en el II, así como del mantenimiento de toda la infraestructura relacionada con cómputo personal.

Por ejemplo:

- Corrección y prevención de problemas de seguridad ocasionados por Spyware y códigos maliciosos.
- Mantenimiento correctivo y preventivo de servicio externo.
- Instalación/desinstalación de SW y HW.
- Configuración de equipos nuevos.

Es con ésta área donde el sistema y los usuarios tendrán mayor interacción con el sistema y viceversa debido a que la mayoría de las solicitudes son del tipo de instalación de sistemas operativos y software (Microsoft, principalmente), así como el de configuración de equipos y puesta a punto para el trabajo del Instituto, ofreciendo mantenimiento a los equipos y asesoría al personal que requiera realizar algún cambio, actualización o traslado de información en su equipo de cómputo.

De la misma forma el personal de área atiende la mayoría de las solicitudes, puesto que parte de su función es re-direccionar el servicio al área donde corresponda el problema.

1.2.4 Portales y páginas web

Encargada del diseño y desarrollo de sitios web, así como de administrar la página del II con la finalidad de divulgar información acerca de eventos, cursos, laboratorios, congresos, publicaciones, etc. Ofrece enlace hacia las diversas opciones de comunicación dentro y fuera del II.

Cuenta con una videoteca en donde se reproducen los videos más sobresalientes producidos dentro del II y un video streaming para observar en tiempo real conferencias en vivo.

Los diversos servicios que el área desarrolla, administra y/o diseña son:

- Portal e Intranet del II.
- Páginas para congresos, cursos, seminarios, talleres, etc.
- Material gráfico y multimedia para cursos y congresos.
- Sitios SharePoint para diversos grupos de investigación.
- Envío de avisos a la comunidad del II.
- Publicaciones en línea (libros, gacetas, series, artículos, tesis, etc.).

1.2.5 Redes y telecomunicaciones

Se encarga de la investigación sobre los últimos avances tecnológicos en redes de cómputo, telefonía y video seguridad de altas especificaciones basadas en tecnologías Gigabit Ethernet.

La red del II responde a la necesidad de colaboración en proyectos de investigadores, profesores, estudiantes y personal administrativo; así como de la necesidad de colaborar con científicos de otros países mediante la red Internet.

Ofrece servicios como:

- Soporte a equipos y asignación de direccionamiento IP.
- Sustitución de cables telefónicos y cambio de conectores.
- Configuración de teléfono, marcación rápida y buzón telefónico.
- Habilitación de nodos existentes y reubicación de rosetas de red por remodelación.
- Servicios de internet.
- Telefonía de larga distancia y celular.
- Instalación de redes.
- Administración remota de redes (Juriquilla, Mérida y Morelia).
- Administración de cámaras de seguridad.

1.2.6 Sistemas Linux-Unix

Se encarga de la investigación de los avances tecnológicos en plataformas de cómputo basados en Linux-Unix y programas de aplicación científica para éstas plataformas. La función primordial del área es administración de servidores Linux.

Algunos servicios solicitados son:

- Conexión remota SSH.
- Importación de entorno gráfico.
- Procesamiento de datos.
- Bases de datos.
- Servicio ftp.
- Instalación remota y en sitio.
- Compartición remota de información.
- Servicio de motor de aplicaciones (tomcat).

- Desarrollo sobre sistemas Linux.
- Auditorías y escaneo de seguridad.
- Procesamiento de datos en clúster.

1.2.7 Sistemas Microsoft Windows

El área de Sistemas Microsoft Windows junto a sus servidores disponibles, realizan tareas dedicadas a proporcionar información, recursos y servicios a los usuarios de la red del II, además se encarga de la investigación de los avances tecnológicos en plataformas de cómputo basados en Windows y programas de aplicación científica y supercómputo para dichas plataformas.

Ofrece servicios como:

- Administración para:
 - Claves de red.
 - Servidores Microsoft.
 - Sitios SharePoint.
 - Bases de datos.
 - Firewall.
- FTP con Shareware y Freeware.
- Cuentas de correo.
- Seguridad (Detección de intrusos).

Con estos antecedentes acerca del II, la CSC y sus diferentes áreas de trabajo que brindan soporte y mantenimiento en cómputo y sistemas de información a la comunidad del Instituto, damos un panorama general de las actividades que serán administradas y controladas por el Sistema de Control de Servicios de Cómputo, que tiene como objetivo ofrecer una aplicación vía pagina web que gestione dichas actividades.

CAPÍTULO 2

ANÁLISIS DEL SISTEMA

2.1 DESCRIPCIÓN DEL PROBLEMA

Al presentarse cualquier problema o requerimiento relacionado con las actividades y servicios que ofrecen las distintas áreas de la CSC mencionadas en el capítulo 1, el usuario se ve en la necesidad de solicitar apoyo a alguna de las siguientes áreas: Cómputo Personal y Periféricos, Sistemas Microsoft Windows, Sistemas Linux-UNIX, Redes y Telecomunicaciones, Sistemas de Bases de Datos, Portales y Páginas Web, y Capacitación Informática; las que se encargan de atender su problema respectivamente.

Las solicitudes se reciben vía telefónica, por correo electrónico y de manera verbal; en algunos casos mediante el llenado de formatos o memoranda internos, lo que implica tener varias fuentes de recepción de solicitudes y por ende se tenga un seguimiento ineficiente, falta de información y que además se supervisen de manera manual, parcial e irregular.

Cada área lleva un seguimiento distinto de reportes, personal que atiende la solicitud, solución del problema, tiempo de respuesta, datos del equipo y ubicación, etc.

Esta forma de trabajo no ha permitido llevar una administración y un control en cuanto al tipo de mantenimiento o servicio que se ha brindado a un mismo equipo o usuario, así como también poder llevar estadísticas de servicio y un control de la eficiencia del personal de la CSC.

2.2 REQUERIMIENTOS

En ingeniería del software y el desarrollo de sistemas, un requerimiento es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio.

Los requerimientos son declaraciones que identifican atributos, capacidades, características y/o cualidades que necesita cumplir un sistema para que tenga valor y utilidad para el usuario. En otras palabras, los requerimientos muestran qué elementos y funciones son necesarias para un proyecto.

A continuación se describen los requerimientos que fueron especificados y clasificados en 4 módulos por el Coordinador de Sistemas de Cómputo de acuerdo a las necesidades de la Coordinación.

Como requerimiento de acceso a cualquiera de los módulos, es necesario que los usuarios y administradores del sistema estén dados de alta en el Active Directory por lo que se validarán y entrarán con el mismo usuario y contraseña asignado en Windows.

2.2.1 Módulo para la recepción, consulta y confirmación de solicitudes recibidas

- **Realizar o cancelar solicitudes de servicio**

Es importante que para realizar una solicitud de servicio se cuente con el número de inventario del equipo para que el sistema recupere los datos detallados del equipo y lo valide en la base de datos de inventarios.

Al concluir la solicitud el usuario deberá recibir una confirmación por correo electrónico con el número de reporte asignado y una liga para consultar posteriormente el avance de la solicitud.

En el caso de solicitudes para equipo y personal que no pertenezca al Instituto, este módulo deberá ligarse con un módulo de registro de personal y equipo externo. Este módulo se detalla más adelante. (Ver tema 2.2.4)

Para poder cancelar una solicitud es necesario que ésta no haya sido asignada.

- **Realizar solicitudes a nombre de otra persona**

Los administradores pueden realizar una solicitud a nombre del usuario para los casos de solicitudes recibidas por buzón de voz, teléfono, correo electrónico; o recibidas por el Director, Secretario, Subdirector o Coordinador.

- **Consultar noticias o tips importantes para los usuarios**

La página de inicio al entrar al sistema, deberá mostrar una sección de noticias o tips importantes para los usuarios.

- **Consultar solicitudes realizadas**

Deberá ser posible la consulta del avance y/o estado de las solicitudes realizadas por el usuario.

Al consultar las solicitudes realizadas, se podrá calificar el servicio del personal de cómputo que atendió el reporte; siempre y cuando la solicitud tenga estado Terminar.

2.2.2 Módulo de administración

- **Estados de solicitudes**

Se podrá actualizar el estado de solicitudes de servicio, generando un correo electrónico para notificar el cambio al personal correspondiente. Los estados que se pueden actualizar son:

- “Asignar” – Cuando se desee asignar la solicitud a personal de cómputo para la atención de la solicitud.
- “Reasignar” Cuando se desee reasignar la solicitud a personal distinto del asignado originalmente para continuar con su atención.

- “Turnar” – Cuando se desee enviar solicitud a otra área.
- “Terminar” – Cuando el personal asignado termine de atender la solicitud.
- “Cerrar” – Cuando se haya hablado telefónicamente con el usuario o hayan pasado 3 días en estado terminado.
- “Cancelar” – Cuando se desee cancelar la solicitud.
- “Atendiendo solicitud” – Cuando se está atendiendo la solicitud.
- “Pendiente” – Cuando se ha dejado pendiente la solución de la solicitud por causas fuera del alcance del personal asignado para su atención.
- “Regresada” – Cuando la solicitud recibida no sea correspondiente al área.

- **Prioridades**

Los jefes de área podrán asignar o reasignar prioridad alta, media o baja.

- **Base histórica**

Todas las solicitudes cerradas deberán pasar a una base histórica, para su posterior consulta y así poder brindar un mejor servicio de manera rápida y eficaz.

- **Consulta de solicitudes**

Se podrán consultar las solicitudes asignadas a cada área, así como las solicitudes asignadas al personal que ingresa al sistema.

- **Seguimiento de solicitudes**

Se deberán manejar alarmas por correo electrónico cuando alguna solicitud esté detenida por exceso de tiempo transcurrido para su atención o solución, inasistencias del personal, falta de software o hardware, el usuario no ha estado, etc.

Se deberá llevar un control del tiempo desde que el usuario ingresó su solicitud, pasando por todos los estados de la solicitud hasta llegar al estado cerrada.

- **Equipo y personal externo**

Se podrán consultar las solicitudes registradas en el sistema por el personal académico correspondiente. Así como se podrán aprobar o rechazar solicitudes.

- **Control de catálogos**

Se podrán corregir, actualizar o ampliar los catálogos existentes en el sistema.

- **Noticias/Tips/Software**

Se deberán permitir altas, bajas y actualizaciones para noticias/tips y software que es mostrado en la página inicial del sistema.

2.2.3 Módulo de estadísticas y reportes

- **Generar reportes**

Se deberán generar reportes y estadísticas generales para la coordinación, por área, personal, equipo, usuario, problema, calificación, periodo, estatus, etc. Los reportes podrán ser exportados a formato “pdf”.

2.2.4 Módulo de registro de personal y equipo externo

- **Solicitud de registro de equipo y/o personal externo**

Los académicos podrán gestionar ante el coordinador la autorización para que la CSC atienda a personal o equipo que no pertenezca al Instituto.

Se deberán proporcionar datos de la persona y/o equipo; y la justificación correspondiente.

El coordinador podrá autorizar o rechazar la solicitud y justificar la razón. El sistema avisará por correo electrónico al solicitante.

- **Control de equipo y personal externo**

Se deberá almacenar información de personal y equipo que no pertenezca al Instituto.

Cada personal y equipo externo deberá estar relacionado con algún Técnico Académico, Secretarios o el Director.

2.3 PROPUESTA DE SOLUCIÓN

Para resolver la problemática mencionada ha sido preocupación del Instituto de Ingeniería y de la Coordinación de Sistemas de Cómputo en particular, utilizar las tecnologías disponibles en el Instituto para beneficio de la comunidad. Por ello se plantea desarrollar un sistema informático que permita a los usuarios generar y enviar sus solicitudes desde cualquier computadora o bien por teléfono, apoyándose en el personal de la CSC para ingresar la solicitud al sistema, llevar un seguimiento y control de los servicios prestados, los problemas más comunes, sus soluciones y probables herramientas que puedan ayudar a resolver aquellos que lleven un patrón.

También será posible llevar un registro completo de los equipos de cómputo propiedad del Instituto y externos para generación de informes.

Se integrará una sección de noticias, tips y software con la finalidad de mantener informados a los usuarios acerca de: problemas o errores en programas, configuración y mantenimiento de equipos, temas actuales relacionados en cómputo y actividades realizadas en el II.

2.4 METODOLOGÍA USADA

El término método se utiliza para el procedimiento que se emplea para alcanzar los objetivos de un proyecto y la metodología es el estudio del método.

Las metodologías de ingeniería de software proporcionan el cómo construir software, técnicamente hablando. Abarcan tareas que incluyen la comunicación, el análisis de requisitos, el modelado del diseño, la construcción del programa, la realización de pruebas y el soporte.

Para el desarrollo de software son extensas y variadas, generalmente parecen tener la misma estructura y no se comprende con claridad sus diferencias al analizarlas conjuntamente. Lo que las hace muchas veces particulares son los elementos que conforman cada una de ellas, como la manera de definir los tiempos, procesos y objetos (disciplinas, roles y responsabilidades).

Algunas de las metodologías más conocidas para el desarrollo de software son las siguientes:

- Cascada. Propone una serie lineal de actividades del marco de trabajo que generalmente resulta inconsistente con la realidad actual en el desarrollo de software debido a, que es muy raro que los proyectos reales sigan el flujo secuencial que propone el modelo. Se puede aplicar en situaciones en las cuales los requisitos están bien definidos y son estables.
- Incremental. Combina elementos del modelo en cascada en forma iterativa. Aplica secuencias lineales de manera escalonada conforme avanza el tiempo en el calendario del proyecto.
- Evolutivo. Es iterativo y lo caracteriza la forma en que permite que los ingenieros de software desarrollen versiones cada vez más completas del software. (Espiral)
- Proceso Unificado. Es un proceso de software guiado por los casos de uso, de arquitectura céntrica, iterativo e incremental. Diseñado como un marco de trabajo para la ingeniería del software orientada a objetos mediante la utilización del UML.

De acuerdo a las características de las metodologías anteriores y tomando en cuenta las necesidades del sistema, elegimos el uso del proceso unificado por ser una metodología actual, orientada a objetos, dirigida por casos de uso y con arquitectura céntrica, iterativa e incremental; la cual, se detalla a continuación.

El Proceso Unificado de desarrollo de software que junto con el UML (del inglés Unified Modeling Language, ‘Lenguaje Unificado de Modelado’), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.^[1]

Esta metodología no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización. Además de que se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema.

Cada ciclo consta de cuatro fases que se desarrollan a lo largo del tiempo: inicio, elaboración, construcción y transición. Dentro de cada fase se pueden descomponer adicionalmente las actividades en flujos de trabajo fundamentales: Requisitos, Análisis, Diseño, Implementación y Prueba.

Durante la fase de inicio, se desarrolla una descripción del sistema final a partir de una buena idea y se presenta el análisis de negocio para la aplicación. La arquitectura es provisional y consiste típicamente en un simple esbozo que muestra los subsistemas más importantes. En esta fase se identifican y

priorizan los riesgos importantes, se planifican en detalle la fase de elaboración, y se estima el sistema de manera aproximada.

Durante la fase de elaboración, se especifican en detalle la mayoría de los casos de uso del sistema y se diseña su arquitectura. Durante esta fase del desarrollo, se realizan los casos de uso más críticos que se identificaron en la fase de inicio. Al final de la fase de elaboración, el director de proyecto está en disposición de planificar las actividades y estimar los recursos necesarios para terminar el sistema.

Durante la fase de construcción se crea el sistema. La arquitectura es estable, aunque se pueden descubrir formas mejores de estructurarlo. Al final de esta fase, el sistema contiene todos los casos de uso que la dirección y el cliente han acordado para el desarrollo de esta versión. Sin embargo puede que no esté completamente libre de defectos. Muchos de estos defectos se descubrirán y solucionarán durante la fase de transición.

La fase de transición cubre el periodo durante el cual el sistema se convierte en versión beta. En esta versión un número reducido de usuarios con experiencia prueban el sistema e informan de defectos y deficiencias. Los desarrolladores corrigen los problemas e incorporan algunas de las mejoras sugeridas en una versión general dirigida a la totalidad de la comunidad de usuarios. Conlleva actividades como la fabricación, formación del cliente, proporcionar una línea de ayuda y asistencia y la corrección de los defectos que se encuentren tras la entrega.

2.5 PLANEACIÓN Y ESTIMACIÓN DE TIEMPOS Y COSTOS

Una de las actividades cruciales del proceso de gestión del proyecto de software es la planificación. La estimación es más arte que ciencia; es una pequeña planeación sobre qué es lo que va a ser nuestro proyecto.

La administración efectiva de un proyecto de software depende de planear completamente su progreso. El administrador del proyecto debe anticiparse a los problemas que podrían surgir, así como preparar soluciones tentativas a esos problemas.

La planeación es un proceso iterativo que solamente se completa cuando el proyecto se termina. Conforme la información se hace disponible, el plan debe revisarse regularmente. Las metas globales de la institución son un factor importante que debe considerarse cuando se formula el plan del proyecto. Si éste cambia, las modificaciones en el proyecto serán necesarias.

Cuando se planifica un proyecto de software se tienen que obtener estimaciones de esfuerzo humano requerido, duración cronológica del esfuerzo humano, del proyecto y del costo. Pero en muchos de los casos las estimaciones se hacen valiéndose de la experiencia pasada como única guía, es decir, si un proyecto es bastante similar en tamaño y función a un proyecto pasado, es probable que el nuevo proyecto requiera aproximadamente la misma cantidad de tiempo y esfuerzo que el anterior. Se prepara un calendario para el proyecto donde se definen las actividades y su revisión. Debido a que la estimación inicial es tentativa, el plan siempre debe actualizarse.

Se han desarrollado varias técnicas de estimación para el desarrollo de software, aunque cada una tiene sus puntos fuertes y sus puntos débiles, todas tienen en común la característica de desglosar el proyecto en partes más pequeñas que se estiman individualmente. Por lo que la estimación de este proyecto la dividimos en partes más pequeñas de acuerdo a las fases de UML como se muestra en la figura 2.1

	i	Nombre de tarea	Trabajo	Duración	Comienzo	Fin
0		- Desarrollo "Sistema de Solicitudes"	942.75 horas	491.33 días?	lun 03/05/10	vie 30/12/11
1		+ Fase de Inicio	7.32 horas	23 días	lun 03/05/10	lun 31/05/10
6		+ Fase de Elaboración	55.63 horas	98.67 días	mar 01/06/10	jue 30/09/10
15		+ Fase de Construcción	106.77 horas	295.33 días?	vie 01/10/10	vie 30/09/11
23		+ Fase de Transición	680.33 horas	72 días?	lun 03/10/11	vie 30/12/11
30		+ Documentación	92.7 horas	49.33 días	lun 01/08/11	vie 30/09/11

Figura 2.1 Estimación de tiempos

La palabra **costo** significa la suma de esfuerzos y recursos que se han invertido para producir algo.^[16]

El costo de producción está formado por tres elementos fundamentales: la materia prima que se emplea en el desarrollo de producción; la mano de obra que se utiliza para la transformación de la materia prima en producto terminado y los gastos indirectos de fabricación.

La materia prima son los insumos adquiridos para utilizarlos en los procesos de elaboración del producto, sin que formen parte del mismo.^[16] Para el caso de sistemas consideramos recursos de hardware y software, como parte de estos costos.

La mano de obra es el trabajo del hombre aplicado a los materiales, con el propósito de transformarlo en un producto o servicio para el consumidor.^[16] En relación a los sistemas de cómputo, los recursos humanos forman parte de la mano de obra.

Los gastos de fabricación son todos los costos de elaboración que no se pueden clasificar como materia prima o mano de obra.^[16] Algunos de ellos como: Agua, Luz, Renta, Teléfono, Mobiliario, Papelería, etc.

A continuación, en la tabla 2.1 se indica a manera de porcentaje el costo de cada uno de los recursos, dado que fueron proporcionados y absorbidos por el II.

Recurso	Porcentaje
Hardware	15%
Software	40%
Recursos Humanos	40%
Otros costos	5%

Tabla 2.1 Recursos generales

Las siguientes tablas detallan los recursos utilizados:

- Hardware

Recurso	
Estación de Trabajo:	470 GB Disco Duro 8 GB Memoria RAM Procesador Intel Core 2 Quad a 2.5 GHz
Estación de Trabajo:	300 GB Disco Duro 8 GB Memoria RAM Procesador Intel Core 2 Quad a 2.4 GHz
Estación de Trabajo:	250 GB Disco Duro 2 GB Memoria RAM Procesador Intel Pentium D a 3.4 GHz
Servidor de Pruebas:	80 GB Disco Duro 1.5 GB Memoria RAM Procesador Intel Pentium 4 a 2.4 GHz
Servidor de Desarrollo:	3 x 80 GB Disco Duro Configuración RAID 5 4 GB Memoria RAM Doble procesador Intel Xeon a 2 GHz

Tabla 2.2 Recursos de hardware

- Software

Recurso
3 Licencias de Windows Vista Ultimate
3 Licencias de Office Ultimate 2007
Office Project 2007
Licencias de Visual Studio Team System 2008
Licencias de Microsoft SQL Server 2008
Licencia de Windows Server Standard Edition 2003

Tabla 2.3 Recursos de software

- Recursos Humanos

Recurso
Ing. Marco Ambríz Maguey Director de Proyecto
Ing. Jonathan Velázquez García Jefe del área de Bases de Datos Coordinador de Proyecto
Ing. Mauricio Velázquez Álvarez Subjefe del área de Bases de Datos Coordinador de Proyecto
Rosaura Apresa Monzón Desarrollador
Liliana González Jiménez Desarrollador
Raúl Trinidad Ramírez y Ramírez Desarrollador

Tabla 2.4 Recursos humanos

- Otros Costos

Instalación del sistema
Mantenimiento del sistema
Capacitación de administradores y usuarios del sistema
Servicios adicionales(luz, agua, teléfono, renta, etc.)

Tabla 2.5 Otros recursos

2.6 CALIDAD DE SOFTWARE

Para el IEEE (del inglés Institute of Electrical and Electronic Engineers, ‘Instituto de Ingenieros en Electricidad y Electrónica’) la calidad de software se define como el grado en que un sistema, componente o proceso cumple con los requerimientos especificados y las necesidades del cliente o usuario.^[26]

Para la ISO (inglés International Organization for Standardization, ‘Organización Internacional para la Estandarización’), la calidad de software es el grado (pobre, bueno o excelente) en que un conjunto de características inherentes del software cumplen con los requisitos del sistema.^[27]

La calidad es uno de los puntos de atención de las organizaciones actuales, ya que el software se ha convertido en un activo que determina en gran medida la operatividad de la organización. Ahora bien, debemos tener muy presente que la calidad de un producto es directamente proporcional a la del proceso con que es generado, por tanto, si queremos garantizar la calidad del producto debemos realizar una correcta selección de un modelo de procesos.

Un modelo de procesos es un conjunto estructurado de elementos que describen las características de procesos efectivos y de calidad, nos dicen qué hacer, no cómo hacerlo ni quién lo hace.^[3]

Existen modelos internacionales como CMMI (del inglés Capability Maturity Model Integration, 'Modelo de Capacidad y Madurez') e ISO difíciles de adaptar en áreas de desarrollo de software mexicanas.

CMMI creado en Estados Unidos describe las prácticas esenciales de un proceso efectivo de desarrollo de software. Define 25 áreas de proceso, las que cubren desde el desarrollo del producto y servicios, hasta el mantenimiento de los mismos. Éstas están organizadas por el nivel de madurez (5 Niveles), lo cual nos asegura que seguiremos un camino progresivo para implementar cada una de ellas, sin tener que preocuparnos por la dependencia de unas con otras, ya que el enfoque de esta representación no nos permite avanzar, sin antes tener cubiertas todas la metas y prácticas que incluye cada área de proceso. Sin embargo, este modelo resulta demasiado complejo para organizaciones PyME (Pequeña y Mediana Empresa) y para proyectos pequeños con menos de 100 empleados.

Por otra parte, ISO 9000 es una familia de normas que sirve como guía en la implantación y operación de Sistemas de Gestión de la Calidad. ISO 9000 en su versión 2000 se enfoca en los procesos y en la satisfacción del cliente; además, es certificable en México. Sin embargo, por su generalidad, es difícil de adecuar al desarrollo de software.

En 2002 la SE (Secretaría de Economía) comenzó la organización de mesas de trabajo para definir las estrategias del programa para el desarrollo de la industria de software (PROSOFT).

La profesora de la Facultad de Ciencias de la UNAM y presidenta en ese entonces de la *AMCIS* (Asociación Mexicana para la Calidad en Ingeniería de Software), Hanna Oktaba, comenzó a coordinar los trabajos de un grupo interesado en el tema, analizando las características básicas de la industria mexicana del software y realizando entrevistas en algunas de ellas. La conclusión de esta investigación fue que ninguno de los modelos internacionales como CMMI e ISO, son aplicables a la industria mexicana. Es así como surge el proyecto de desarrollo de un modelo de procesos y un método de evaluación "a la medida de la industria mexicana" hoy llamado MoProSoft y EvalProSoft.

El proyecto a cargo de Hanna Oktaba a petición de la SE, se concretó cuando la norma fue aprobada por la NYCE (Normalización y Certificación Electrónica) el 5 de julio de 2005 y el 15 de agosto fue publicada en el Diario Oficial de la Federación. El nombre completo de la norma es: NMX-I-059/04-NYCE-2005 Tecnología de la Información-Software-Modelos de procesos y evaluación para desarrollo y mantenimiento de software.

Parte 01: Definición de conceptos y productos

Parte 02: Requisitos de procesos (MoProSoft)

Parte 03: Guía de implantación de procesos

Parte 04: Directrices para la evaluación (EvalProSoft)

Este modelo, a diferencia de otros como CMMI e ISO, está dirigido a las micro y pequeñas empresas y surge bajo las premisas: barato, práctico, fácil de entender y aplicar.

Definir la estructura y las actividades de las PYMES no es algo fácil. Ante todo, es importante separarlas del resto de las empresas. De un lado tenemos a la micro empresa, la cual es, por lo general, un simple establecimiento en el que una persona o un número muy pequeño de individuos producen un bien o prestan un servicio. En el extremo opuesto se encuentran las grandes empresas.

La clasificación de las empresas de acuerdo a su tamaño en México, según el Diario Oficial de la Federación publicado el 30 de Junio de 2009, se determinará a partir del obtenido del número de trabajadores multiplicado por 10%; más el monto de las ventas anuales por 90%, como se muestra en la Tabla 2.6.^[23]

Tamaño	Sector	Rango del número de trabajadores	Rango de monto de ventas anuales(mdp)	Tope máximo combinado*
Micro	Todas	Hasta 10	Hasta \$4	4.6
Pequeña	Comercio	Desde 11 hasta 30	Desde \$4.01 hasta \$100	93
	Industria y servicios	Desde 11 hasta 50	Desde \$4.01 hasta \$100	95
Mediana	Comercio	Desde 31 hasta 100	Desde \$100.01 hasta \$250	235
	Servicios	Desde 51 hasta 100	Desde \$100.01 hasta \$250	235
	Industria	Desde 51 hasta 250	Desde \$100.01 hasta \$250	250

* Tope máximo combinado = (Trabajadores) x 10% + (Ventas anuales) x 90%

Tabla 2.6 Clasificación de empresas

Por lo que podemos considerar a la CSC como una mediana empresa, ya que cuenta con una comunidad de 40 miembros y puede definirse en el sector de servicios así que es posible aplicarle el método de evaluación de MoProSoft que se aborda a continuación, para asegurar un sistema de calidad, haciendo uso de herramientas mexicanas disponibles para este tipo de empresas.

MoProSoft tiene tres categorías de procesos:

- Alta Dirección (DIR)

Categoría de procesos que aborda las prácticas relacionadas con la Gestión del Negocio. Proporciona los lineamientos a los procesos de la Categoría de Gestión y se retroalimenta con la información generada por ellos.

- Gestión (GES)

Categoría integrada por los procesos de Gestión de Procesos, Gestión de Proyectos y Gestión de Recursos (Recursos Humanos y Ambiente de Trabajo, Bienes, Servicios e Infraestructura y Conocimiento de la Organización) en función de los lineamientos establecidos en la categoría de Alta Dirección.

Proporciona los elementos para el funcionamiento de los procesos de la Categoría de Operación, recibe y evalúa la información generada por éstos y comunica los resultados a la Categoría de Alta Dirección.

- Operación (OPE)

Integrada por los procesos de Administración de Proyectos Específicos y de Desarrollo y Mantenimiento de Software.

Esta categoría realiza las actividades de acuerdo a los elementos proporcionados por la Categoría de Gestión y entrega a ésta la información y productos generados.

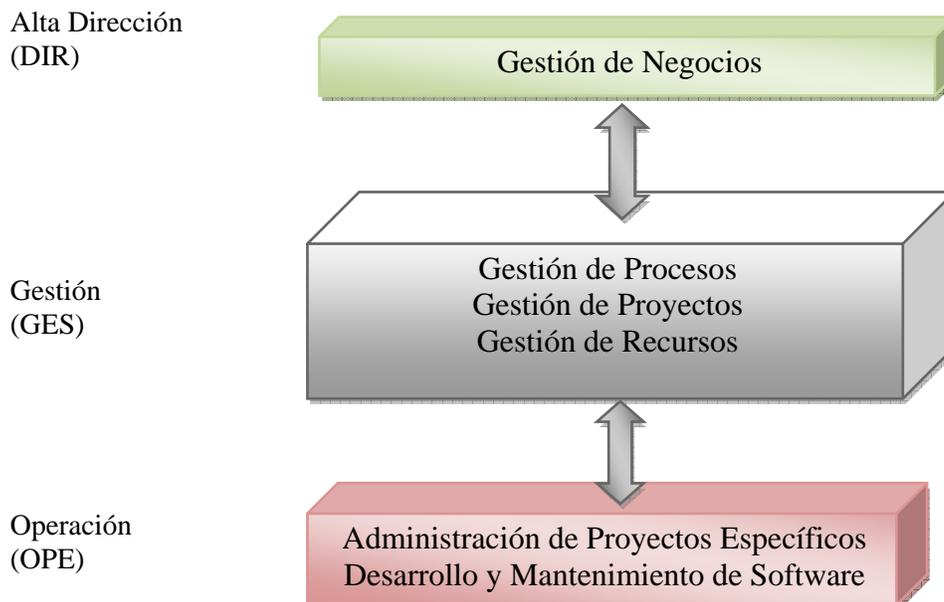


Figura 2.2 Diagrama de categorías del modelo MoProSoft

Como podemos ver, se trata de un modelo integrado, en el cual las salidas de un proceso están claramente dirigidas como entradas a otros procesos. En cada proceso están definidos los roles responsables para la ejecución de las prácticas. Los roles se asignan al personal de la organización de acuerdo a habilidades y capacitación para desempeñarlos. Se clasifican en Grupo Directivo, Responsable de Proceso y otros roles involucrados, además, se considera al cliente y al usuario como roles externos a la organización.

Los objetivos que para nuestro proyecto perseguiremos en cada categoría de este modelo son:

ALTA DIRECCIÓN

- Gestión de Negocios

- O1 Lograr una planificación estratégica exitosa mediante el cumplimiento del Plan Estratégico.
- O2 Lograr que el equipo trabaje en función del plan estratégico mediante la correcta comunicación e implantación del mismo.
- O3 Mejorar el plan estratégico mediante la implementación de la propuesta de mejoras.

El Instituto de Ingeniería ha definido un plan estratégico de desarrollo para el periodo 2008- 2012. La propuesta tiene como propósito institucional constituirse como el instrumento que oriente y defina las acciones que permitan al Instituto alcanzar la superación y excelencia académicas; la formación de mejores profesionistas; su consolidación como centro de investigación en ingeniería más relevante, de mayor calidad y pertinencia en México.

El cuerpo de la propuesta del Plan de Desarrollo 2008-2012, está integrada a partir del Plan de Trabajo que el Instituto presento a la Junta de Gobierno de la UNAM y por numerosas ideas resultado de las opiniones y propuestas de los académicos, subdirectores y secretarios, donde se expone:

- La razón de ser de la institución.
- Los elementos para un diagnóstico.
- A dónde queremos llegar como institución y los retos que enfrentaremos.
- Los grandes ejes que lo guiarán, los programas y los proyectos que están vinculados con cada uno de ellos, donde se incluyen los objetivos.
- Procuración y distribución de recursos.
- Seguimiento y evaluación (indicadores).

GESTIÓN

- Gestión de Procesos

O1 Planificar las actividades de definición, implantación y mejora de los procesos en función del plan estratégico.

O2 Dar seguimiento a las actividades de definición, implantación y mejora de los procesos mediante el cumplimiento del plan de mejora.

O3 Mejorar el desempeño de los procesos mediante el cumplimiento del plan de mejora.

O4 Mantener informado a gestión de negocios mediante el reporte cuantitativo y cualitativo.

- Gestión de Proyectos

O1 Mantener bajo control las actividades de gestión de proyectos mediante el cumplimiento del plan de gestión de proyectos.

O2 Proveer la información del desempeño de los proyectos a gestión de negocio mediante la generación del reporte cuantitativo y cualitativo.

O3 Atender los comentarios y quejas del cliente mediante la definición y ejecución de acciones correctivas o preventivas.

- Gestión de Recursos

O1 Lograr los objetivos del plan estratégico mediante la provisión de los recursos suficientes y calificados. (Objetivo cumplido bajo el patrocinio del II)

O2 Proveer a los miembros del proyecto de los medios y mecanismos adecuados para el uso y resguardo de la información mediante la base de conocimiento.

- Gestión de Recursos Humanos y Ambiente de Trabajo

O1 Proveer al proyecto recursos humanos calificados mediante la capacitación adecuada a los roles que se les asignen.

- Bienes, Servicios e Infraestructura

O1 Proporcionar al proyecto los bienes y servicios requeridos por los procesos y los proyectos mediante la selección y evaluación de los proveedores. (Objetivo cumplido por el II)

O2 Mantener la infraestructura de la organización mediante el cumplimiento del Plan de Mantenimiento. (Objetivo cumplido por el II)

- Conocimiento de la Organización

O1 Proporcionar a la organización la Base de Conocimiento de forma confiable, oportuna y segura mediante el cumplimiento del Plan de Administración de la Base de Conocimiento. (Objetivo cumplido por el II)

OPERACIÓN

- Administración de Proyectos Específicos

O1 Lograr los objetivos del proyecto en tiempo y costo mediante la coordinación y el manejo de los recursos del mismo.

O2 Mantener informado al cliente mediante la realización de reuniones de avance del proyecto.

O3 Atender las solicitudes de cambio del cliente mediante la recepción y análisis de las mismas.

- Desarrollo y Mantenimiento de Software

O1 Lograr que los productos de salida sean consistentes con los productos de entrada en cada fase de un ciclo de desarrollo mediante las actividades de verificación, validación o prueba.

O2 Sustentar la realización de ciclos posteriores o proyectos de mantenimiento futuros mediante la integración de la configuración del ciclo actual.

O3 Llevar a cabo las actividades de las fases de un ciclo mediante el cumplimiento del plan de desarrollo actual.

CAPÍTULO 3

ANÁLISIS DE RECURSOS DISPONIBLES

3.1 RECURSOS DE SOFTWARE

Los recursos de software brindados por el Instituto de Ingeniería y establecidos para los desarrolladores en la realización de este proyecto son: Microsoft SQL Server, Microsoft Visual Studio y IIS; explicados a lo largo de este capítulo.

3.1.1 Manejador de base de datos - Microsoft SQL Server

Un Sistema manejador de Bases de Datos (SMBD) es un módulo de programa constituido por la interfaz entre los datos de bajo nivel almacenados en la Base de Datos (BD) y los programas de aplicaciones. Tiene como propósito general facilitar el proceso de definir, construir y manipular BD que se utilizan para diferentes tipos de aplicaciones.^[5]

Entre las funciones principales del SMBD están:

- Crear, almacenar, recuperar, eliminar y modificar los datos.
- Guardar la consistencia de los datos.
- Solucionar los problemas de concurrencia.
- Seguridad en el almacenamiento de los datos.

TIPOS DE SMBD

La clasificación de los SMBD se basa en tres criterios, los cuales son: según el modelo lógico de datos, según el número de usuarios a los que da servicio y según el número de sitios en los que está distribuida la BD.

- Según modelos de datos:
 - Relacional.
 - Orientado a Objetos.
 - Objeto-Relacional o relacional extendido.
 - Jerárquico.
- Según Número de Usuarios:
 - Monousuario.
 - Multiusuario.
- Según Número de Sitios:
 - Centralizado.
 - Distribuido.

El modelo de datos relacional representa una base de datos como una colección de tablas, cada una de las cuales se puede almacenar en forma de archivo individual.

El usuario del sistema de bases de datos puede consultar, borrar y actualizar (tuplas). Existen varios lenguajes para expresar estas operaciones, como el álgebra relacional, que forma la base del lenguaje de consulta SQL. El álgebra relacional define un conjunto de operaciones algebraicas que operan sobre tablas y devuelven tablas como resultado.

MICROSOFT SQL SERVER

La historia de SQL (del inglés Structured Query Language, ‘Lenguaje de Consulta Estructurado’) empieza en 1974 con la definición, por parte de Donald Chamberlin y de otras personas que trabajaban en los laboratorios de investigación de IBM (del inglés International Business Machines), de un lenguaje para la especificación de las características de las bases de datos que adoptaban el modelo relacional, este lenguaje se llamaba SEQUEL (del inglés Structured English Query Language) y se implementó en un prototipo llamado SEQUEL-XRM entre 1974 y 1975, para 1977 por motivos legales cambio su nombre convirtiéndose en SQL.

IBM basándose en este lenguaje adoptó y utilizó “System R”; gracias al éxito de este sistema, que no estaba todavía comercializado, también otras compañías empezaron a desarrollar sus productos relacionales, compañías como Oracle y Sybase comercializaron productos basados en SQL. En 1986, el ANSI adoptó SQL como estándar para los lenguajes relacionales y en 1987 se transformó en estándar ISO.

Microsoft SQL Server es un SDBD basado en el lenguaje Transact-SQL, y en Sybase IQ, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea, Microsoft SQL Server es la alternativa de Microsoft a otros sistemas manejadores de bases de datos como Oracle, Sybase ASE, PostgreSQL, entre otros.

Microsoft SQL Server incluye varias tecnologías de análisis y administración de datos, las que organiza en las siguientes categorías (Ver Figura 2.1) ^[29]

- Motor de base de datos de SQL Server
- Analysis Services
- SQL Server Integration Services (SSIS)
- Réplica de SQL Server
- SQL Server Reporting Services
- SQL Server Service Broker

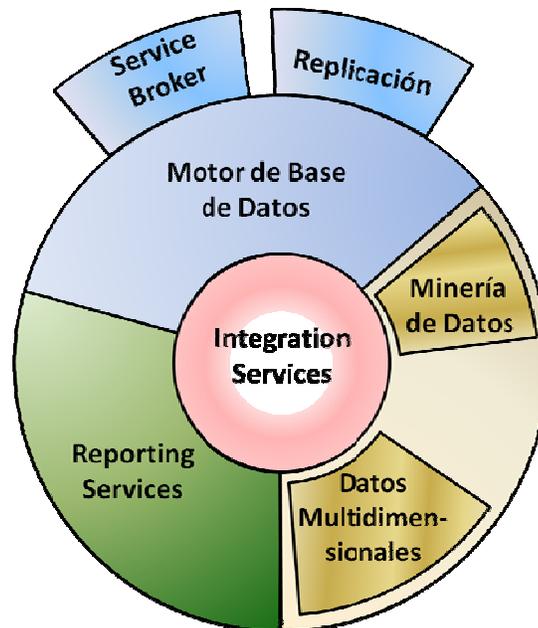


Figura 2.1 Estructura de Microsoft SQL Server 2008

Motor de base de datos (Database Engine)

Es el servicio principal para almacenar, procesar y proteger datos. El Motor de base de datos proporciona acceso controlado y procesamiento de transacciones rápido para cumplir con los requisitos de las aplicaciones consumidoras de datos. Se usa para crear bases de datos relacionales para el procesamiento de transacciones en línea o datos de procesamiento analítico en línea. Éste incluye la creación de tablas para almacenar datos y objetos de base de datos (índices, vistas y procedimientos almacenados) para ver, administrar y proteger datos.

Analysis Services

Permite diseñar, crear y administrar estructuras multidimensionales que contienen datos de detalle y agregados procedentes de varios orígenes de datos. Proporciona un análisis rápido, intuitivo y descendente de grandes cantidades de datos, que se puede entregar a los usuarios en varios idiomas y monedas. Trabaja con almacenes de datos, mercado de datos, bases de datos de producción y almacenes de datos operativos, y admite el análisis de datos históricos y en tiempo real.

Integration Services

Es una plataforma para la creación de soluciones empresariales de transformaciones de datos e integración de datos, sirve para resolver complejos problemas mediante la copia o descarga de archivos, el envío de mensajes de correo electrónico como respuesta a eventos, la actualización de almacenes de datos, la limpieza y minería de datos, y la administración de objetos y datos de SQL Server.

Puede extraer y transformar datos de muchos orígenes distintos, como archivos de datos XML (del inglés Extensible Markup Language, Lenguaje de marcas Extensible), archivos planos y orígenes de datos relacionales, y, posteriormente, cargarlos en uno o varios destinos.

Replicación

Es un conjunto de tecnologías destinadas a la copia y distribución de datos y objetos de base de datos desde una base de datos a otra, para luego sincronizar ambas y mantener su coherencia. Permite

distribuir datos entre diferentes ubicaciones y entre usuarios remotos o móviles mediante redes locales y de área extensa, conexiones de acceso telefónico, conexiones inalámbricas e Internet.

Reporting Services

Es una plataforma de creación de informes para una gran variedad de orígenes de datos, puede crear informes interactivos, tabulares, gráficos o de forma libre a partir de orígenes de datos relacionales, multidimensionales o basados en XML. Puede publicar informes, programar el procesamiento de informes u obtener acceso a informes a petición. Puede elegirse entre varios formatos de visualización, exportar informes a otras aplicaciones y suscribirse a los informes publicados. Los informes creados se pueden ver mediante una conexión web o como parte de una aplicación de Microsoft Windows o un sitio de SharePoint. Dispone herramientas y servicios para crear, implementar y administrar, así como de características de programación que le permiten extender y personalizar la funcionalidad de los informes.

Service Broker

Los programadores pueden usar Service Broker para crear con facilidad aplicaciones distribuidas y confiables. Pueden distribuir las cargas de trabajo de datos en varias bases de datos sin tener que programar complejas funciones internas de comunicación y mensajería, reduciendo así el trabajo de programación y realización de pruebas.

3.1.2 Servidor – IIS

El IIS (del inglés Internet Information Services) con el sistema operativo Microsoft Windows Server proporciona una funcionalidad de servidor web integrado, confiable, escalable, seguro y administrable a través de una intranet, Internet o una extranet.^[21]

El IIS es una herramienta para crear una sólida plataforma de comunicaciones de aplicaciones de red dinámicas. Organizaciones de todos los tamaños usan IIS para alojar y gestionar páginas Web en Internet o en su intranet, para alojar y gestionar sitios FTP y para dirigir las noticias o el correo utilizando el Protocolo de transferencia de noticias en la red (NNTP) y el Protocolo simple de transferencia de correo (SMTP). Aprovecha los últimos estándares Web como Microsoft ASP.NET, XML y el Protocolo simple de acceso a objetos (SOAP) para el desarrollo, la implementación y la administración de aplicaciones Web.

Este servicio convierte a un ordenador en un servidor de Internet o Intranet es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente (servidor web).

Seguridad

IIS 6.0 añade una característica llamada "extensiones de servicio Web" que impide que IIS ejecute cualquier programa sin el permiso explícito de un administrador. Con la versión actual de IIS 7.0, los componentes son modulares de manera que sólo los componentes necesarios tienen que ser instalados, reduciendo aún más el área de ataque.^[21]

Hay varias características integradas de seguridad de Microsoft. Muchas empresas ofrecen herramientas de seguridad de terceros y características, también conocido como "Web App cortafuegos, o firewalls de aplicaciones web." La ventaja de estas herramientas es que ofrecen elementos mucho más amplios, por ejemplo, facilidad de usar interfaz gráfica de usuario; que ayudan en la protección de una instalación de IIS con una capa adicional de protección a un nivel superior.

Mecanismos de autenticación

- Autenticación anónima.
- Autenticación básica.
- Autenticación de texto implícita.
- Autenticación de Windows integrada.
- Autenticación .NET Passport.
- Autenticación de sitios FTP.

En los sistemas desarrollados en la CSC usaremos la Autenticación de Windows integrada.

3.1.3 Lenguaje de programación - Plataforma .NET

En 1998 un equipo de trabajo de Microsoft comenzó a trabajar en el proyecto llamado Next Generation Windows Services (Servicios Windows de Próxima Generación), esta nueva estrategia de Microsoft surge de las siguientes necesidades:

- El auge y rápido crecimiento de Internet en las comunicaciones Internacionales.
- Reforzar el desarrollo y distribución simplificado de servicios basados en Internet.
- Una plataforma de desarrollo independiente del lenguaje de programación orientada a objetos.
- Una plataforma de desarrollo de software eficiente, sencillo y eficaz.

Finalmente a principios del año 2000 Microsoft dio a conocer este trabajo al que denominaron Microsoft .NET. Después de 2 años, Microsoft creó dos productos que serían clave en el desarrollo de esta nueva tecnología:

- .NET Framework
Es un componente integral de Windows que admite la compilación y la ejecución de aplicaciones y servicios Web XML. ^[22]
- Visual Studio.NET
Es un IDE (del inglés Integrated Development Environment, 'Entorno Integrado de Desarrollo') basado en componentes para crear aplicaciones eficaces de alto rendimiento, aprovechando todas las características de .Net Framework. ^[22]

La plataforma .NET es una capa de software que se coloca entre el Sistema Operativo (SO) y el programador. Las características fundamentales de esta plataforma son:

- Portabilidad: Una aplicación .NET puede ser ejecutada en cualquier SO de cualquier máquina que disponga de una versión de la plataforma.
- Multilenguaje: Cualquier lenguaje de programación puede adaptarse a la plataforma .NET y ejecutarse en ella.
- Interoperabilidad: Es total entre diferentes trozos de código escritos en diferentes lenguajes.

Los componentes principales de .NET Framework son:

- CLR (del inglés Common Language Runtime, 'Entorno de Ejecución')
El CLR administra la memoria, ejecución de subprocesos, ejecución de código, comprobación de la seguridad del código, compilación y demás servicios del sistema. ^[22]

- Biblioteca de clases

Es una colección de tipos reutilizables que se integran estrechamente con CLR. La biblioteca de clases está orientada a objetos, lo que proporciona tipos de los que su propio código administrado puede derivar funciones. Esto ocasiona que los tipos .NET Framework sean sencillos de utilizar y reducir el tiempo asociado con el aprendizaje de sus nuevas características.^[22]

Como en cualquier biblioteca de clases orientada a objetos, los tipos de .NET Framework permiten realizar diversas tareas de programación comunes, como son la administración de cadenas, recolección de datos, conectividad de bases de datos y acceso a archivos. Además de estas tareas habituales, la biblioteca de clases incluye tipos adecuados para diversos escenarios de desarrollo especializado de aplicaciones de consola, Windows, ASP. NET, orientadas a servicios, habilitadas para el flujo de trabajo, servicios Web y Windows.

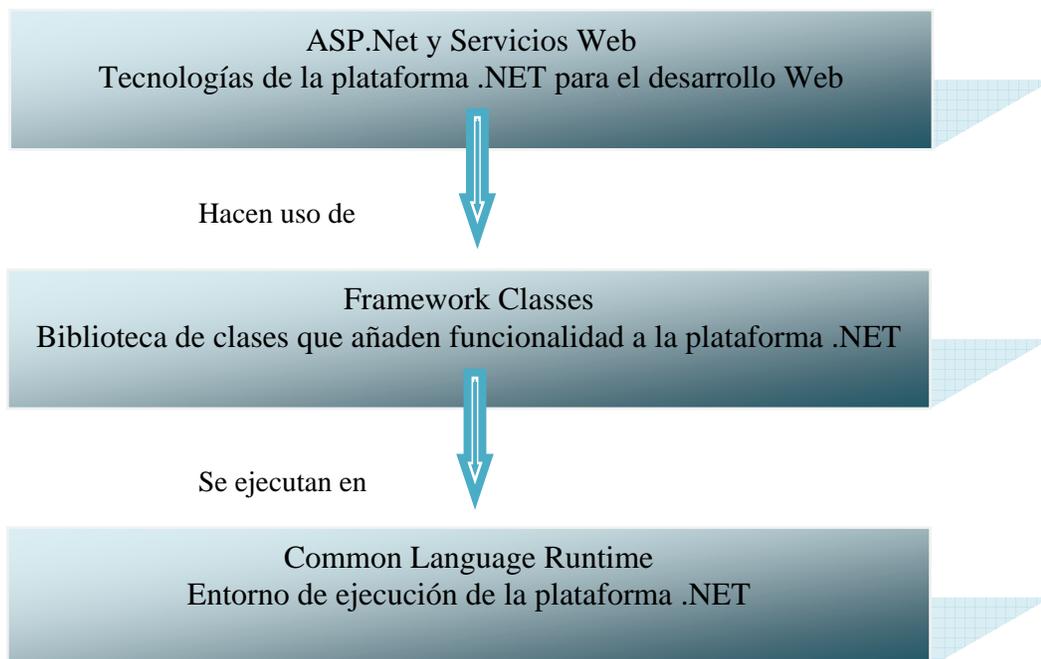


Figura 2.2 Plataforma .NET

3.2 RECURSOS DE HARDWARE

El entorno que soporta un proyecto de software incorpora hardware y software; el hardware proporciona una plataforma que soporta las herramientas (software) con que se producen los productos de trabajo basados en una buena práctica de la ingeniería del software. Puesto que la mayor parte de las organizaciones de software tienen múltiples constituyentes que requieren acceso al entorno, el planificador del proyecto debe prescribir la ventana de tiempo requerida por el hardware y el software, y verificar que estos recursos estarán disponibles.

Dadas las características del Instituto de Ingeniería, como centro de investigación, en cuestión de sistemas de información, se cuenta con hardware y software para el desarrollo y prueba de sus propias aplicaciones, como lo son equipos de cómputo, servidores y red.

Cuando un sistema basado en computadora (que incorpora software y hardware especializados) se somete a ingeniería, el equipo de software quizá requiera acceso a elementos de hardware que están desarrollando otros equipos. Por ejemplo, el software para administración de solicitudes utilizado por la comunidad del II requiere una máquina-herramienta específica que en este caso es equipo de cómputo y red. El planificador del proyecto de software debe especificar cada elemento de hardware.

En el caso del Sistema realizado, contamos con equipo de cómputo ya proporcionado por el Instituto los cuales cubren los requerimientos de las herramientas de software.

3.2.1 Hardware de desarrollo

Como se menciona en el punto anterior, el hardware de desarrollo está basado en especificaciones sobre las herramientas de software que se utilizaron para desarrollar el sistema, como lo es el Sistema Manejador de Bases de Datos y plataforma .NET, software utilizado como base para los sistemas desarrollados en el Instituto, principalmente sistemas de información como lo es nuestro proyecto. Por lo que dadas las características de nuestras herramientas y basados en las especificaciones recomendadas para dichas herramientas nuestro hardware de desarrollo se resume en equipos:

Con un procesador: recomendado a 2.2 GHz o superior.
Memoria RAM: recomendado con 2048 MB o superior.
Monitor: recomendado 1280x1024 display.
Disco duro: 7200 RPM o superior.

Las características de los equipos de cómputo con los que actualmente cuenta el Instituto, cumplen ampliamente con estas especificaciones, proporcionando así un entorno adecuado tanto en hardware como en software, para el desarrollo del sistema.

3.2.2 Hardware de producción

Un servidor debe garantizar que el acceso, el almacenamiento y el intercambio de datos, satisfagan nuestras necesidades, a la vez que ofrecen una gran seguridad y rendimiento.

Bajo los mismos argumentos que en el hardware de desarrollo, establecemos las siguientes características del software de producción tomando en cuenta ahora las herramientas del servidor (IIS) y el sistema manejador de bases de datos.

Procesador: recomendado a 2.0 GHz o superior.
Memoria RAM: recomendado con 2048 MB o superior.
Disco duro: 7200 RPM o superior.

Especificaciones, con las que el servidor proporciona un buen rendimiento para los usuarios del sistema

3.2.3 Hardware y software recomendado para los usuarios

Para los usuarios tenemos características de hardware y software con mayor diferencia, puesto que los recursos de los equipos para el buen funcionamiento del sistema son menores. Recomendando las siguientes:

Hardware

Procesador: de 1 GHz o superior.

Memoria RAM: 1024 MB o superior.

Pantalla: SuperVGA (1024 x 768) o monitor de mayor resolución con 256 colores.

Software

Sistema operativo: Plataforma de Microsoft Windows.

Microsoft Windows XP SP2 o superior.

Navegador*: Internet Explorer.
Mozilla Firefox.
Google Chrome.

* Ver análisis en tema 6.3

3.3 RECURSOS HUMANOS

El administrador del proyecto comienza evaluando el ámbito del software y seleccionando las habilidades requeridas para completar el desarrollo. Se especifican tanto la posición organizacional (gestor, ingeniero de software, etc.) como la especialidad (telecomunicaciones, bases de datos, cliente/servidor, etc.). El perfil del personal para el desarrollo de este proyecto es Ingeniero en Computación y es el siguiente:

Recurso	
Ing. Marco Ambríz Maguey	Director de Proyecto
Ing. Jonathan Velázquez García	Coordinador de Proyecto
Ing. Mauricio Velázquez Álvarez	Coordinador de Proyecto
Rosaura Apresa Monzón	Desarrollador
Liliana González Jiménez	Desarrollador
Raúl Trinidad Ramírez y Ramírez	Desarrollador

Tabla 2.3 Recursos humanos

CAPÍTULO 4

DISEÑO DEL SISTEMA

4.1 DISEÑO DE LA BASE DE DATOS

Conceptos básicos

Las Bases de Datos (BD) surgen por la necesidad de gestionar cada vez mayores cantidades de información, al referirnos a información nos referimos no sólo a los datos, sino también a las relaciones entre ellos y las restricciones que deben cumplir.

Una BD es una colección organizada de datos, relativa a un problema concreto, que puede ser compartida por un conjunto de usuarios/aplicaciones. ^[5]

DBA (del inglés Database Administrator, ‘Administrador de la Base de Datos’). Administra el uso de la BD y garantiza que funcione apropiadamente. Dentro de sus deberes se encuentra la recuperación de desastres (respaldos y pruebas de respaldos), análisis de desempeño y optimización, y asistencia en el diseño de la base de datos. ^[5]

Diseñador de Base de Datos. Diseña la estructura de la BD, llamado también arquitecto de BD. Actualmente, como las organizaciones se esfuerzan por optimizar sus recursos, el trabajo del diseñador de bases de datos se ha ampliado hasta abarcar más responsabilidades.

Modelo

Los modelos de datos son un grupo de herramientas conceptuales para describir los datos, sus relaciones, su semántica y sus limitantes, no son cosas físicas, son abstracciones que permiten la implementación de un sistema eficiente de BD, siendo este, el primer paso para su diseño; por lo general se refieren a algoritmos, y conceptos matemáticos.

La evolución de los sistemas de BD ha sido impulsada por la búsqueda de nuevas formas de modelar datos del mundo real. Haciendo un resumen de los modelos más reconocidos mostramos la figura 4.1

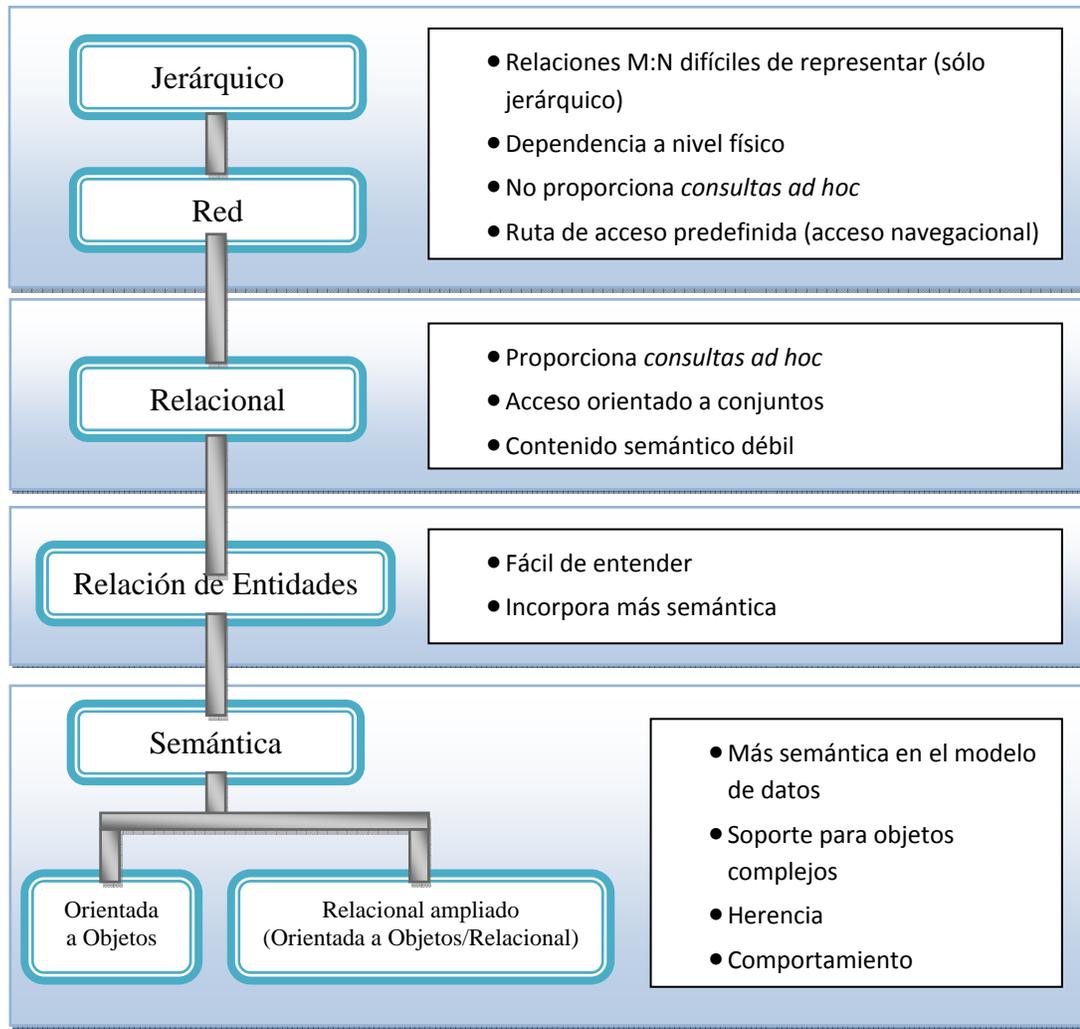


Figura 4.1 Diagrama de los modelos más reconocidos.

El modelo de red comparado con el jerárquico facilitó la representación de relaciones de muchos a muchos, el modelo relacional representa los datos de una forma más simple que los anteriores y un lenguaje de consulta relativamente fácil.

Analizando los modelos de datos, se pueden determinar algunas características que éstos deben tener para que sean ampliamente aceptados:

1. Un modelo de datos debe mostrar un cierto grado de simplicidad conceptual sin que se comprometa el contenido semántico* de la BD.
*La semántica tiene que ver con el comportamiento dinámico de los datos.
2. Un modelo de datos debe representar tanto como sea posible el mundo real.
3. La representación de las transformaciones en el mundo real (comportamiento) debe ajustarse a las características de consistencia e integridad de cualquier modelo de datos.

El modelo Entidad Relación (ER) se ha establecido como el principal modelo de datos para las aplicaciones de procesamiento de datos. Fue propuesto por Peter Chen en 1976 para la definición del esquema conceptual de una base de datos.

Términos del modelo ER:

Entidad

Representa un objeto con existencia física como: una persona, un animal, una casa, etc. (entidad concreta), o un objeto con existencia conceptual como: un puesto de trabajo, una asignatura de clases, un nombre, etc. (entidad abstracta).

Una entidad está descrita y se representa por sus características o atributos. Por ejemplo, la entidad Persona puede llevar consigo las características: Nombre, Apellido, Género, Estatura, Peso, Fecha de nacimiento, por nombrar algunas.

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

Atributos

Los atributos son las propiedades que describen a cada entidad en un conjunto de entidades.

Un conjunto de entidades dentro de una entidad, tiene valores específicos asignados para cada uno de sus atributos, de esta forma, es posible su identificación unívoca. Dos o más entidades diferentes pueden tener los mismos valores para algunos de sus atributos, pero nunca para todos.

En particular, los atributos identificativos son aquellos que permiten diferenciar a una instancia de la entidad de otra distinta. Por ejemplo, el atributo identificativo que distingue a un alumno de otro es su número de cuenta.

Relación

Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su función.

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

Una relación recursiva es una relación donde la misma entidad participa más de una vez en la relación con distintos papeles. El nombre de estos papeles es importante para determinar la función de cada participación.

La cardinalidad con la que una entidad participa en una relación, especifica el número mínimo y el número máximo de correspondencias en las que puede tomar parte cada ocurrencia de dicha entidad.

Dependencia de Existencia

Ésta implica que la existencia de una entidad depende de la existencia de otra. Si la existencia de la entidad a depende de la existencia de entidad b , se dice entonces que a es dependiente por existencia de b . Ésto implica que si se elimina b , también se debe eliminar a . Es entonces como podemos decir que a es una entidad subordinada y b es una entidad dominante. ^[13]

Llaves

Las entidades y las relaciones deben diferenciarse unas de otras en términos de sus atributos, para hacer la distinción se asigna una superllave a cada conjunto de entidades.

Superllave

Es un conjunto de uno o más atributos, que juntos permiten identificar en forma única a una entidad dentro del conjunto de entidades.

Llave Candidato

Es una superllave en la que ningunos de sus subconjuntos es una superllave de la relación. El atributo o conjunto de atributos K de la relación R si y sólo si satisface las siguientes propiedades:

- Unicidad: nunca hay dos entidades en la relación R con el mismo valor de K .
- Irreducibilidad: ningún subconjunto de K tiene la propiedad de unicidad, es decir, no se pueden eliminar componentes de K sin destruir la unicidad.

Llave Primaria

Es una llave candidato que el diseñador de la base de datos elije para identificar una entidad dentro de un conjunto de identidades. Es posible que una entidad no cuente con suficientes atributos para definir una llave primaria, a éstas también las podemos llamar entidades débiles, en consecuencia a las entidades con llave primaria son llamadas entidades fuertes.

Diseño

El diseño de las bases de datos se realiza por lo general en tres fases:

1. Diseño conceptual
Representación abstracta y de alto nivel de la realidad.
2. Diseño lógico
Convierte esta presentación en especificaciones que pueden implantarse en un sistema de cómputo.
3. Diseño físico
Determina las estructuras de almacenamiento físico y los métodos de consulta requeridos para un acceso eficaz a los contenidos de una base de datos.

En el diseño de una base de datos existen algunos puntos importantes a tomar en cuenta para lograr integridad y no redundancia de datos como son los siguientes:

- Correcta definición de objetos (Tablas y Catálogos)
- Establecer convenciones en el nombrado de la base de datos y sus objetos, estandarizarlo y apegarse a él.
- Utilizar el tipo de dato adecuado.
- Todas las tablas y catálogos deberán contener llaves primarias.
- Definir las relaciones entre tablas.
- Asegurarnos de normalizar los datos cuando menos hasta la tercer forma normal, así no comprometeremos el rendimiento.
- Definir aquellos tipos de datos cuya longitud y precisión se ajuste mejor al rango de valores requeridos validos.

Normalización

El objetivo del diseño de las bases de datos relacionales es la generación de un conjunto de esquemas relacionales que permite almacenar la información sin redundancias innecesarias, pero que también nos permita recuperar fácilmente esa información. Un enfoque es el diseño de esquemas que se hallen en una forma normal adecuada. ^[13]

La **primera forma normal (1FN)**, impone un requisito muy elemental a las relaciones; a diferencia de las demás formas normales, no exige información adicional como las dependencias funcionales.

Un dominio es atómico si se considera que los elementos de dominio son unidades indivisibles. Se dice que el esquema de una relación R está en la primera forma normal si los dominios de todos los atributos de R son atómicos.

Un conjunto de nombres es un ejemplo de valor no atómico. Por ejemplo, si el esquema de la relación *empleado* incluyera el atributo *hijos*, los elementos de cuyo dominio son conjuntos de nombres, el esquema no se hallaría en la 1FN.

Un esquema de relación R está en la **segunda forma normal (2FN)** si cada atributo A de R cumple uno de los criterios siguientes:

- Aparece en una clave candidata.
- No es parcialmente dependiente de una clave candidata.

Una dependencia funcional $x \rightarrow y$ se denomina dependencia parcial si hay un subconjunto adecuado z de x tal que $z \rightarrow y$. Se dice que y es parcialmente dependiente de x .

Un esquema de relación R está en **tercera forma normal (3FN)** respecto a un conjunto F de dependencias funcionales si, para todas las dependencias funcionales de F^+ de la forma $x \rightarrow y$, se cumple al menos una de las siguientes condiciones:

- $X \rightarrow y$ es una dependencia funcional trivial.
- X es una superclave de R .
- Cada atributo A de $y-x$ está contenido en alguna clave candidata de R .

Se dice que algunas dependencias funcionales son triviales porque las satisfacen todas las relaciones. Por ejemplo, $A \rightarrow A$ las satisfacen todas las relaciones que impliquen el atributo A .

Un esquema de relación R está en la **cuarta forma normal (4FN)** con respecto a un conjunto F de dependencias funcionales y multivaloradas si, para todas las dependencias multivaloradas de F^+ de la forma $x \twoheadrightarrow y$, se cumple, como mínimo, una de las condiciones siguientes

- $x \twoheadrightarrow y$ es una dependencia multivalorada trivial.
- X es una superclave del esquema R .

Las dependencias funcionales impiden que ciertas tuplas estén en una relación. Si $A \rightarrow B$, entonces no puede haber dos tuplas con el mismo valor de A y diferentes valores de B . Las dependencias multivaloradas, por otro lado, no impiden la existencia de esas tuplas. En lugar de eso, exigen que estén presentes en la relación otras tuplas de una cierta forma.

4.1.1 Modelo entidad relación

Con base en los conceptos anteriores, a continuación se muestran las entidades que conforman la base de datos que finalmente darán forma al modelo ER (Figura 4.n):

Ctl_Areas: Contiene las áreas correspondientes a la CSC

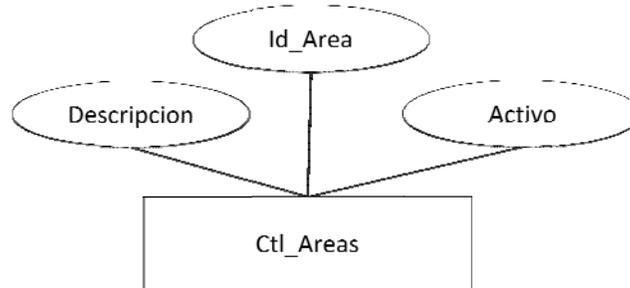


Figura 4.2 Entidad “Ctl_Areas”

Ctl_CalifUsuario: Contiene las calificaciones que el usuario puede asignar a la atención de su solicitud.

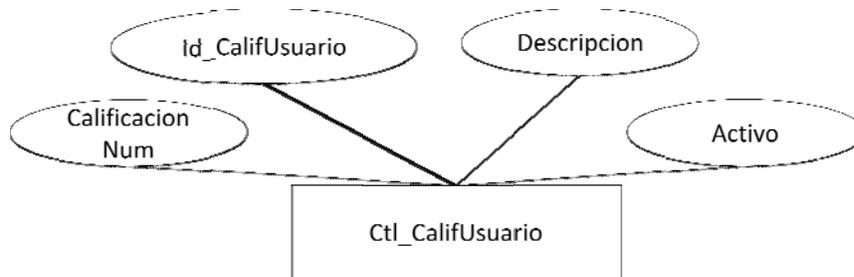


Figura 4.3 Entidad “Ctl_CalifUsuario”

Ctl_Empresa: Contiene las empresas externas que pueden realizar servicios de alguna área de la CSC.

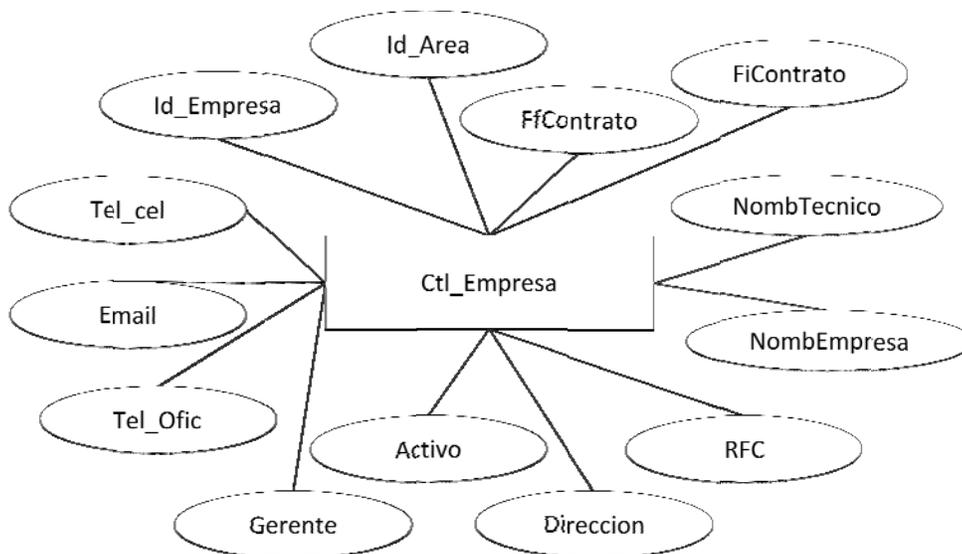


Figura 4.4 Entidad “Ctl_Empresa”

Ctl_Prioridad: Contiene las prioridades que se pueden asignar a una solicitud.

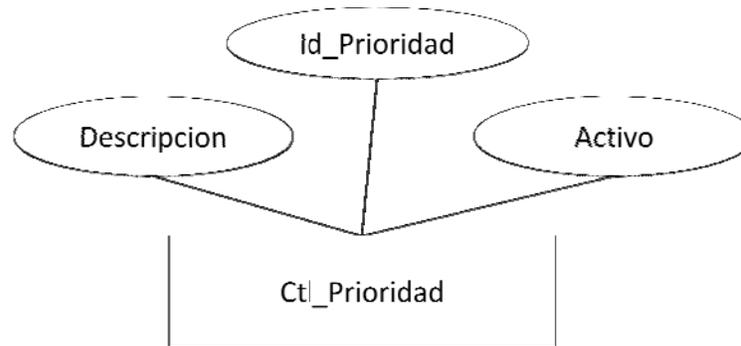


Figura 4.5 Entidad "Ctl_Prioridad"

Ctl_Problemas: Contiene los problemas correspondientes a las diferentes áreas de la CSC.

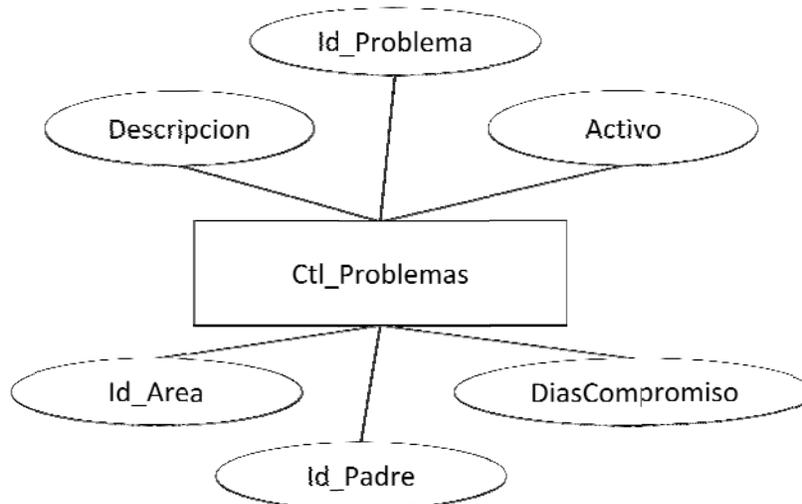


Figura 4.6 Entidad "Ctl_Prioridad"

Ctl_StatusSolicitud: Contiene los status que se pueden asignar a una solicitud.

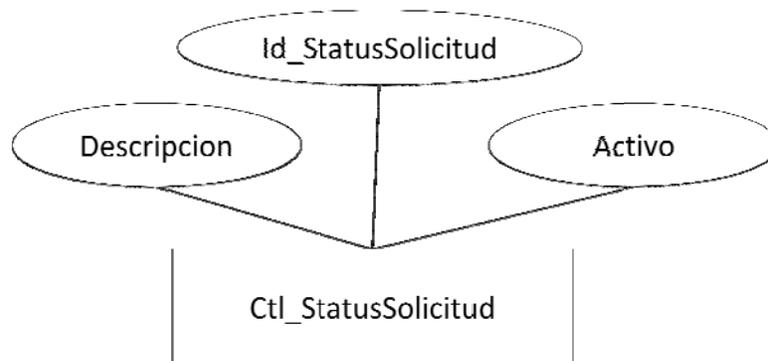


Figura 4.7 Entidad "Ctl_StatusSolicitud"

Ctl_Tipo_Admin: Contiene los tipos de administradores del sistema.

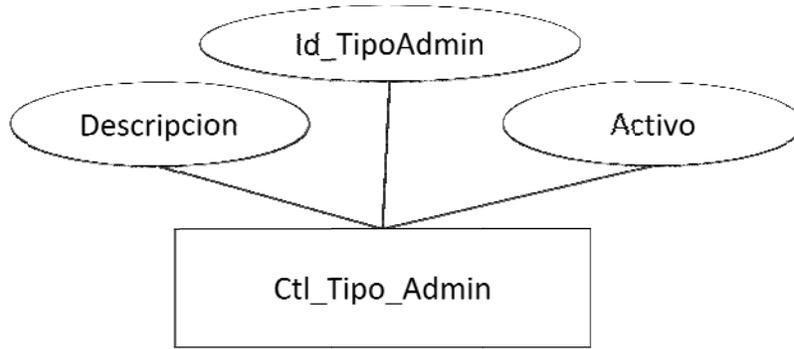


Figura 4.8 Entidad "Ctl_Tipo_Admin"

Ctl_TipoSolicitud: Contiene el tipo de solicitud que se puede realizar en el sistema.

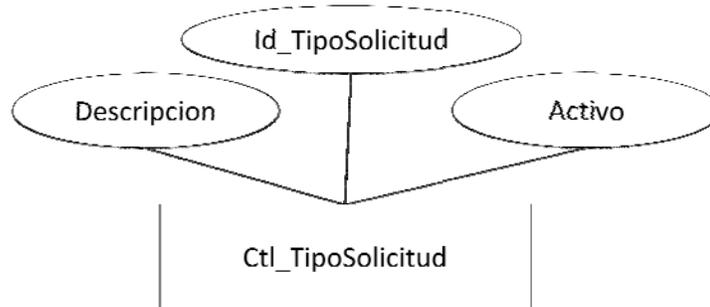


Figura 4.9 Entidad "Ctl_TipoSolicitud"

Edificio: Contiene los edificios existentes en el II.

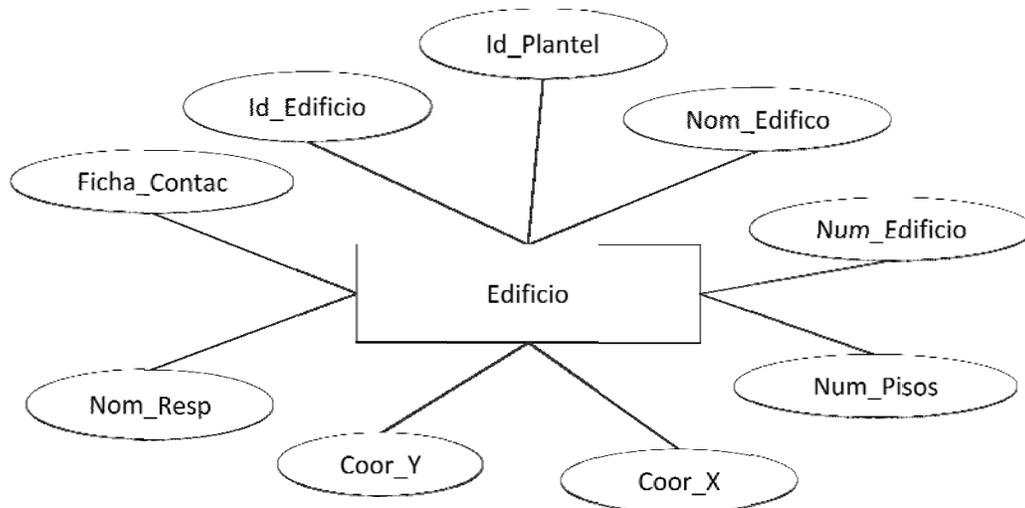


Figura 4.10 Entidad "Edificio"

Espacio: Contiene los espacios existentes en Ciudad Universitaria.

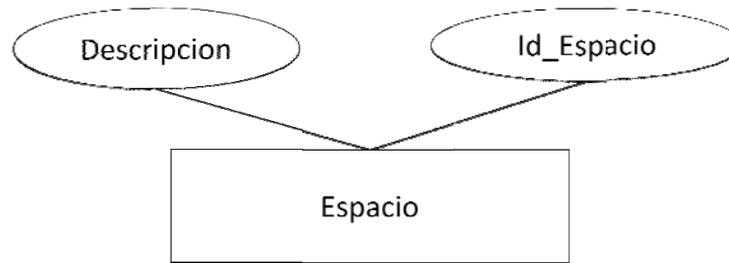


Figura 4.11 Entidad "Espacio"

Número: Contiene los números de los espacios.

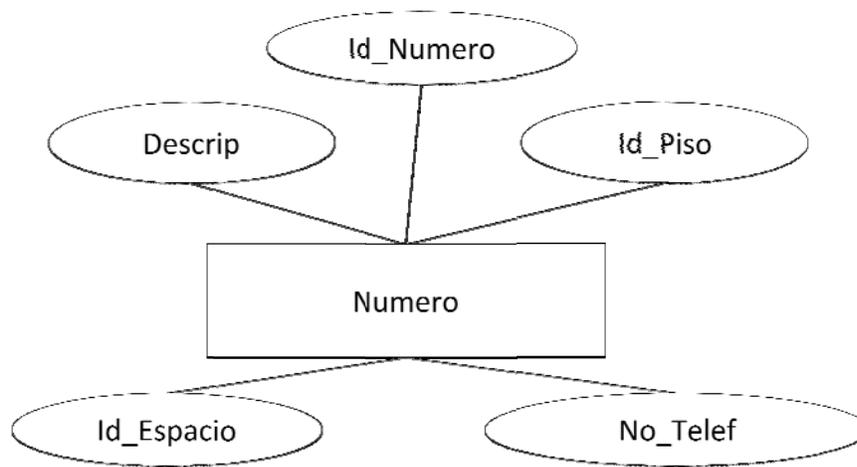


Figura 4.12 Entidad "Numero"

Piso: Contiene los pisos de los números de edificios.

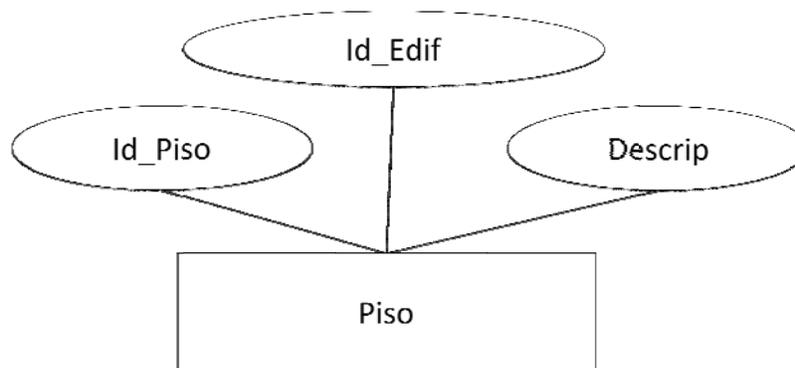


Figura 4.13 Entidad "Piso"

Plantel: Contiene los planteles existentes en Ciudad Universitaria.

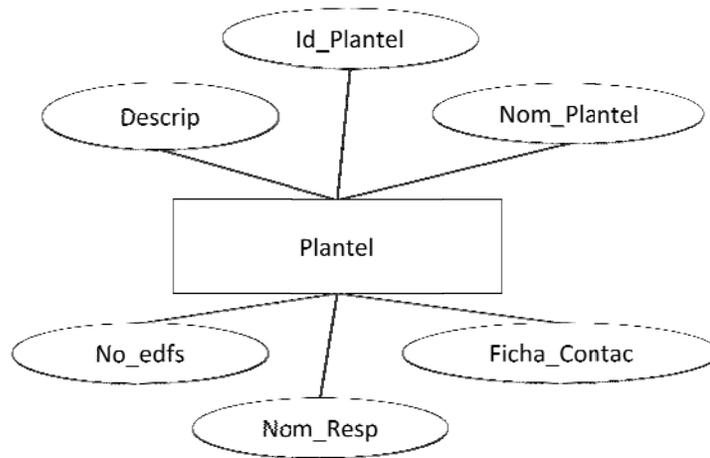


Figura 4.14 Entidad “Plantel”

Tbl_Administrador: Contiene los administradores del sistema.

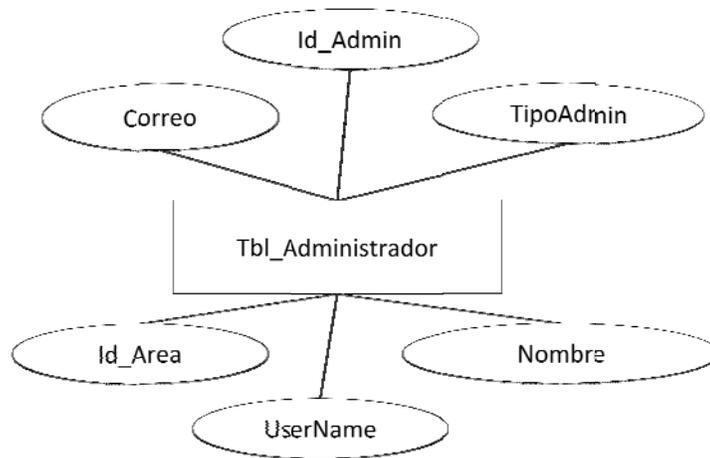


Figura 4.15 Entidad “Tbl_Administrador”

Tbl_AdminiStatusSolic: Contiene los status que los administradores pueden asignar a una solicitud.

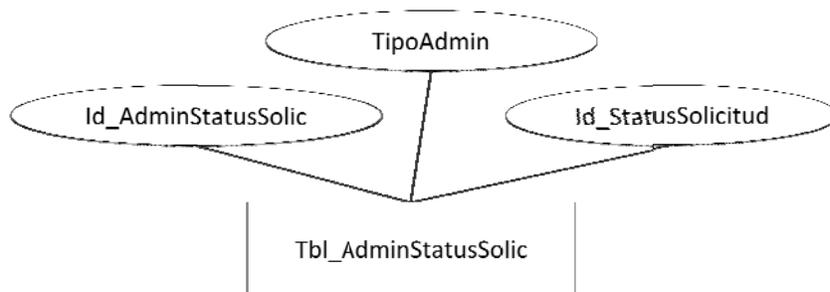


Figura 4.16 Entidad “Tbl_AdminiStatusSolic”

Tbl_Asignacion: Contiene los status que los administradores pueden asignar a una solicitud.

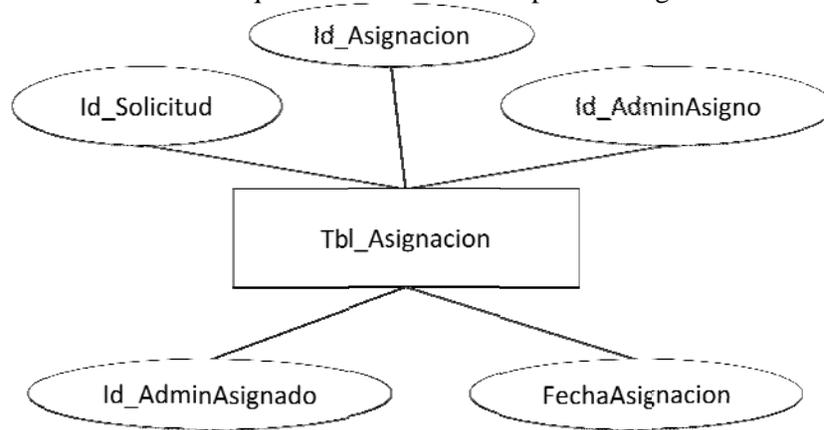


Figura 4.17 Entidad “Tbl_Asignacion”

Tbl_Aux_Not_Area: Relaciona las noticias por área.

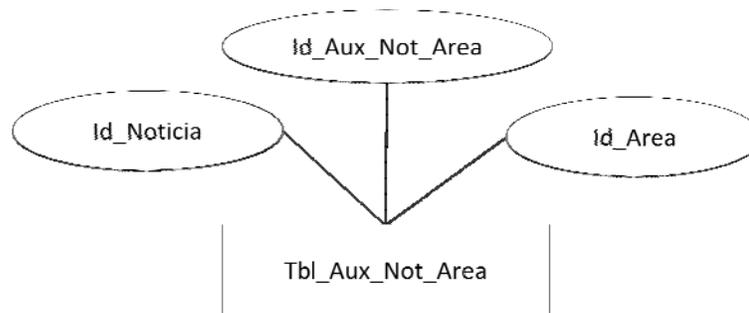


Figura 4.18 Entidad “Tbl_Aux_Not_Area”

Tbl_DatosUsrHist: Contiene los datos históricos de los usuarios.

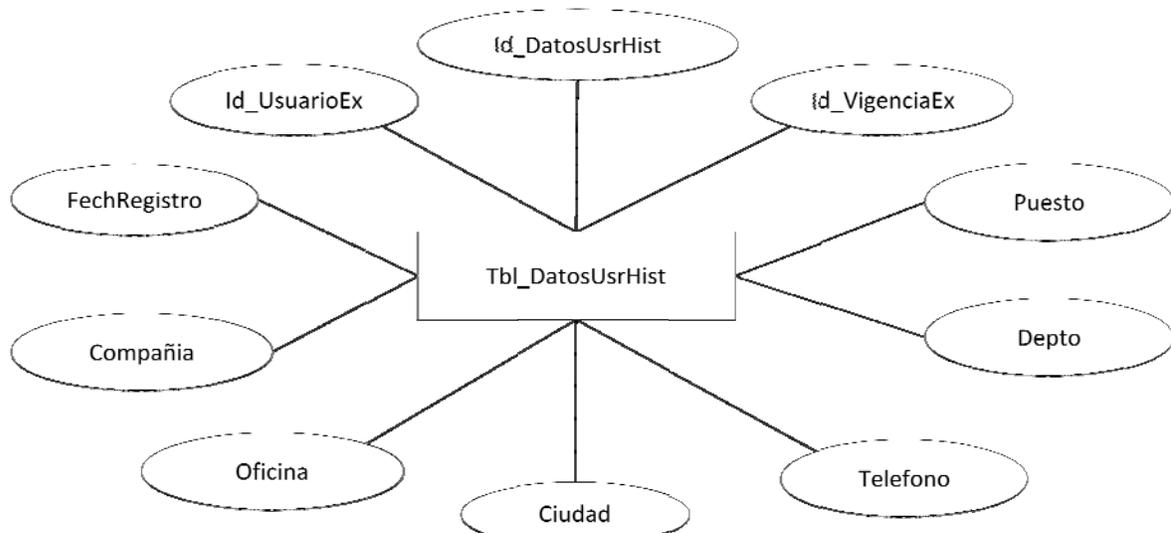


Figura 4.19 Entidad “Tbl_DatosUsrHist”

Tbl_EquipoComputo: Contiene los datos del equipo de computo registrado en el sistema de inventarios.

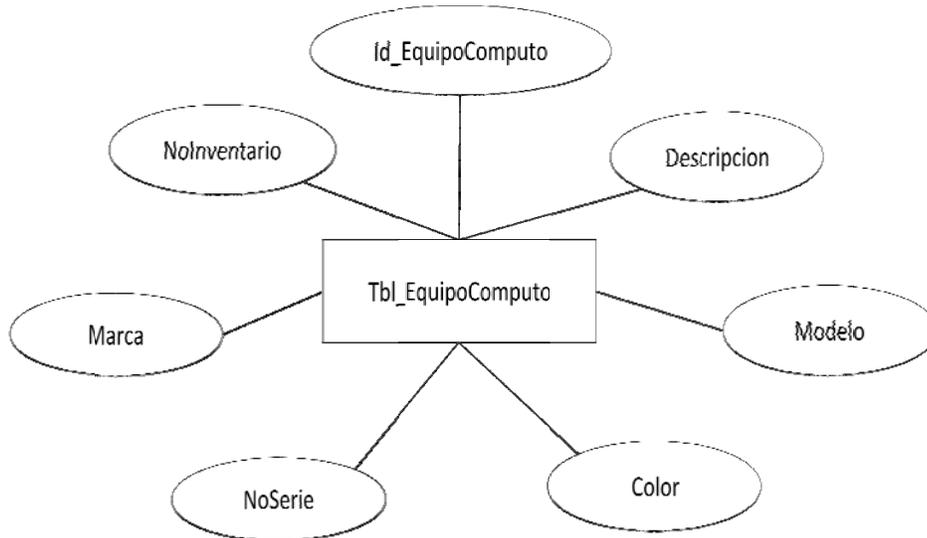


Figura 4.20 Entidad "Tbl_EquipoComputo"

Tbl_EquipoEx: Contiene los datos del equipo externo.

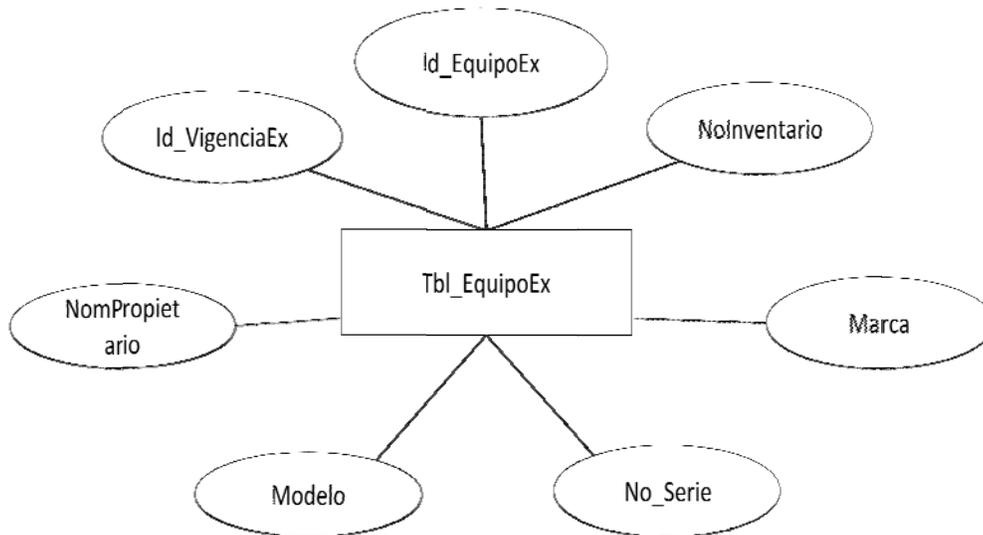


Figura 4.21 Entidad "Tbl_EquipoEx"

Tbl_HistoricoStatus: Contiene los datos históricos de los status de una solicitud.

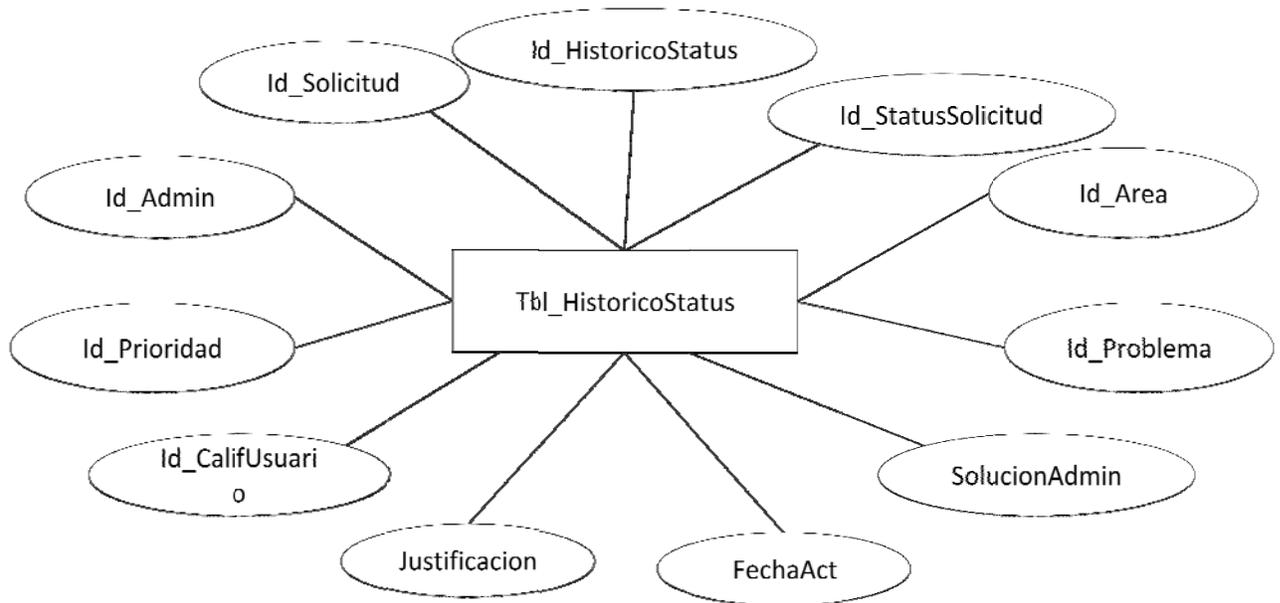


Figura 4.22 Entidad “Tbl_HistoricoStatus”

Tbl_Menu: Contiene las opciones de menú por tipo de usuario.

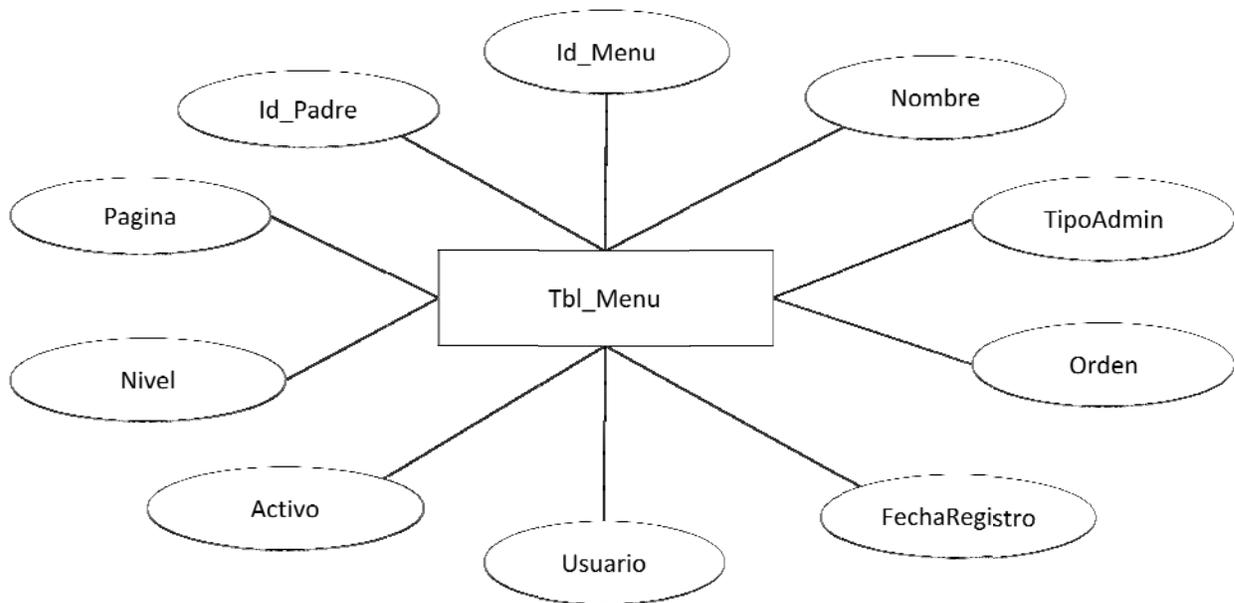


Figura 4.23 Entidad “Tbl_Menu”

Tbl_Noticias: Contiene las noticias para los usuarios.

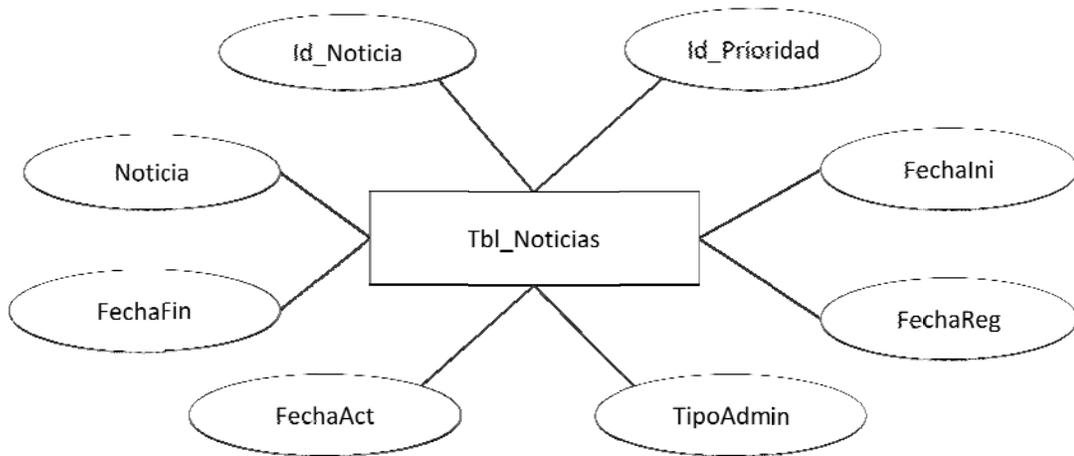


Figura 4.24 Entidad "Tbl_Noticias"

Tbl_ServicioEx: Contiene las solicitudes que son enviadas a servicio externo.

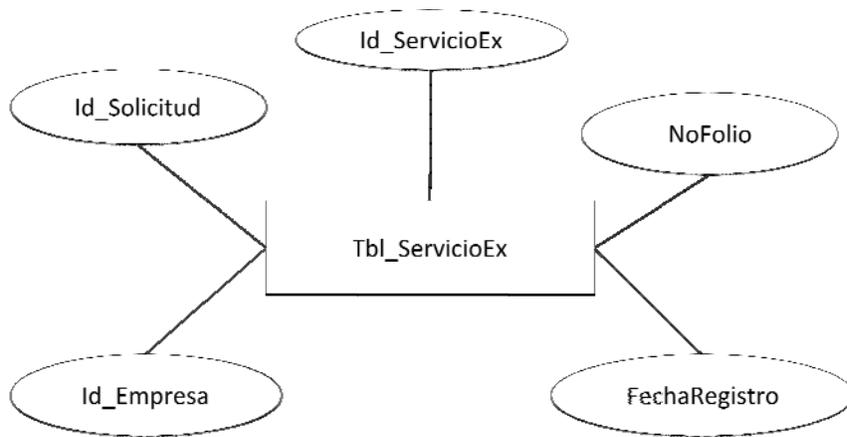


Figura 4.25 Entidad "Tbl_ServicioEx"

Tbl_Software: Contiene el software de referencia para los usuarios.

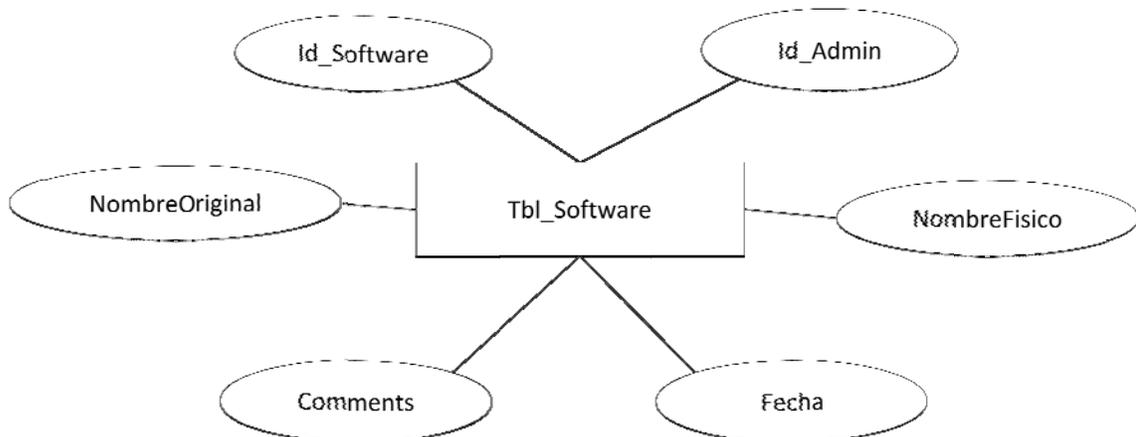


Figura 4.26 Entidad "Tbl_Software"

Tbl_Solicitud: Contiene las solicitudes de servicio registradas en el sistema.

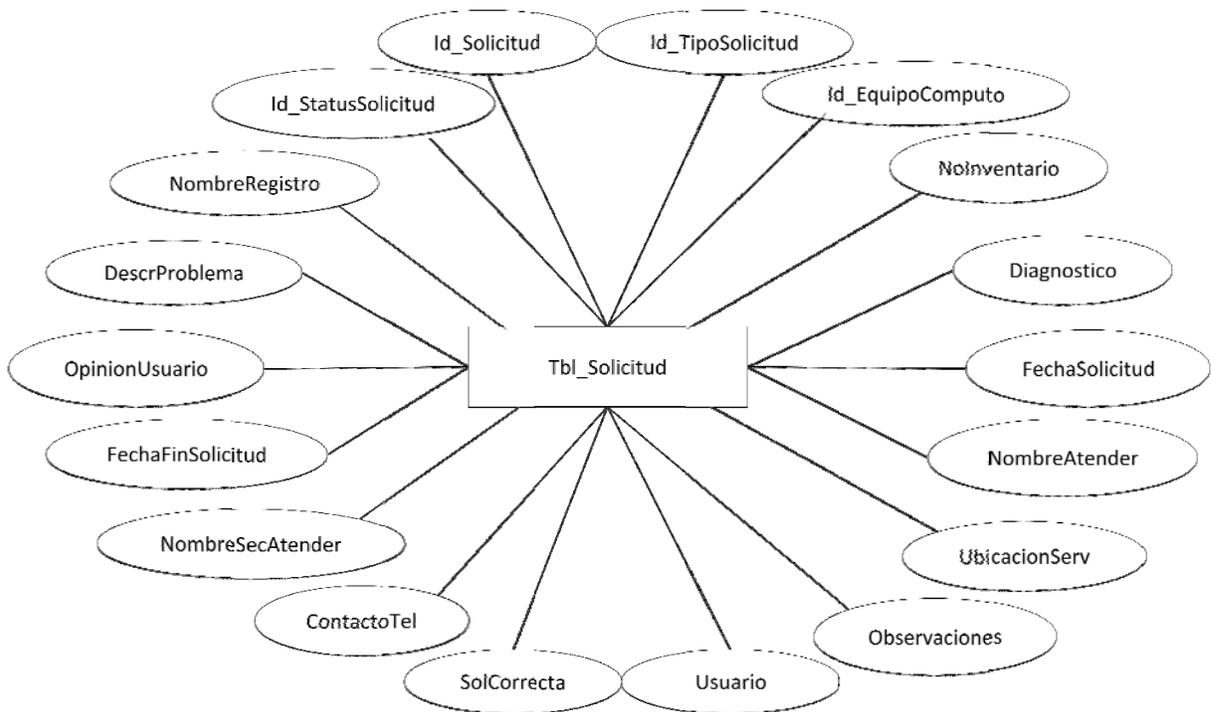


Figura 4.27 Entidad “Tbl_Solicitud”

Tbl_UsuarioEx: Contiene el registro de usuarios externos.

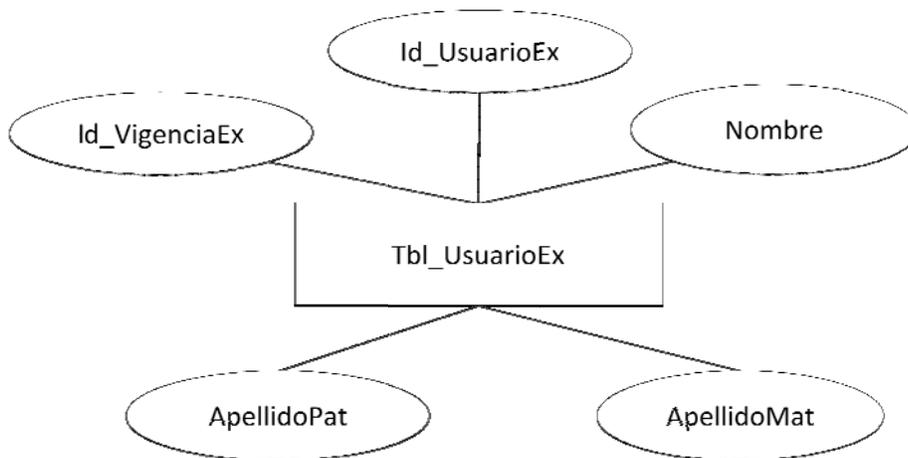


Figura 4.28 Entidad “Tbl_UsuarioEx”

Tbl_VigenciaEx: Contiene el registro de vigencia externa.

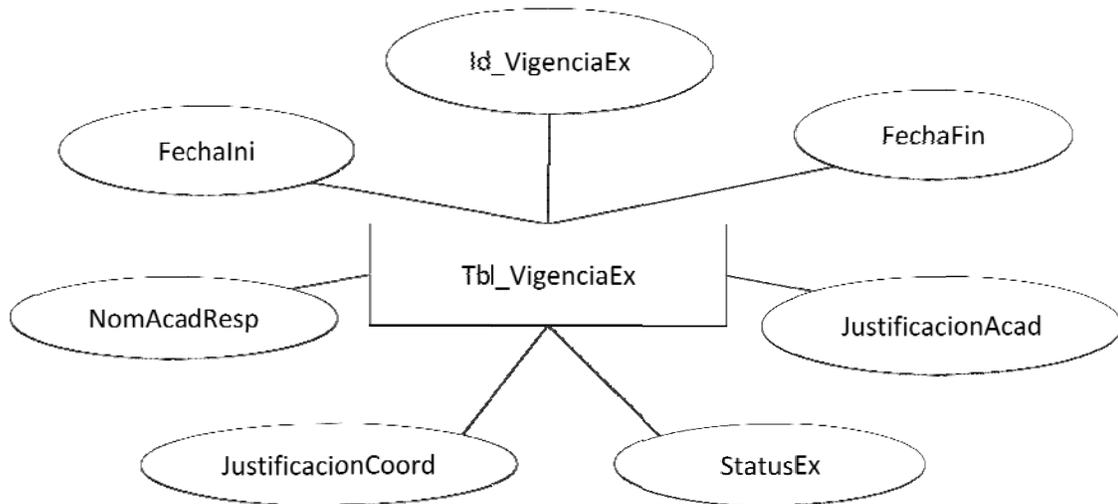


Figura 4.29 Entidad "Tbl_VigenciaEx"

Tbl_VigHistEquipo: Contiene el registro histórico de vigencias de equipos externos.

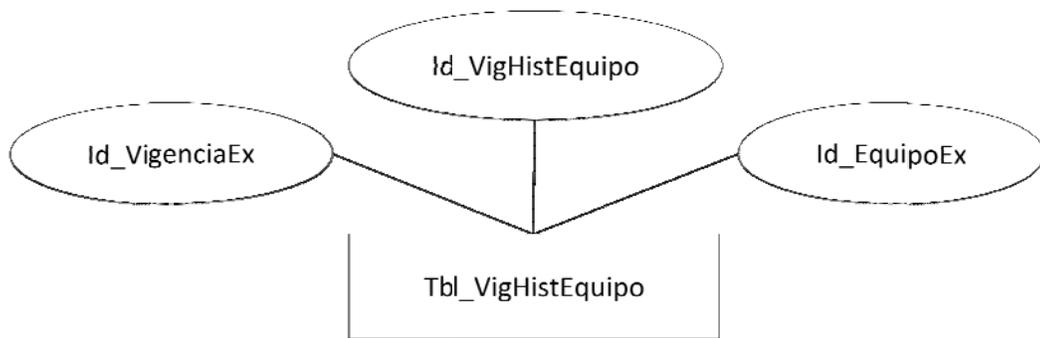


Figura 4.30 Entidad "Tbl_VigHistEquipo"

Tbl_VigHistUsuario: Contiene el registro histórico de vigencias de usuarios externos.

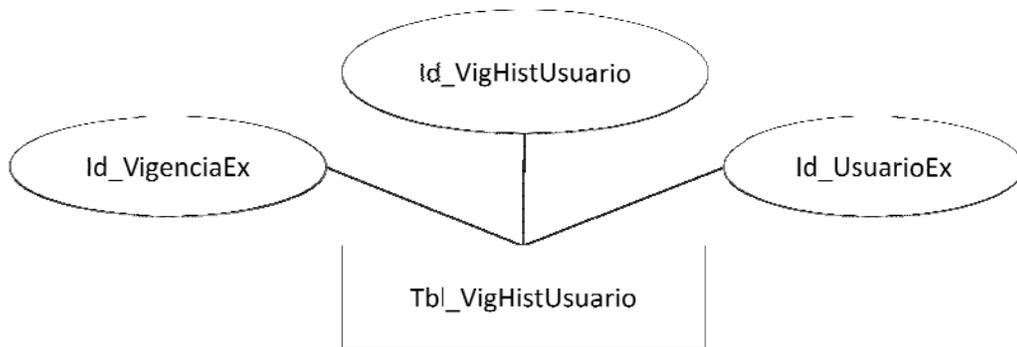


Figura 4.31 Entidad "Tbl_VigHistUsuario"

4.2 DISEÑO DE CASOS DE USO

Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo, y que producen un resultado observable de valor para un actor concreto. ^[1]

Los casos de uso fueron introducidos por Jacobson en 1992, sin embargo, la idea de especificar un sistema a partir de su interacción con el entorno es original de Mc Menamin y Palmer, dos precursores del análisis estructurado. Gran parte de los más reconocidos especialistas en métodos orientados a objetos coincidieron en considerar a los casos de uso como una excelente forma de especificar el comportamiento externo de un sistema. De esta forma, la notación de los casos de uso fue incorporada al lenguaje estándar de modelado UML propuesto por Ivar Jacobson, James Rumbaugh y Grady Booch, tres de los precursores de las metodologías de análisis y diseño orientado a objetos, y avalado por las principales empresas que desarrollan software en el mundo. UML va en camino de convertirse en un estándar para modelado de sistemas de software de amplia difusión.

Los requisitos funcionales se expresan como casos de uso en un modelo de casos de uso. Este modelo ayuda al cliente, a los usuarios y a los desarrolladores a llegar a un acuerdo sobre cómo utilizar el sistema. La mayoría de los sistemas tienen muchos tipos de usuarios. Cada tipo de usuarios se representa mediante un actor. Los actores utilizan el sistema al interactuar con los casos de uso. Todos los actores y casos de uso del sistema forman un modelo de casos de uso.

Los actores se comunican con el sistema mediante el envío y recepción de mensajes hacia y desde el sistema. Para el Sistema de Solicitudes, se identifican los siguientes actores:

- Usuario: Miembros de la comunidad del Instituto que pueden realizar una solicitud de servicio.
- Administrador 1: Jefes de área y coordinador de la CSC.
- Administrador 2: Personal de área de la CSC.

Los diagramas de caso de uso para cada uno de los actores del sistema se detallan del punto 4.2.1 al 4.2.3 de acuerdo a la notación básica de UML(Figura 4.32)

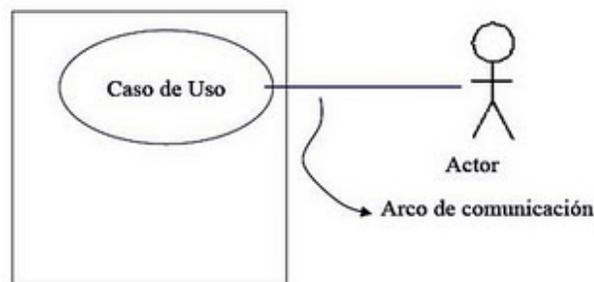


Figura 4.33 Notación básica de UML

4.2.1 Casos de uso para administrador 1

Para el administrador 1, las actividades identificadas de acuerdo a los requerimientos recabados, se muestran en el diagrama de casos de uso de la figura 4.34

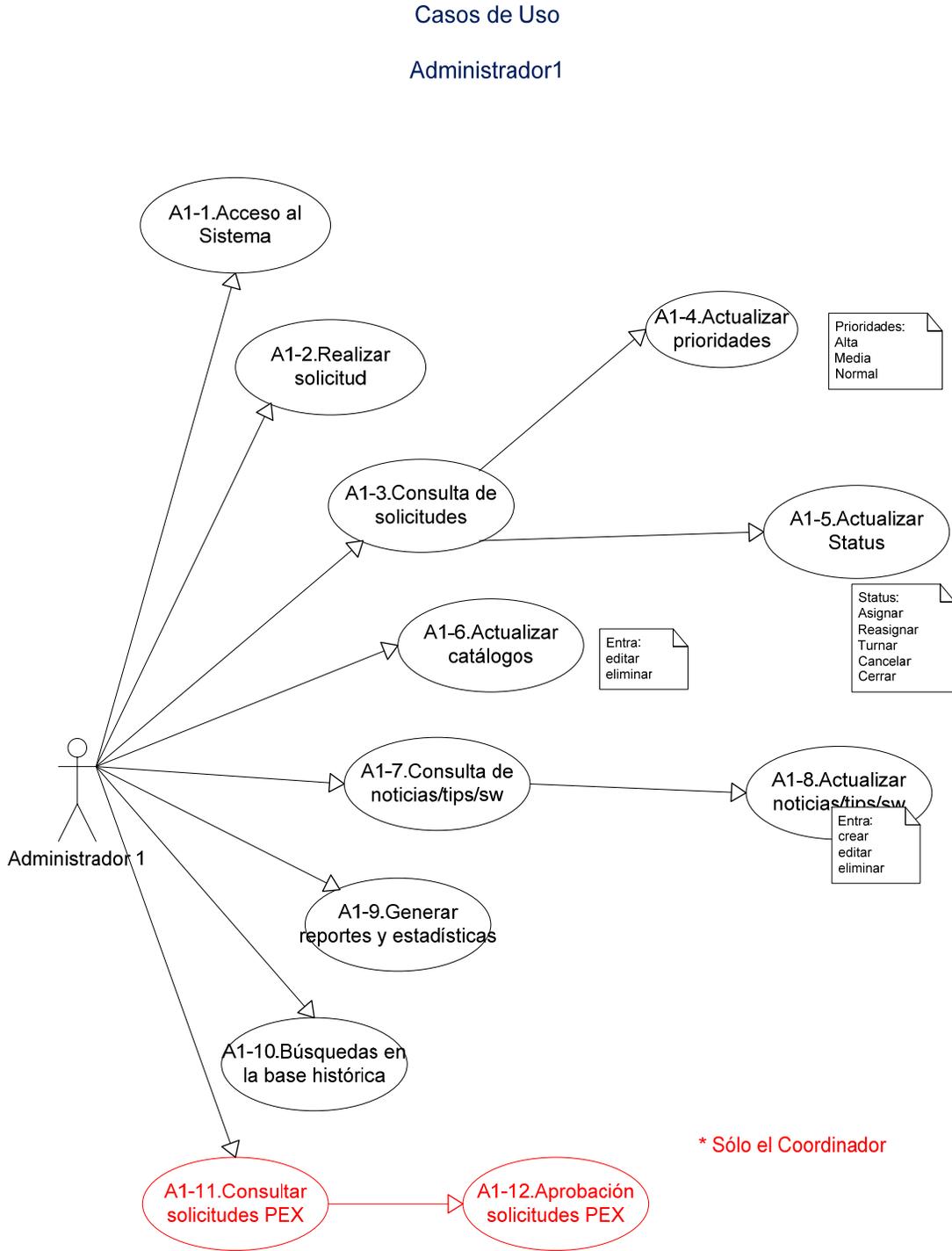


Figura 4.34 Casos de uso del Administrador 1

Un ejemplo del formato que se usó para la descripción de cada uno de los casos de uso del Administrador1 es el siguiente:

Caso de uso: Generar reportes y estadísticas.

Actor: Administrador

Descripción: Se pueden generar reportes y estadísticas de la información más relevante y requerida por el Coordinador de Sistemas de Cómputo.

Pre-requisitos:

- Deben existir solicitudes cerradas. Sólo tendrán acceso el Coordinador y jefes de área.
- Haber ejecutado el caso de uso Ingresar al sistema.

Flujo del caso de uso:

Actor		Sistema		
Paso	Acción	Paso	Acción	Id Excepción
1	Selecciona la opción “Reportes y estadísticas”.	2	Muestra los tipos de reportes y/o estadísticas.	
3	Selecciona el tipo de reporte y/o estadística.	4	Genera y muestra el reporte y/o estadísticas.	
5	Selecciona la opción “Generar en Word o PDF”	6	Genera el reporte en Word o PDF	

Excepciones del caso de uso:

Ninguna.

Post-condiciones:

Obtener información de la actividad realizada por parte del personal de la CSC.

El correspondiente diagrama de flujo para el caso de uso “Generar reportes y estadísticas” se muestra en la figura 4.35

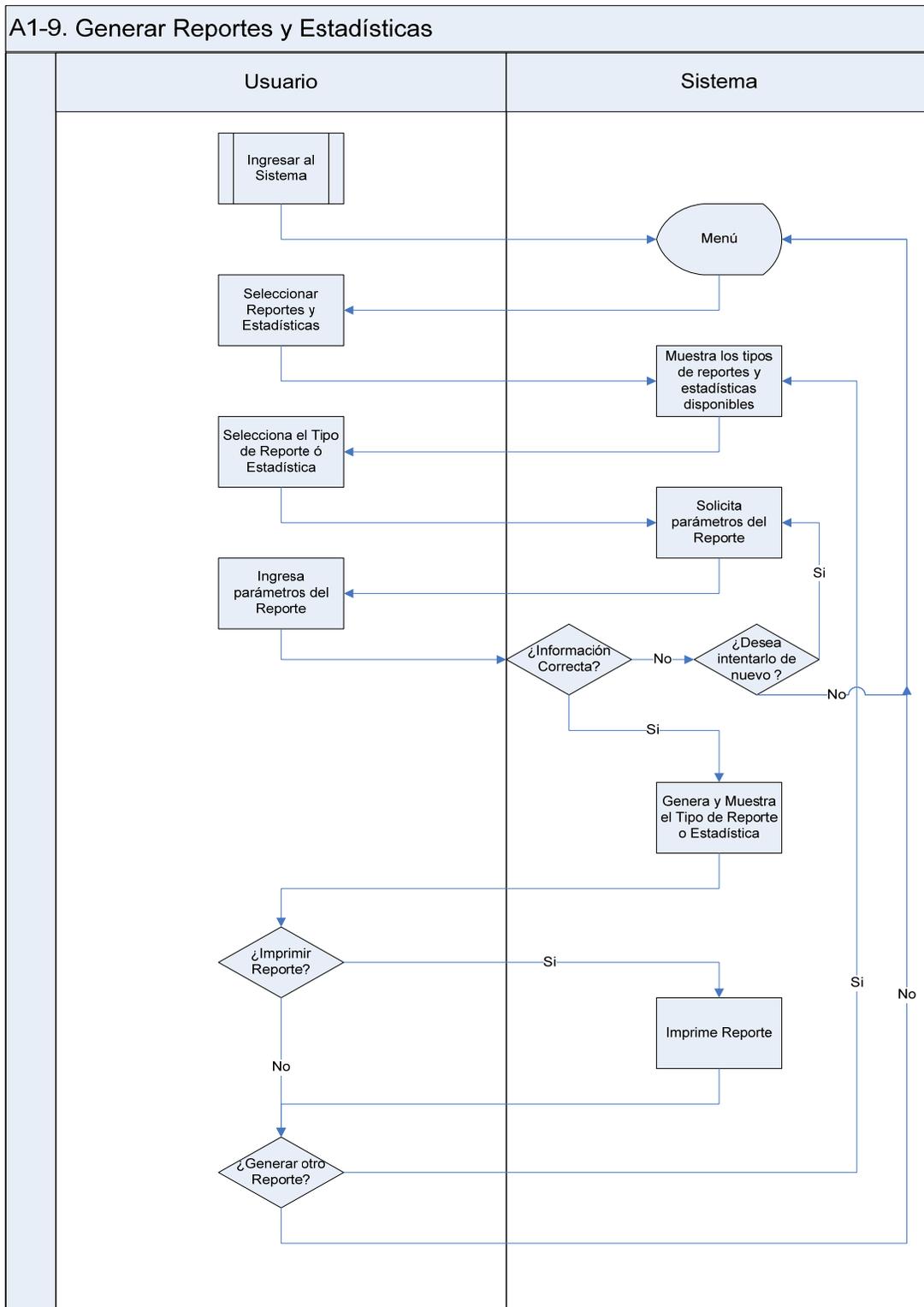


Figura 4.35 Diagrama de caso de uso “Generar reportes y estadísticas”

4.2.2 Casos de uso para administrador 2

Para el administrador 2, las actividades identificadas de acuerdo a los requerimientos recabados, se muestran en el diagrama de casos de uso de la figura 4.36

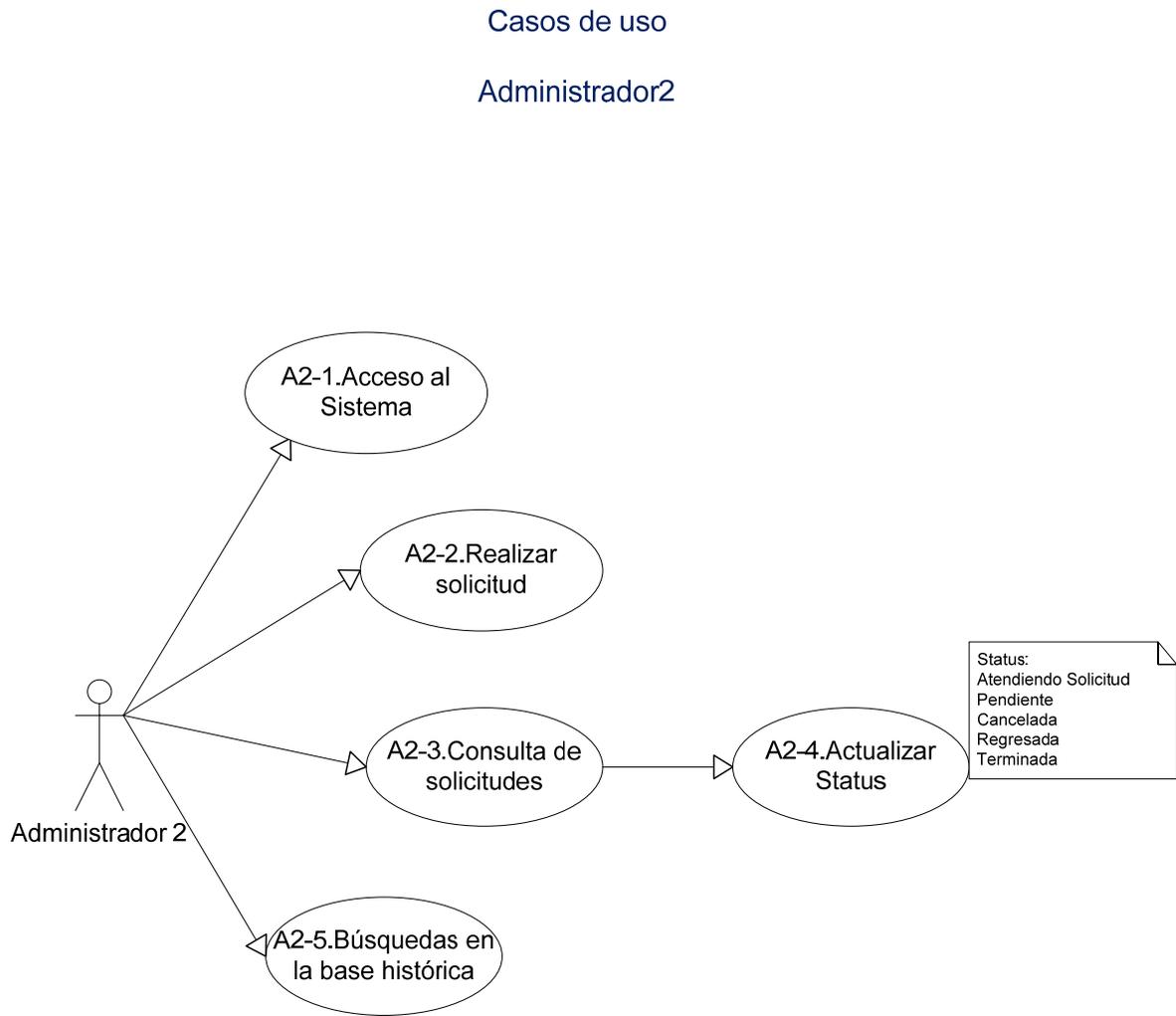


Figura 4.36 Casos de uso del Administrador 2

Un ejemplo del formato que se usó para la descripción de cada uno de los casos de uso del Administrador2 es el siguiente:

Caso de uso: Búsquedas en la base histórica.

Actor: Administrador

Descripción: El personal de cómputo podrá realizar búsquedas en la base histórica de todas las solicitudes concluidas que correspondan al área, por nombre de usuario, inventario equipo, etc.

Pre-requisitos:

- Debe existir por lo menos un registro de solicitudes concluidas.
- Haber ejecutado el caso de uso Ingresar al sistema.

Flujo del caso de uso:

Actor		Sistema		
Paso	Acción	Paso	Acción	Id Excepción
1	Selecciona la opción “Búsquedas históricas”.	2	Muestra el tipo de búsquedas que puede realizar.	
3	Selecciona el tipo de búsqueda.	4	Muestra las solicitudes encontradas.	

Excepciones del caso de uso: Ninguna.

Post-condiciones:

Consulta de solicitudes históricas.

El correspondiente diagrama de flujo para el caso de uso “Búsquedas en la base histórica” se muestra en la figura 4.37

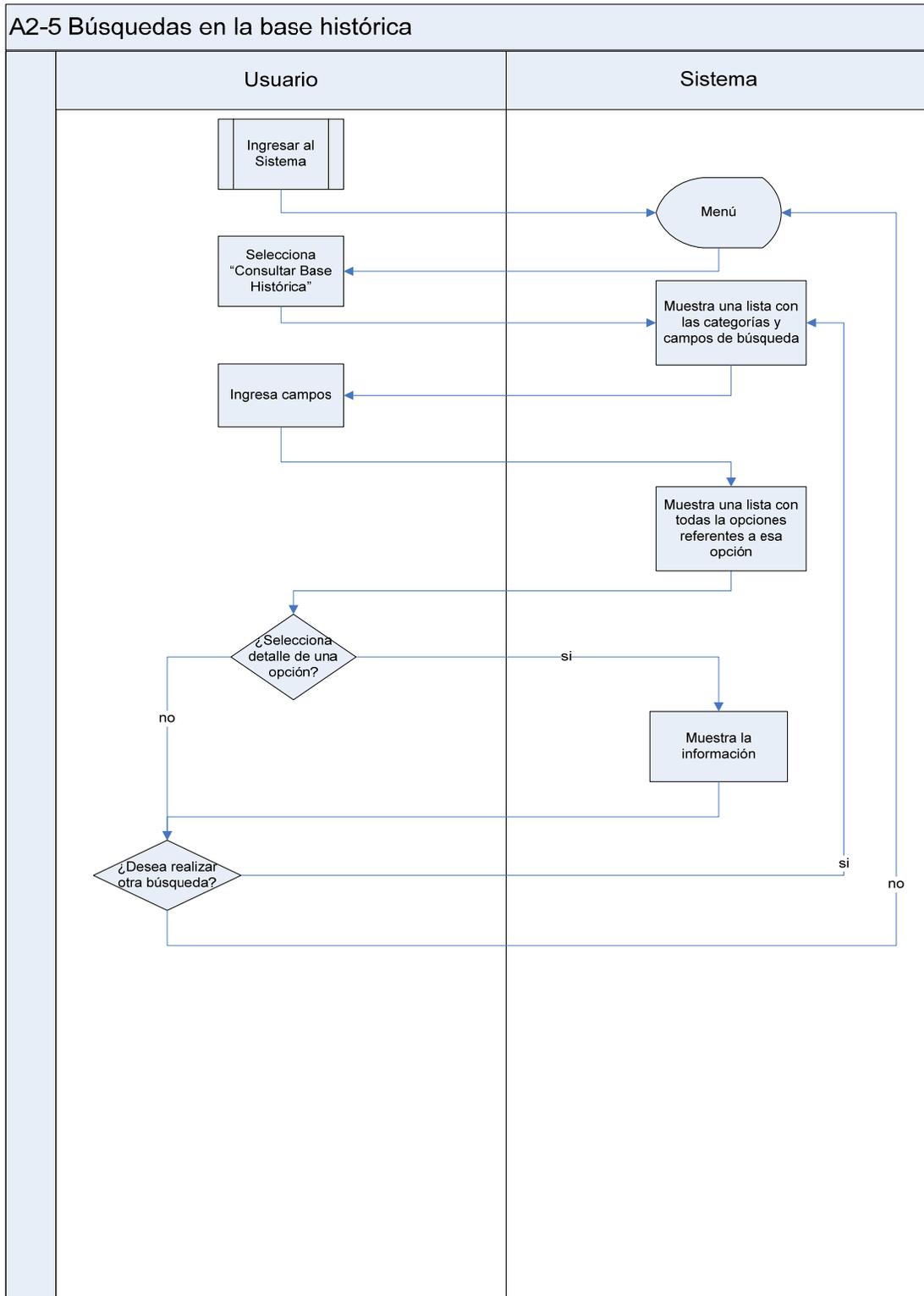


Figura 4.37 Diagrama de caso de uso “Búsquedas en la base histórica”

4.2.3 Casos de uso para usuarios

Para los usuarios, las actividades identificadas de acuerdo a los requerimientos recabados, se muestran en el diagrama de casos de uso de la figura 4.38

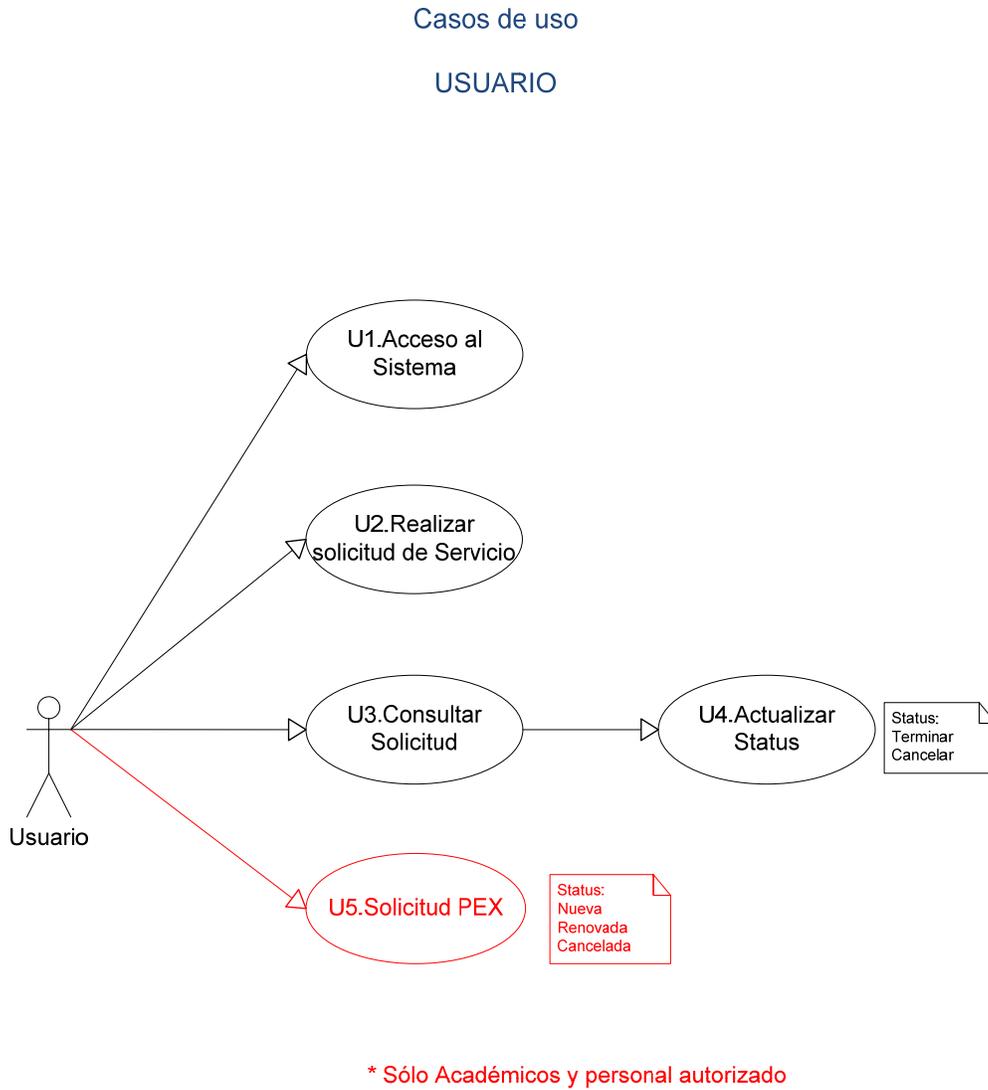


Figura 4.38 Casos de uso del usuario

Un ejemplo del formato que se usó para la descripción de cada uno de los casos de uso del Usuario es el siguiente:

Caso de uso: Realizar solicitud

Actor: Usuario.

Descripción: El usuario realiza una solicitud con la descripción de sus necesidades.

Pre-requisitos:

- Ser usuario validado.
- El equipo a ser revisado debe estar registrado en el Sistema de Inventarios, de lo contrario deberá solicitar autorización para registrar el equipo externo.
- Haber ejecutado el caso de uso Ingresar al sistema.

Flujo del caso de uso:

Actor		Sistema		
Paso	Acción	Paso	Acción	Id Excepción
1	Selecciona la opción “Realizar solicitud” del menú.	2	Muestra en pantalla los datos personales del usuario.	
3	Proporciona el número de inventario del equipo y presiona el botón “Continuar”.	4	Valida el número de inventario proporcionado.	e1, e2
		5	Muestra en pantalla los datos del equipo.	
6	Captura la información necesaria para realizar la solicitud.			
7	Presiona el botón “Enviar solicitud”.	8	Valida la información capturada.	e3
		9	Guarda la información capturada.	
		10	Envía al usuario y administradores notificación vía correo electrónico con un número de registro para poder consultar el avance de la solicitud realizada.	
		11	Muestra el detalle de la solicitud.	
12	Selecciona la opción	13	Muestra las opciones de impresión.	

	“Imprimir comprobante”			
14	Imprime el comprobante.			

Excepciones del caso de uso:

Id	Descripción	Acción
e1	Información incorrecta.	Revisar y corregir información capturada.
e2	Número no existente en la base de datos.	Hacer petición de registro del equipo.
e3	Información incorrecta.	Revisar y corregir información capturada.

Post-condiciones:

Registro de la información de la solicitud realizada.

El correspondiente diagrama de flujo para el caso de uso “Realizar solicitud” se muestra en la figura 4.39

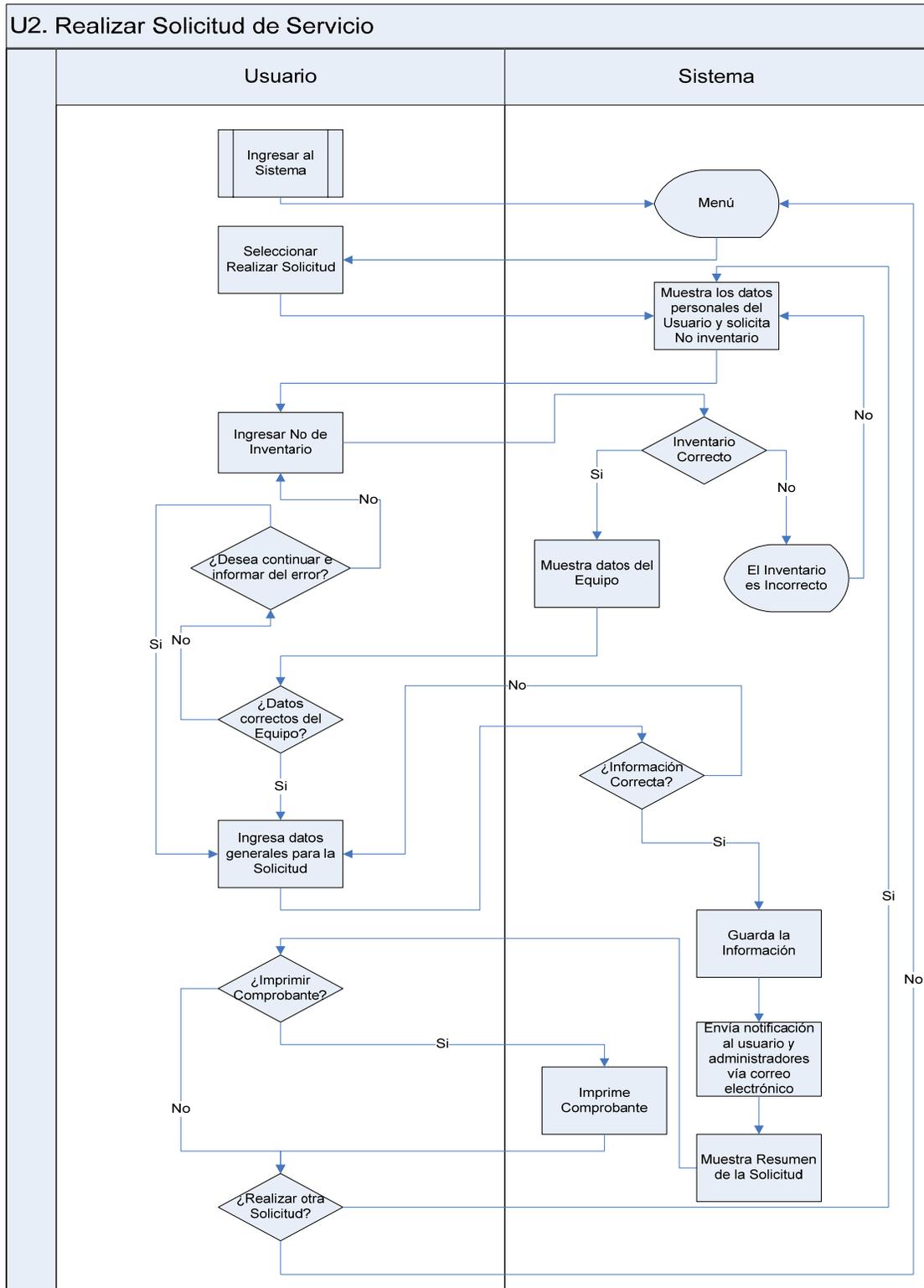


Figura 4.39 Diagrama de caso de uso “Realizar solicitud”

4.2.4 Diseño de la interfaz

Interfaz es el conjunto de elementos gráficos y/o multimedia que permite la interacción del usuario con la computadora o más concretamente con nuestro software. Al realizar un buen diseño de interfaz de usuario, debemos realizar una investigación previa que nos proporcione información para asegurar el éxito de la misma, tal como:

- **Tipo de Usuario**
La interfaz será diseñada de acuerdo al tipo de usuario. Si es un usuario inexperto, la interfaz deberá ser muy amigable e intuitiva, llevando de la mano al usuario; por el contrario si el usuario es experto, la interfaz será más flexible con opciones de ajustes y parámetros de acuerdo a su conocimiento.
- **Tipo de trabajo**
Si el software será usado para una actividad repetitiva, la interfaz deberá ser ágil, dando prioridad a que todo se realice presionando teclas fáciles de recordar y de preferencia maximizando el uso del ratón.
- **Dispositivos de interfaz**
El tipo de dispositivo de interfaz empleado afecta también la interfaz de usuario para explotar las características propias del dispositivo.

Buscando una interfaz que le resulte familiar a los usuarios, se planteo una solución en la que la navegación por el sistema sea intuitiva y sencilla, cabe señalar que dependiendo del rol que se juegue en el sistema, el menú de opciones será un árbol diferente de navegación como ya lo pudimos notar en los diagramas de casos de uso del capítulo anterior.

También tomamos en cuenta algunos principios básicos que nos guiaron en el diseño:

- **Uniformidad**
Es fundamental no saturar nuestra interfaz de colores, debemos mantener en todas las pantallas los mismos colores, tipos de letras y tamaños.
- **Alineación**
La correcta alineación permite identificar el flujo de entrada de datos, que el usuario seguirá, y guiar la vista del usuario hacia donde queremos que vea.
- **Encuadre**
Es conveniente delimitar áreas que permitan identificar los distintos tipos de información a capturar, ya que esto le permite al usuario tener una lectura eficiente.

MAPA DE NAVIGACIÓN

Las siguientes figuras ilustran la navegación que puede realizar el usuario y los administradores en el sistema:

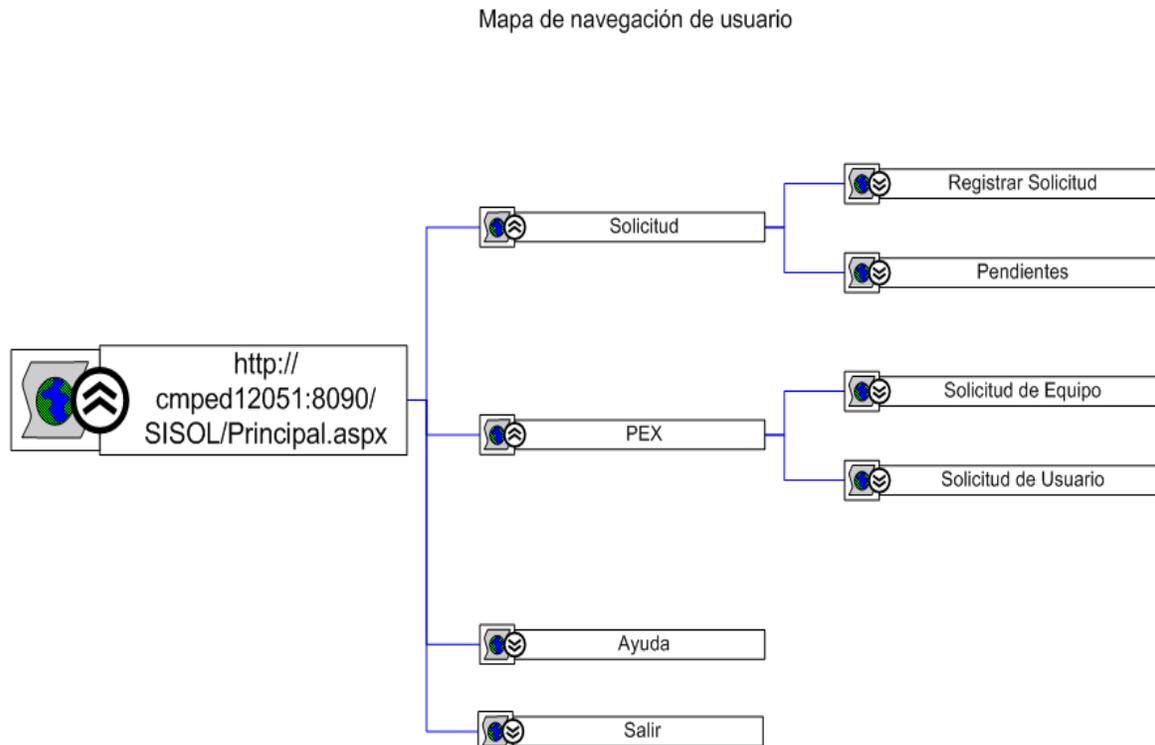


Figura 4.40 Mapa de navegación para usuarios

Mapa de navegación de administrador 1

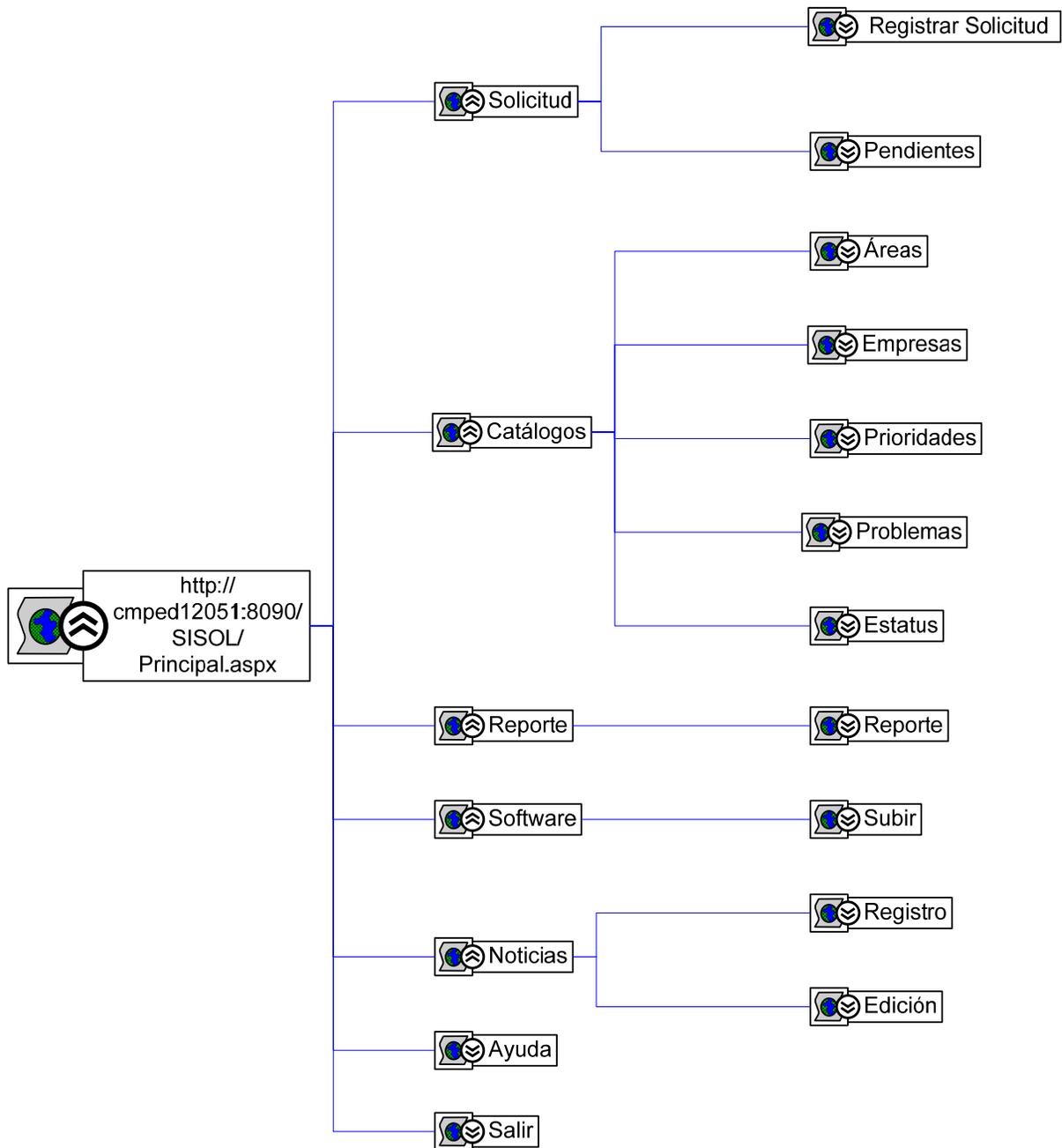


Figura 4.41 Mapa de navegación para administrador 1

Mapa de navegación de administrador 2

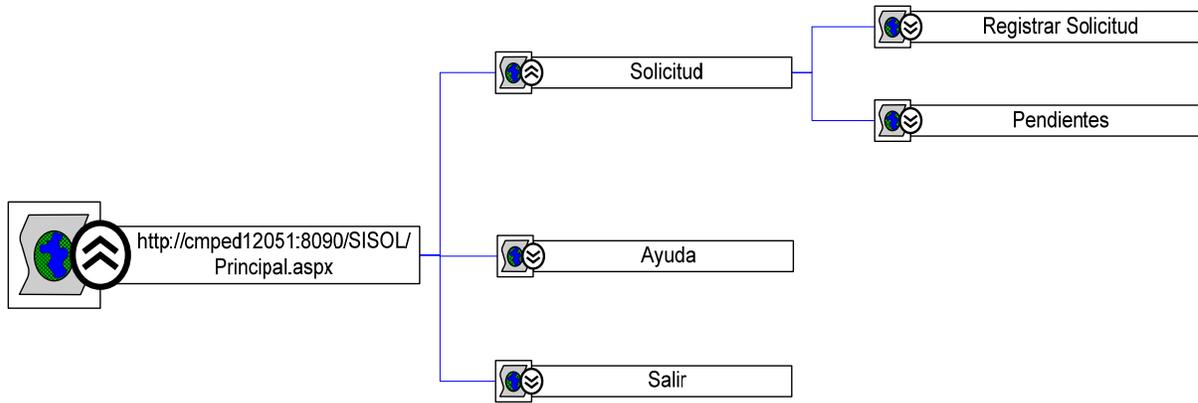


Figura 4.42 Mapa de navegación para administrador 2

CAPÍTULO 5

DESARROLLO DEL SISTEMA

5.1 IMPLEMENTACIÓN DE BASE DE DATOS

La implementación de la base de datos se realizó usando el manejador de Bases de datos Microsoft SQL Server 2008 como se mencionó en el capítulo 3.

Para llevar a cabo la implementación física de una base de datos definimos su nombre, su tamaño y los archivos y grupos de archivos usados para almacenarla.

Se usan tres tipos de archivos para almacenar una base de datos: archivos principales, archivos secundarios y registro de transacciones. Toda base de datos tiene al menos 2 archivos: un archivo de datos principal y un registro de transacciones como mínimo.^[6]

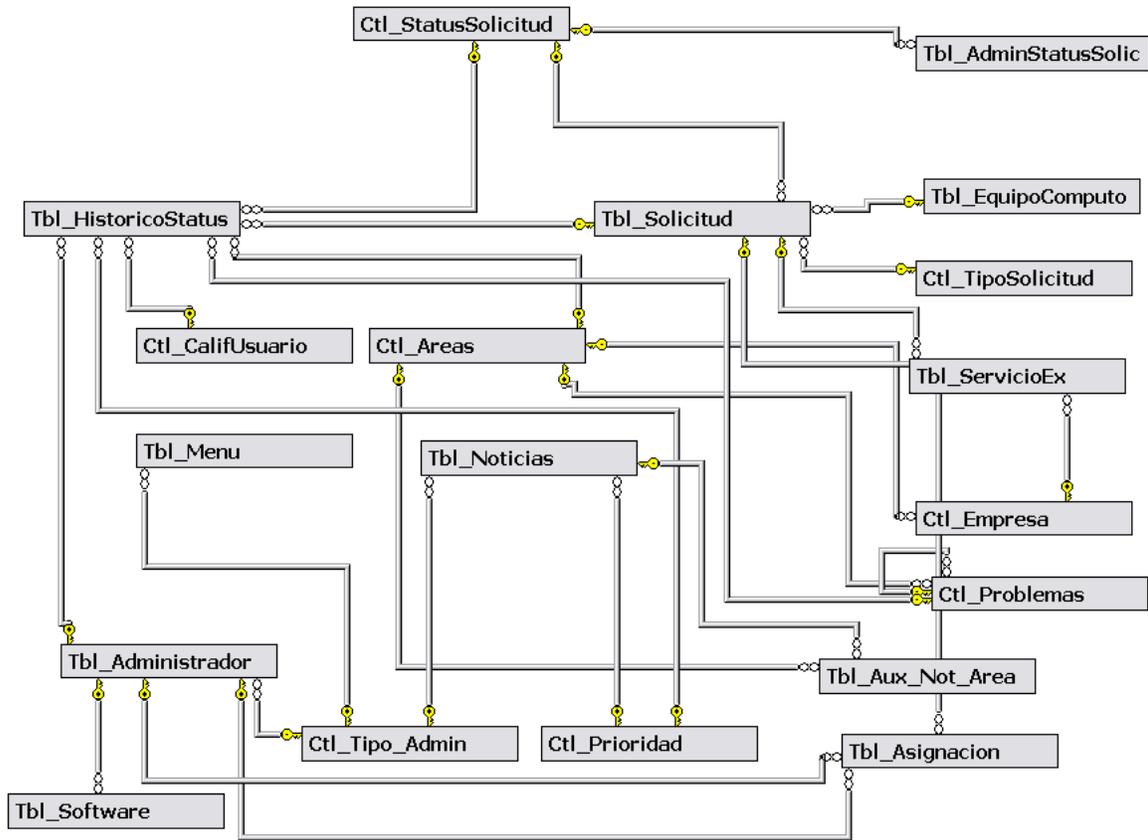
- Archivos principales: contienen la información de inicio de la base de datos. Se utilizan también para almacenar datos. Cada base de datos tiene un archivo principal.
- Archivos secundarios: contienen todos los datos que no caben en el archivo de datos principal, éstos son opcionales si el archivo principal es lo suficientemente grande como para contener todos los datos.
- Registro de transacciones: contienen la información de registro que se utiliza para recuperar la base de datos. Cada base de datos debe tener al menos un archivo de registro de transacciones, aunque puede tener más de uno.

Para crear la base de datos “CMP_SISOL” tomamos en cuenta lo siguiente:

- 1 Disposición de permisos para CRATE DATABASE, CREATE ANY DATABASE O ALTER ANY DATABASE.
- 2 En ciertos casos, algunos permisos se establecen en los archivos de datos y de registro de cada base de datos. Los permisos evitan que los archivos se modifiquen accidentalmente si residen en un directorio sin restricción de permisos.
- 3 Todos los objetos definidos por el usuario de la base de datos **model** se copian en todas las bases de datos recién creadas. Se puede agregar a la base de datos model todos los objetos (tablas, vistas, procedimientos almacenados, tipos de datos, etc.) que se desee incluir en todas las bases de datos nuevas.

SQL server provee métodos que se pueden utilizar para crear bases de datos: el comando Transact-SQL CREATE DATA BASE, el árbol de la consola Enterprise Manager, y el asistente Create Database, al cual se puede acceder a través de SQL Server Enterprise Manager.

Transact-SQL es el lenguaje que se utiliza para administrar instancias del motor de base de datos de SQL Server, para crear (CREATE) y administrar objetos de base de datos, y para insertar (INSERT INTO), recuperar (SELECT), modificar (ALTER) y eliminar (DELETE) datos. Transact-SQL es una extensión del lenguaje definido en los estándares de SQL publicados por International Standards Organization (ISO) y American National Standards Institute (ANSI). Lenguaje que se utilizó para la implementación de la base de datos en el Sistema de Solicitudes, tomando en cuenta la estructura de las tablas especificadas en el diccionario de datos^[6] (Ver Anexo 1).



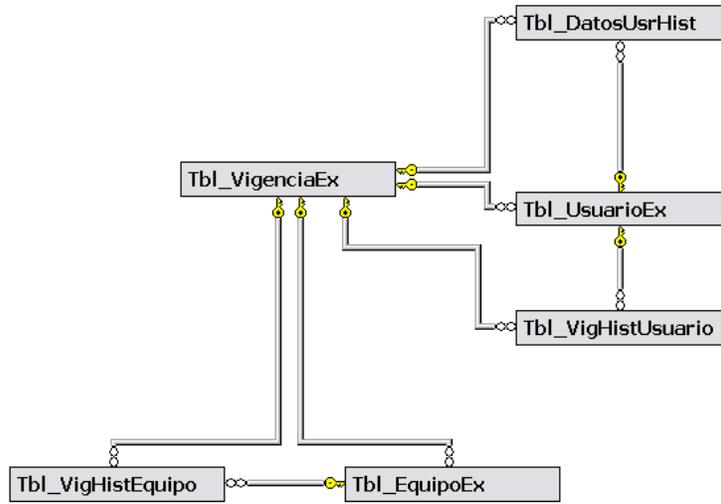
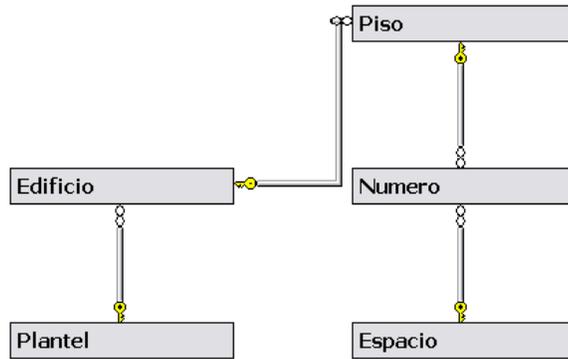


Figura 5.1 Diagrama de implementación en SQL Server 2008

5.2 PROGRAMACIÓN E INTERFAZ

Como se mencionó en los capítulos anteriores, la herramienta de desarrollo para este sistema es Microsoft Visual Studio 2008 y C#, el cual maneja una programación orientada a objetos, que se explica a continuación.

Un objeto es una estructura con datos y métodos que manipulan los datos. En general, una clase es una representación abstracta de algo, mientras que un objeto es un ejemplo utilizable de ese algo representado por la clase. La única excepción a esta regla la constituyen los miembros de clases compartidas, que se pueden usar tanto en caso de una variable de clase como de una variable de objeto declaradas como el tipo de la clase.

Las clases constan de campos, propiedades, métodos y eventos. Los campos y propiedades representan la información que contiene un objeto. Los campos son similares a las variables en cuanto que se pueden leer o definir directamente. Por ejemplo, si se tiene un objeto llamado "Automóvil", se puede almacenar su color en un campo llamado "Color".

Las propiedades se recuperan y definen igual que los campos, pero se implementan mediante los procedimientos de las propiedades Get y Set, que proporcionan más control sobre el modo en que los valores se definen o se devuelven.

Los métodos representan acciones que puede realizar un objeto. Por ejemplo, un objeto "Automóvil" puede tener definidos los métodos "MotorArranque," "Tracción" y "Parada". Los métodos se definen al agregar procedimientos (subrutinas o funciones) a la clase.

Los eventos son notificaciones que un objeto recibe de otros objetos u otras aplicaciones o que transmite a ellos. Los eventos permiten a los objetos realizar acciones cada vez que se da un caso concreto. Un ejemplo de evento para la clase "Automóvil" podría ser "Comprobar_motor".

Los campos, las propiedades, los métodos y los eventos forman sólo la mitad de la programación orientada a objetos, el resto requiere que los objetos admitan tres cualidades: encapsulación, herencia y polimorfismo.

La encapsulación implica el tratamiento de un grupo de propiedades, métodos y otros miembros como una única unidad u objeto. Los objetos pueden controlar el modo en que se cambian las propiedades y se ejecutan los métodos. Por ejemplo, un objeto puede validar valores antes de permitir cambios en propiedades. La encapsulación facilita también el cambio de implementación posterior al permitir ocultar los detalles de implementación de los objetos (ocultación de la información).

Herencia describe la capacidad de crear clases nuevas a partir de una clase existente. La nueva clase hereda todas las propiedades, métodos y eventos de la clase base, y se puede personalizar con propiedades y métodos adicionales. Por ejemplo, puede crear una nueva clase llamada "Camión" basada en la clase "Automóvil". La clase "Camión" hereda la propiedad "Color" de la clase "Automóvil" y puede tener propiedades adicionales, como "TracciónCuatroRuedas".

El polimorfismo implica la posibilidad de tener varias clases que se pueden usar de forma intercambiable, incluso si cada clase implementa las mismas propiedades o métodos de formas distintas. El polimorfismo es esencial para la programación orientada a objetos, ya que permite usar elementos con los mismos nombres, sin importar qué tipo de objeto esté en uso en ese momento. Por ejemplo, si se tiene una clase base "Automóvil", el polimorfismo permite al desarrollador definir distintos métodos "MotorArranque" para cualquier número de clases derivadas. El método

"MotorArranque" de una clase derivada llamada "AutomóvilDiesel" puede ser completamente distinto del método con el mismo nombre en la clase base.

Para la programación del sistema se utilizó una arquitectura de 3 capas que consta de capa de datos, capa de negocios y capa de presentación que permite su respuesta de una manera rápida y eficaz. Así mismo, permite futuras modificaciones de código para la ampliación del sistema en caso de ser necesario.

La figura 5.2 ilustra la arquitectura de programación en capas. Las Figuras 5.3, 5.4 y 5.5 muestran la implementación de las 3 capas, respectivamente.

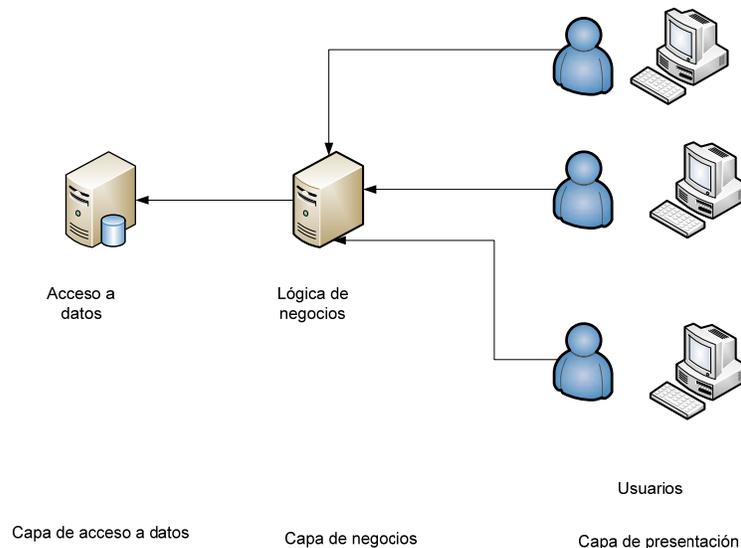


Figura 5.2 Programación en capas

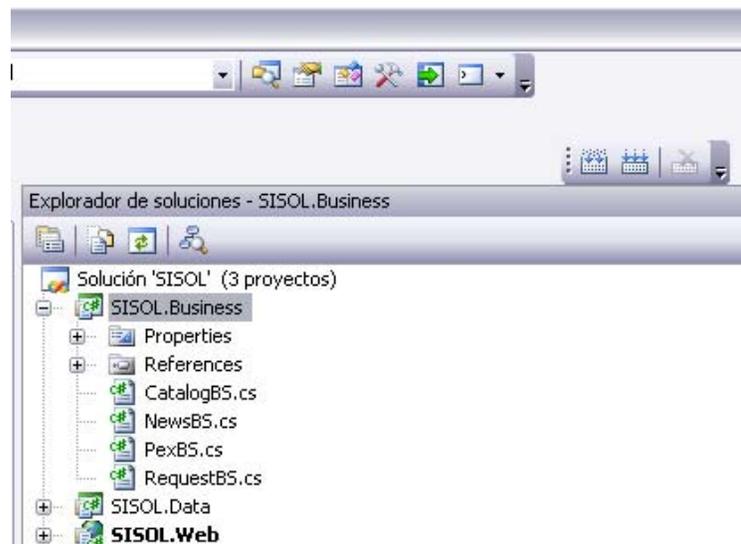


Figura 5.3 Implementación de la capa de Negocio

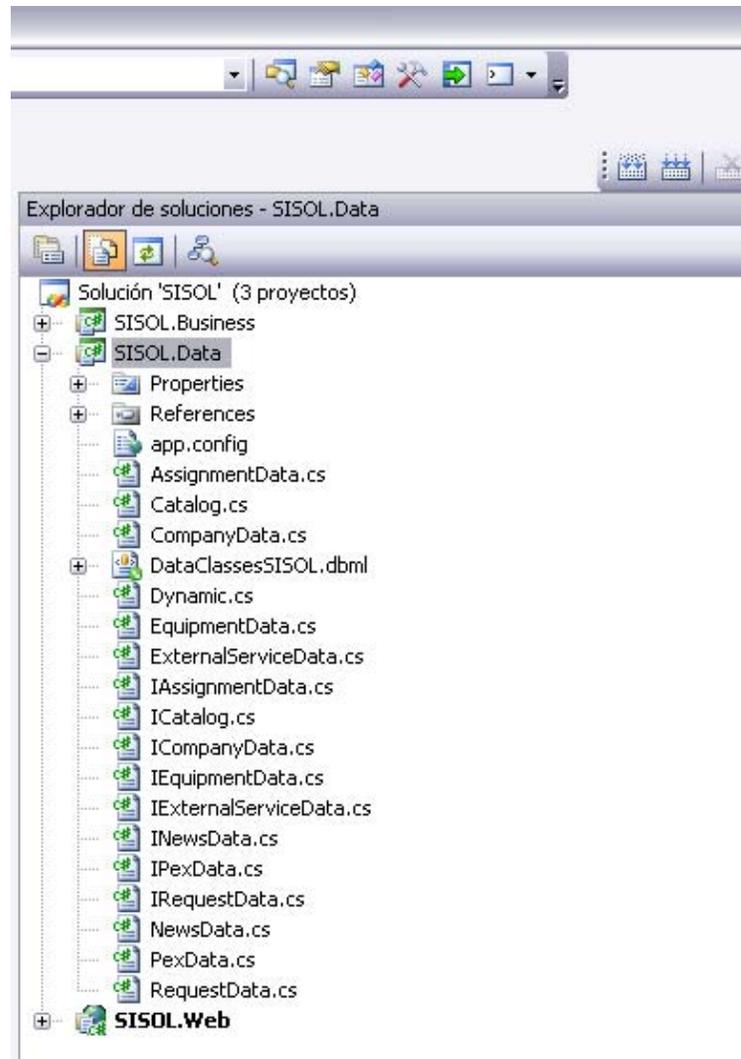


Figura 5.4 Implementación de la capa de Datos

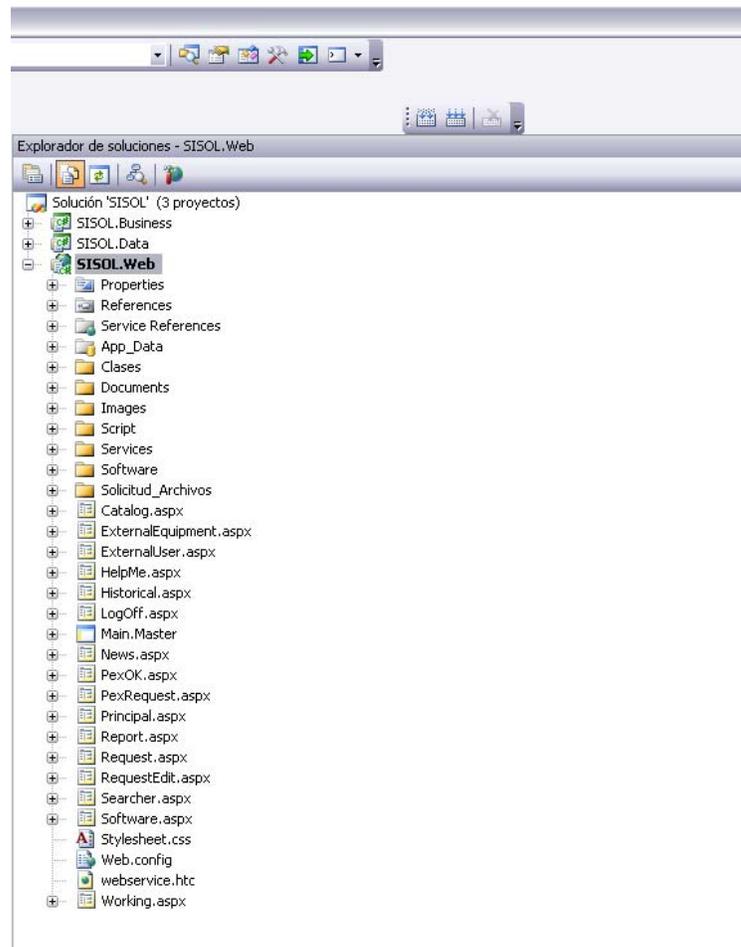


Figura 5.5 Implementación de la capa de Presentación

En la capa de presentación se diseñaron las páginas que integran el sistema, haciendo uso de páginas maestras o principales que proporcionan una funcionalidad para no repetir código, texto y elementos de control. En los siguientes puntos se muestran algunas pantallas principales implementadas en los módulos sistema.

5.2.1 Recepción, consulta y confirmación de solicitudes recibidas

- **Realizar solicitudes de servicio y solicitudes a nombre de otra persona**

En la siguiente pantalla (Figura 5.6) podemos observar la página principal para realizar una solicitud de servicio con la opción de poder realizarla a nombre de otra persona.



Figura 5.6 Realizar solicitud

- **Consulta de noticias o tips importantes para los usuarios y consulta de solicitudes realizadas**

En la siguiente pantalla (Figura 5.7) es posible consultar las solicitudes realizadas por el usuario que ingresó al sistema así como también consultar las noticias registradas y activas.



Figura 5.7 Consulta de noticias, tips y solicitudes

5.2.2 Administración

- **Estados de solicitudes, Prioridades**

En la siguiente pantalla (Figura 5.8) de edición de una solicitud es posible actualizar el estado y la prioridad de la misma.



Figura 5.8 Actualizar estado de solicitudes

- **Consulta de solicitudes**

En la siguiente pantalla (Figura 5.9) es posible consultar las solicitudes asignadas al administrador que ingresó al sistema.

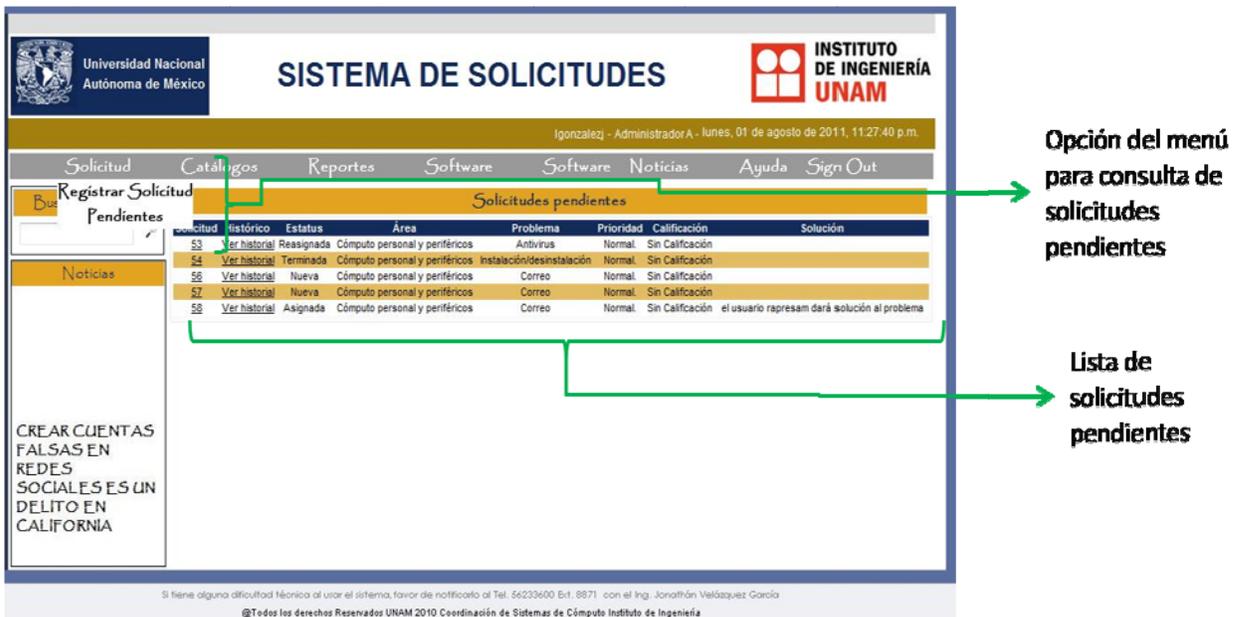


Figura 5.9 Consulta de solicitudes asignadas

- **Equipo y personal externo**

En la siguiente pantalla (Figura 5.10) es posible consultar las solicitudes de equipo y personal externo para aprobarlas o rechazarlas según sea el caso.

Si tiene alguna dificultad técnica al usar el sistema, favor de notificarlo al Tel. 56233600 Ext. 8871 con el Ing. Jonathan Velázquez García
 @Todos los derechos Reservados UNAM 2010 Coordinación de Sistemas de Cómputo Instituto de Ingeniería

Figura 5.10 Consulta de equipo y personal externo

- **Control de catálogos**

En la siguiente pantalla (Figura 5.11) es posible consultar, editar y agregar registros a los catálogos existentes en el sistema.

Si tiene alguna dificultad técnica al usar el sistema, favor de notificarlo al Tel. 56233600 Ext. 8871 con el Ing. Jonathan Velázquez García
 @Todos los derechos Reservados UNAM 2010 Coordinación de Sistemas de Cómputo Instituto de Ingeniería

Figura 5.11 Consulta de catálogos

- **Noticias/Tips**

En la siguiente pantalla (Figura 5.12) es posible consultar, editar y agregar registros de noticias que son mostradas al usuario al ingresar al sistema.

Univerisidad Nacional Autónoma de México

SISTEMA DE SOLICITUDES

INSTITUTO DE INGENIERÍA UNAM

mambrizm - Administrador del sistema - lunes, 01 de agosto de 2011, 11:27:40 p.m.

Solicitud PEX Catálogos Reportes Software Noticias Ayuda Sign Out

Buscar solicitudes

Noticias

Registro Edición

Noticia	Fecha Inicio	Fecha Fin	Tipo Admin	Prioridad	Editar
prueba	01/03/2011	30/07/2011	Super Usuario	Alta	
prueba	01/03/2011	09/04/2011	Super Usuario	Alta	
Noticia utilizando la aplicación	01/04/2011	09/07/2011	Super Usuario	Media	
CREAR CUENTAS FALSAS EN REDES SOCIALES ES UN DELITO EN CALIFORNIA	01/01/2010	01/01/2012	Usuario	Alta	
Los jugadores del Pontevedra español se arrodillaron al unisono durante un partido oficial. http://mx.yahoo.com/?p=us	01/01/2010	27/11/2011	Administrador B	Alta	
Prueba Rosaura	01/03/2011	10/09/2011	Administrador A	Media	
Prueba de VHCR	01/06/2011	30/06/2011	Administrador A	Alta	
Prueba de VHCR	01/06/2011	09/07/2011	Administrador A	Alta	
Noticia utilizando la aplicación.***	01/04/2011	30/06/2011	Administrador A	Alta	
Noticia utilizando la aplicación.	01/04/2011	04/06/2011	Super Usuario	Alta	
Google Maps Navigation ha dado un paso de gigante al añadir a su navegador la posibilidad de crear rutas alternativas en la carretera por las que evitar atascos	01/03/2011	07/05/2011	Administrador A	Alta	

Si tiene alguna dificultad técnica al usar el sistema, favor de notificarlo al Tel. 54233600 Ext. 8871 con el Ing. Jonathan Velázquez García

©Todos los derechos Reservados UNAM 2010 Coordinación de Sistemas de Cómputo Instituto de Ingeniería

Opción de menú para administración de noticias.

Figura 5.12 Consulta y edición de noticias/tips

- **Software**

En la siguiente pantalla (Figura 5.13) es posible consultar, agregar y eliminar el software de apoyo al usuario.

Univerisidad Nacional Autónoma de México

SISTEMA DE SOLICITUDES

INSTITUTO DE INGENIERÍA UNAM

lgonzalezj - Administrador A - lunes, 01 de agosto de 2011, 11:27:40 p.m.

Solicitud Catálogos Reportes Software Noticias Ayuda Sign Out

Buscar solicitudes

Noticias

Subir Software

Archivo: Examinar...

Comentarios:

Archivo	Usuario	Comentario	Fecha
avatar77321_1.gif	lgonzalezj	Test	04/05/2011 05:48:26 p.m.
Balon-de-Oro.jpg	lgonzalezj	Prueba	04/05/2011 05:56:27 p.m.
Test2.rar	lgonzalezj	addas	04/05/2011 06:37:39 p.m.

CREAR CUENTAS FALSAS EN REDES SOCIALES ES UN DELITO EN CALIFORNIA

Opción de menú para consulta y carga de software

Figura 5.13 Consulta y edición de software

5.2.3 Estadísticas y reportes

- **Reportes**

En la siguiente pantalla (Figura 5.14) se deben ingresar área y fechas para la generación del reporte de solicitudes atendidas en el periodo indicado al área seleccionada.

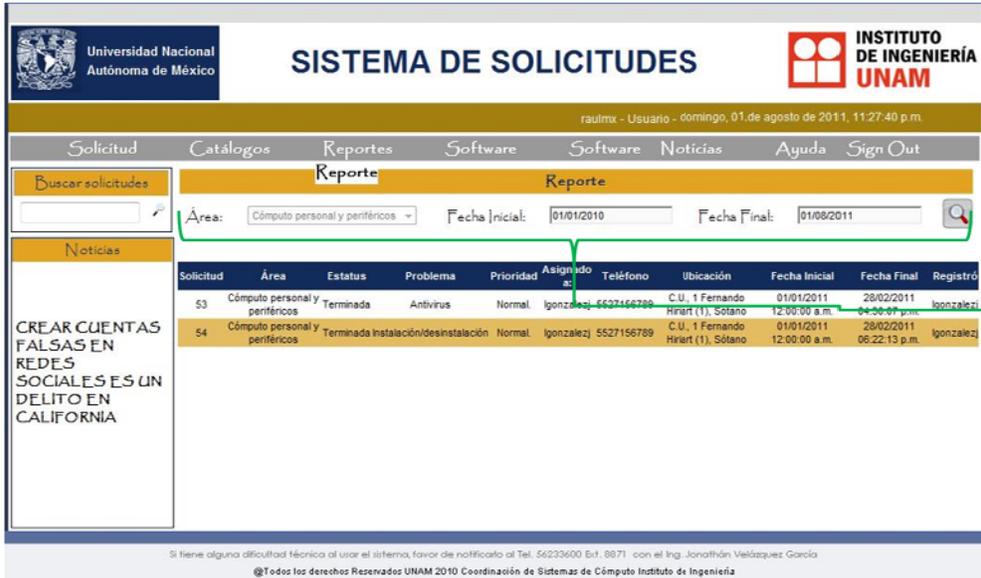


Figura 5.14 Generación de reportes

5.2.4 Registro de personal y equipo externo

- **Solicitud y control de equipo externo**

En la siguiente pantalla (Figura 5.15) es posible la captura de información para el registro de equipo externo.



Figura 5.15 Registro de equipo externo

- **Solicitud y control de personal externo**

En la siguiente pantalla (Figura 5.16) es posible la captura de información para el registro de usuario externo.

Universidad Nacional Autónoma de México

SIST

INSTITUTO DE INGENIERÍA UNAM

raulmx - Usuario - domingo, 01 de agosto de 2011, 11:27:40 p.m.

Solicitud PEX Ayuda Sign Out

Buscar solicitudes

Noticias

CREAR CUENTAS FALSAS EN REDES SOCIALES ES UN DELITO EN CALIFORNIA

Solicitud de Equipo tipo externo

Solicitud de Usuario

Académico Responsable: raulmx

Nombre:

Apellido Paterno:

Apellido Materno:

Periodo de vigencia

Fecha inicio:

Fecha fin:

Justificación Académico:

Capturar usuario externo

Si tiene alguna dificultad técnica al usar el sistema, favor de notificarlo al Tel. 56233600 Ext. 8971 con el Ing. Jonathan Velázquez García
© Todos los derechos Reservados UNAM 2010 Coordinación de Sistemas de Cómputo Instituto de Ingeniería

Figura 5.16 Registro de personal externo

CAPÍTULO 6

PRUEBAS Y LIBERACIÓN DEL SISTEMA

6.1 TIPOS DE PRUEBAS

La meta de probar las aplicaciones Web es ejercitar cada una de las muchas dimensiones de la calidad en la aplicación con la finalidad de encontrar errores o descubrir conflictos que pudieran conducir a fallas de uso, navegabilidad, desempeño, compatibilidad, capacidad y seguridad. Las pruebas también incorporan revisiones que ocurren conforme se diseña la aplicación. ^[12]

La estrategia de prueba de la aplicación ejercita cada una de las dimensiones de calidad al examinar inicialmente “unidades” de contenido, funcionalidad o navegación. Una vez que las unidades individuales han sido validadas, el enfoque se cambia a pruebas que ejerciten la aplicación como un todo. Ésto se logra derivando muchas pruebas de las perspectivas de los usuarios y se alimentan con la información contenida en los casos de uso. Se desarrolla un plan de prueba de ingeniería web y se identifican los pasos de prueba, los productos de trabajo y los mecanismos para la evaluación de los resultados de prueba, el proceso de prueba abarca diferentes tipos de pruebas, que se mencionan a continuación:

- La prueba del contenido (y las revisiones) se centra en varias categorías de contenido. La finalidad es descubrir errores tanto semánticos como sintácticos que afecten la precisión del contenido o la forma en la que se presenta al usuario final. ^[12]
- La prueba de la interfaz ejercita los mecanismos de interacción que permiten que un usuario se comunique con la aplicación y valida los aspectos estéticos de la interfaz. El objetivo es descubrir errores que resulten de mecanismos de interacción mal implementados, u omisiones, inconsistencias o ambigüedades en la semántica de la interfaz. ^[12]
- La prueba de navegación aplica casos de uso, derivados como parte de la actividad de análisis, en el diseño de casos de prueba que ejercitan cada uno de los escenarios de uso frente al diseño de navegación. Los mecanismos de navegación se prueban para garantizar que se identifican y corrigen los errores que impiden el completar un caso de uso. ^[12]
- La prueba de componentes ejercita las unidades de contenido y funcionales dentro de la aplicación. Cada página web encapsula contenido, vínculos de navegación y elementos de procesamiento que forman una “unidad” dentro de la arquitectura de la aplicación web. Se deben probar dichas unidades. ^[12]
- La prueba de configuración intenta descubrir los errores o los problemas de compatibilidad específicos de un ambiente particular de cliente servidor. Entonces se llevan a cabo pruebas para descubrir los errores asociados con cada posible configuración. ^[12]
- La prueba de la seguridad incorpora una serie de pruebas diseñadas para explotar las vulnerabilidades de la aplicación y en su ambiente. La finalidad es encontrar hoyos de seguridad. ^[12]

6.2 DOCUMENTACIÓN DE PRUEBAS

Prueba de Contenido

Para este tipo de pruebas los mismos programadores y administradores funcionaron como examinadores del sistema, debido a que, al mismo tiempo que se construía, se comprobaba la calidad y veracidad del contenido. Así mismo se cotejaba que al consultar o modificar la base de datos, la información se almacenara en el lugar correcto de acuerdo a la operación realizada.

Un ejemplo de éstas pruebas, es verificar que los datos se muestren correctamente al agregar una solicitud:

Paso 1: Datos personales del Active Directory de acuerdo al usuario que ingresó al sistema. (Ver figura 6.1).

The screenshot displays the 'SISTEMA' web application interface. At the top, it shows the logos for 'Universidad Nacional Autónoma de México' and 'INSTITUTO DE INGENIERÍA UNAM'. The user is logged in as 'raulmx' on 'domingo, 01 de agosto de 2011, 11:27:40 p.m.'. The main navigation includes 'Solicitud', 'PEX', 'Ayuda', and 'Sign Out'. The current page is 'Registro de Solicitud', which is the first of four steps. The form is divided into two sections: 'Datos Personales del Solicitante' and 'Datos de Atención del Servicio'. The 'Datos Personales' section includes a checkbox for 'Solicitud a nombre de otra persona' and several input fields: 'Nombre: raulmx', 'Teléfono: 5527156789', 'Subdirección: DGSCA', 'Coordinación: UNAM', and 'Correo: raul@unam.mx'. A green arrow points from the text 'Datos cargados del Active Directory' to these fields. The 'Datos de Atención del Servicio' section includes a 'Ubicación:' label and dropdown menus for 'Campus', 'Edificio', 'Piso', and 'Cubículo'. A sidebar on the left contains a search bar and a news section with the text 'CREAR CUENTAS FALSAS EN REDES SOCIALES ES UN DELITO EN CALIFORNIA'. The footer contains the copyright notice: '@Todos los derechos Reservados UNAM 2010 Coordinación de Sistemas de Cómputo Instituto de Ingeniería'.

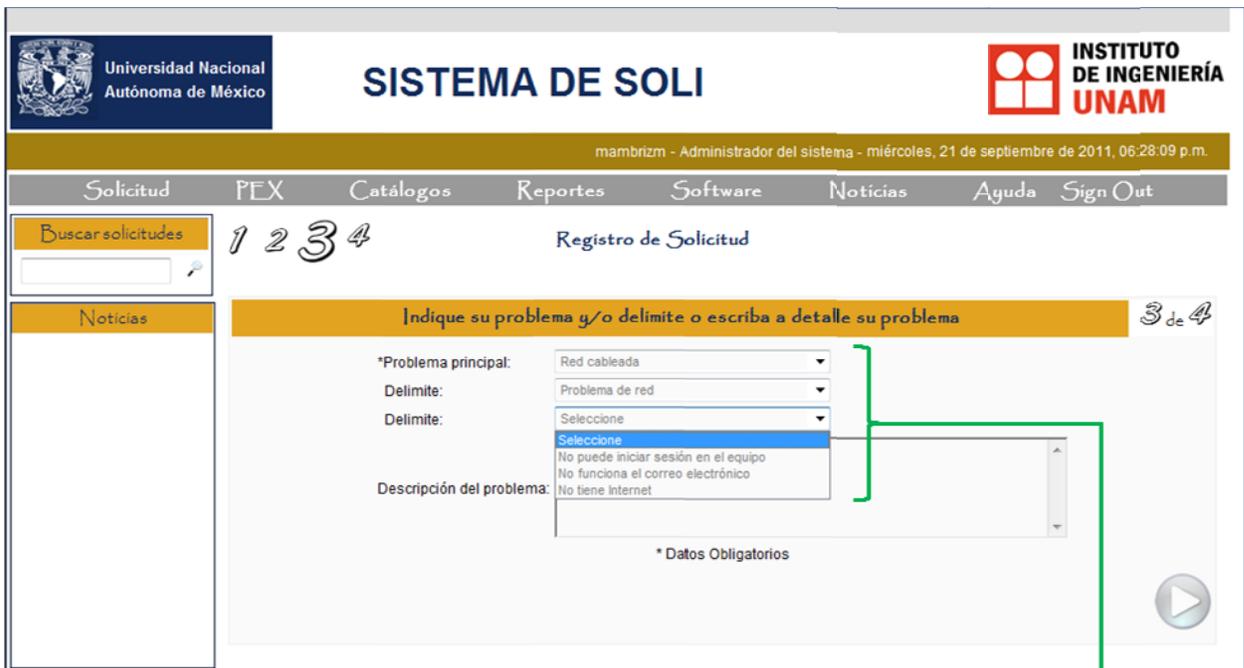
Figura 6.1 Carga de datos mediante Active Directory

Paso 2: Datos del equipo de acuerdo al no. de inventario proporcionado por el usuario. (Ver figura 6.2).



Figura 6.2 Carga de datos por base de datos

Paso 3: Niveles de problemas de acuerdo al servicio requerido. (Ver figura 6.3).



Catálogos de problemas

Figura 6.3 Niveles de problemas en registro de solicitud

Paso 4: Resumen de los datos recabados en los pasos anteriores y guardados en la tabla Tbl_Solicitud. Se envía un correo con la información a los jefes de área correspondientes y al usuario. (Ver Figura 6.4)

Universidad Nacional Autónoma de México

SISTEMA DE SOLICITUDES

INSTITUTO DE INGENIERÍA UNAM

mambrizm - Administrador del sistema - jueves, 22 de septiembre de 2011, 09:05:08 p.m.

Solicitud PEX Catálogos Reportes Software Noticias Ayuda Sign Out

Buscar solicitudes 1 2 3 4

Registro de Solicitud

Noticias

Resumen de la solicitud 4 de 4

Nombre Solicitante: mambrizm
 Nombre Responsable: mambrizm
 Correo: MAmbrizM@ingen.unam.mx
 Teléfono: 5527156789
 Subdirección: DGSCA
 Coordinación: UNAM
 Ubicación: C.U., Sótano, 1 Fernando Hiriar (1), Cubículo, 120
 Núm Inventario: 01400313
 Problema: Red cableada - Problema de red - No tiene Internet

Datos de resumen de la solicitud

Figura 6.4 Resumen de la solicitud

Al realizar y completar este tipo de pruebas en cada uno de los módulos del sistema, se logró verificar que la información solicitada es concreta y sólo la necesaria, consultando y obteniendo datos personales del Active Directory, lo que reduce la captura de información sin sentido para el usuario.

Prueba de Interfaz

Para esta prueba hay varios factores a considerar, y no solo al final, sino que en tres etapas distintas, del desarrollo: en el análisis y formulación de requerimientos, en el diseño y en las pruebas finales, ésto con el objetivo de llevar un buen control de lo que el usuario observara y conducirá en el sistema. Los factores a considerar son:

Mecanismos de la interfaz: Cuando se interactúa con el sistema se hace mediante uno o más mecanismos de interfaz, los cuales pueden ser: vínculos de navegación, formatos, ventanas pop-up, mapas de sitio, etc. Los cuales se verifican probándolos para asegurarse que se alcance el objeto de contenido o función debida. En el caso de los vínculos redirigir la pagina al lugar correcto, en el tema de los formatos garantizar que el usuario identifique etiquetas, campos obligatorios, valores por defecto y el sistema cumpla con el envío correcto de la información, que las funciones del navegador no corrompan la información, que los menús se muestren completos y que las verificaciones de error de datos sean correctas. Ver figura 6.4

Semántica de la Interfaz: Conforme se prueba cada caso de uso, se revisa la información en pantalla que el actor obtendrá, principalmente de los menús y formularios, que sea entendible y coherente con el actor que hace uso del sistema. Un ejemplo de ello es que el usuario no vea información sólo permitida para los Administradores. Ver figura 6.4

Facilidad de Uso: Es una prueba similar a la de semántica, cuya diferencia radica en que son para determinar el grado de facilidad de uso para el usuario, ya sea que se pruebe por mecanismos, por páginas completas o por el sistema entero. Se pueden dividir las pruebas en categorías y establecer objetivos específicos como: interactividad (mecanismos fáciles de entender y usar), plantillas (mecanismos localizables rápidamente), legibilidad (texto e imágenes comprensibles), estética (colores, tipografía y características relacionadas), personalización (según el tipo de actor: Administrador 1, Administrador 2 ó Usuario, sean las pantallas correspondientes), etc. (Ver Figura 6.4)

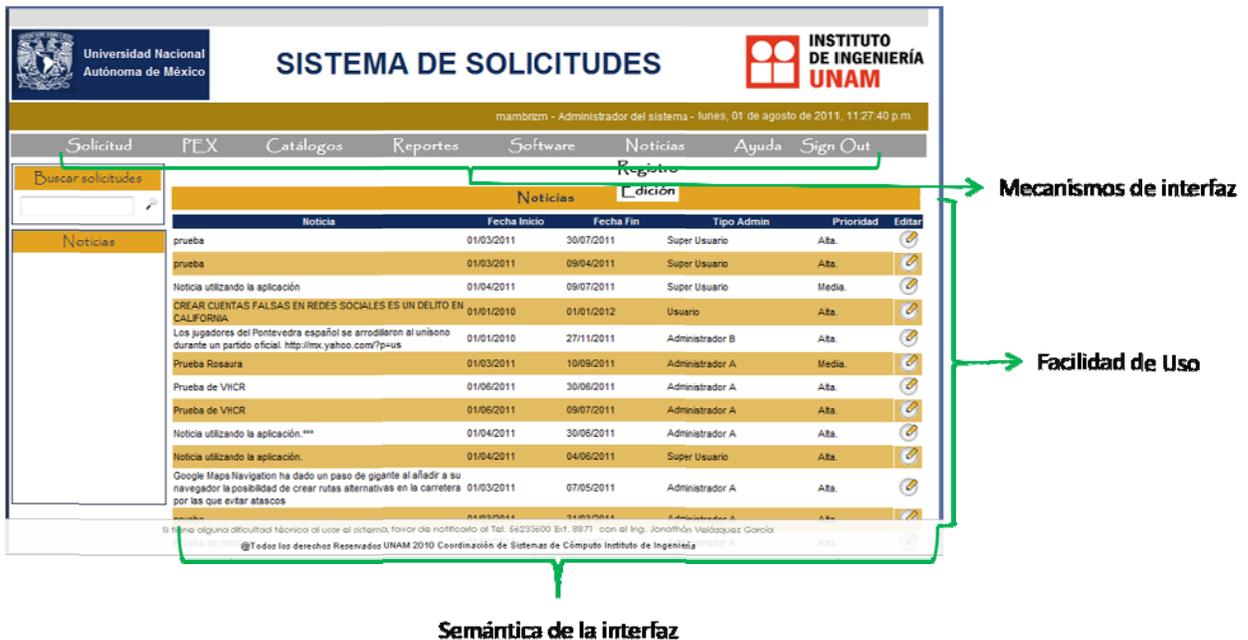


Figura 6.4 Factores a considerar en las pruebas de interfaz

En la formulación y análisis de requerimientos se planteó la manera en que el usuario se desenvolverá en la interfaz, así como los datos que se le pedirán para llenar los formularios de petición de servicios y con lo que podría llevar el seguimiento de su servicio, que se determinó como un numero consecutivo llamado ticket. En esta etapa se realizaron pruebas de mecanismos y semántica de interfaz puesto que se definió la estructura del sistema y, los casos de uso, lo cuales sirvieron para definir los vínculos, formularios, ventanas, etc., de los diferentes actores que intervienen en el sistema, dándonos como resultado una idea clara de cómo diseñar el sistema y los casos de uso. Ver tema 4.2, como ejemplo de casos de uso definidos.

Para las pruebas durante el diseño se formularon los posibles bocetos de las páginas que interactúan con los usuarios así como su formato y seguimiento, esto es que los formularios a rellenar sean congruentes y tengan una secuencia lógica. Ofreciendo la oportunidad de hacer pruebas de mecanismos, en su mayoría, y semántica, pero con una mayor idea sobre el sistema en desarrollo ya que en este punto se tienen diseños más en forma y estructurados. Las pruebas nos corrigieron errores en el diseño físico del sistema, dándonos la oportunidad de re-establecer vínculos y formularios que pudieran ser incongruentes según el actor y el proceso que llevara, por ejemplo, que los

administradores 1 en general pudieran validar personal externo, siendo función solamente del Coordinador de la CSC. De la misma manera se empezó a llevar un control de la semántica de la interfaz para así poder definir correctamente los roles e información correspondiente a cada actor, según los casos de uso. Ver tema 4.3, para ejemplificar el diseño de la interfaz.

En el caso de las pruebas finales se plantearon guiones para los posibles usuarios, tomando como referencia los casos de uso que se diseñaron al principio, ésto contempla errores de lógica en los datos introducidos para verificar las pantallas de alerta y su reacción al error. Por lo que podemos implementar las 3 pruebas más importantes en ésta etapa (mecanismo, semántica de la interfaz y facilidad de uso) en una sola prueba, como por ejemplo el siguiente guión:

Proyecto: Sistema de Solicitud de Servicios para la CSC

CASO DE PRUEBA	Registra solicitud		
CASO DE USO de referencia	Realizar Solicitud		
PROPÓSITO	El usuario captura la información para registrar una solicitud de forma correcta.		
PRE – REQUISITOS	<ol style="list-style-type: none"> 1. Se cuenta con perfil de usuario. 2. El usuario accedió al sistema de solicitudes. 3. El usuario cuenta con un número de inventario del equipo para el que solicita servicio. 4. El usuario seleccionó el menú principal “Solicitud”→”Registrar Solicitud” 		
DATOS DE PRUEBA	Usuario de dominio. Número de inventario del equipo.		
PASOS	Actividad	Resultado	
		✓	X
	<ol style="list-style-type: none"> 1. Paso 1: La pantalla está integrada por los componentes que a continuación se listan: <ol style="list-style-type: none"> 1.1. Casilla para realizar solicitud a nombre de otra persona. Aparece deshabilitada. Al habilitar casilla, se muestra lista Académico responsable. ✓ 1.2. Caja de texto donde aparece el Nombre del usuario que ingresó al sistema. Aparece inhabilitada. ✓ 1.3. Caja de texto donde aparece el Teléfono del usuario que ingresó al sistema. Aparece inhabilitada. ✓ 1.4. Caja de texto donde aparece la Subdirección del usuario que ingresó al sistema. Aparece inhabilitada. ✓ 1.5. Caja de texto donde aparece el Correo del usuario que ingresó al sistema. Aparece inhabilitada. ✓ 1.6. Caja de texto para capturar el nombre del Responsable Alterno para la solicitud. Aparece habilitada. ✓ 1.7. Lista Campus aparece habilitada y no presenta ✓ 		

	<p>información seleccionada.</p> <p>1.8. Lista Edificio aparece habilitada y no presenta información seleccionada.</p> <p>1.9. Lista Piso aparece habilitada y no presenta información seleccionada.</p> <p>1.10. Lista Cubículo aparece habilitada y no presenta información seleccionada.</p> <p>1.11. Botón Siguiente, aparece habilitado. (Ver paso 2)</p>	<p>✓</p> <p>✓</p> <p>✓</p> <p>✓</p>	
	<p>2. Paso 2: La pantalla está integrada por los componentes que a continuación se listan:</p> <p>2.1. Caja de texto para capturar el Número de Inventario para la solicitud. Aparece habilitada. Ingresar número de inventario existente.</p> <p>2.2. Botón Buscar para consultar los datos del equipo con el número de inventario proporcionado.</p> <p>2.3. Caja de texto donde aparece la Descripción del equipo con el número de inventario proporcionado. Aparece inhabilitada.</p> <p>2.4. Caja de texto donde aparece la Marca del equipo con el número de inventario proporcionado. Aparece inhabilitada.</p> <p>2.5. Caja de texto donde aparece el Modelo del equipo con el número de inventario proporcionado. Aparece inhabilitada.</p> <p>2.6. Caja de texto donde aparece el Número de Serie del equipo con el número de inventario proporcionado. Aparece inhabilitada.</p> <p>2.7. Caja de texto donde aparece el Color del equipo con el número de inventario proporcionado. Aparece inhabilitada.</p> <p>2.8. Botón Siguiente, aparece habilitado. (Ver paso 3)</p>	<p>✓</p> <p>✓</p> <p>✓</p> <p>✓</p> <p>✓</p> <p>✓</p> <p>✓</p> <p>✓</p>	
	<p>3. Paso 3: La pantalla está integrada por los componentes que a continuación se listan:</p> <p>3.1. Lista Problema Principal aparece habilitada y no presenta información seleccionada. Seleccionar para</p>	<p>✓</p>	

	<p>mostrar la siguiente lista.</p> <p>3.2. Lista Delimite aparece habilitada y no presenta información seleccionada. Seleccionar para mostrar la siguiente lista. ✓</p> <p>3.3. Lista Delimite aparece habilitada y no presenta información seleccionada. ✓</p> <p>3.4. Caja de texto para capturar la Descripción del problema para la solicitud. Aparece habilitada. ✓</p> <p>3.5. Botón Siguiente, aparece habilitado. (Ver paso 4) ✓</p> <p>4. Paso 4 “Resumen de la Solicitud”: La pantalla está integrada por los componentes que a continuación se listan:</p> <p>4.1. Caja de texto donde aparece Nombre de Solicitante. Aparece inhabilitada. ✓</p> <p>4.2. Caja de texto donde aparece Nombre de Responsable. Aparece inhabilitada. ✓</p> <p>4.3. Caja de texto donde aparece Correo. Aparece inhabilitada. ✓</p> <p>4.4. Caja de texto donde aparece Teléfono. Aparece inhabilitada. ✓</p> <p>4.5. Caja de texto donde aparece Subdirección. Aparece inhabilitada. ✓</p> <p>4.6. Caja de texto donde aparece Coordinación. Aparece inhabilitada. ✓</p> <p>4.7. Caja de texto donde aparece Ubicación. Aparece inhabilitada. ✓</p> <p>4.8. Caja de texto donde aparece Número de Inventario. Aparece inhabilitada. ✓</p> <p>4.9. Caja de texto donde aparece Problema. Aparece inhabilitada. ✓</p> <p>4.10. Botón Guardar, aparece habilitado. ✓</p> <p>4.11. Se guarda la solicitud y muestra mensaje de solicitud guardada. X</p> <p>4.12. Botón Word, aparece habilitado. Para consultar el resumen de la solicitud en archivo de tipo Word X</p>		
NOTAS Y PREGUNTAS			

Prueba de Navegación

En las pruebas de navegación al igual que la mayoría de las demás pruebas se pueden realizar conforme se desarrolla el sistema, en este caso las pruebas se le realizan a la congruencia y facilidad de manejo del sistema, es decir, una secuencia lógica y amigable para los usuarios durante el uso del sistema.

La primera fase de la prueba de navegación comienza durante la prueba de interfaz. los mecanismo de navegación se prueban para asegurar que cada uno realiza la función que se busca, como ya se vio en la prueba de interfaz.

La prueba de la semántica de navegación se realiza para probar los conjuntos de estructuras de información y navegación relacionadas que colaboran en el cumplimiento de los requisitos de usuario, se pueden realizar mediante los casos de uso, esto es, con guiones formados de los casos de uso.

Para probar el sistema, se determinaron guiones para posibles usuarios y diferentes problemáticas se implementaron dichos guiones y se verifico que tan coherente era el desarrollo de una solicitud, en el caso de un usuario y una asignación de servicio en el caso de un administrador, en ambos casos mostrando problemas de navegación y re direccionamiento, para los links, y otros componentes webs. Se puede ejemplificar la prueba mediante el guión utilizado en el punto anterior.

Prueba de Configuración

La labor de probar la configuración no es ejercitar toda posible configuración del lado del cliente. Más bien, es probar un conjunto de probables configuraciones de los lados del cliente y del servidor para garantizar que la experiencia del usuario será la misma en todos ellos y para aislar errores que puedan ser especificados de una configuración particular.

Esencialmente, la aplicación se instaló en el ambiente del servidor y se probó con la intención de encontrar errores relacionados con la configuración. Por lo que podemos decir que se probaron servidores de Aplicaciones y de base de datos, dándonos como resultado:

- Compatibilidad de la aplicación con el sistema operativo del servidor.
- Creación correcta de los archivos de sistema, directorios y datos de sistema relacionados cuando la aplicación es operativa.
- Las medidas de seguridad como cortafuegos, permiten la ejecución del sistema sin interferencia ni bajo rendimiento.
- El sistema se integró correcta y completamente con la base de datos.

Cabe resaltar que los servidores son manejados y administrados por el área de servidores Windows de la CSC, por lo que constantemente son sometidos a pruebas de configuración, desempeño y errores, lo que nos asegura el correcto funcionamiento de nuestro sistema del lado del servidor.

En el lado del cliente las pruebas de configuración se centran principalmente en la compatibilidad de la aplicación con las configuraciones que contienen los siguientes elementos:

- Hardware: CPU, memoria, almacenamiento y dispositivos de impresión. (Ver tema 3.2.3)
- Sistemas operativos y software de navegación. (Ver tema 3.2.3 y 4.3)
- Conectividad. Red proporcionada por el II.

Pruebas de Seguridad

Las pruebas de seguridad están diseñadas para probar las vulnerabilidades en el ambiente del lado del cliente, las comunicaciones de red y el ambiente del lado del servidor

Nos corresponden las del lado del cliente y parte de las del servidor, debido a que el servidor es administrado por el área de Servidores Windows de la CSC. En el lado del cliente, las vulnerabilidades con frecuencia provienen de errores preexistentes en los navegadores, programas de correo electrónico o software de comunicación. Por ejemplo desbordamiento de buffer, acceso no autorizado a cookies, simulación de páginas web.

En el lado del servidor, las vulnerabilidades incluyen ataques de negación de servicio y guiones maliciosos que pueden pasar al lado del cliente o empleados para deshabilitar las operaciones del servidor y acceso, no autorizado, a la base de datos.

La protección contra estas vulnerabilidades requiere implementar los siguientes elementos de seguridad.

- Cortafuegos: Mecanismo de filtrado que examina cada paquete de información entrante para garantizar que llega de una fuente legítima y bloquea cualquier dato sospechoso (Firewall Administrado por el área de Servidores Windows de la SCS).
- Autenticación: Mecanismo de verificación que valida la identidad de todos los clientes y servidores, y permite que la comunicación ocurra solo cuando ambos lados son verificados. La autenticación se realiza mediante el Active Directory y solo permite el acceso a miembros del II con cuenta y contraseña validadas por el área de Servidores Windows.
- Detección de intrusos: Es el seguimiento y la notificación de una actividad no autorizada en un equipo o red monitorizados. Implica funcionalidades como: alertas, registro, elaboración de informes y prevención.

La finalidad de esta prueba es exponer hoyos en dichos elementos de seguridad que podrían ser explotadas por aquellos que tengan intenciones maliciosas. En muchos casos, las pruebas de seguridad se subcontratan con firmas que se especializan en dichas tecnologías. En el caso del II, la misma CSC se ha visto en la necesidad de adquirir tecnologías y desarrollar políticas de seguridad, administradas por algunas áreas de la coordinación; un ejemplo de ello es el área de servidores Windows que, como se ha mencionado, mantiene mucha de la seguridad en lo que a cómputo y redes se refieren.

6.3 NAVEGADORES MÁS UTILIZADOS

Los tres navegadores más utilizados en la actualidad son: Internet Explorer (Microsoft), Firefox (Mozilla) y Chrome (Google).

Conforme avanza la tecnología, las aplicaciones web son cada vez más complejas, por lo que se ha buscado reducir su tiempo de desarrollo y a su vez, tener la seguridad de que las aplicaciones funcionan igual de bien en los navegadores más populares. Es por esta razón que han surgido algunas librerías y frameworks específicos para el desarrollo de aplicaciones con JavaScript, y un ejemplo de éste es el Framework .Net 3.5 de Microsoft.

Microsoft .NET 3.5 incluye la biblioteca ASP.NET AJAX para desarrollar aplicaciones web más eficientes, interactivas y altamente personalizadas que funcionen para la mayoría de los navegadores y utilicen las últimas tecnologías y herramientas Web, incluyendo Silverlight y Popfly.^[22]

A continuación presentamos una lista de los navegadores que mantienen una compatibilidad con la tecnología AJAX, debe tenerse en cuenta que su soporte dependerá de las características que el navegador permita.

- Microsoft Internet Explorer para Windows versión 6.0 ó superiores.
- Navegadores basados en Gecko como Mozilla Firefox versión 1.5 ó superiores.
- Navegadores basados en WebKit como Google Chrome y Apple Safari versiones 2.0 ó superiores.
- Navegadores basados en Presto como Opera versión 9.0 ó superiores.

En el desarrollo del Sistema de Control de Solicitudes se utilizó la biblioteca de AJAX incluida en el Framework 3.5 de Visual Studio 2008, por lo que el sistema fue probado y diseñado para su uso en el navegador más utilizado, Windows Internet Explorer, y que recomendamos para la correcta visualización del sistema. En el caso de los otros dos navegadores más populares se recomienda verificar su funcionamiento y de ser necesario adecuar el sistema conforme a las necesidades de los usuarios.

6.4 LIBERACIÓN

Una aplicación Web está compuesta por un conjunto de archivos, páginas, módulos y código ejecutable, que se invocan o ejecutan dentro del ámbito de un directorio virtual (y sus subdirectorios) en el servidor Web de aplicaciones IIS.

Para poner en marcha el sistema debemos seguir las políticas del II, las cuales incluyen, la coordinación y apoyo con el área de servidores Windows, que como se detalló en el capítulo es la encargada de administrar servidores de bases de datos y aplicaciones desarrolladas para y por el II, pasos que se describen a continuación.

1.- Base de datos

Para cargar la base de datos “CMP_SISOL”, se creó un script con la información necesaria para generar la base de datos que incluye tablas, vistas, procedimientos almacenados, funciones, etc.; así como creación de usuarios y sus correspondientes permisos. Para ejemplificar este paso, la imagen siguiente muestra el script ejecutado, en el entorno SQL Server 2008 del servidor de pruebas.

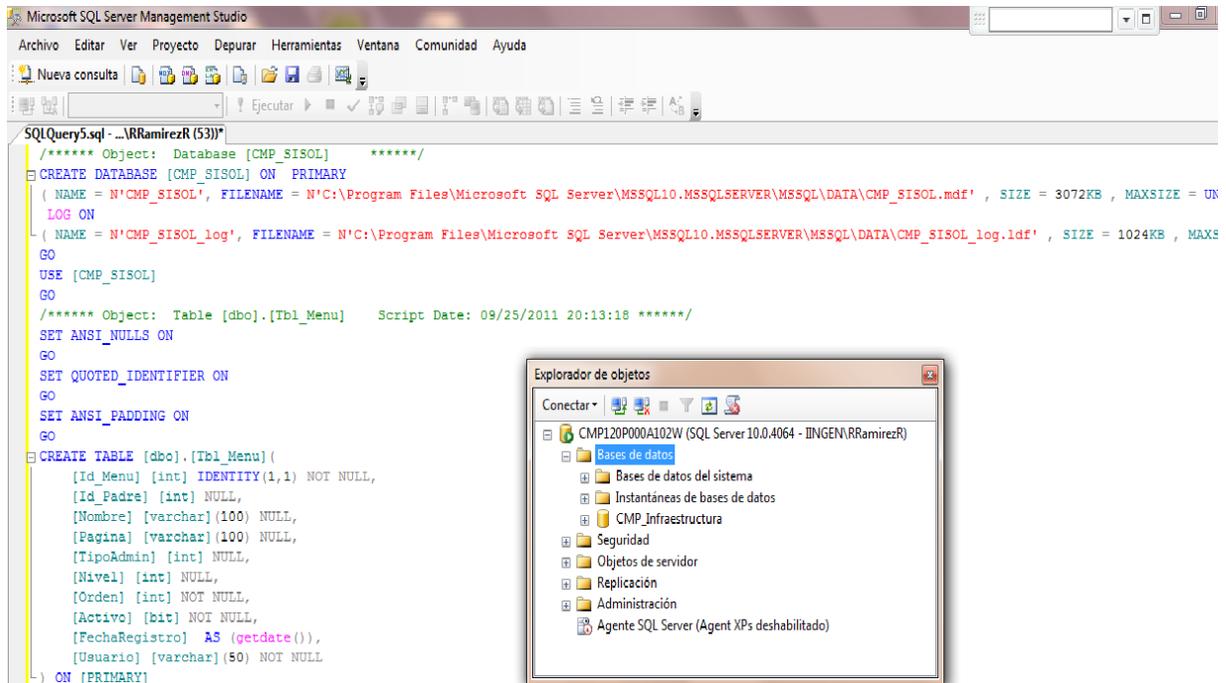


Figura 6.5 Script para la creación de la base de datos

2.- Probar conexión

La cuenta de usuarios necesaria para el acceso a la base de datos es: II_CMP_SISOL, la cual es creada y especificada en el script del paso 1. Para probar la conexión debemos conectarnos al motor de base de datos indicando tipo de autenticación de SQL Server, cuenta y contraseña de usuario II_CMP_SISOL para inicio de sesión. Ver figura 6.6 para ejemplificar este punto.

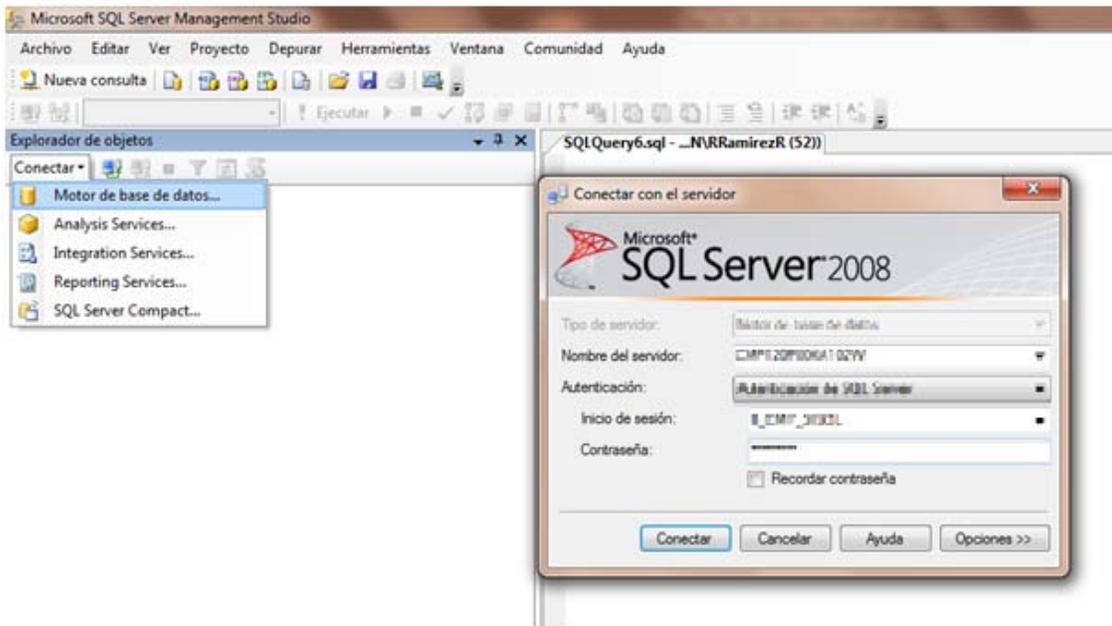


Figura 6.6 Acceso y autenticación a la base de datos

La conexión exitosa se confirma con una pantalla similar a la figura 6.7.

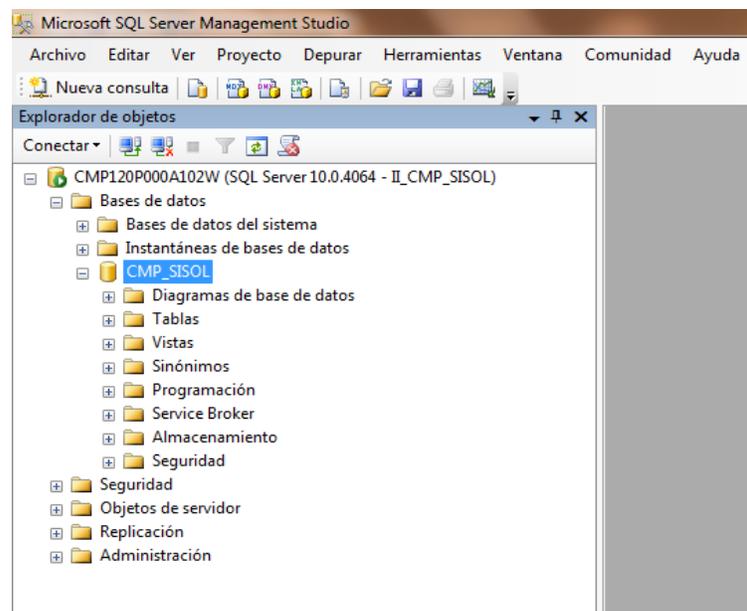


Figura 6.7 Prueba de conexión a la base de datos

3.- Organización de aplicación en servidor de producción

La organización de carpetas y permisos para nuestra aplicación, se detalla en el documento de especificaciones requerido por el área de Windows para su adecuada creación y administración del sitio virtual y grupos de control de usuarios. Dicho documento se puede consultar en el anexo B.

4.- Mantenimiento

Una vez realizados los puntos anteriores, el sistema está liberado y es necesario establecer un plan de mantenimiento de acuerdo con las políticas del II y los integrantes de la CSC responsables de su administración.

Según la terminología ANSI-IEEE, el mantenimiento del software es: “la modificación de un producto software después de su entrega al cliente o usuario para corregir defectos, para mejorar el rendimiento u otras propiedades deseables, o para adaptarlo a un cambio de entorno”.^[30]

Tipos de Mantenimiento

En la definición de mantenimiento aparecen indicados, directa o indirectamente, cuatro tipos de mantenimiento:

- Corregir defectos → Mantenimiento correctivo
- Mejorar el rendimiento → Mantenimiento preventivo/perfectivo
- Adaptar a un cambio de entorno → Mantenimiento adaptativo

Mantenimiento Correctivo

A pesar de las pruebas y verificaciones que aparecen en etapas anteriores del ciclo de vida del software, los programas pueden tener defectos. El mantenimiento correctivo tiene por objetivo localizar y eliminar los posibles defectos de los programas.

Un defecto es una característica del sistema con el potencial de causar un fallo. Un fallo ocurre cuando el comportamiento de un sistema es diferente del establecido en las especificaciones. Entre otros, los fallos en el software pueden ser de:

- Procesamiento, por ejemplo, salidas incorrectas del sistema.
- Rendimiento, por ejemplo, tiempo de respuesta demasiado alto en una búsqueda de información.
- Programación, por ejemplo, inconsistencias en el diseño del sistema.
- Documentación, por ejemplo, inconsistencias entre la funcionalidad del sistema y el manual de usuario.

Mantenimiento Adaptativo

Este tipo de mantenimiento consiste en la modificación del sistema debido a cambios en el entorno (hardware o software) en el cual se ejecuta.

La envergadura del cambio necesario puede ser muy diferente: desde un pequeño retoque en la estructura de un módulo hasta tener que reescribir prácticamente toda la programación del sistema.

Los cambios pueden afectar a:

- El sistema operativo (cambio a una versión actual).
- El entorno de desarrollo del software (incorporación de nuevos elementos o herramientas), los cambios pueden ser de dos clases: en el entorno de los datos y de los procesos.

Este tipo de mantenimiento es cada vez más frecuente debido principalmente al cambio, cada vez más rápido, en los diversos aspectos de la informática: nuevas generaciones de hardware, nuevos sistemas operativos, y mejoras en los periféricos o en otros elementos del sistema (frente a esto, la vida útil de un sistema de software puede superar fácilmente los diez años).

Mantenimiento Perfectivo

Cambios en la especificación, normalmente debidos a cambios en los requerimientos de un producto de software, implican un nuevo tipo de mantenimiento llamado perfectivo, estos casos son muy variados. Desde algo tan simple como cambiar el formato de impresión de un informe, hasta la incorporación de un nuevo módulo funcional.

Entonces podemos definir el mantenimiento perfectivo como el conjunto de actividades para mejorar o añadir nuevas funcionalidades requeridas por el usuario.

Mantenimiento Preventivo

Consiste en la modificación del software para mejorar sus propiedades (por ejemplo, aumentando su calidad y/o su mantenibilidad) sin alterar sus especificaciones funcionales.

Algunas maneras de hacerlo son:

- Incluir sentencias que comprueben la validez de los datos de entrada.
- Reestructurar los programas para mejorar su legibilidad.
- Incluir nuevos comentarios que faciliten la posterior comprensión del programa.

En algunos casos se ha planteado el Mantenimiento para la Reutilización, consistente en modificar el software (buscando y modificando componentes para incluirlos en bibliotecas) para que sea más fácilmente reutilizable. En realidad este tipo de mantenimiento es preventivo, especializado en mejorar la propiedad de reusabilidad del software.

Las actividades de mantenimiento de software se pueden agrupar en tres categorías funcionales:

- Comprensión del software y de los cambios a realizar: para poder modificar un programa, los programadores necesitan conocer su funcionalidad y objetivos, su estructura interna y los requisitos de operación. De no ser así, se corre un gran riesgo de introducir nuevos defectos que en el futuro supondrán un coste de mantenimiento adicional.
- Modificación del software: para incorporar los cambios necesarios se deben crear y modificar las estructuras de datos, la lógica de los procesos, las interfaces y la documentación. Los programadores deben conocer lo mejor posible las repercusiones que tienen en el sistema los cambios que están realizando, con el fin de evitar al máximo posible los efectos secundarios.
- Realización de pruebas: para validar los cambios se deben realizar pruebas selectivas que nos permitan comprobar la corrección del software. Esta actividad es necesaria siempre, ya que incluso un cambio muy pequeño no verificado puede producir defectos en el software que reduzcan su calidad y fiabilidad.

Una vez definidos los puntos anteriores, recomendamos establecer un programa de mantenimiento correctivo, adaptativo, preventivo y perfectivo, de acuerdo al siguiente plan:

- **Correctivo:** A pesar de haber tenido un plan de pruebas y corrección de errores, el sistema podría presentar alguna falla. Por lo que los desarrolladores deberán resolver las incidencias lo más pronto posible durante un periodo de 3 meses a partir de su liberación.
- **Adaptativo:** El sistema fue probado en diferentes equipos, la mayoría con el sistema operativo Windows Vista (una de las tecnologías más actuales) y navegador Internet Explorer 8.0. Se recomienda evaluar el aprovechamiento para nuevos sistemas operativos y versiones posteriores de Internet Explorer, en caso de ser necesario realizar su correspondiente mantenimiento adaptativo.
- **Perfectivo:** Se realizará cuando el sistema sea conocido a la perfección por los usuarios y llegue a la madurez correspondiente para poder realizar cambios que hagan mucho más eficiente al sistema. Es recomendable que el sistema sea usado aproximadamente por 1 año para que los usuarios se familiaricen y puedan aportar mejoras o peticiones de integración con tecnología móvil u otras.
- **Preventivo:** Durante el año de madurez propuesto, se recomienda realizar el monitoreo de la actividad en el sistema para poder identificar, analizar y llevar a cabo las modificaciones necesarias, reestructuraciones de código y diseño del sistema.

CONCLUSIONES

En apariencia la recopilación de requerimientos es una tarea relativamente sencilla pero en muchas ocasiones el desarrollador se enfrenta a procesos difíciles de recabación de información, como sucedió en nuestro caso; donde la tarea de reunir a los jefes de área de la CSC, plantear la estructura del sistema y obtener de manera específica el catálogo de problemas más comunes por área, fue un proceso lento que se llevó a cabo en varias sesiones. Cabe destacar que el ser miembros de la comunidad del II, nos ofrece la ventaja de conocer la estructura de la CSC, sus procesos y actividades en los servicios que ofrece y así poder agilizar esta etapa fundamental en el éxito del proyecto.

Al utilizar la metodología UML e implementar sus 4 fases para el desarrollo del Sistema de Control de Servicios, pudimos corroborar porque constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos; ésto debido a que su capacidad de adaptar las metodologías al contexto y necesidades de cada organización, se lograron aplicar al II y nos fueron de gran utilidad al facilitar y permitir la entrega del sistema con las características especificadas en los requerimientos.

Tomando siempre en cuenta las necesidades, objetivos y recursos disponibles del Instituto de Ingeniería, el Sistema de Control de Servicios se desarrolló con las herramientas SQL Server 2008, Visual Studio 2008 y lenguaje C#, implementando una estructura de 3 capas; lo cual nos permitió reutilizar componentes, mejorar y acortar el tiempo de construcción, así como facilitar su mantenimiento y la posibilidad de escalar el sistema a futuras versiones, de acuerdo a las nuevas tecnologías adoptadas por la CSC.

El Modelo de Procesos de Software (MoProSoft) nos fue de mucha utilidad como una guía de referencia para mantener la calidad del Sistema de Control de Servicios, al que pudimos aplicar el modelo que se adapta perfectamente a las PyMES categoría dónde se encuentra la CSC del II. También nos permitió considerar los procesos importantes del II y optimizar los resultados de la CSC procurando mejorar la productividad y la calidad de sus servicios para aumentar su competitividad. Por lo que podemos recomendar a la CSC el uso del modelo para su continua implementación para mantener y aumentar la calidad del software desarrollado en la Coordinación.

Las diferentes pruebas realizadas al Sistema de Control de Servicios, fueron de gran utilidad para corregir a tiempo los errores no detectados que se cometieron involuntariamente en cada una de las etapas de desarrollo. Aunque sabemos que el sistema en su implementación no está cien por ciento libre de errores, el seguir un guión de pruebas nos ayudó en esta etapa y consideramos que es muy recomendable para poder entregar un sistema donde su liberación sea con la menor posibilidad de incidencias de error.

Las herramientas disponibles actualmente en el II, además de ser unas de las tecnologías más modernas, son también de las más usadas en el desarrollo, implementación y mantenimiento de software, nos permitieron completar este proyecto bajo las especificaciones requeridas y con la posibilidad de aumentar sus alcances.

Esperamos que la herramienta desarrollada por nosotros, para el Instituto de Ingeniería, sea de muchísima utilidad a la Coordinación de Sistemas de Cómputo en el control y administración de los servicios que presta hacia su comunidad y viceversa; puesto que el sistema necesita de la retroalimentación de las experiencias de los usuarios en su uso para así lograr un madurez y en su momento poder ofrecer nuevas funcionalidades.

Con la finalización de nuestro trabajo de tesis también podemos concluir, que nos ayudó a reforzar la práctica del trabajo en equipo, que es una actividad que en la actualidad nos ayuda a desenvolvernos en el área laboral, puesto que es algo cotidiano y esencial en las empresas en las que trabajamos. De igual manera tenemos cómo experiencia que la constante comunicación entre los desarrolladores y el cliente es fundamental para entregar un buen sistema de acuerdo a sus necesidades. Así como también es un orgullo poder concluir esta meta que el equipo se propuso desde el inicio del proyecto, haciéndonos crecer y ser personas responsables tanto en el ámbito personal como profesional.

Finalmente podemos concluir que al realizar este trabajo tanto práctico como teórico, hemos podido poner en práctica varios de los conocimientos adquiridos: a lo largo de nuestra carrera en la Facultad de Ingeniería, durante la realización del servicio social y a la estancia como becarios en el Instituto de Ingeniería. Así mismo la experiencia y conocimientos obtenidos en estas instituciones nos dieron la oportunidad de ubicarnos en una ambiente laboral para desarrollarnos como profesionistas.

BIBLIOGRAFÍA

- [1] Booch Grady, Rumbaugh James y Jacobson, Ivar. "El Lenguaje Unificado de Modelado." España Addison Wesley, 2000.
- [2] Booch Grady, Rumbaugh James y Jacobson, Ivar. "El Proceso Unificado de Desarrollo de Software." España Addison Wesley, 2000.
- [3] Chrissis M B, Konrad Mike, Shrum Sandy. "CMMI: guidelines for process integration and product improvement".
- [4] Del Río González Cristóbal, "Costos Históricos" 21ª. Edición, Editorial Thomson, México, 2003.
- [5] Elmasri y Navathe: "Fundamentos de Sistemas de Bases de Datos" 3ª Ed. Addison Wesley Iberoamericana 2002.
- [6] Gabillaud Jérôme SQL Server 2008 SQL, Transact SQL Diseño y creación de una base de datos, Ediciones ENI Abril 2009.
- [7] Garcia-Molina, Ullman y Widom: "Database System: the complete book" Prentice-Hall.
- [8] Ocampo Sámano, José Eliseo "Costos y Evaluación de Proyectos" 1ª Edición, Compañía editorial Continental, México, 2002.
- [9] Oktaba Hanna, Claudia Alquicira Esquivel Modelo de Procesos para la Industria de Software MoProSoft Versión 1.3
- [10] Olivier Dewit: "ASP.NET: programación web con Visual Studio y Web Matrix" Ediciones ENI, 2003.
- [11] Osorio Rivera Fray León "Bases de datos relacionales teoría y práctica" 1ra. Edición diciembre 2008, Fondo Editorial ITM Instituto Tecnológico Metropolitano.
- [12] Pressman Roger S "Ingeniería del software. Un enfoque practico." 3ª Edición. Ed. McGraw-Hill. España 1993.
- [13] Rob Peter, Coronel Carlos "Sistemas de bases de datos: diseño, implementación y administración" Edición 5, Editor Cengage Learning Editores, 2003.
- [14] Soto Pineda, Eduardo y Dolan, Simon L. "Las PYMES ante el desafío del siglo XXI: los nuevos mercados globales" Editorial Thomson, México, 2004.
- [15] Weitzenfeld, Alfredo "Ingeniería de software orientada a objetos con UML, Java e Internet".
- [16] "Estimación de costos". 1ª Edición, Editorial Aconagua Ediciones y publicaciones, México, 1995.

Melografía

- [17] <http://www.iingen.unam.mx>
- [18] http://www.eqsoft.net/presentas/modelos_de_calidad_y_software_libre.pdf
- [19] http://modelosdegestiondelacalidad.blogspot.com/2008/01/modelo-cmmi_12.html
- [20] <http://www.unalmed.edu.co/~mstabare/Dbms.htm>
- [21] <http://msdn.microsoft.com>
- [22] <http://msdn.microsoft.com/netframework/>
- [23] <http://www.observatoriopyme.com>
- [24] <http://www.enterate.unam.mx/Articulos/2006/marzo/moprosoft.htm>
- [25] <http://www.comunidadmoprosoft.org.mx/>
- [26] <http://www.ieee.org>
- [27] <http://www.iso.org>

Otra bibliografía

- [28] Documento Pdf Plan estratégico 2005-2010 ProSoft ,Gobierno del Estado de Veracruz
- [29] Libros en pantalla de SQL Server 2008
- [30] Presentación Pdf “Mantenimiento del Software” (S1), Francisco Ruiz, Macario Polo Dpto. de Informática, Grupo Alarcos, Escuela Superior de Informática, Universidad de Castilla la Mancha, 2001.

ANEXO A

DICcionario DE DATOS

Tabla	Descripción de tabla					
Ctl_Areas	Tabla para almacenar áreas de la coordinación					
Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)	
Id_Area	int	IDENTITY	Clave principal de Ctl_Areas	Sí	No	
Descripcion	nvarchar(50)	NO	Descripción del área	No	No	
Activo	bit	NO	Estado	No	No	
Ctl_CalifUsuario	Tabla para almacenar las calificaciones que el usuario puede asignar a la atención del servicio.					
Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)	
Id_CalifUsuario	int	IDENTITY	Clave principal de Ctl_CalifUsuario	Sí	No	
Descripcion	nvarchar(200)	NO	Descripción de calificación	No	No	
CalificacionNum	int	NO	Número de la calificación	No	No	
Activo	bit	NO	Estado	No	No	
Ctl_Empresa	Tabla para almacenar las empresas de servicio externo.					
Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)	
Id_Empresa	int	IDENTITY	Clave principal de Ctl_Empresa	Sí	No	
Id_Area	int	NO	Identificador de área	No	Sí	

FiContrato	datetime	SÍ	Fecha inicio del contrato	No	No
FfContrato	datetime	SÍ	Fecha fin del contrato	No	No
NombTecnico	nvarchar(50)	SÍ	Nombre del técnico	No	No
NombEmpresa	nvarchar(100)	NO	Nombre de la empresa	No	No
RFC	nvarchar(25)	NO	RFC de la empresa	No	No
Direccion	nvarchar(100)	NO	Dirección	No	No
Gerente	nvarchar(50)	NO	Nombre del gerente	No	No
Email	nvarchar(50)	NO	Correo electrónico	No	No
Tel_cel	nvarchar(20)	NO	Teléfono celular	No	No
Tel_Ofic	nvarchar(20)	NO	Teléfono de oficina	No	No
Activo	bit	NO	Estado	No	No

Ctl_Prioridad Tabla para almacenar las prioridades.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Prioridad	int	IDENTITY	Clave principal de Ctl_Prioridad	Sí	No
Descripcion	nvarchar(50)	NO	Descripción de prioridad	No	No
Activo	bit	NO	Estado	No	No

Ctl_Problemas Tabla para almacenar problemas por área.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Problema	int	IDENTITY	Clave principal de Ctl_Problemas	Sí	No
Descripcion	nvarchar(300)	NO	Descripción del problema	No	No
Id_Padre	int	NO	Identificador de padre del problema	No	Sí

Id_Area	int	NO	Identificador de área	No	Sí
DiasCompromiso	int	NO	Dias compromiso para la atención del problema	No	No
Activo	bit	NO	Estado	No	No

Ctl_StatusSolicitud Tabla para almacenar estados de la solicitud.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_StatusSolicitud	int	IDENTITY	Clave principal de Ctl_StatusSolicitud	Sí	No
Descripcion	nvarchar(50)	NO	Descripción	No	No
Activo	bit	NO	Estado	No	No

Ctl_Tipo_Admin Tabla para almacenar tipo de administrador.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
IdTipoAdmin	int	NO	Clave principal de Ctl_Tipo_Admin	Sí	No
Descripcion	varchar(150)	SÍ	Descripción de tipo de administrador	No	No
Activo	bit	NO	Estado	No	No

Ctl_TipoSolicitud Tabla para almacenar los tipos de solicitud.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_TipoSolicitud	int	IDENTITY	Clave principal de Ctl_TipoSolicitud	Sí	No
Descripcion	nvarchar(50)	NO	Descripción de tipo de solicitud	No	No
Activo	bit	NO	Estado	No	No

Edificio

Tabla para almacenar los edificios.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
ID_Edif	int	NO	Clave principal de Edificio	Sí	No
ID_Plantel	int	NO	Identificador de plantel	No	Sí
Nom_Edif	varchar(50)	NO	Nombre del edificio	No	No
Num_Edif	varchar(10)	NO	Número del edificio	No	No
No_Pisos	int	NO	Número de piso	No	No
Coor_X	varchar(30)	NO	Coordenadas X	No	No
Coor_Y	varchar(30)	NO	Coordenadas Y	No	No
Nom_Resp	varchar(50)	SÍ	Nombre del responsable	No	No
Ficha_Contac	varchar(50)	SÍ	Ficha de contacto	No	No

Espacio

Tabla para almacenar los espacios disponibles para ubicación.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
ID_Espacio	int	NO	Clave principal de Espacio	Sí	No
Descrip	varchar(40)	NO	Descripción de espacio	No	No

Numero

Tabla para almacenar el número de pisos por espacio.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
ID_Numero	int	NO	Clave principal de Número	Sí	No
ID_Piso	int	NO	Identificador de piso	No	Sí
ID_Espacio	int	NO	Identificador de espacio	No	Sí
No_Telef	varchar(25)	SÍ	Número telefónico	No	No
Descrip	varchar(50)	SÍ	Descripción	No	No

Piso Tabla para almacenar los pisos por espacio

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
ID_Piso	int	NO	Clave principal de Piso	Sí	No
ID_Edif	int	NO	Identificador de edificio	No	Sí
Descrip	varchar(50)	SÍ	Descripción	No	No

Plantel Tabla para almacenar planteles.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
ID_Plantel	int	NO	Clave principal de Plantel	Sí	No
Nom_Plantel	varchar(50)	NO	Nombre de plantel	No	No
No_Edfs	int	NO	Número de edificio	No	No
Nom_Resp	varchar(50)	SÍ	Nombre del responsable	No	No
Ficha_Contac	varchar(50)	SÍ	Ficha de contacto	No	No
Descrip	varchar(50)	SÍ	Descripción	No	No

Tbl_Administrador Tabla para almacenar los administradores del sistema.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Admin	int	IDENTITY	Clave principal de Tbl_Administrador	Sí	No
TipoAdmin	int	NO	Identificador de tipo administrador	No	Sí
Id_Area	int	NO	Identificador de área	No	No
UserName	nvarchar(50)	NO	Nombre de usuario en dominio	No	No
Nombre	nvarchar(50)	NO	Nombre del administrador	No	No

Correo nvarchar(50) NO Correo electrónico No No

Tbl_AdminStatusSolic Tabla para almacenar los estados que puede asignar un administrador.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_AdminStatusSolic	int	IDENTITY	Clave principal de Tbl_AdminStatusSolic	Sí	No
TipoAdmin	int	NO	Descripción de tipo de administrador	No	No
Id_StatusSolicitud	int	NO	Identificador de estado de la solicitud	No	Sí

Tbl_Asignacion Tabla para almacenar la asignación de solicitudes.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Asignacion	int	IDENTITY	Clave principal de Tbl_Asignacion	Sí	No
Id_Solicitud	int	NO	Identificador de solicitud	No	Sí
Id_AdminAsigno	int	NO	Identificador de administrador que asigno la solicitud	No	Sí
Id_AdminAsignado	int	NO	Identificador de administrador asignado a la solicitud	No	Sí
FechaAsignacion	datetime	NO	Fecha de asignación	No	No

Tbl_Aux_Not_Area Tabla para almacenar las noticias por área.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Aux_Not_Area	int	IDENTITY	Clave principal de Tbl_Aux_Not_Area	Sí	No

Id_Noticia	int	NO	Identificador de noticia	No	Sí
Id_Area	int	NO	Identificador de área	No	Sí

Tbl_DatosUsrHist Tabla para almacenar los datos históricos de usuarios.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_DatosUsrHist	int	IDENTITY	Clave principal deTbl_DatosUsrHist	Sí	No
Id_UsuarioEx	int	NO	Identificador de usuario externo	No	Sí
Id_VigenciaEx	int	NO	Identificador de vigencia externa	No	Sí
FechRegistro	datetime	NO	Fecha de registro	No	No
Puesto	nvarchar(50)	NO	Puesto	No	No
Compañía	nvarchar(50)	NO	Compañía	No	No
Depto	nvarchar(50)	NO	Departamento	No	No
Oficina	nvarchar(50)	NO	Número telefónico de oficina	No	No
Telefono	nvarchar(15)	NO	Número telefónico personal	No	No
Ciudad	nvarchar(50)	NO	Ciudad	No	No

Tbl_EquipoComputo Tabla para almacenar las características del equipo de cómputo.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_EquipoComputo	int	IDENTITY	Identificador de Tbl_EquipoComputo	Sí	No
NoInventario	nvarchar(50)	NO	Número de inventario	No	No
Descripcion	nvarchar(50)	SÍ	Descripción del equipo	No	No
Marca	nvarchar(50)	SÍ	Marca	No	No
Modelo	nvarchar(50)	SÍ	Modelo	No	No
NoSerie	nvarchar(50)	SÍ	Número de serie	No	No

Tbl_EquipoEx Color nvarchar(50) SÍ Color No No
 Tabla para almacenar equipo externo.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_EquipoEx	int	IDENTITY	Clave principal de Tbl_EquipoEx	Sí	No
Id_VigenciaEx	int	NO	Identificador de vigencia externa	No	Sí
NoInventario	nvarchar(50)	SÍ	Número de inventario del equipo	No	No
NomPropietario	nvarchar(100)	NO	Nombre del propietario del equipo	No	No
Marca	nvarchar(50)	NO	Marca	No	No
Modelo	nvarchar(50)	NO	Modelo	No	No
No_Serie	nvarchar(50)	NO	Número de serie	No	No

Tbl_HistoricoStatus Tabla para almacenar histórico de estado de la solicitud.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_HistoricoStatus	int	IDENTITY	Clave principal de Tbl_HistoricoStatus	Sí	No
Id_Solicitud	int	NO	Identificador de solicitud	No	Sí
Id_StatusSolicitud	int	NO	Identificador de estado de la solicitud	No	Sí
Id_Admin	int	NO	Identificador de administrador	No	Sí
Id_Area	int	NO	Identificador de área	No	Sí
Id_Prioridad	int	NO	Identificador de prioridad	No	Sí
Id_Problema	int	NO	Identificador de problema	No	Sí
Id_CalifUsuario	int	NO	Identificador de calificación de	No	Sí

SolucionAdmin	nvarchar(4000)	NO	usuario	Solución del administrador	No	No
Justificacion	nvarchar(200)	NO		Justificación	No	No
FechaAct	datetime	NO		Fecha de actualización	No	No

Tbl_Menu

Tabla para almacenar las opciones de menú.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Menu	int	IDENTITY	Clave principal de Tbl_Menu	Sí	No
Id_Padre	int	SÍ	Identificador de padre	No	No
Nombre	varchar(100)	SÍ	Nombre	No	No
Pagina	varchar(100)	SÍ	Nombre de la página	No	No
TipoAdmin	int	SÍ	Identificador de tipo administrador	No	Sí
Nivel	int	SÍ	Nivel	No	No
Orden	int	NO	Orden	No	No
Activo	bit	NO	Estado	No	No
FechaRegistro	datetime	NO	Fecha de registro	No	No
Usuario	varchar(50)	NO	Usuario	No	No

Tbl_Noticias

Tabla para almacenar las noticias

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Noticia	int	IDENTITY	Clave principal de Tbl_Noticias	Sí	No
Id_Prioridad	int	NO	Identificador de prioridad	No	Sí
Noticia	nvarchar(300)	NO	Descripción de noticia	No	No
FechaIni	datetime	NO	Fecha de inicio para mostrar la noticia	No	No
FechaFin	datetime	NO	Fecha de fin para mostrar la noticia	No	No

FechaReg	datetime	NO	Fecha de registro	No	No
FechaAct	datetime	NO	Fecha de actualización	No	No
idTipoAdmin	int	NO	Identificador de tipo de administrador	No	Sí

Tbl_ServicioEx

Tabla para almacenar las solicitudes en servicio externo

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_ServicioEx	int	IDENTITY	Clave principal de Tbl_ServicioEx	Sí	No
Id_Solicitud	int	NO	Identificador de solicitud	No	Sí
NoFolio	varchar(50)	NO	Número de folio	No	No
Id_Empresa	int	NO	Identificador de empresa	No	Sí
FechaRegistro	datetime	NO	Fecha de registro de solicitud en servicio externo	No	No

Tbl_Software

Tabla para almacenar software.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Software	int	IDENTITY	Clave principal de Tbl_Software	Sí	No
Id_Admin	int	NO	Identificador de administrador	No	Sí
NombreOriginal	varchar(350)	NO	Nombre del software	No	No
NombreFisico	varchar(250)	NO	Nombre físico del software	No	No
Comments	varchar(350)	SÍ	Comentarios	No	No
Fecha	datetime	NO	Fecha de registro	No	No

Tbl_Solicitud

Tabla para almacenar las solicitudes registradas.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_Solicitud	int	IDENTITY	Clave principal de Tbl_Solicitud	Sí	No
Id_TipoSolicitud	int	NO	Identificador de tipo solicitud	No	Sí
Id_StatusSolicitud	int	NO	Identificador de estado de la solicitud	No	Sí
Id_EquipoComputo	int	NO	Identificador de equipo de cómputo	No	Sí
NombreRegistro	nvarchar(100)	NO	Nombre de quien registro la solicitud	No	No
NoInventario	nvarchar(50)	NO	Número de inventario	No	No
DescrProblema	nvarchar(500)	SÍ	Descripción del problema	No	No
Diagnostico	nvarchar(300)	SÍ	Diagnostico del problema reportado	No	No
OpinionUsuario	nvarchar(500)	SÍ	Opinion del usuario	No	No
FechaSolicitud	datetime	SÍ	Fecha de registro de solicitud	No	No
FechaFinSolicitud	datetime	SÍ	Fecha fin de la solicitud	No	No
NombreAtender	nvarchar(100)	SÍ	Nombre del personal que se va a atender	No	No
NombreSecAtender	nchar(50)	SÍ	Nombre de personal alternativo a atender	No	No
UbicacionServ	nvarchar(100)	NO	Ubicación del servicio	No	No
ContactoTel	nvarchar(100)	SÍ	Contacto telefónico	No	No
Observaciones	nvarchar(300)	SÍ	Observaciones	No	No
SolCorrecta	bit	SÍ	Identificador de solución correcta	No	No

Usuario	varchar(20)	SÍ	Nombre de usuario en dominio	No	No
---------	-------------	----	------------------------------	----	----

Tbl_UsuarioEx

Tabla para almacenar los usuarios externos.

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_UsuarioEx	int	IDENTITY	Clave principal de Tbl_UsuarioEx.	Sí	No
Id_VigenciaEx	int	NO	Identificador de vigencia externa.	No	Sí
Nombre	nvarchar(50)	NO	Nombre de usuario.	No	No
ApellidoPat	nvarchar(50)	NO	Apellido Paterno.	No	No
ApellidoMat	nvarchar(50)	NO	Apellido Materno.	No	No

Tbl_VigenciaEx

Tabla para almacenar la vigencia externa de equipo o usuarios

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_VigenciaEx	int	IDENTITY	Clave principal de Tbl_VigenciaEx	Sí	No
FechaIni	datetime	NO	Fecha inicio de la vigencia	No	No
FechaFin	datetime	NO	Fecha fin de la vigencia	No	No
NomAcadResp	nvarchar(50)	NO	Nombre del académico responsable	No	No
JustificacionAcad	text	NO	Justificación del académico	No	No
JustificacionCoordinador	text	NO	Justificación del coordinador	No	No
StatusEx	bit	NO	Estado de la vigencia	No	No

Tbl_VigHistEquipo Tabla para almacenar histórico de la vigencia de equipo

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_VigHistEquipo	int	IDENTITY	Clave principal de Tbl_VigHistEquipo	Sí	No
Id_VigenciaEx	int	NO	Identificador de vigencia externa	No	Sí
Id_EquipoEx	int	NO	Identificador de equipo externo	No	No

Tbl_VigHistUsuario Tabla para almacenar histórico de la vigencia de usuarios

Columna	Tipo de dato	Nulos	Descripción de campo	Llave primaria (PK)	Llave foránea (FK)
Id_VigHistUsuario	int	IDENTITY		Sí	No
Id_VigenciaEx	int	NO	Identificador de vigencia externa	No	Sí
Id_UsuarioEx					

ANEXO B



Documentos de especificaciones para un sistema

Sistema de Control de Servicios para el Instituto de Ingeniería

Instituto de Ingeniería, UNAM

Coordinación de Sistemas de Cómputo

Área de Bases de Datos

Autores

Rosaura Apresa Monzón

Liliana González Jiménez

Raúl Trinidad Ramírez y Ramírez

Documentos de especificaciones para liberación de un sistemas

Revisor por: RApresaM (09/11/2011 12:36 A.M.)

Documento generado por el Instituto de Ingeniería

Índice

I.Descripción general del sistema.....	115
a) Nombre del Sistema	115
b) Descripción.....	115
c) Responsables del sistema	115
d) Directorio de contacto	115
II.Requerimientos técnicos del sistema	115
e) Roles de Usuario.....	115
f) Sitio web de alojamiento	115
g) Tipo de Acceso al sitio web.....	115
h) Listado de Carpetas que conformar toda la aplicación.....	115
i) Permisos de Carpetas.....	116
j) Requisitos de la Base de Datos.....	116
k) Permisos de la Base de Datos	116
III.Consideraciones adicionales.....	116

Descripción general del sistema

Nombre del Sistema

Sistema de Control de Servicios de cómputo para el Instituto de Ingeniería

Descripción

El objetivo principal del sistema es apoyar, agilizar, facilitar y administrar la atención de solicitudes de servicio, a través de páginas Web en Internet.

Responsables del sistema

Coordinación de Sistemas de Cómputo

Directorio de contacto

En este punto se muestran los datos para contactar a los administradores del Sistema de Control de Servicios.

Requerimientos técnicos del sistema

Roles de Usuario

Hay dos tipos de Usuario:

Usuarios: En este punto se enlistan a los usuarios que tendrán acceso al sistema como "Usuarios".

Su función en el sistema es de realizar una solicitud de atención de servicio vía página Web y recibir notificación vía correo electrónico cuando la solicitud es realizada y atendida.

Administradores: En este punto se enlistan a los usuarios que tendrán acceso al sistema como "Administradores".

Su función es la de entrar al sistema vía página Web para administrar la lista de solicitudes del *Usuario* una vez que recibe notificaciones vía correo electrónico. Así como aprobar, rechazar o cancelar solicitudes PEX.

Sitio web de alojamiento

Se especifica la carpeta dentro del servidor donde se alojara el Sistema

Tipo de Acceso al sitio web

Para acceder a cualquier página Web del sistema, se debe pedir cuenta y contraseña de Windows, por lo que es necesario que la carpeta virtual en el IIS tenga autenticación de Windows. Sólo acceden el *Usuario* y el *Administrador*.

Listado de Carpetas que conformar toda la aplicación

- En este punto, se muestra la lista de carpetas y subcarpetas que contienen al sistema

Documentos de especificaciones para liberación de un sistemas

Revisor por: RApresaM (09/11/2011 12:36 A.M.)

Documento generado por el Instituto de Ingeniería

Permisos de Carpetas

- SISOL (Raíz de la aplicación)

Lista de permisos en las diferentes carpetas que conforman el Sistema de Control de Servicios.

Requisitos de la Base de Datos

SQL 2008 en adelante
Tamaño: 4 MB mínimo

Permisos de la Base de Datos

Base de datos: (Nombre de la Base de Datos en el Sistema)
Cuentas de usuario de SQL: Grupos de cuentas con permiso a la Base de datos

Consideraciones adicionales

Sin consideraciones.

GLOSARIO

Active Directory: Servicio de Directorio de Microsoft.

ActiveX: Entorno para definir componentes de software reusables de forma independiente del lenguaje de programación.

AJAX: Asynchronous JavaScript And XML.

AMCIS: Asociación Mexicana para la Calidad en Ingeniería de Software.

BD: Base de Datos.

CLR: Common Language Runtime, 'Entorno de Ejecución'.

Cluster: Conjuntos de computadoras construidos mediante la utilización de hardware comunes y que se comportan como si fuesen una única computadora.

CMMI: Capability Maturity Model Integration, 'Modelo de Capacidad y Madurez'.

CPU: Unidad Central de Procesamiento.

CSC: Coordinación de Sistemas de Cómputo.

CU: Ciudad Universitaria.

DBA: Database Administrator, 'Administrador de la Base de Datos'.

DIR: Alta Dirección.

ER: Entidad Relación.

EvalProSoft: Evaluación de Procesos de Software.

Firewall: Cortafuegos que se utiliza con frecuencia para evitar que los usuarios de Internet no autorizados tengan acceso a redes privadas.

Freeware: Software que se distribuye sin costo.

Ftp: File Transfer Protocol, 'Protocolo de Transferencia de Archivos'.

GES: Gestión.

Gecko: Motor de los programas de la fundación Mozilla.

Gigabit Ethernet: GigaE, es una ampliación del estándar Ethernet que consigue una capacidad de transmisión de 1 gigabit por segundo.

Hardware: Corresponde a todas las partes tangibles de un sistema informático.

IBM: International Business Machines

IDE: Integrated Development Environment, 'Entorno Integrado de Desarrollo'.

IE: Internet Explorer.

IEEE: Institute of Electrical and Electronic Engineers, 'Instituto de Ingenieros en Electricidad y Electrónica'.

IIS: Internet Information Services

ISO: International Organization for Standardization, 'Organización Internacional para la Estandarización'.

Mantenibilidad: Facilidad con la que un sistema o componente software puede ser modificado para corregir fallos, mejorar su funcionamiento u otros atributos o adaptarse a cambios en el entorno

MoProsoft: Modelo de Procesos de Software.

MS: Mantenimiento de Software.

NNTP: Network News Transport Protocol.

NYCE: Normalización y Certificación Electrónica.

ODBC: Open Database Connectivity.

OPE: Operación.

Popfly: Sitio web que permite al usuario crear páginas web usando runtimes para aplicaciones ricas de internet basadas en Microsoft Silverlight.

Presto: Motor de los programas de Opera Software.

PyME: Pequeña y Mediana Empresa.

SE: Secretaría de Economía

SEQUEL: Structured English Query Language.

Sharepoint: Plataforma web de trabajo colaborativo y gestión documental, especialmente orientada a documentos Microsoft Office.

Shareware: Distribución de software, en la que el usuario puede evaluar de forma gratuita el producto.

Silverlight: Estructura para aplicaciones web que agrega nuevas funciones multimedia como la reproducción de vídeos, gráficos vectoriales, animaciones e interactividad.

SISAL: Unidad Académica Sisal del Instituto de Ingeniería de la UNAM.

SMBD: Sistema Manejador de Bases de Datos.

SMTP: Simple Mail Transfer Protocol, 'Protocolo Simple de Transferencia de Correo'

SO: Sistema Operativo.

SOAP: Simple Object Access Protocol.

Software: Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

Spyware: Software que rastrea la información personal o confidencial.

SSH: Secure SHell, 'Intérprete de órdenes segura'.

Streaming: Distribución de multimedia a través de una red de computadoras de manera que el usuario consume el producto al mismo tiempo que se descarga.

SQL: Structured Query Language, 'Lenguaje de Consulta Estructurado'.

Tomcat: Servicio de motor de aplicaciones.

Tupla:Fila.

UML: Unified Modeling Language, 'Lenguaje Unificado de Modelado'.

WebKit: Plataforma para aplicaciones que funciona como base para el navegador web Google Chrome.

XML: Extensible Markup Language, 'Lenguaje de marcas Extensible'.