

Capítulo 2

Análisis del sistema

2.1 Propuesta de solución

A partir de la definición del problema y la descripción de la situación actual se plantea como solución el desarrollo de un sistema integral que reduzca las fallas y brinde un mejor servicio.

El sistema estará basado en una arquitectura cliente – servidor donde el elemento de presentación será el único del lado del cliente a través de una interfaz de usuario constituida por una página Web que sea visible por cualquier navegador Web evitando así la instalación de software adicional.

Las partes de lógica de negocio y gestión de datos se encontrarán del lado del servidor, donde un lenguaje de programación será la herramienta necesaria para crear los módulos encargados de la seguridad, de generar las páginas dinámicas que muestren y obtengan información, de darle persistencia en una base de datos, así como consultarla y modificarla, etc.

En concreto el sistema permitirá mediante un portal Web la gestión, asignación, visualización y revalidación de las calificaciones de laboratorios pertenecientes al Departamento de Ingeniería en Computación indicados como (L+) en el mapa curricular, lo que significa que deben inscribirse por separado en un horario diferente pero carentes de valor en créditos.

Las calificaciones serán asignadas cada semestre por los profesores de laboratorio (L+) a quienes se asignará un usuario y una contraseña para acceder vía Web al sistema, una vez confirmada su identidad podrán elegir los grupos que imparten con las listas de alumnos inscritos regularmente y ASDRI, así como las posibles calificaciones numéricas de 5 a 10 o NP que podrán indicar de forma sencilla a cada uno.

Capítulo 2

Para el caso de los ASDRI es indispensable realicen un proceso de registro por medio del cual serán incorporados a la lista de laboratorio y así puedan ser evaluados por el profesor como el resto de los alumnos. El trámite consistirá de forma similar al realizado en las teorías en descargar un formato disponible en el portal de asignación de calificaciones, llenar los campos requeridos con los datos personales del ASDRI y del laboratorio donde ha sido aceptado por el profesor quien deberá firmar el formato de registro para avalar la inscripción.

Una vez completado el formato con la información requerida éste será entregado en el Departamento de Ingeniería en Computación donde el administrador con acceso al sistema registrará al alumno en un periodo de tiempo coincidente con el proceso de alta de ASDRI en teoría. Cabe señalar que el profesor de laboratorio puede reservarse el derecho de aceptar o no alumnos en esta situación.

Una vez evaluados todos los grupos de laboratorio e incorporadas las calificaciones acumuladas de semestres previos como resultado de la revalidación, se procede a las opciones de visualización y descarga de calificaciones por los profesores de teoría que contarán también con privilegios de acceso al sistema mediante un usuario y una contraseña en el mismo portal Web.

El desarrollo del sistema y los elementos necesarios para su funcionamiento estarán a cargo del trabajo conjunto del Departamento de Ingeniería en Computación y la USECAD, donde el primero es responsable del proceso de calificaciones de laboratorio y su revalidación, mientras que la USECAD proveerá el alojamiento del sistema en uno de sus servidores y aportará las listas de alumnos de teorías y laboratorios resultantes del proceso de inscripción de cada semestre.

El software propuesto está ubicado dentro del contexto de un problema real por lo cual su análisis, desarrollo e implementación deben ser llevados bajo criterios formales dada su importancia, complejidad e impacto sobre las personas a las que está destinado el sistema.

Ingeniería de software

En vista de lo anterior el proceso del proyecto debe seguirse bajo el marco de la ingeniería de software concebida como el área de las ciencias de la computación que ofrece métodos, técnicas y herramientas para el desarrollo, operación y mantenimiento de software con el fin de crear un producto confiable y de calidad.

Se conforma por capas como se observa en la figura 2.1 donde la base es la calidad lograda bajo ciertos estándares y donde los requerimientos del cliente son satisfechos; la capa del proceso o también conocida como ciclo de vida del software define el marco de trabajo donde se aplican los métodos técnicos para obtener productos definiendo las actividades necesarias en las etapas de definición, desarrollo y mantenimiento.

El método especifica cómo construir técnicamente el software abarcando tareas como el análisis de requerimientos, el modelado del diseño que contempla las estructuras de datos, la arquitectura de programación y los algoritmos; la construcción del programa, las pruebas y el mantenimiento.

La capa de herramientas brinda el soporte automatizado o semiautomático para las capas de proceso y método logrando que la información procedente de una pueda ser usada en la otra.

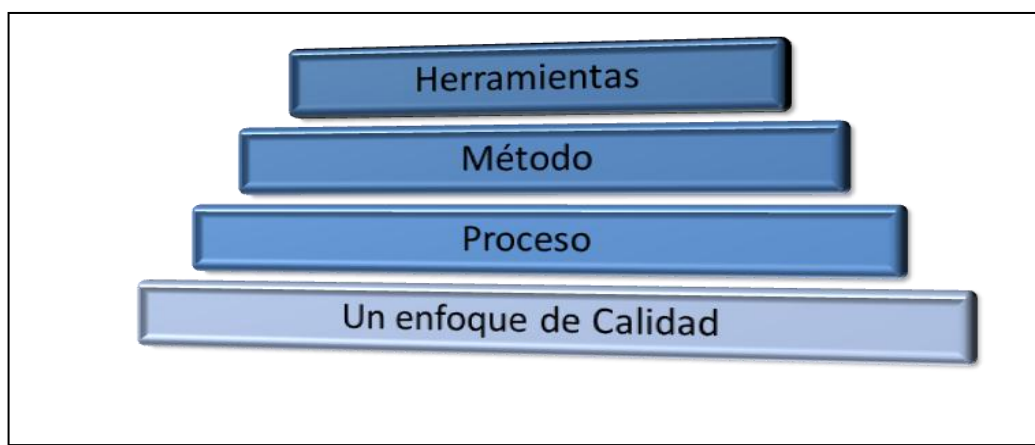


Figura 1.1 Estratos de la Ingeniería de Software

Capítulo 2

El proceso de desarrollo de software puede resumirse en 3 fases genéricas con diversas actividades cada una:

1. Definición: requisitos, análisis y planeación (¿qué?).
2. Desarrollo: diseño, codificación y pruebas (¿cómo?).
3. Mantenimiento: correcciones, adaptación y mejoras (cambio).

Además de abarcar las fases del ciclo de vida del desarrollo de cualquier sistema, la ingeniería de software considera aspectos de la gestión de proyectos como el manejo de personal, costos, etc.

Es recomendable en la creación de software incorporar una estrategia de desarrollo que defina el marco de actividades que acompañen a las fases genéricas, al proceso, los métodos y las herramientas, esta estrategia recibe el nombre de paradigma del software o modelo de proceso.

De acuerdo a las necesidades y naturaleza del proyecto se determinó emplear el modelo en espiral debido a su naturaleza de tipo evolutivo que toma en cuenta el cambio del software con el tiempo y los nuevos requerimientos del usuario, adaptándose y aplicándose a lo largo del ciclo de vida de la aplicación lo que resulta en versiones cada vez más completas.

El uso del modelo aporta además un proceso bien definido con mayor planeación, control y comunicación con el cliente.

El modelo espiral

El modelo en espiral propuesto por B. Boehm en 1988 combina la naturaleza iterativa de la construcción de prototipos (en donde tan pronto se completa una iteración o esfuerzo de desarrollo se procede a la siguiente iteración) con las características sistemáticas del modelo lineal secuencial pero añadiendo el análisis de riesgos.

El software se desarrolla en una serie de versiones incrementales a través de iteraciones, en cada una se atienden 4 aspectos:

- Determinar el objetivo a lograr.
- Plantear las alternativas.
- Analizar los riesgos y resultados esperados para cada una.
- Elegir la mejor alternativa evaluando sus alcances.

En la figura 2.2 se muestra la estructura de un modelo espiral típico que se compone del eje del punto de entrada en cuyo centro parte la espiral que gira en el sentido de las manecillas del reloj donde el primer circuito puede producir la especificación del producto que es usada en giros posteriores para el desarrollo del prototipo y progresivamente mejores versiones.

El modelo se divide en un conjunto de actividades llamadas regiones de tareas que pueden ir de 3 a 6 que se adaptan a las características del proyecto y son:

1. Comunicación entre el equipo de desarrollo y el cliente.
2. Planificación de recursos y tiempos.
3. Análisis de riesgos técnicos y de gestión.
4. Ingeniería o construcción de modelos de la aplicación.
5. Construcción, pruebas, instalación, entrega de documentación y soporte al usuario.
6. Evaluación y reacciones del cliente ante cada versión entregada resultado de la ingeniería y las pruebas.



Figura 2.2 Modelo espiral típico

Capítulo 2

Cada ciclo de la espiral comienza con la recolección de requerimientos y la identificación de objetivos, luego se realiza un análisis de riesgo evaluando las alternativas respecto a los objetivos y restricciones; posteriormente se determina si es conveniente seguir a pesar de los riesgos o se plantean estrategias para resolverlos; se obtiene un prototipo que es presentado al cliente y una vez resueltos los riesgos se sigue el ciclo en cascada.

El giro concluye con la revisión del ciclo anterior y la planeación del siguiente hasta que la evaluación del cliente sea positiva al cumplirse todos los requerimientos.

Ventajas:

- Enfoque más realista de ingeniería de software.
- Permite reducir riesgos en cada ciclo.
- No requiere una definición completa de los requisitos para comenzar a funcionar.
- Incorpora planeación, objetivos de calidad y gestión de riesgos.
- Permite vuelta atrás al identificar problemas.

Desventajas:

- Requiere de experiencia en la identificación de riesgos.
- Necesita participación activa del cliente.
- Si un riesgo importante no se descubre y atiende causará problemas.
- Es difícil convencer al cliente que es un modelo estable.

2.2 Beneficios de la automatización

Una vez desarrollado, probado e implementado el sistema propuesto se logrará un impacto benéfico en profesores de teoría y laboratorios del tipo L+ del Departamento de Ingeniería en Computación, alumnos que cursen estas asignaturas como regulares o ASDRI, y en el personal del Departamento responsable del proceso de calificaciones y revalidaciones de laboratorios. Las ventajas estimadas del sistema son:

- Los profesores de teorías y laboratorios tendrán acceso a una interfaz sencilla para asignar o visualizar calificaciones según sea el caso.
- Con el registro de los ASDRI se tendrá un mejor control de su inscripción en los grupos de laboratorio.
- Los datos de calificaciones serán conservados en la base de datos acumulándose con los de semestres anteriores para cumplir con la revalidación.
- Reducción de errores humanos por parte de los profesores y del personal del Departamento.
- El proceso de asignación, registro, visualización y revalidación será automático, confiable y seguro.
- El profesor de laboratorio ya no escribirá los datos de sus alumnos y sus calificaciones en un archivo para después enviarlo por correo reduciendo el factor de error.
- Ya no será necesaria la verificación del archivo de calificaciones de cada profesor para completar datos, estandarizar o redondear calificaciones.
- El sistema mostrará al profesor de laboratorio su información precargada, la lista de alumnos regulares y ASDRI con sus números de cuenta y las calificaciones válidas sólo en espera de seleccionar la que corresponda a cada uno.
- Se eliminan los errores relacionados con la escritura de la dirección de correo electrónico o del servidor encargado de gestionarlo derivados del envío del archivo de calificaciones al ya no ser éste necesario.
- Contemplará servicios de seguridad como la confidencialidad de la información que viaje por la red y la verificación de identidad de los usuarios que ingresen al sistema.
- El sistema será abierto a los profesores de laboratorio para calificar durante un lapso de tiempo definido y comunicado a todos a principio del

Capítulo 2

semestre con el fin de garantizar que el proceso esté completo en las fechas establecidas.

- Los profesores de teoría tendrán a su disposición oportunamente las calificaciones de laboratorio de su grupo para realizar su evaluación final sin demoras como sucede en el método actual.
- Consulta y corrección de información de forma rápida y sencilla bajo reglas de control de acceso.
- Se llevará un registro de las correcciones de calificación gracias a la entrega al Departamento de Computación del formato correspondiente descargable del portal Web.

2.3 Requerimientos

Los requerimientos identifican las necesidades de funcionalidad y comportamiento del producto, restricciones de diseño e interfaces de usuario, además ayudan a comprender el impacto del software y lo que realmente quiere el cliente.

La etapa de análisis permite a través de la comunicación directa con el cliente definir las funciones del software, el flujo de los datos del sistema al usuario o sistemas externos, su interacción y su almacenamiento.

Esta etapa es indispensable en el planteamiento y resolución efectiva de problemas al fijar la base del diseño a partir del entendimiento de lo que el cliente necesita, obteniendo así una descripción de los requisitos y lista de funcionalidades en lenguaje natural, especificaciones o escenarios de usuario que posteriormente podrán ser verificados durante el proceso en reuniones con el cliente.

2.3.1 Sistema

Categorías de usuarios del sistema

- Profesor Teoría (PT). Accede al sistema para visualizar y descargar las listas de los grupos que imparta con las calificaciones de laboratorios incluidas.
- Profesor de Laboratorio (PL).
- Alumno sin derecho a reinscripción (ASDRI).
- Administrador de USECAD (AU).
- Administrador del Departamento de Computación (ADC).

Requerimientos generales

- Crear un portal que permita asignar, revalidar y visualizar calificaciones de laboratorios L+ para alumnos inscritos regularmente y ASDRI (serán registrados por el sistema) del Departamento de Ingeniería en Computación las cuales serán almacenadas para su posterior consulta, descarga y corrección. Ciertas funciones estarán disponibles por periodos de tiempo determinados por el Departamento de Computación.

Capítulo 2

- Únicamente las teorías con laboratorios L+ coordinados por el Departamento de Computación serán contemplados en el funcionamiento del sistema y son:

Plan de Estudios 2005

- ✓ Computación para Ingenieros.
- ✓ Programación Avanzada y Métodos Numéricos.
- ✓ Administración de Redes.
- ✓ Computación Gráfica.
- ✓ Diseño de Sistemas Digitales.
- ✓ Dispositivos de Almacenamiento Entrada / Salida.
- ✓ Redes de Datos.

Plan de Estudios 1994

- ✓ Microcomputadoras.
- ✓ Memorias y Periféricos.

- El acceso al sistema por parte de los usuarios PT, PL y ASDRI se efectuará desde cualquier navegador Web sin importar el sistema operativo del equipo de cómputo mientras cumpla con las condiciones mínimas indicadas más adelante, ya sea desde su casa, un local de renta de computadoras o salas de cómputo de la Facultad.
- Las interfaces para asignación de calificaciones deben ser semejantes a las mostradas por el sistema existente para asignaturas teóricas con la finalidad de hacerlas más amigables y aceptadas por los usuarios PL.
- El diseño de las interfaces debe estar orientado a que los usuarios digiten la menor información posible para reducir errores.
- Dada la naturaleza del proyecto no se cuenta con presupuesto para la compra de equipo o el pago de licencias de software, por lo tanto el desarrollo e implementación deberá cubrirse con aplicaciones de distribución libre.
- La administración del sistema estará a cargo de personal de USECAD y del Departamento de Ingeniería en Computación.
- El proyecto deberá operar sin problema alguno en el servidor de USECAD garantizando la compatibilidad con su arquitectura, sistema operativo y sistema manejador de bases de datos.

Requerimientos funcionales por módulos

Registro de ASDRI.

Módulo dedicado al alta de ASDRI en las listas de laboratorios a partir de la información contenida en el formato descargable y entregado al Departamento de Computación por el ASDRI. Los datos son:

- Nombre del Alumno.
- Número de Cuenta.
- Clave del Laboratorio.
- Nombre del Laboratorio.
- Grupo del Laboratorio.
- Nombre del Profesor de Laboratorio.
- Firma del Profesor de Laboratorio.
- Grupo de Teoría.

Acceso al sistema.

Módulo destinado a validar el usuario y contraseña de los usuarios PT, PL AU y ADC introducidos en la interfaz Web.

Alta de usuarios.

Módulo en el cual los administradores podrán crear usuarios PT y PL ante la incorporación de nuevos profesores cada semestre para impartir alguna de las asignaturas abarcadas por el sistema.

Descarga de formatos.

Módulo destinado a proporcionar los formatos en el portal sin necesidad de iniciar sesión en el sistema y podrán ser descargados para la corrección de calificaciones de laboratorio o el registro de ASDRI.

Consulta de listas de PL.

Módulo que muestra en una interfaz Web por grupo la lista de alumnos inscritos y ASDRI registrados.

Capítulo 2

Consulta de listas de PT.

Módulo que muestra en una interfaz Web por grupo la lista de alumnos inscritos y ASDRI con su calificación de laboratorio.

Corrección de calificación.

Módulo dedicado a la modificación de calificaciones de laboratorio posterior al proceso de asignación llevando una bitácora de cada petición realizada mediante el formato descargable del portal y entregado al Departamento de Computación por el usuario PL con los siguientes datos:

- Nombre del alumno.
- Número de cuenta.
- Clave del laboratorio.
- Nombre del laboratorio.
- Grupo de laboratorio.
- Nombre del profesor de laboratorio.
- Firma del profesor de laboratorio.
- Exposición de motivos del profesor de laboratorio para la corrección.
- Calificación asignada erróneamente.
- Calificación correcta.

Calificación.

Módulo encargado de mostrar en una interfaz Web por grupo de laboratorio la lista de inscritos con ASDRI y los campos con las calificaciones válidas frente a cada alumno para seleccionarla según corresponda sin necesidad de que el usuario PL la ingrese por teclado. Al concluir la información será almacenada en la Base de Datos.

Distribución grupal.

Módulo de búsqueda de calificación de alumno asignada en el presente semestre u obtenida en semestres previos para ser revalidada, en caso de existir más de una se elegirá bajo el siguiente criterio:

Se asignará la calificación aprobatoria mayor sin importar el semestre de procedencia, de no existir tomar la más reciente sea o no aprobatoria.

El proceso se repite sucesivamente para todos los alumnos de un grupo de teoría y así para todos los grupos y asignaturas reconocidas por el sistema.

El módulo puede ejecutarse automáticamente y por petición del usuario ante la corrección de calificaciones.

Descarga de listas.

Módulo para crear y proporcionar la descarga de archivos de hoja de cálculo con los datos del profesor y la asignatura, pero además la lista de teoría por grupo con calificaciones de laboratorio, o la lista de laboratorio por grupo completa con inscritos y ASDRI.

Interfaz usuario PT.

Módulo encargado de generar la interfaz Web propia del usuario PT mostrando sus datos, así como las opciones de consulta y descarga de listas por grupo de teoría que incluyen las calificaciones de laboratorio.

Interfaz usuario PL.

Módulo encargado de generar la interfaz Web propia del usuario PL mostrando sus datos, así como las opciones de consulta y descarga de listas por grupo posterior al registro de ASDRI, pero además la opción de calificar listando los grupos impartidos en el semestre e indicando gráficamente el estado pendiente o exitoso del proceso para cada uno.

Interfaz usuario ADC.

Módulo enfocado al usuario de tipo administrador para el cual debe generarse una interfaz Web diferente al resto de los usuarios, debe incluir mayores parámetros de consulta y opciones que sólo éste tipo de usuario puede realizar como el registro de ASDRI, correcciones de calificación, creación de archivos con calificaciones finales, respaldos, alta de usuarios, etc.

Consulta de usuario ADC.

Módulo de consultas de calificación de laboratorio por parte del administrador bajo los siguientes criterios de búsqueda:

- *Por grupo de teoría.* Mostrando todos los alumnos, su calificación y el grupo de laboratorio del que fue tomada.

Capítulo 2

- *Por alumno.* Ya sea por su nombre o número de cuenta y filtrando además por semestre y asignatura.
- *Por grupo de laboratorio.* Mostrará la lista disponible a visualizar y descargar por el profesor de laboratorio.
- *Por semestre.* Mostrando todos los resultados o filtrados por asignatura.

Respaldo periódico de información.

Con el objetivo de evitar la pérdida de información por fallas del sistema u ocurridas en el servidor relacionadas con el sistema operativo, con el sistema manejador de bases de datos o con el disco duro, se requiere de un módulo que realice respaldos de información vital como el acumulado de calificaciones en otras tablas dentro del gestor o en medios externos a petición del usuario y automáticamente por periodos de tiempo establecidos.

Respaldo del sistema.

Módulo responsable de la creación del respaldo del propio sistema en caso de sufrir alteraciones y sea necesario restablecer versiones previas más estables o ante mejoras posteriores.

2.3.2 Hardware

Servidor SunFire v880

El proyecto está destinado a operar en un ambiente Web donde los usuarios accedan al sistema a través de cualquier equipo conectado a Internet, por tanto el sistema debe estar alojado en un servidor de alto desempeño, seguro y confiable que integre además un gestor de bases de datos.

Tal servidor no será adquirido sin embargo la USECAD ha puesto a disposición uno de sus servidores que cumple con todas las condiciones y que además contiene la información necesaria para el completo funcionamiento del sistema.

El servidor en cuestión es un Sun Fire modelo v880 de alto desempeño, memoria compartida, escalable, con sistema de multiprocesamiento simétrico y soporte de hasta 8 procesadores Sun UltraSparc III, haciéndolo ideal para estaciones de trabajo y hospedaje de sitios Web.

Su poder de procesamiento es proporcionado por un máximo de 4 tarjetas CPU/Memoria, cada tarjeta incorpora:

- 2 procesadores UltraSparc III.
- 8 Megabytes de memoria caché por cada procesador.
- 16 ranuras para módulos de memoria con capacidad máxima por cada una de 1GB.
- A lo anterior se añaden otras características de almacenamiento y conectividad:
- Máximo de 12 discos duros de 18, 36, 72 o 146 GB.
- Canal de fibra óptica y SCSI para almacenamiento.
- Protocolos de red Ethernet, Fast Ethernet y Gigabit Ethernet.
- Unidad óptica DVD-ROM.
- 9 ranuras de expansión PCI, 7 x 33 MHz y 2 x 33 o 66 MHz.
- Interfaces USB y serial.
- RSC o Remote System Control, es una tarjeta de administración segura de servidor sobre líneas de modem o red para monitorear sus condiciones de ambiente, temperatura y alimentación, controlar funciones mediante consola del sistema como diagnósticos, encendido y reinicio, emitir notificaciones de problemas.

El procesador UltraSparc III implementa el SPARC V9 Instruction Set Architecture (ISA) y el Visual Instruction Set (VIS) para acelerar los procesos multimedia, de redes, encriptación y procesamiento de Java. Desarrollado por Sun Microsystems y fabricado por Texas Instruments basado en el SPARC, ésta fue la primera arquitectura RISC abierta con especificaciones de diseño disponibles y presentaba un número reducido de instrucciones. Algunas otras características del UltraSparc III son:

- Microprocesador super-escalar.
- Rango de operación de 750 a 1200 MHz.
- Controlador de memoria integrado y bus de multiproceso dedicado para lograr el rendimiento de multiprocesamiento de memoria compartida.
- Interfaz externa con un bus de datos de 128 bits y bus de direcciones de 43 bits operando a 150 MHz.
- unidades aritmético lógicas (ALUs), 2 unidades de punto flotante y 1 unidad de carga y almacenamiento (LOAD/STORE).
- Transferencia de hasta 9 GB de información por segundo.

Capítulo 2

- Procesamiento de 64 bits.
- Detección y corrección de errores en cachés.
- 29 millones de transistores.

Equipo cliente

Los usuarios requieren una PC con acceso a Internet cuyas especificaciones satisfagan la carga de procesos de navegadores Web para acceder al sitio pero se recomiendan los siguientes requisitos para no afectar el desempeño:

- *Disco duro:* no hay requisitos debido a que el sistema no se instala en los equipos cliente.
- *Memoria RAM:* 256 MB (mínimo) y 512 (recomendado) para Microsoft Windows XP, 1 GB (mínimo) y 2GB (recomendado) para Microsoft Windows Vista y 7.
- *Procesador:* 750 MHz (mínimo) equivalente a Pentium III en adelante.

Equipo de desarrollo

La instalación de herramientas y entornos de desarrollo, de sistemas manejadores de bases de datos, así como el software que emule el comportamiento de un servidor Web demandan los siguientes recursos de cómputo:

- *Disco duro:* 80 GB (mínimo) para las instalaciones y bases de datos de prueba.
- *Memoria RAM:* 1 GB (mínimo) o 2 GB (recomendado).
- *Procesador:* Pentium 4 (mínimo) o doble núcleo (recomendado).
- *Red:* Tarjeta de red 10/100 Mbps.

2.3.3 Software

El desarrollo del sistema para la gestión de calificaciones de laboratorios y su revalidación tiene una naturaleza Web, que involucra lenguajes de programación, creación de interfaces de usuario, así como el almacenamiento y consulta de información, por tanto son indispensables herramientas de software que cubran los aspectos del sistema en todas sus etapas, desde entornos de desarrollo hasta servidores Web y manejadores de bases de datos.

Existen diversas herramientas de tipo privativo o libre especializadas en cada área descritas a continuación.

Java

Java no es sólo un lenguaje de programación, constituye también las herramientas para desarrollarlo y ejecutarlo, es decir, incluye el entorno de programación (API y compilador) y el entorno de ejecución (JRE).

El entorno de desarrollo necesario es el JDK de Sun o Java Development Kit y en la versión 2 toma el nombre de J2SDK.

La plataforma Java 2 se divide en 3 categorías dependiendo el mercado al cual está enfocado:

Java 2 Standard Edition (J2SE)

Orientado a equipos de escritorio para el desarrollo de aplicaciones y applets que poseen generalmente una interfaz de usuario.

Java 2 Enterprise Edition (J2EE)

Toma al J2SE y añade clases para el desarrollo de aplicaciones empresariales orientadas a servidores basadas en Enterprise JavaBeans, Servlets, Java Server Pages y Extensible Markup Language (XML).

Java 2 Micro Edition (J2ME)

Enfocado a dispositivos electrónicos con pocos recursos como teléfonos celulares, PDA's, sistemas GPS, etc., que poseen poca memoria y baja capacidad de procesamiento, así como limitadas resoluciones gráficas de pantalla.

IDE (Integrated Development Environment)

El IDE o Entorno de Desarrollo Integrado es una aplicación dedicada a la creación de proyectos de software bajo uno o más lenguajes de programación y consta de una serie de herramientas como un editor de texto de código fuente, un compilador, un depurador y un constructor de interfaces gráficas de usuario GUI brindando al desarrollador un marco de trabajo completo y más amigable.

Capítulo 2

Eclipse

Es un IDE de código abierto basado en plugins creado originalmente por IBM para el desarrollo de aplicaciones Java, es ideal para la gestión de proyectos grandes con un buen número de clases y cuenta con una versátil opción de depuración de programas.

Está disponible en la página de Fundación Eclipse en diferentes versiones compiladas que agrupan herramientas útiles para situaciones o lenguajes de programación específicos.

Su instalación es bastante sencilla, la versión seleccionada se descarga en un archivo comprimido con extensión zip, éste se descomprime en la carpeta deseada y sólo es necesario invocar el archivo ejecutable eclipse.exe para iniciar la aplicación, cabe señalar que es indispensable tener instalado previamente el J2SDK.

Algunas otras características de Eclipse son:

- Multiplataforma (Linux, Windows, MacOS).
- Soporte de Arquitecturas x86 y 64.
- Estructura de plugins para agregar nuevas características o compatibilidad con otros lenguajes de programación como C, C++ y PHP.
- Control de versiones con CVS o con subversion.
- Resaltado de sintaxis y autocompletado.
- Compilación en tiempo real.
- Asistentes o Wizards para la creación de proyectos, clases y platillas de código.

NetBeans

Es un IDE de carácter libre para escribir, compilar, depurar y ejecutar programas en Java. Tiene una base modular que le permite extenderse para cubrir otro tipo de aplicaciones o desarrollar herramientas propias.

Fue fundado por Sun Microsystems y en la actualidad cuenta con una gran comunidad de usuarios que aportan módulos que interactúan con las APIs de NetBeans y pueden ser incorporados para extender el IDE.

Su descarga inicial ya contiene los módulos con las funciones orientadas al soporte de Java y para el sistema de control de versiones. Presenta algunas otras características importantes:

- Multiplataforma (Windows, MacOS, Linux, Solaris).
- Amplia fuente de módulos de terceros.
- Soporte para lenguajes C, C++, PHP, JavaScript, Ajax.
- Gratuito y sin restricciones de uso.
- Herramientas de esquemas XML.
- Soporte a persistencia.
- Desarrollo de aplicaciones empresariales.

Dreamweaver

Aunque no es considerado un IDE como tal, es una aplicación útil para la construcción bajo estándares de páginas, sitios y aplicaciones Web, fue creado por Macromedia y está cargo actualmente de Adobe Systems.

Además del editor de código añade un entorno gráfico más amigable a los diseñadores para crear visualmente páginas Web estáticas y dinámicas mediante la inserción de elementos dispuestos en paneles de herramientas, tal peculiaridad ubica también la aplicación en el ámbito de los editores “lo que ves es lo que obtienes” (WYSIWYG).

Algunas otras características importantes son:

- Integración con Adobe Flash y Adobe Fireworks.
- Soporte de tecnologías CSS y JavaScript.
- Disponible para las plataformas MacOS y Windows.
- Conexión a bases de datos MySQL y Microsoft Access.
- Contenido dinámico con ASP, ASP.NET, ColdFusion, JSP y PHP.
- Interfaz personalizable.
- Cliente de FTP integrado.
- Comparación de contenido en diferentes navegadores.
- Integración de contenido interactivo y videos de forma sencilla como FLV.

Capítulo 2

Servidores Web

Un servidor Web hace referencia al programa y al equipo donde se ejecuta encargado de transferir páginas Web (generalmente en código HTML) con textos, vínculos y contenido incrustado como imágenes, animaciones, audio y video.

El servidor se mantiene a la espera de peticiones a los sitios que proporciona, el usuario solicita la página indicando su dirección en la red o URL (Uniform Resource Locator) en un navegador Web entablando comunicación por el protocolo HTTP, si el recurso se encuentra disponible el servidor responde con el código HTML de la página que es recibido e interpretado por el navegador y la muestra finalmente en pantalla.

Sin embargo, un servidor Web no se limita a la distribución de páginas HTML, dado que un navegador puede invocar una aplicación integrada por fragmentos de código que se ejecuta a partir de la petición o como parte de la respuesta tanto del lado del cliente o del lado del servidor.

En tal situación el servidor recibe el nombre de contenedor Web o contenedor Servlet/JSP, que puede operar por sí mismo respondiendo con recursos estáticos, dinámicos y aplicaciones, o bien mediante un conector trabajar en conjunto con un servidor HTTP encargado de responder o re-direccionar al contenedor según sea el caso.

Servidor HTTP Apache

Servidor Web de código abierto desarrollado bajo la Apache Software Foundation que ha sido empleado por gran parte de los sitios Web del mundo debido a su facilidad de instalación y a la posibilidad de convertir cualquier PC de bajos recursos en servidor corriendo bajo sistemas Unix o Windows.

Es un software de arquitectura modular organizado en módulos base que contienen las funciones elementales y los de multiproceso responsables de los puertos y las peticiones.

Si se desean añadir utilidades basta con agregar los módulos necesarios sin volver a instalar el software que en la mayoría de los casos corre sin realizarle cambios, sin embargo es altamente configurable a pesar de la carencia de una interfaz gráfica apropiada para ello. Otras características son la capacidad de conectarse directamente con una base de datos y la inclusión de módulos

compatibles con los lenguajes PHP, Perl y Python para el desarrollo de aplicaciones o funciones avanzadas.

Su versión 2 ofrece mejoras respecto a versiones previas:

- Un nuevo conjunto de interfaces de programación de aplicaciones (API's).
- Los módulos pueden actuar como filtros de contenido.
- Soporte para el formato de direcciones IPv6.
- Las páginas de respuesta a errores ahora se presentan en diversos idiomas.
- Es más rápido y estable en sistemas que no son tipo Unix debido a la incorporación de los módulos de multiprocesamiento y API nativa para cada plataforma.
- Posibilidad de servir en distintos protocolos.

Existen paquetes para el desarrollo de aplicaciones como WampServer y EasyPHP de gran ayuda para la fase de pruebas antes de su implementación final que integran Apache como servidor Web junto con el manejador de bases de datos MySQL y el lenguaje de programación PHP en una misma interfaz, son de fácil instalación y configuración para sistemas Windows.

Apache Tomcat

También llamado Jakarta Tomcat es un servidor Web de código abierto diseñado para trabajar con Java, desarrollado originalmente por Sun como parte del Java Server Web Development Kit y a cargo actualmente de la Apache Software Foundation.

Es un contenedor de Servlets que implementa las especificaciones de Sun Microsystems, no es más que un Shell de ejecución que maneja e invoca Servlets por cuenta del usuario o los convierte resultado de la compilación de JSP.

Puede funcionar como servidor Web independiente, sin embargo no es tan rápido para servir páginas estáticas como Apache, no es tan configurable ni maneja otros lenguajes además de Java, por lo cual su uso autónomo se recomienda para entornos de desarrollo o para sitios bajo condiciones mínimas de transacciones y velocidad.

Dado lo anterior se recomienda emplear en forma de plugin para otros servidores Web ya sea Apache o IIS. De esta forma se complementa la

Capítulo 2

funcionalidad del servidor Web con las ventajas que implica el contenedor como threading, manejo de sesiones, conectividad, etc.

Ambos comportamientos de Tomcat se observan en la figura 2.3:

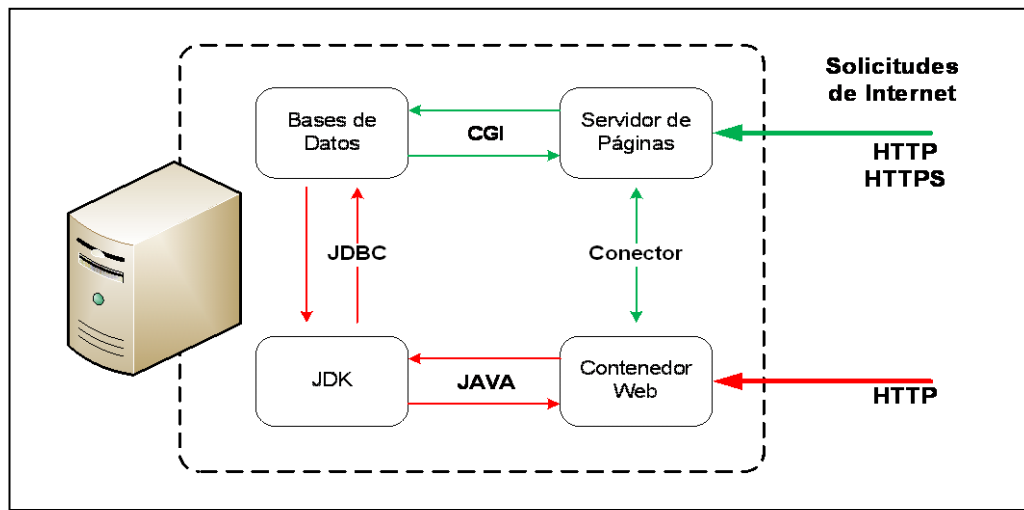


Figura 2.3 Funcionamiento Servidor Tomcat

Sistema Manejador de Bases de Datos (SMBD)

El SMBD es el software que sirve de interfaz entre la base de datos física, el usuario y las aplicaciones que la utilizan. Permite crear, definir y estructurar una base de datos, guardando la información en algún medio controlado por el SMBD para después consultarla o modificarla.

Existen productos libres o con licencia propietario disponibles que cubren su función primordial pero además incorporan mecanismos de seguridad, respaldo y recuperación, manejo de transacciones y tiempo de respuesta variable.

Sybase ASE

Sybase es una compañía que crea productos y servicios dedicados a la gestión de información, herramientas de desarrollo y almacenamiento de datos.

En el ramo de los SMBS relacionales cuenta con Adaptive Server Enterprise (ASE), reconocido como un sistema altamente escalable de alto rendimiento y

bajo costo, con soporte a grandes volúmenes de datos, transacciones y usuarios. Está disponible en las siguientes ediciones:

- ASE Enterprise Edition
- ASE Small Business Edition
- ASE Developer's Edition
- ASE Express Edition para Linux
- ASE Cluster Edition

En su versión empresarial 15 se distinguen opciones de seguridad únicas y funciones que incrementan el rendimiento reduciendo los costos y riesgos de operación mientras responde a las exigencias de tamaño de varios terabytes y millones de transacciones por minuto en diversas plataformas. Como parte de estas funciones están:

- In-Memory. Es una opción enfocada a la virtualización de los datos y el escalamiento crítico de grandes volúmenes de datos y usuarios concurrentes para ambientes que demandan tiempos de respuesta inmediatos.
- Servicios de recuperación más rápidos con menor uso de recursos de red y almacenamiento.
- Posibilidad de 4 tipos de particionamiento de datos ubicados en varios dispositivos físicos, si uno de éstos falla las particiones restantes pueden usarse.
- Sistema de encriptación a través de permisos de usuarios.
- Conexiones seguras SSL.
- Soporte a herramientas de desarrollo y lenguajes como Power Builder, Visual Basic, Java, PHP, etc.
- Auto-administración, manejo automático de recursos y bases de datos transportables.
- Soporte a protocolos ODBC, ADO.NET, JDBC y OpenClient de Sybase.
- Mejoras en servicios de Java y XML.
- Tecnología de consultas para transacciones más inteligentes.
- Cómputo distribuido con tecnologías grid y cluster.
- Arquitectura orientada a servicios Web.
- Compatibilidad inalámbrica.

2.4 Criterios de elección de software

Las herramientas de software fueron elegidas considerando en primer lugar los requisitos restrictivos del servidor donde será colocado el proyecto y el sistema manejador de bases de datos. El resto de las herramientas se seleccionaron de acuerdo a la compatibilidad entre ellas, desempeño y seguridad.

A continuación se muestran en la tabla 2.1 las herramientas y software a utilizar, así como la justificación de su elección.

SOFTWARE	JUSTIFICACIÓN
Java	<ol style="list-style-type: none">1. Permite a través del uso de sesiones un manejo seguro de los usuarios.2. Mediante los Servlets y jsp se obtienen datos y generan las páginas dinámicas para responder a las peticiones con información procedente de una base de datos.3. La conexión a la BD y el intercambio de datos es posible con el driver adecuado que también está hecho con Java.4. El API del lenguaje Java brinda las herramientas necesarias para desarrollar el proyecto en la mayoría de sus etapas sin recurrir a otros lenguajes para una función específica.
Eclipse	<ol style="list-style-type: none">1. Es una aplicación ligera que no se instala ideal para equipos de desarrollo con bajos recursos.2. Elegida la versión requerida y por su estructura de plugins instalables se tiene una aplicación con sólo lo necesario.3. Puede asociarse al contenedor Apache Tomcat indispensable para el proyecto integrando un marco completo de desarrollo y pruebas para aplicaciones Web basadas en Servlets y jsp dentro del mismo IDE antes de ser llevado al equipo final.4. Su editor de código soporta el HTML de las páginas que servirán de interfaz a los diferentes tipos de usuario eliminando la necesidad de Macromedia Dreamweaver para su diseño que si requiere pago de licencia.

<p>Apache Tomcat</p>	<ol style="list-style-type: none"> 1. La generación de contenido dinámico y obtención de información de una base de datos puede realizarse con scripts incrustados en la página Web, lo cual puede considerarse riesgoso al mostrar datos privados del servidor o la base. Al emplear un contenedor de Servlets/JSP se pueden desarrollar aplicaciones y almacenarlas ahí en espera de ser solicitadas por la página Web, éstas se ejecutan del lado del servidor y pueden realizar las mismas funciones de forma más segura. 2. Al recibir la petición de un Servlet o jsp, la JVM responde creando la primera vez sólo una instancia y a partir de ésta se crean hilos para peticiones posteriores reduciendo el consumo de recursos en comparación con la tecnología CGI que responde a cada solicitud con un proceso.
<p>Sybase ASE</p>	<ol style="list-style-type: none"> 1. El sistema y la base de datos estarán alojados en un servidor de USECAD el cual ya posee la licencia de éste sistema manejador de base de datos. 2. Las tablas requeridas con alumnos inscritos en teorías con laboratorio (+L) y de los propios laboratorios se encuentran bajo éste SMBD por lo cual será más sencilla su manipulación.

Tabla 1.1 Justificación de Software

2.5 Análisis de factibilidad

Teniendo presente la problemática existente y la justificación para el desarrollo de un proyecto de software integral, es indispensable realizar un estudio de factibilidad, el cual además de evitar una conclusión desastrosa y fallida del proyecto, determina en varios rubros si es posible su diseño y desarrollo evaluando aspectos operacionales, técnicos, económicos, los costos y beneficios, legales, entre otros.

Si la evaluación resulta positiva en cada uno se dice que el proyecto es viable permitiendo una mejor toma de decisiones bajo estudios reales de posibilidades.

Factibilidad operacional

Examina la posibilidad del cumplimiento del sistema en cuanto a sus funciones y objetivos al ejecutarse tal y como fue pensado, además de la aceptación por parte de todos los usuarios.

El proyecto cuenta con puntos a favor en este sentido, debido a que ciertas funciones serán semejantes a las integradas en sistemas conocidos por los usuarios desde hace algún tiempo, con los que ya están familiarizados como el caso de la asignación de calificaciones vía Web y la descarga de listas de alumnos inscritos, por lo cual se estima no habrá demasiados conflictos al migrar del método actual al nuevo sistema que ofrecerá una interfaz amigable y utilidades para reducir la dificultad y tiempo de los procesos.

Sin embargo, algunas funciones del sistema son nuevas y tal vez sean cuestionadas en un inicio al involucrar trámites no hechos hasta ahora como el llenado y entrega de formatos de corrección de calificación y registro de ASDRI, pero necesarios para una correcta administración en el Departamento de Computación al llevar una bitácora formal para estas acciones.

Lo que respecta a la administración y ejecución de funciones del sistema por el cliente (Personal del Departamento de Computación), éste deberá aprenderlo dejando atrás su ya conocido método pero se estima lograr su aceptación de forma rápida gracias a la comunicación continua y su inclusión durante todas las etapas del proyecto.

Con el fin de ampliar la aceptación de los usuarios y el cliente se pretende preparar materiales de apoyo a ser divulgados con suficiente tiempo para evitar confusiones en momentos críticos.

Considerando que el desarrollo e implementación de hace por petición del cliente bajo sus requisitos de funcionalidad, no existirán conflictos por su uso y por el contrario aumentará su productividad reflejada en menores tiempos y fallas. Sin embargo perderá cierto grado de control de información, pues anteriormente era almacenada en el equipo de cómputo personal del cliente y con la puesta en marcha del sistema ahora estará en el servidor de USECAD.

Finalmente, cabe destacar que todas las funciones solicitadas se consideran posibles de desarrollar sin implicar la creación de nuevas y riesgosas tecnologías.

Factibilidad técnica

Consistió en un análisis de la tecnología existente en el Departamento de Computación donde labora el cliente evaluando si era suficiente para implementar y soportar el sistema, arrojando datos negativos ante la ausencia del equipo indispensable para ello.

Tomando en cuenta los requerimientos técnicos se solicitó la cooperación de USECAD para proporcionar información vital para el funcionamiento del sistema, así como los recursos de hardware y software.

Dicha entidad cuenta con la experiencia, la tecnología y el personal calificados para el desarrollo y mantenimiento de aplicaciones de tipo Web con la capacidad suficiente de procesamiento, almacenamiento de información y atención a peticiones de usuarios.

El hecho de correr sobre el servidor de USECAD presenta 2 situaciones de ámbito técnico: la primera es la compatibilidad del sistema con el hardware y software del servidor; la segunda radica en una reducción significativa en el tamaño del proyecto dada la ubicación del sistema en el mismo sitio que la información requerida, evitando así el desarrollo de un módulo de interconexión entre sistemas incrementando la complejidad de programación y operación.

En cuanto al equipo de desarrollo, éste si se ubica en el Departamento de Computación y cumple con las especificaciones mínimas para la instalación de software.

Factibilidad económica

El análisis económico brinda un panorama de los recursos, costos y beneficios asociados al proyecto, la adquisición de equipo y la operación del sistema.

Aunque el proyecto carece de presupuesto para la compra de equipo, licencias de software o la contratación de personal para el desarrollo y operación, todos los aspectos se prevé sean cubiertos.

En primer lugar, el equipo servidor necesario será proporcionado por la USECAD, entidad responsable de otros servicios de cómputo, almacenamiento, publicación Web y sistemas institucionales, con lo cual se logra un beneficio mutuo al evitar la compra de equipo y la justificación de mayores recursos para la USECAD para el mantenimiento de sistemas.

Además, el equipo de desarrollo tampoco será adquirido al encontrarse uno disponible en el Departamento de Computación dedicado únicamente para el proyecto.

En lo que respecta al software, los de tipo propietario a emplear están instalados en el servidor y ya no es necesario pagar por su uso, los requeridos para el desarrollo y pruebas son de carácter libre evitando así la compra de software operativo, gestor de bases de datos, entornos de desarrollo y diseño, herramientas de administración de proyectos, entre otros.

Por otro lado tampoco es preciso el pago del personal por el desarrollo en vista de que es un proyecto de tesis ni la contratación adicional para su operación, la cual recae dentro de las actividades laborales del cliente.

El proyecto es viable financieramente al encontrarse los medios para cubrir todos los costos y ofrecer beneficios tanto para el cliente como para los usuarios, reduciendo la carga de trabajo en los equipos de escritorio del Departamento de Computación, las horas/hombre que eran destinadas a esta actividad, el riesgo de pérdida de información vital, así como los costos de operación y papelería.

Factibilidad legal

Análisis para determinar alguna infracción o responsabilidad legal en la que se incurra durante el proceso del proyecto, y a pesar de ser una tesis no se exime de cumplir reglamentos para ser aprobada e implementada.

Bajo el ámbito de la normatividad de la UNAM, en específico del Reglamento General de Exámenes de la Legislación Universitaria, se dicta en su Artículo 19 que la titulación en nivel licenciatura se expedirá al acreditar en su totalidad el plan de estudios, realizado el servicio social y cumplido con alguna de las opciones de titulación propuestas en el Artículo 20, de las cuales destaca en su Apartado “A” opción “a” la referida a la tesis y su evaluación.

Elegida esta forma de titulación y determinado el tema, objetivos, metas y alcances se registra en la Coordinación de Seminarios y Servicio Social de la Facultad de Ingeniería, donde se valora si cumple con el nivel académico y no afecta entidad o tema existente para finalmente ser aprobada por un Comité Evaluador de Exámenes Profesionales.

El presente proyecto superó esas etapas y fue aprobado constatando su viabilidad legal dentro del marco de la Legislación Universitaria.

2.6 Análisis de riesgo

El proceso del software no se encuentra aislado, también se ve afectado por su entorno y situaciones externas e inciertas llamadas riesgos que pueden provocar que el proyecto resulte mal como cambios en los requerimientos, tecnologías o personal.

Un riesgo es un problema potencial cuyo acontecimiento es impredecible, sin embargo debe ser identificado, clasificado, evaluada la probabilidad de que ocurra, estimar su impacto por nivel de daño, y planearse una estrategia con medidas enfocadas a evitarlo, contrarrestarlo cuando se presente y darle solución respondiendo de forma controlada y efectiva.

Se distinguen 3 categorías de acuerdo al grado de incertidumbre de ocurrencia y del grado de pérdidas:

✚ *Riesgos del proyecto.*

Afectan el presupuesto, la planeación y el personal, pueden ser derivados de problemas con la complejidad, tamaño del proyecto, los recursos, el cliente y los requisitos.

✚ *Riesgos técnicos.*

Ocurren ante dificultades en el diseño, implementación y mantenimiento debido a que el problema a resolver resultó más complejo de lo esperado en un principio o por factores de tecnología.

✚ *Riesgos de negocios.*

Afectan la viabilidad del software y se deben principalmente a los siguientes problemas:

- Riesgo de mercado. Nadie requiere el producto.
- Riesgo estratégico. El producto no corresponde con la estrategia comercial de la empresa.
- Riesgo de ventas. Construir un producto del cual no se tenga idea de cómo vender.
- Riesgos administrativos. Pérdida de apoyo de parte de la dirección o ejecutivos.
- Riesgos de presupuesto. Pérdida de recursos económicos o de personal.

Identificación del riesgo

Además de clasificarlos en categorías, un método recomendable para identificar y gestionar riesgos es realizar una lista de verificación de riesgos asociados en subcategorías genéricas:

✚ *Tamaño del producto (PS).*

Relacionados con el grado de certeza en la estimación del tamaño del software y base de datos, cantidad de módulos, archivos y código.

✚ *Impacto en el negocio (BU).*

Restricciones impuestas por la compañía o el gobierno, viabilidad del producto, costos debidos a retrasos y fallas, efecto en los ingresos.

✚ *Características del cliente (CU).*

Comunicación entre equipo de desarrollo y el cliente, conocimiento sobre cuestiones técnicas y del proceso de software por parte del cliente y su disposición para comentar requisitos y revisiones.

✚ *Definición del proceso (ES).*

Riesgos provocados por falta de conocimiento del proceso de software seleccionado, experiencias de implementación en otros proyectos o mal seguimiento documental de sus etapas. En la parte técnica refiere a la carencia de métodos o herramientas para el análisis, diseño, pruebas y medidas de calidad.

✚ *Entorno de desarrollo (DE).*

Los riesgos son causados por una mala herramienta o la ausencia de ésta para la gestión del proyecto o el desarrollo del software, por su falla en la integración con otras herramientas, una deficiente documentación y formación del equipo en su uso.

✚ *Tecnología a construir (TE).*

Riesgos latentes en la creación de nuevas tecnologías, en el uso de nuevos métodos de análisis y diseño, por la interacción del producto con otros sistemas de software y hardware no probados, y en la incertidumbre de conseguir la funcionalidad deseada.

Capítulo 2

- ✚ *Tamaño y experiencia de la plantilla (ST).*

Afines a la experiencia en proyectos y formación del equipo de trabajo, así como su disponibilidad a lo largo de todo el proceso.

Componentes de riesgo

- Rendimiento. Grado de incertidumbre respecto a si el producto cumplirá con los requisitos y el uso que se pretende darle.
- Costo. Grado de incertidumbre del mantenimiento del presupuesto del proyecto.
- Soporte. Grado de incertidumbre relacionado a la facilidad del software para adaptarse, corregirse y mejorarse.
- Planeación Temporal. Grado de incertidumbre para respetar el calendario del proyecto y fechas de entrega.

Proyección del riesgo

Una vez definidas las categorías y la lista de comprobación de riesgos se procedió a medir la probabilidad de que éstos ocurran, así como el impacto sobre el proyecto y los problemas que ocasionen.

La estimación del riesgo partió de cuatro actividades:

1. Establecer la probabilidad del riesgo.
2. Definir las consecuencias del riesgo.
3. Estimar el impacto del riesgo en el proyecto y en el producto a partir de la naturaleza, porcentaje de afectación y el tiempo que dure el problema.
4. Apuntar la exactitud general de la proyección del riesgo para no dejar lugar a confusiones.

Posteriormente se decidió proyectar todos los riesgos pensados sin importar la etapa donde podrían presentarse a través de una tabla de riesgos que se observa en la tabla 2.2.

RIESGOS	CATEGORÍA	PROBABILIDAD	IMPACTO
Dado que el proyecto es una tesis el personal está limitado.	ST	80	2
Falta de conocimiento en las herramientas, lenguajes de programación, bases de datos y sistemas operativos indispensables para el proyecto.	ST	70	2
Las tareas no técnicas como la redacción de capítulos y documentación necesitan más tiempo.	BU	70	2
La fecha final ha cambiado sin considerar si los recursos disponibles son suficientes ni el avance hasta el momento.	BU	60	1
La instalación en el entorno final puede causar fallas que no ocurrieron en el entorno de pruebas.	TE	60	2
El periodo de familiarización con las herramientas de software y los lenguajes de programación toma más tiempo del estimado.	BU	60	2
El cliente (Depto. Comp.) añade requisitos que no son viables o provocan cambios en el diseño.	PS	60	3
Se necesita regresar a una versión anterior más estable pero ya no está disponible.	ES	60	3
Falta de compromiso reduce el rendimiento.	ST	60	3
Los recursos de hardware ya no son suficientes para sustentar el desarrollo.	TE	50	2
El diseño es poco flexible para adaptarse al cambio de requisitos o la corrección de errores.	ES	40	2
La integración de los módulos es compleja o provoca fallas.	ES	40	2
El requisito de compatibilidad con el SMBD Sybase ASE implica cambios en el diseño.	TE	40	3
Faltaron especificaciones escritas de requisitos, diseño y código.	ES	40	3
No se llevó un seguimiento de las pruebas y errores para después darles solución.	ES	40	3
El entorno de desarrollo de programación elegido no cubre todos los aspectos necesarios.	DE	40	3
Los requisitos de hardware y software restrictivos retrasan el proceso de desarrollo.	BU	40	3
Existen retrasos en la entrega de prototipos y versiones.	BU	40	3
Las tablas de listas proporcionadas por USECAD están en un formato no contemplado provocando rediseño de módulos o de la base de datos.	PS	30	2
LÍNEA DE CORTE			

Capítulo 2

El requisito de compatibilidad con el SO Solaris puede implicar nuevos módulos.	TE	30	3
No se detectaron riesgos críticos en la etapa de análisis y el darles solución demanda más tiempo.	ES	30	3
El usuario final (Profesor de Teoría) está en desacuerdo con la interfaz que presenta las calificaciones.	BU	30	4
La falta de uso de herramientas de planificación y seguimiento de actividades provoca demoras.	ES	30	4
El número de usuarios finales (Profesor de Laboratorio) que asignen calificación de forma simultánea es mayor al esperado.	PS	20	2
Existen conflictos con tablas existentes en el SMBD.	TE	20	2
La presión reduce la productividad.	BU	20	3
La entrega de la información solicitada a USECAD demora más de lo previsto.	BU	20	3
El modelo de desarrollo de software elegido no fue el más adecuado.	ES	20	3
Los trámites de entrega y aprobación tardan más de lo esperado.	BU	20	4
El diseño de la interfaz para asignación de calificaciones no es del agrado del usuario final (Profesor de Laboratorio).	BU	20	4

Tabla 2.2 Proyección de Riesgos

En la primera columna se colocaron los riesgos resultantes de revisar toda la lista de verificación, en la segunda se indicó su clasificación de acuerdo a las categorías, en la tercera su probabilidad estimada, en la cuarta el impacto del riesgo evaluado y promediado para los cuatro componentes de riesgo representado como: 1-catastrófico, 2-crítico, 3-marginal, 4-despreciable.

Completadas las cuatro columnas se ordenaron priorizando las de mayor probabilidad y de ahí en segundo nivel por su impacto. Posteriormente se marcó una línea de corte horizontal separando los riesgos dejando sobre la línea los de alta a moderada probabilidad para ser tratados mediante un plan que los reduzcan, supervisen o administren.