



UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

**“MÉTODO ALTERNATIVO PARA EL DESARROLLO DE
SOFTWARE EDUCATIVO BASADO EN PRÁCTICAS EFECTIVAS”**

TESIS

**PARA OBTENER EL TÍTULO DE
MAESTRO EN ELECTRÓNICA
Y COMPUTACIÓN**

PRESENTA

ING. JOSUÉ NEFTALÍ GARCÍA MATÍAS

**DIRECTOR DE TESIS
DR. IVÁN ANTONIO GARCÍA PACHECO**

HUAJUAPAN DE LEÓN, OAX.; DICIEMBRE DE 2008

**Tesis presentada el 12 de Diciembre de 2008.
Ante los siguientes sinodales:**

**Dra. Virginia Berrón Lara
Dr. Antonio Orantes Molina
Dra. Carla Leninca Pacheco Agüero
Dra. Irma Salinas Pérez**

Director de Tesis:

Dr. Iván Antonio García Pacheco

Dedicatoria

A mi tesoro más pequeño Ximena, gracias por venir a refrendarnos el amor que hay entre tu mamá y yo.

A mi ya no tan pequeña hija Ariadna, primer motivo enviado por Dios para ayudarme a cumplir esta meta.

A mis padres Bon y Arnu, por su constante apoyo, el cual he sentido siempre a mi lado.

Sobre todo a mi esposa Patricia, quien me ha brindado amor, cariño, comprensión, confianza y apoyo, sin importar lo lejos que me encuentre.

Agradezco especialmente a:

Al Dr. Iván Antonio García Pacheco, por darme la oportunidad de trabajar con el, aprender de su experiencia para concluir este trabajo y por su amistad sincera durante estos 30 años, créeme que admiro tu trabajo, espero no haberte defraudado, amigo.

Más vale lento pero seguro.

Josué.

Agradecimientos

A mis hermanos, Pao, Moi y Mary, por el amor, apoyo, cariño, enojos y solidaridad que siempre nos hemos tenido. A mis cuñados Jorge y Jesús, mis sobrinos Román, Valeria y mi pequeña sobrina Belén por llegar a fortalecer a la familia García Matías.

A mis abuelos, por estar siempre pendientes de mí y de mis hermanos.

A los sinodales Dra. Virginia Berrón Lara, Dr. Antonio Orantes Molina, Dra. Carla Leninca Pacheco Agüero y la Dra. Irma Salinas Pérez por el tiempo dedicado a la revisión del presente trabajo.

A mis amigos: Heriberto, por el apoyo y confianza que siempre me haz tenido, Enrique y Esteban, por el apoyo y la gran amistad que nos unen; Carla, por ser una gran hermana, amiga y comadre; José Manuel por la ayuda prestada mientras culminaba mis estudios, a Mario por convivir en Ixtlán y cultivar su amistad en este periodo; a Ana laura, Carmen, Sayde y Rosario, grandes amigas y compañeras de trabajo y a todas las personas que en algún momento me han ayudado desinteresadamente.

A todos los profesores que me aportaron conocimientos durante mi época de estudiante.

A la Universidad Tecnológica de la Mixteca, por permitir realizarme primero como Ingeniero en Computación y ahora como Maestro en Electrónica y Computación dentro de sus salones de clase, y brindarme la oportunidad de presentar este trabajo, siempre la tendré presente.

A la Universidad de la Sierra Juárez por brindarme la oportunidad de compartir mi experiencia y conocimientos hacia la creación de nuevos profesionistas.

A mis alumnos de la Universidad de la Sierra Juárez que me han enseñado que ser un buen profesor es gracias a ustedes y no gracias a que tenemos un título universitario.

Pero sobre todo a Dios, que me ha cuidado durante este tiempo, además de que no me ha alejado del buen camino y me permitió volver a vivir este momento.

Josué.

Índice

Índice	xi
Lista de tablas	xv
Lista de figuras	xvii
Resumen	xix
Abstract.....	xxi
1. Introducción.....	1
1.1. Importancia del problema y necesidad de la solución.....	5
1.2. Delimitaciones de la tesis	7
1.3. Objetivos del trabajo.....	8
1.3.1. Objetivo general	8
1.3.2. Objetivos específicos.....	8
1.4. Solución propuesta	8
1.5. Estructura de la tesis	9
1.6. Publicaciones generadas	10
2. Estudio de las Principales Metodologías utilizadas para el Desarrollo de Software Educativo.....	13
2.1. Calidad en la educación.....	13
2.2. La mejora del proceso enseñanza-aprendizaje mediante la tecnología.	15
2.3. Ingeniería de Software Educativo con Modelado Orientado a Objetos	17
2.3.1. Propósito.....	17
2.3.2. Enfoque.....	17
2.3.3. Estructura.....	17
2.3.3.1. Análisis	17
2.3.3.2. Diseño.....	19
2.3.3.3. Desarrollo	20
2.3.3.4. Prueba	20
2.4. Metodología para desarrollo de Software Educativo de Pere Marqués.....	20
2.4.1. Propósito.....	20
2.4.2. Enfoque.....	20
2.4.3. Estructura.....	20

2.4.3.1. Génesis de la idea	20
2.4.3.2. Prediseño o diseño funcional	21
2.5. Metodología Extendida para la Creación de Software Educativo desde una Visión Integradora	24
2.5.1. Propósito	24
2.5.2. Enfoque	24
2.5.3. Estructura	25
2.6. Metodología Dinámica para el Desarrollo de Software Educativo	29
2.6.1. Propósito	29
2.6.2. Enfoque	29
2.6.3. Estructura	29
2.6.3.1. Diseño Educativo	29
2.6.3.2. Producción	30
2.6.3.3. Realización	31
2.6.3.4. Implementación	31
2.6.3.5. Eje Transversal de Evaluación	31
2.7. Propuestas de Sistemas de Educación basados en WEB	31
2.7.1. Propósito	31
2.7.2. Enfoque	32
2.7.3. Estructura	32
2.8. Desarrollo de Objetos de Aprendizaje Reutilizables Inteligentes para los Sistemas Educativos Basados en Web	33
2.8.1. Propósito	33
2.8.2. Enfoque	33
2.8.3. Estructura	34
2.9. Sistemas de Educación Adaptativos e Inteligentes Basados en Web Usando Programación Orientada a Componentes	35
2.9.1. Propósito	35
2.9.2. Enfoque	35
2.9.3. Estructura	36
2.9.3.1. Plataforma IRLCOO	36
2.9.3.2. Arquitectura orientada a servicio	37
2.10. Estudio Comparativo de las Metodologías y Modelos	37
3. Diseño de MeSoFT	43
3.1. Prácticas efectivas	43
3.2. Áreas de proceso	45
3.2.1. Áreas de proceso de la Gestión del Proyecto	48
3.2.2. Áreas de proceso de la Gestión Pedagógica	48
3.2.3. Áreas de proceso de la Gestión de la Información	48
3.3. Relaciones entre Áreas de Proceso	49

3.4. Definición de las Áreas de Proceso de la Gestión del Proyecto	51
3.4.1. Planificación del Proyecto	51
3.4.1.1. Establecer estimaciones	52
3.4.1.2. Desarrollar el plan del proyecto.....	56
3.4.1.3. Obtener el compromiso	62
3.4.2. Supervisión del Proyecto	65
3.4.2.1. Supervisar el proyecto	66
3.4.2.2. Gestionar las acciones correctivas.....	72
3.4.3. Medición y Análisis.....	75
3.4.3.1. Establecer las actividades de la medición.....	76
3.4.3.2. Proporcionar los resultados de la medición.....	78
3.5. IADIAL	81
3.6. Definición de las Áreas de Proceso de la Gestión Pedagógica.....	83
3.6.1. Gestión de Requisitos	83
3.6.1.1. Identificar de la necesidad educativa.....	84
3.6.1.2. Establecer la génesis de la idea	86
3.6.1.3. Realizar el Diseño educativo	88
3.6.2. Verificación	95
3.6.2.1. Verificar los requisitos educativos y del sistema.....	96
3.6.3. Validación.....	99
3.6.3.1. Validar los requisitos educativos y de sistema	100
3.7. Definición del Área de Proceso de la Gestión de la Información	102
3.7.1. Definición del proceso.....	102
3.7.1.1. Establecer el proceso estándar	103
3.7.1.2. Establecer la descripción de los ciclos de vida.....	106
3.7.1.3. Establecer el repositorio de activos	108
4. Resultados obtenidos	109
4.1. Limitaciones	¡Error! Marcador no definido.
4.2. Validación del trabajo.....	¡Error! Marcador no definido.
4.3. Datos históricos recopilados.....	109
4.4. Definición de procesos	110
4.5. Implementación del software educativo	111
4.6. Datos recogidos	117
5. Conclusiones.....	120
6. Anexo A.- Activos de MeSoFT	123
6.1. Plantilla de MeSoFT para la Planificación de Tareas: (ACTI)	124
6.2. Plantilla de MeSoFT para Entrevista a Alumnos: (ALU)	125
6.3. Plantilla de MeSoFT para el Resumen del Plan de Calidad: (CAL)	126
6.4. Plantilla de MeSoFT para el Calendario: (CALEN)	127

6.5. Plantilla de MeSoFT para el Plan de Acciones Correctivas: (CORRE).....	128
6.6. Plantilla de MeSoFT para describir la Lista de Objetivos Curriculares: (CUR).....	129
6.7. Plantilla de MeSoFT para el Plan de la Gestión de los Datos: (DAT).....	130
6.8. Plantilla de MeSoFT para el Plan de la Gestión de los Datos: (DAT) (continuación).....	131
6.9. Plantilla de MeSoFT para el Diseño Conceptual: (DIS).....	132
6.10. Plantilla de MeSoFT para describir la Lista de Objetivos Educativos: (EDU).....	133
6.11. Plantilla de MeSoFT para el Inventario de Habilidades: (HAB).....	134
6.12. Plantilla de MeSoFT para el Storyboard: (HIS).....	135
6.13. Plantilla de MeSoFT para el Plan de Participación: (INV).....	136
6.14. Plantilla de MeSoFT para describir el Marco Educativo y Comunicacional: (MAR).....	137
6.15. Plantilla de MeSoFT para el Diseño Computacional: (MOD).....	138
6.16. Plantilla de MeSoFT para el Diseño Computacional: (MOD).....	139
6.17. Plantilla de MeSoFT para el Diseño Computacional: (MOD).....	140
6.18. Plantilla de MeSoFT para las Necesidades y Objetivos de Medición: (NEC).....	141
6.19. Plantilla de MeSoFT para la Matriz participante/actividad: (PARTI).....	142
6.20. Plantilla de MeSoFT para describir la Estrategia Pedagógica: (PED).....	143
6.21. Plantilla de MeSoFT para describir la Estrategia Pedagógica: (PED) (continuación).....	144
6.22. Plantilla de MeSoFT para describir el Problema educativo Identificado: (PROB).....	145
6.23. Plantilla de MeSoFT para Entrevista a Profesores: (PROF).....	146
6.24. Plantilla de MeSoFT para el Plan del Proyecto: (PROY).....	147
6.25. Plantilla de MeSoFT para el Reporte de la Validación: (REPVALI).....	148
6.26. Plantilla de MeSoFT para el Seguimiento de los Riesgos: (RIE).....	149
6.27. Plantilla de MeSoFT para el Resumen de Tamaño del Producto (TAM).....	150
6.28. Taxonomía de Riesgos de MeSoFT.....	151
6.29. Plantilla de MeSoFT para los Criterios de la Validación: (VALI).....	152
7. Anexo B.- Acrónimos.....	153
8. Anexo C.- Actas de publicaciones.....	155
9. Bibliografía.....	161
9.1. Sitios de Internet.....	163
9.2. Otras referencias.....	164

Lista de tablas

Tabla 1. Matrícula de las diferentes carreras de Ingeniería de la UTM.	6
Tabla 2. Procesos a tomar en cuenta dentro del uso de la metodología propuesta por Cataldi.	27
Tabla 3. Procesos a tomar en cuenta dentro del uso de la metodología propuesta por Cataldi (continuación).....	28
Tabla 4. Procesos a tomar en cuenta dentro del uso de la metodología propuesta por Cataldi (continuación).....	29
Tabla 5. Cuadro comparativo entre metodologías y modelos para el desarrollo de SW Educativo.	41
Tabla 6. Objetivos del proyecto.....	109
Tabla 7. Principales procesos de desarrollo de MeSoFT implementados.	110
Tabla 8. Datos estimados contra datos reales.	117
Tabla 9. Buscando el Objetivo 1.	117
Tabla 10. Cumpliendo el Objetivo 2.	118
Tabla 11. Buscando el Objetivo 3.	119

Lista de figuras

Figura 1.1. Arquitectura de MeSoFT	9
Figura 2.1. Metodología ISE propuesta por Galvis.....	18
Figura 2.2. Arquitectura del sistema LTSA-IEEE-LTSC 184.1. [URL11].....	33
Figura 3.1. Definición del ciclo de vida del SE.....	43
Figura 3.2. Un mecanismo para mejorar el proceso de aprendizaje continuo.....	44
Figura 3.3. Área de Proceso de la Gestión del Proyecto	46
Figura 3.4. Áreas de Proceso de la Gestión Pedagógica.	46
Figura 3.5. Definición de Proceso de la Gestión de la Información.....	47
Figura 3.6. Categorías de áreas de procesos de MeSoFT.....	47
Figura 3.7. Áreas de proceso de MeSoFT.	49
Figura 3.8. Modelo IADIAL.	82
Figura 4.1. Estructura física del LabCD.....	111
Figura 4.2. Plantilla ACTI para la planificación de tareas.	112
Figura 4.3. Plantilla ALU para la entrevista de los alumnos.....	113
Figura 4.4. Plantilla CORRE para el Plan de Acciones Correctivas.	114
Figura 4.5. Plantilla HIS para el Storyboard.	115
Figura 4.6. Software del LabCD.....	116
Figura 4.7.GUI para el ejercicio de la ley de Ohm (virtual).....	116
Figura 4.8.GUI para el ejercicio del filtro pasa-bajas (programable).....	117
Figura 4.9. Monitoreando el valor ganado para los ejercicios seleccionados.	118
Figura 4.10. Reducción de tiempo en la fase de prueba.....	118
Figura 4.11. Mejora del % de alumnos aprobados.	119

Resumen

El software educativo es uno de los pilares de los sistemas de enseñanza-aprendizaje a distancia que será utilizado como una herramienta para las generaciones futuras de estudiantes. Sin embargo, las recientes metodologías para el desarrollo de software educativo tienen demasiados problemas como la carencia de marcos de trabajo comunes los cuales puedan ser utilizados para cualquier proyecto, además de la excesiva formalidad de dos factores, el técnico y el pedagógico. En la actualidad, las actividades para el desarrollo de software educativo son complejas porque el proceso se enfoca en la experiencia del desarrollador, así como en los aspectos técnicos de la Ingeniería de Software y la adquisición e implementación del conocimiento pedagógico. Este trabajo propone la introducción de las “prácticas efectivas” en un método alternativo para desarrollar software educativo. La identificación de prácticas efectivas está enfocada en asegurar que el desarrollo del proceso sea conducido con eficacia y orientado a la supervisión pedagógica del proyecto. Las prácticas efectivas que aquí se proponen proporcionan las bases de un método alternativa para desarrollar software educativo con el rigor necesario para desarrollar software comercial, eso nos permitirá obtener un proceso que se puede repetir con altos niveles de éxito en el área de la instrumentación electrónica, específicamente.

Abstract

Educational Software is one of the pillars of distance learning and educational systems and has become the basic tool for future generations of students. However, recent methodologies used in this development have too many problems: a lack of common theoretical frameworks which can be used by anyone in the project, and excessive formality in both technical and pedagogical factors. Activities employed in the development of educational software are complex because the process depends on the developer's expertise, aspects of software engineering, and the acquisition and implementation of pedagogical knowledge.

We propose the introduction of "effective practices" within an alternative method to develop this kind of software. The identification of effective practices is focused internally to ensure the effective realization of the development process, and externally to guide the pedagogical monitoring of a project. Our effective practices provide the basis of an alternative method for the development of educational software under rigorous conditions that enable us to achieve a highly successful and repeatable process in the field of electronic instrumentation.

1. Introducción

La educación, como función fundamental de la sociedad, está obligada no sólo a inculcar los valores, proporcionar los conocimientos, desarrollar las habilidades y educar a la gente, si no también tiene la obligación de generar y mantener los conocimientos, elaborar y aplicar las diferentes modalidades del proceso aprendizaje-enseñanza, crear las herramientas que satisfagan las necesidades del estudiante, pero el aspecto más importante, debe ser capaz de responder a las continuas transformaciones sociales de un mundo en permanente evolución.

Un problema tradicional en los medios de enseñanza consiste en determinar cómo pueden ser diseñados para cumplir eficazmente con la función para la cual son creados; es decir, para que la comunicación de sus mensajes sea eficaz y la interacción que se establezca con el usuario sea lo más útil posible. En definitiva, siempre se ha buscado facilitar el aprendizaje y almacenar la información que propicie entornos de aprendizaje más variados [Cabero92].

Sin embargo, existen problemas desde el punto de vista de las relaciones profesor-alumno que nos dejan entrever fracasos durante el aprendizaje:

- Demostrar respeto y educación al alumno, lo que implica escucharlo, dedicarle tiempo y voluntad de atenderlo personalmente.
- Valorarlo como individuo, es decir, mantener una relación personal con él, reconociendo sus avances y sus logros.
- Ayudarle en sus problemas personales, es decir, estar disponible para dedicarle tiempo y ayuda en sus dificultades no académicas (tutorías).
- Apoyarle en sus problemas académicos, animándole y guiándole en sus dificultades o explicándole la materia de forma más comprensible.

Para ayudar a solventar estos problemas, causados por la frustración al estudiar una materia por primera vez, se han desarrollado una infinidad de laboratorios virtuales dentro de la modalidad de **educación a distancia** o educación no presencial. El desarrollo de la educación a distancia tiene sus orígenes en los países con problemas de cobertura en la educación, haciéndose indispensable el desarrollo y utilización de nuevas tecnologías que permitan expandir la oferta de educación a un mayor número de estudiantes.

De forma similar, la tendencia de las ventajas tecnológicas que ofrecen tanto el software como el hardware, permiten aprovechar mejor los recursos en la creación de sistemas interactivos de enseñanza que puedan cumplir con el objetivo de capacitar a los estudiantes que requieran entrenamiento en áreas determinadas.

Actualmente, muchas de las universidades públicas destinadas a formar a los estudiantes no cuentan con la infraestructura necesaria para brindar una formación adecuada, es por eso que el desarrollo de la educación a distancia representa una solución alternativa para dichos problemas [Giraldo02].

El *Software Educativo* (SE), también conocido como *courseware*, se define como una herramienta software orientada a la educación y creada con la finalidad específica de ser usada como un medio didáctico que facilite los procesos de enseñanza-aprendizaje en sus modalidades tradicional, presencial o a distancia. De acuerdo a los trabajos de Marqués [Marques95], Squires [Squires01], Cataldi [Cataldi03], García [Garcia02A], Lipeikine [Lipeikine03], Adomavicius [Adomavicius04] y Kocijancic [Kocijancic04] se sabe que es posible mejorar el desempeño académico en forma sustancial y que estos resultados pueden variar según el tipo de software utilizado y el tipo de metodología aplicada para su desarrollo.

Sin embargo, en la actualidad el término “educativo” se agrega a cualquier producto que sea diseñado con la intención de enseñar un tema específico. El SE no se ha utilizado como un proceso formal de enseñanza que posea un diseño específico para adquirir el conocimiento, las habilidades y los procedimientos que aseguren que todos los estudiantes obtengan solamente el conocimiento necesario [Gros00]. El desarrollo de SE requiere de las condiciones deseadas que incluyan características didácticas y pedagógicas junto con el análisis y las fases de diseño para su construcción.

Desde hace algunos años, el SE ha constituido la principal evidencia del impacto de la tecnología en la educación [Fernandez99], ya que muestra características particulares en cuanto a la comunicación con el usuario, las cuales no se pueden cuantificar mediante métricas porque están relacionadas con conductas de aprendizaje.

Sin embargo, la evolución de las Tecnologías de Información (TI) ha proporcionado una nueva alternativa en el área educativa. Esto ha originado cambios importantes en la comunidad educativa que han proporcionado ventajas a todos sus miembros, incluso con la modificación de los modelos tradicionales de enseñanza los *laboratorios virtuales* son una alternativa para mejorar el concepto de la “educación de la distancia”.

Las ventajas tecnológicas del software y hardware permiten una gestión adecuada de los recursos cuando el sistema interactivo de enseñanza es desarrollado. Estos sistemas deben ser capaces de proporcionar exclusivamente el conocimiento necesario de acuerdo al nivel del estudiante y el campo de interés. Sin embargo, algunas instituciones académicas no tienen la infraestructura necesaria para esto (equipamiento, maestros, instalaciones, y demás). El desarrollo de los laboratorios virtuales, por ejemplo, proporciona una solución alternativa a estos problemas específicamente en el área de la Electrónica [Garcia04][Garcia07].

De acuerdo a Galvis [Galvis98], hasta hace algunos años la demanda de SE de alta calidad era cada vez mayor y las metodologías convencionales de Ingeniería de Software Educativo (ISE) presentaban mecanismos robustos para realizar un análisis de necesidades y diseño educativo completo, pero poco habían evolucionado en cuanto a la tecnología, específicamente en lo relacionado con el diseño computacional.

Para construir un SE bajo las condiciones deseadas dentro de las fases de análisis y diseño del mismo se deben incorporar aspectos didácticos y pedagógicos, que faciliten y garanticen la satisfacción de necesidades educativas tal como lo propone Galvis en su *Metodología para el Desarrollo de Software Educativo* [Gavis93].

La *Metodología propuesta por Marqués*, define 11 etapas para facilitar el proceso de diseño y desarrollo de SE y menciona que la elaboración del mismo no es un proceso lineal, sino iterativo: en

determinados momentos de la realización se comprueba el funcionamiento, el resultado, se evalúa el producto y frecuentemente se detecta la conveniencia de introducir cambios [Marqués95].

La *Metodología Extendida para la Creación de Software Educativo desde una Visión Integradora* propuesta por Cataldi, hace hincapié en que las metodologías propias de la Ingeniería del Software no toman en cuenta aspectos pedagógicos-didácticos del producto software ha desarrollar, muy pocas metodologías toman en cuenta los aspectos de comunicación con el usuario que en principio son un componente elemental y básico en el desarrollo de SE [Cataldi03].

La metodología propuesta por Díaz de Feijoo en [Díaz02], denominada "*Propuesta de una Metodología de Desarrollo y Evaluación de Software Educativo bajo un Enfoque de Calidad Sistémica*" sigue la metodología de Proceso Unificado de Rational (RUP, *Rational Unified Process*) adaptada a la producción de software educativo y se apoya en el Modelo Sistémico de Calidad (MOSCA) propuesto por el Laboratorio de Información de Sistemas (LISI) de la Universidad Simón Bolívar.

Una propuesta más para desarrollar SE es la de Arias [Arias02], llamada *Metodología Dinámica para el Desarrollo de Software Educativo*, que se basa en la necesidad de concebir el medio instruccional, es decir, utilizar la computadora como un medio dinámico aprovechando todo el poder multimedia de esta, la metodología está compuesta por cuatro fases (diseño educativo, producción, realización e implementación) [Arias02].

Entre las nuevas iniciativas para desarrollar herramientas educativas se encuentra el uso de la Inteligencia Artificial con aspectos cognitivos y las ciencias pedagógicas para integrar los sistemas, programas y herramientas que proporcionen recursos y servicios de acuerdo con el alumno; a través de la creación de repositorios de conocimientos, planificación curricular, entornos de colaboración, asistencia individual, y los mecanismos de adaptación, esto con el fin de aprovechar las contribuciones de una comunidad internacional y que sean adecuadas para un perfil de estudiante a través de Internet, mediante la creación de programas informáticos educativos basados en la Web (SEBW).

Según Sheremetov y Uskov [Sheremetov01]: "*la primera generación de SEBW se caracteriza por el desarrollo de cursos on-line con la adición de correo electrónico, grupos de usuarios, manejar los mensajes, gestión de páginas de información y transferencia*". En el caso de la Universidad Tecnológica de la Mixteca (UTM) esta primera generación de SEBW se ha estado utilizando como plataforma de aprendizaje en la Maestría Virtual; sin embargo, una limitación importante de esta propuesta ha sido la falta de integración e interacción entre los componentes. Para evitar estas deficiencias una segunda generación promueve el curso en línea de gestionar, planificar y controlar el proceso de aprendizaje, proporcionando un acceso fácil a los temas de educación, la gestión del rendimiento de los estudiantes, y el control de asignación de tareas. Entre las más importantes herramientas eReading fueron: Blackboard y WebCT.

La tercera generación creó el paradigma de aprendizaje llamado "Web Lecturing" [Brusilovksy, 2000] el cual desarrolla y proporciona material didáctico en multimedia (texto, audio, vídeo y realidad virtual) para enviarlo a través de Internet. Entre los resultados más significativos se encuentran los siguientes: INTERLABS de Bradley University, Universidad Virtual de Carnegie Mellon, Mi UCLA, y UW de la Universidad de Washington.

Hoy en día, la nueva generación de SEBW está diseñada con muchos niveles de servicio, algunos son: de cooperativa de trabajo dedicada a compartir los repositorios de conocimiento [ADL, 2007], de educación orientada al usuario a través del paradigma de aprendizaje adaptativo [Rebak, 2000], de colaboración de desarrollo de conocimientos entre los estudiantes [Cumming, 1998], la aplicación de entornos inteligentes en la planificación y el control de aprendizaje [Sheremetov,

2002], el desarrollo de la web semántica, el uso de herramientas en la Inteligencia Artificial Distribuida [Kimovski, 2003] y las ontologías [Miltiadis, 2003].

Así, no se puede considerar que actualmente exista un conjunto de fundamentos, principios, métodos y herramientas claramente definidas y ampliamente aceptadas, que constituyan la base de una ISE lo suficientemente completa [Garcia02A]. Además, el construir SE nunca es fácil. Según Cabero “*uno de los grandes problemas para el uso e introducción de la informática en el terreno educativo ha sido la calidad del software*”. Esto se reduce a factores técnicos, de actitud, disponibilidad del software adecuado, y otro tipo de factores educativos [Cabero92].

Un problema tradicional en los instrumentos educativos es diseñarlos de tal manera que logren efectivamente los fines para los que fueron creados, lo que significa que la comunicación y la interacción con el usuario deben ser eficaces. El diseñador, sin embargo, se ha enfrentado a una cuestión diferente, a saber, la forma de mejorar el proceso de aprendizaje y almacenar toda la información de una manera que pueda crear una variedad de entornos de aprendizaje [Cabero92].

En la actualidad, la matrícula de estudiantes se incrementa muy rápidamente, se requiere mayor flexibilidad en los horarios de los maestros, el número de estudiantes con acceso a Internet está creciendo exponencialmente, y el material de aprendizaje cambia rápidamente [Lipeikiene03].

Según García Roselló [Garcia02A] existen 3 problemas fundamentales que afectan al desarrollo de SE como un enfoque didáctico, en el marco de la tecnología educacional:

- *La falta de impacto educativo*, es decir la falta de integración de las innovaciones en investigación educativa en un conjunto de herramientas prácticas basadas en la computadora que permitan apoyar las reformas de los planes de estudio.
- *La necesidad de reducir los costos de desarrollo*, para incrementar el impacto del SE en los planes de estudio, como se mencionaba en el punto anterior, se requiere, entre otras cosas, reducir drásticamente los costos de producción. Hoy en día, a medida que el coste del hardware va siendo menor, el gasto real para desarrollar SE de buena calidad se ve incrementado por el riesgo de fracaso asociado a la producción de ese software.
- Por último, *la dificultad de estimación de los recursos necesarios para su producción*, es decir, la creciente demanda de SE de calidad, tecnológicamente más complejo y por lo tanto más caro, ha generado la necesidad, por parte de los desarrolladores, tanto comerciales como de otros ámbitos, de realizar estimaciones realistas del esfuerzo requerido para un proyecto de desarrollo de SE.

La calidad del SE está determinada no solo por los aspectos técnicos del producto sino por el diseño pedagógico y los materiales de soporte. Este aspecto es uno de los más problemáticos ya que existen pocos programas que ofrezcan un soporte didáctico [Gros00].

Tradicionalmente, el SE tiene el predominio del paradigma generativo, el cual permite un alto grado de reutilización, y tiene la ventaja de que los patrones reutilizados generalmente son diseñados e implementados por expertos en el dominio de la aplicación, su mayor desventaja es la inflexibilidad, es decir, lo cual limita su adaptación a requisitos que no encajen bien dentro del patrón generativo, otro inconveniente es su claro sesgo hacia la generación de aplicaciones completas. Por todo ello las aplicaciones resultantes son poco integrables y reutilizables, una vez ya generadas [Garcia02B].

La propuesta de esta tesis es desarrollar un método para el desarrollo de SE que esté basado en el concepto de *prácticas efectivas para el desarrollo de software*. De acuerdo a Jones [Walker03] “*las prácticas efectivas son todas aquellas metodologías y herramientas que mejoran*

consistentemente la productividad y calidad de los productos software cuando son implementadas, y al aplicarlas se tiene un beneficio directo en el proyecto”.

Por lo tanto, las prácticas efectivas de software son aquellas prácticas que las personas con experiencia reconocida en el área específica tienen identificadas a través de la experiencia y esa contribución se hace significativa para el buen término del proyecto de software [Adams04].

De acuerdo a las definiciones anteriores sobre el concepto de “prácticas efectivas”, la experiencia al introducir estas actividades no ha obligado a las personas o a las compañías a establecer una práctica inflexible y sin cambios. En lugar de eso, las prácticas efectivas han sido un método basado en el aprender continuo y la mejora continua. Las prácticas efectivas para el desarrollo software son, por lo tanto, prácticas que las personas con reconocida experiencia en la creación de software tiene identificadas a través su experiencia y que representan contribuciones significativas para la correcta construcción del mismo.

Sin embargo, no existe una metodología para SE definida a su conveniencia en totalidad, cada desarrollador cuenta con su propia experiencia y cada quien utiliza las metodologías tradicionales creadas para la programación de sistemas. Tal es el caso de la falta de formas para adquirir los requisitos del usuario, las formas erróneas de presentar los resultados obtenidos, la mala interpretación de los requisitos del sistema, el error de no elegir correctamente el sistema de desarrollo para realizar el software, la mala planificación del proyecto, y demás.

1.1. Importancia del problema y necesidad de la solución

De acuerdo con la Universidad Autónoma de México: *"A nivel nacional, el sistema educativo mexicano ha presentado las siguientes características: (1) la falta de un plan estratégico nacional, sólo son buenas intenciones, (2) la falta de continuidad en los programas de educación y estrategias; (3) centrarse en los intereses políticos; (4) una contradicción entre los planes y las acciones desarrolladas; (5) los profesores tienen el papel central en el proceso educativo en la escuela, y (6) que los intelectuales, políticos y tecnócratas deben jugar un papel protagonista en México "* [UNAM, 2000]. En el modelo tradicional de educación la "exposición de temas" es la principal técnica didáctica. Los profesores hacen preguntas a los estudiantes y alientan su participación y proponen algunos proyectos que deben desarrollarse dentro o fuera del aula y de forma individual o en equipo. Los estudiantes se concentran en la toma de notas, reflexionar sobre lo que el maestro dice, participar en discusiones de grupo y pedir al profesor que aclare los conceptos que ellos no entienden [Ramírez, 2004].

Este sistema de educación tradicional ha sido eficaz para muchos profesores a través de los años, y ha respondido a las necesidades de la sociedad en ese momento dado. Por otra parte, el modelo tradicional de educación no está explícitamente declarado; por lo tanto, las habilidades, valores y actitudes para ser desarrolladas por los estudiantes no están previstos de antemano. De este modo, los estudiantes pueden o no desarrollar su capacidad óptima. En esta situación no es común para el profesor definir los métodos para medir el desarrollo, los valores, habilidades y actitudes de los estudiantes. En este modelo de enseñanza el profesor toma el papel central en el proceso de aprendizaje, decidiendo lo que debe ser aprendido por los estudiantes y la forma en que serán evaluados. A lo largo de la historia, los maestros mexicanos han desempeñado una función central en el proceso educativo sin dar a los estudiantes su lugar en el proceso de aprendizaje. Se podría pensar que se ha olvidado que los maestros y estudiantes trabajan juntos, de que el conocimiento surge en el proceso de enseñanza-aprendizaje. Es importante subrayar que este sistema de educación todavía existe en muchas escuelas públicas y privadas en México. El objetivo de esta tesis es

proponer el desarrollo de herramientas educativas de calidad que permitan la mejora del sistema de enseñanza mediante el uso de las TI.

Un problema principal en los países en vías de desarrollo es el limitado presupuesto asignado a las universidades públicas. Esta cuestión se refleja en la infraestructura mínima y la falta de equipo para enseñar cursos de especialización como las áreas de Instrumentación Electrónica Programable e Instrumentación Electrónica Virtual (PVEI), por ejemplo.

Los profesores tienen que encontrar una forma de ofrecer los conocimientos técnicos de los cursos a todos los estudiantes de la misma manera que podía hacerlo con los métodos tradicionales. En nuestro caso concreto, la UTM es una institución mexicana con más de 15 años de experiencia. El laboratorio de Electrónica de Investigación cuenta con una infraestructura limitada para satisfacer las necesidades de los 8 grupos (un total de 160 estudiantes) en esta área específica. Sin embargo, en los últimos años la Universidad ha aumentado el número de estudiantes y por lo tanto se han encontrado dos nuevos problemas;

- El equipo tecnológico actual no es suficiente para evaluar las capacidades de todos los estudiantes, al mismo tiempo, específicamente en las áreas PVEI (véase el Tabla 1), y
- El SEBW creado en una primera fase mostró deficiencias puesto que no fue desarrollado tomando en cuenta las prácticas pedagógicas correctas y presentó problemas técnicos al no tomar en cuenta un proceso metodológico específico para herramientas educativas.

Los esfuerzos de este trabajo de tesis se enfocan en las áreas de Instrumentación Electrónica Programable (IEP) e Instrumentación Electrónica Virtual (IEV). Las áreas de IEP e IEV tienen la intención de automatizar el proceso de adquisición de parámetros eléctricos a través de Sistemas Automatizados de Prueba (ATE, *Automated Test Equipment*).

Actualmente, el desarrollo de los ATE es un área de investigación para las compañías de software y hardware quienes desarrollan herramientas nuevas para controlar el hardware a distancia, proporcionando así, soluciones alternativas para la industria y centros educativos. La firma National Instruments ha desarrollado la herramienta conocida como NIELVIS (*NI Educational Laboratory Virtual Instrumentation Suite*) o Laboratorio Educativo de Instrumentación virtual de National Instruments, que controla hardware a través de software y proporciona un mecanismo formal para realizar mediciones precisas. Sin embargo, la herramienta de National Instruments no contiene las bases pedagógicas que motiven a los estudiantes a aprender solamente una parte específica de la IEV a través del software, un sitio Web por ejemplo.

La infraestructura tecnológica orientada a la IEP e IEV que tiene la UTM, se está quedando disminuida ante el crecimiento de la matrícula en las carreras de Ingeniería en Electrónica, Ingeniería en Computación e Ingeniería en Mecatrónica en la UTM. La Tabla 1 presenta el incremento anual de la matrícula.

Tabla 1. Matrícula de las diferentes carreras de Ingeniería de la UTM.

	2001	2002	2003	2004	2005	2006	2007
Electrónica	104	107	123	121	116	182	193
Computación	168	234	220	201	174	302	360
Mecatrónica	0	0	0	0	0	101	142
Total	272	341	343	322	290	585	695

El Laboratorio de Comunicaciones Digitales (LabCD) de la UTM cuenta con 10 ATE's, los cuales son insuficientes para satisfacer las necesidades de servicio de los alumnos quedando así limitada la realización de prácticas. Por cada ATE hay 50 alumnos, en promedio, trabajando en prácticas, este problema trae una limitación considerable para optimizar el grado de aprendizaje de cada alumno; aunado a esto se encuentra el problema de que no existe una metodología correcta de trabajo para el desarrollo de programas de este tipo, existen muchas pérdidas de tiempo y el uso del equipo está limitado.

En el campo de la educación se desarrollan una infinidad de laboratorios virtuales en la modalidad de educación a distancia o educación no presencial. El desarrollo de la educación a distancia tiene sus orígenes en los países con problemas de cobertura en la educación, haciéndose indispensable el desarrollo y utilización de nuevas tecnologías que permitan expandir la oferta de educación a un mayor número de población.

De forma similar, la tendencia de las ventajas tecnológicas, que ofrece tanto software como hardware, permiten aprovechar mejor los recursos en la creación de sistemas interactivos de enseñanza que puedan cumplir con el objetivo de capacitar a los estudiantes que requieran entrenamiento en áreas determinadas.

Actualmente, muchas de las escuelas destinadas a capacitar al personal no cuentan con la infraestructura necesaria para brindar una formación adecuada, es por eso que el desarrollo de la instrumentación virtual y los laboratorios virtuales vienen a ser una solución a dichos problemas [Garcia04].

El desarrollo de los ATE's aplicados al campo de la IEP se encuentra en la etapa de desarrollo de nuevas herramientas software que controlen equipo hardware y den solución a problemas presentados en la industria, laboratorios de investigación, pero sobre todo a la educación. Tal es el caso de la herramienta diseñada por la firma National Instruments denominada NIELVIS, la cual es un desarrollo hardware y software que en la actualidad es utilizado en los cursos de varias instituciones, desde la Universidad de Georgia Tech en Estados Unidos de America hasta la Universidad de Keio en Japón.

Por último, en México, la Secretaria de Educación Publica (SEP) y el Instituto Nacional de Estadística Geografía e Informática (INEGI) en su encuesta de formación de recursos humanos en informática publicada el 13 de agosto del 2003 informó que la participación porcentual de las escuelas con posgrado en TI y comunicaciones con apoyo de la teleinformática en el proceso de enseñanza aprendizaje en el periodo 2001/2002 fue de un 70% de un total de 4347 escuelas universitarias y posgrado a nivel nacional. Esto muestra que el desarrollo de SE tiene un amplio campo de desarrollo y aplicación.

1.2. Delimitaciones de la tesis

Las delimitaciones de la investigación son planteadas a continuación.

- El método que se propondrá estará orientado a apoyar específicamente en el desarrollo de aplicaciones educativas para educación a distancia.
- El método que se propondrá cubrirá la planificación y el diseño de SE exclusivamente, la implementación dependerá del lenguaje que el programador seleccione.
- La validación del método se centrará en probar la implementación del método en el área de la instrumentación electrónica específicamente.

- El grupo de estudiantes que será tomado como muestra para aprobar el método será de tercer semestre de la Licenciatura en Informática de la Universidad de la Sierra Juárez.
- El modelo pedagógico que se utilizará se centrará en la técnica de Thomas D'Zurilla y Marvin Goldfried para la resolución de problemas.
- El método generará un conjunto de plantillas que deberán utilizarse para obtener resultados exitosos.

1.3. Limitaciones

- No existe un modelo de calidad orientado a establecer prácticas efectivas para el desarrollo de SE, por lo que se tendrá que adaptar un modelo de calidad ya existente enfocado a SW comercial.
- MeSoFT será probado en la Universidad de la Sierra Juárez, sin embargo se aclara que ésta no posee la infraestructura ideal para obtener datos significativos sobre la fiabilidad del método construido. A pesar de esto consideramos que esta tesis será el primer paso para iniciar una serie de proyectos piloto que nos permitirá generar un histórico de datos.

1.4. Objetivos del trabajo

Los objetivos del trabajo están compuestos de un objetivo general y varios objetivos secundarios, los cuales son descritos a continuación:

1.4.1. Objetivo general

Diseñar, construir y desarrollar un método alternativo para el desarrollo de Software Educativo basado en la introducción e implantación de prácticas efectivas (MeSoFT).

Para alcanzar el objetivo general será necesario conseguir ciertos objetivos secundarios. Estos establecerán las aportaciones esperadas al final de la tesis:

1.4.2. Objetivos específicos

- *Adaptar un modelo de calidad para controlar el proceso de desarrollo del SE.*
- *Proponer un modelo pedagógico orientado al desarrollo de software educativo.*
- *Diseñar documentos definidos a través de un ciclo de vida de desarrollo para controlar el aprendizaje.*
- *Desarrollar un conjunto de prácticas de IEP e IEV, en base a la implementación del método desarrollado*

1.5. Solución propuesta

Se propone el desarrollo de un método que proporcione el soporte y la guía en el desarrollo de aplicaciones destinadas a cubrir el proceso de enseñanza-aprendizaje mediante el uso de la computadora, también llamadas Software Educativo. Un punto importante será la intervención de

los expertos de las dos áreas que convergen dentro de esta solución propuesta, tanto los expertos del área de pedagogía como los expertos de la Ingeniería de Software que proporcionarán el conocimiento para la construcción del método objetivo de esta tesis; de esta forma, se estaría construyendo un mecanismo capaz de identificar un conjunto de prácticas efectivas que al seguirse, generará productos educativos de calidad para un área específica.

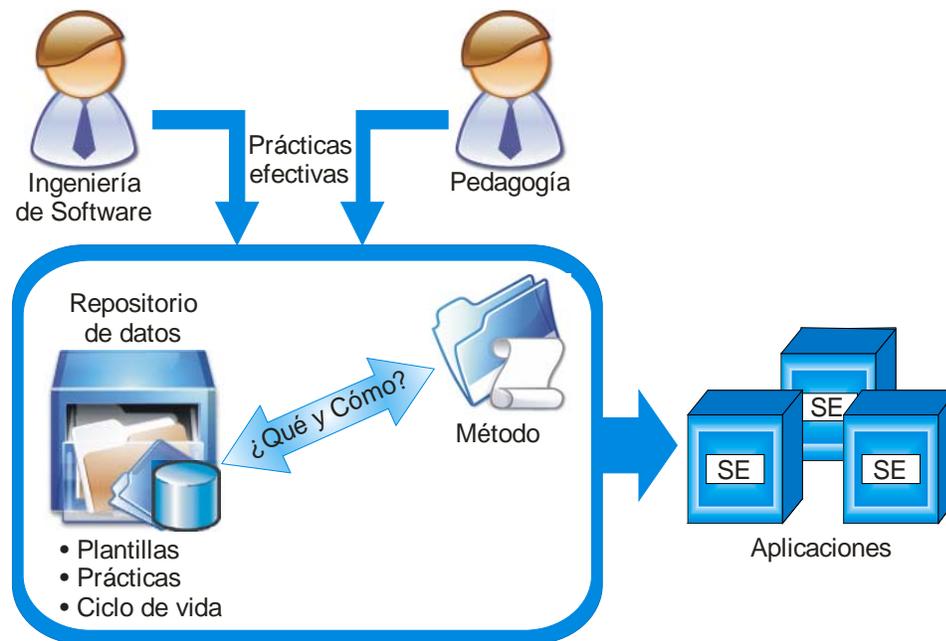


Figura 1.1. Arquitectura de MeSoFT

La arquitectura propuesta para el desarrollo de MeSoFT (ver Figura 1.1) resumirá el conocimiento de expertos en las áreas de la Ingeniería de Software y la Pedagogía, los cuales brindarán los conocimientos y experiencia necesarios para identificar un conjunto de prácticas efectivas. Se pretende que la arquitectura conste de un repositorio de datos en donde se almacene toda la información necesaria para el desarrollo de las aplicaciones educativas.

Al identificar las prácticas efectivas la arquitectura deberá almacenarlas en un repositorio de datos junto con sus plantillas, evaluaciones, documentos, diagramas, y demás. Se tiene la idea que con este mecanismo, el método proporcionará calidad al reproducir el conocimiento en diversos proyectos orientados bajo el enfoque de SE.

1.6. Estructura de la tesis

La estructura del documento de tesis se detalla a continuación.

El capítulo 2 presenta el marco conceptual acerca de metodologías y modelos de desarrollo de SE, se expone un análisis sobre los modelos y metodologías revisadas.

En el capítulo 3 se describe el diseño del método para desarrollo de SE basado en prácticas efectivas.

El capítulo 4 presenta la validación del método para el desarrollo de SE basado en prácticas efectivas mediante la implementación en el área de la Instrumentación Electrónica.

El capítulo 5 presenta las conclusiones y líneas futuras sobre este trabajo.

El anexo A presenta el repositorio de activos generado, como son las platillas definidas para el uso del método.

El anexo B presenta una lista de acrónimos que se mencionan durante el desarrollo del presente trabajo.

El anexo C resume las publicaciones generadas durante el desarrollo de esta tesis.

Por último se presentan las referencias bibliográficas utilizadas en el desarrollo de esta tesis.

1.7. Publicaciones generadas

A continuación se enlistan algunas de las publicaciones que se generaron durante el desarrollo del presente trabajo.

Autores:	Josué N. García , Heriberto I. Hernández
Título:	An alternative methodology for developing virtual instruments in virtual labs
Congreso:	IEEE Instrumentation and Measurement Technology Conference, IMTC07
Publicación:	2007 IEEE Instrumentation and Measurement Technology Conference Proceedings IEEE Catalog number: 07EX1720C ISBN: 1-4244-1080-0 ISSN: 1091-5281 Library of Congress: 2007921695
Lugar:	Warsaw, Poland
Año:	1,2 y 3 de Mayo 2007

Autores:	Iván A. García, Josué N. García
Título:	MeSoft: Una Metodología Alternativa para el Desarrollo de Software Educativo Basada en Prácticas Efectivas
Congreso:	VIII Encuentro Internacional Virtual Educa 2007
Publicación:	Memorias del VIII Encuentro Internacional Virtual Educa 2007
Lugar:	São José dos Campos, São Paulo, Brasil.
Año:	18-22 de Junio, 2007

Autores: Iván A. García, **Josué N. García**
Título: Implementing Virtual Practices using an Alternative Methodology to Develop Educational Software
Congreso: The Second International Conference on Systems and Networks Communications, ICSNC 2007
Publicación: ICSNC, The Second International Conference on Systems and Networks Communications Proceedings
IEEE Computer Society Number: 0-7695-2938-0/07
Lugar: Cap Esterel, French Riviera, France.
Año: August 25-31, 2007

Autores: Iván A. García, **Josué N. García**
Título: A Methodology based on Effective Practices to develop Educational Software
Una Metodología basada en Prácticas Efectivas para desarrollar Software Educativo
Publicación: Revista Iberoamericana de Computación y Sistemas (CIC IPN)
Lugar: México, D.F. No.: 4
Año: Abril – Junio 2008 Volumen: 11
ISSN: 1405-5546 Pags. 313-332

Autores: Carla Pacheco, Iván García and **Josué García**
Título: Developing Educational Software with Effective Practices
Congreso: 38th ASEE/IEEE Frontiers in Education Conference
Publicación: 38th ASEE/IEEE Frontiers in Education Conference Proceedings
Lugar: Saratoga Springs, NY
Año: October 22 – 25, 2008

2. Estudio de las Principales Metodologías utilizadas para el Desarrollo de Software Educativo

2.1. Calidad en la educación

La educación es un fenómeno que puede tener las formas y las modalidades más diversas, según sean los diversos grupos de personas y su correspondiente grado de desarrollo. Desde un punto de vista particular, la educación se puede dividir en dos tipos de educación: la *educación cultural* que consiste precisamente en la transmisión de la cultura del grupo, o bien la *educación institucional* que tiene como fin llevar al grupo al nivel de los modos de vida o las técnicas propias del grupo de personas.

La función y el concepto de *escuela* y *educación* han sufrido muchos cambios; esto probablemente se deba al desarrollo social, a los factores económicos y, por supuesto, a los factores políticos. De esta manera la práctica docente ha sido afectada de una u otra manera y se ha visto obligada a asumir nuevas concepciones y diferentes formas de actuar en relación a su tarea educativa.

La educación también ha sido señalada o acusada de reproducir y apoyar o reforzar los intereses de la clase que en ese momento tiene el poder, por lo tanto se puede considerar que actualmente debe generarse un cambio que provoque como resultado, una educación y una didáctica que comprometa a los profesores con responsabilidad y conciencia.

En sus distintos niveles, la educación puede presentar el problema de que gran parte de los profesores prefieran un método histórico y tradicional a utilizar un nuevo paradigma como lo es la educación a distancia o el empleo de nueva tecnología como herramienta de trabajo que permita una mejor interacción entre el profesor y el alumno, el *software educativo*, o un nuevo modelo que propicie la reflexión crítica del alumno, su genuina participación pero principalmente, un verdadero aprendizaje que mejore su nivel de vida.

El proceso de innovación educativa fue legitimado mediante el uso de varias estrategias, como por ejemplo las evaluaciones a los docentes, a los alumnos sobre el nivel de aprendizaje, a la escuela en relación a las respuestas a situaciones conflictivas tales como deserción, fracaso escolar, violencia, reinserción, permanencia y demás. Este proceso generó un sentimiento de rechazo hacia las innovaciones educativas. Ciertamente se ha comprobado que se necesitan cambios, mejoras y calidad educativa, pero uno de los problemas es que el personal docente no se encuentra capacitado para aprovechar estas innovaciones tecnológicas, dejando así que la educación siga siendo de forma tradicional y un poco rudimentaria.

Modernizar la escuela y actualizar a los profesores para actuar en un mundo cambiante en lo cultural, social, tecnológico y laboral, se torna sin lugar a dudas en algo necesariamente importante. Pero cuando la producción de conocimiento y la interacción entre quienes lo producen y quienes lo aplican, mediante la utilización de modernas estrategias tecnológicas, es orientada a un consumidor final, parece difícil obtener mejoras en el ámbito del desarrollo de las personas [Gomez01].

En los últimos años de la década de los 90's la política educativa de los países latinoamericanos introduce un proceso de innovación en busca de mejoras, cambios y perfeccionamiento de las prácticas educativas, como es la educación a distancia y el empleo de software educativo.

La educación a distancia es una modalidad que permite el acto educativo mediante diferentes métodos, técnicas, estrategias y medios, en una situación en que alumnos y profesores se encuentran separados físicamente y sólo se relacionan de manera presencial de vez en cuando [URL8][URL9].

Durante los últimos años, la educación a distancia se ha ido abriendo un lugar dentro del campo de la educacional mundial. Actualmente existen desde la capacitación laboral hasta los posgrados universitarios que figuran en la oferta de la mayoría de las instituciones educativas. El tema principal de esta tesis busca ayudar a mejorar la calidad de la educación a través de la introducción de las herramientas tecnológicas, en nuestro caso la educación a distancia a través de SEBW, pero antes es necesario analizar la situación del uso de la tecnología en la educación.

Se debe comenzar por definir el concepto de *tecnología de la educación* como “*la suma total de las actividades que hacen que la persona modifique sus ambientes externos (materiales) o internos (de comportamiento). Una aplicación sistemática de los recursos del conocimiento científico del proceso que necesita cada individuo para adquirir y utilizar los conocimientos*” [URL9].

El sentido tradicional confunde a la tecnología educativa con máquinas, salones audiovisuales, y demás, limitando su ámbito solamente a lo que se refiere a medios mecánicos, eléctricos o recursos electrónicos. En este sentido, la educación se orienta por la denominada “*artesanía pedagógica*”, enfocándose en fines instrumentales y de apoyo. Se ha mitificado, además de las máquinas, todos los procesos técnicos rigurosos, las programaciones rígidas, los objetivos cerrados y concretos, las pruebas objetivas en evaluación, y todo aquello que en apariencia da seguridad y eficacia [URL9].

Por otra parte, en los últimos años la tecnología ha sufrido un gran avance; sin embargo es muy difícil que este avance tenga detenimiento, tanto en lo que se refiere a la investigación y el conocimiento del hombre y de la naturaleza, así como en lo que tiene que ver con los medios y recursos. El movimiento que la informática ha impuesto al mundo actual genera la necesidad de revisar todos los esquemas científicos y filosóficos que se atañen directamente a la misma estructura mental del ser humano.

Un sistema tecnológico en educación es el que se pone en funcionamiento para lograr que las realidades educativas existentes, sus objetivos, los medios y recursos al alcance, tanto humanos como materiales, se organicen, estructuren y sistematicen en un proceso único y global que permita la realización plena del individuo en su sociedad. Este proceso único es el que se llama sistema de enseñanza-aprendizaje [URL8][URL9].

Los avances tecnológicos aplicados a la educación en los últimos años han sido verdaderamente grandes. Está comprobado que el uso de la tecnología y el uso del software educativo mejora el aprendizaje de los alumnos y al mismo tiempo reduce el tiempo de instrucción y los costos de la enseñanza [CINEA03].

Los alumnos necesitan para su futuro profesional la utilización de los medios tecnológicos, ya que éstos varían enormemente su habilidad de percepción y aprendizaje, y por lo tanto, sus requerimientos didácticos individuales mejoran. Algunos alumnos aprenden fácil y rápidamente a través de informaciones orales o impresas y con un mínimo de experiencias más directas. Sin embargo, muchos otros requieren experiencias más concretas que incluyan los medios audiovisuales.

Existen muchos factores culturales que afectan el aprendizaje; los alumnos necesitan de un amplio catálogo de experiencias que incluya aspectos reales, representaciones visuales y símbolos abstractos. Las nuevas necesidades y expectativas laborales que el alumno tiene aconsejan una mayor participación del mismo en el aprendizaje mediante los métodos activos de investigación y experimentación.

Los programas educativos necesitan ser apreciados en términos de eficacia y flexibilidad de aplicación en cuanto a tiempo, personal y recursos de que se disponga. La demanda de empleo, exige una preparación que obliga al conocimiento de todo aquello que tiene que ver con la sociedad de la información, las nuevas tecnologías, la multiplicidad y variación profesional, la interacción de recursos, y en fin, de todo aquello que facilita la inserción en el campo laboral y profesional [URL8][URL9].

Los profesores, utilizando las nuevas tecnologías, pueden sentirse más libres para realizar trabajos de orientación. Los nuevos patrones didácticos en los que se toma en cuenta a las nuevas tecnologías y los medios de comunicación para mejorar el aprendizaje suponen nuevas funciones para los profesores. No es el profesor el que debe proporcionar toda la información, ya que ésta se puede presentar eficazmente por los medios apropiados, ya sea para distribuirla en grandes grupos o para que cada alumno la amplíe por sí mismo en forma individual o en equipo. Con esta medida, los profesores se liberan de trabajos rutinarios y pueden centrarse en la preparación de los alumnos, que hasta ahora se había descuidado, de manera profesional y creativa. Este trabajo de orientación y guía supone: amplia participación en la planificación y producción de materiales audiovisuales para adecuarlos a las necesidades de los grupos o de los individuos a que van destinados.

El uso de la tecnología para mejorar la comunicación obliga a cambiar los métodos rutinarios por otros más ágiles para alcanzar las metas educativas. La gran resistencia del personal docente a estos cambios se debe a que, el romper con la rutina, perturba una situación habitual. También, inexactamente se cree que los nuevos medios deshumanizan la enseñanza, desplazan a los profesores y que la educación en consecuencia se hará automática sin la calidad humana que le da la comunicación entre profesor y alumno.

En la actualidad, con el aumento de las disciplinas de estudio, del contenido de las mismas, la especialización y el aumento de la demanda profesional, lo que realmente sucede y de lo que generalmente se olvidan los profesores, es la verdadera humanización y el sello personal, esto sólo pueden conseguirse con el uso adecuado de los recursos tecnológicos, que liberan al profesor del trabajo rutinario y de la mera transmisión de información, para permitirle realizar el trabajo de orientador y guía en la formación de sus alumnos [URL8].

2.2. La mejora del proceso enseñanza-aprendizaje mediante la tecnología.

Los métodos didácticos requieren una planificación cuidadosa y detallada que generalmente se descuida en los procedimientos tradicionales. Ahora, debe atenderse no solamente a la asignatura, contenido y diferencias individuales de los alumnos sino a otros muchos factores que influyen para el éxito del proceso del aprendizaje. Reuniendo todos estos elementos, podemos desarrollar una

planificación o diseño sistemático de la enseñanza. Uno de los elementos indispensables del diseño, es la previsión, organización y producción de recursos didácticos entre ellos la creación de software educativo [URL9].

Los especialistas en el uso de los medios tecnológicos deberían trabajar con los profesores para ayudarles a desarrollar planes o diseños didácticos para sus clases diarias, para unidades de trabajo o para un curso completo. El éxito de estos diseños didácticos exige una planificación cuidadosa y se debe afrontar con realismo muchos problemas que deben ser resueltos. Esto no se logra por casualidad e improvisación; sino que supone un método riguroso que el profesor conoce aquello sobre lo que debe actuar, toma las decisiones oportunas y realiza sistemáticamente la acción.

Los medios tecnológicos que deberán usarse en el diseño didáctico son los que requieran los objetivos, el contenido y los métodos de aprendizaje. Los medios tecnológicos no son suplementarios a la enseñanza, ni su soporte: son el estímulo mismo. Bajo este concepto no puede aceptarse la concepción obsoleta de los medios tecnológicos como auxiliares o ayudas. Debe pues determinarse cuáles medios, cómo y cuándo van a proporcionar las experiencias más efectivas y eficaces para los alumnos.

Así como diferentes objetivos requieren diferentes clases de aprendizaje, así también los recursos para ser adecuados necesitan corresponder a las tareas requeridas. Ciertos medios tecnológicos pueden ser mejores que otros para ciertos propósitos (software educativo, aplicaciones multimedia; películas, Internet). En otros casos, uso del equipo disponible, conveniencia de los costos y otros muchos factores pueden ser los determinantes de la elección del medio tecnológico. Sin embargo, no existe alguna guía que permita determinar que medio tecnológico será útil para cada situación y dominio.

Este enfoque de la enseñanza y el aprendizaje se desarrolla con relación a los objetivos de comportamiento y para atender a las necesidades específicas de los alumnos. Querer producir localmente todos los materiales que se necesiten es imposible por lo costoso y por el tiempo que se pierde. En las nuevas tecnologías, son muy escasas todavía las disponibilidades de programas educativos que tengan la suficiente capacidad de abarcar la aparte didáctica y la parte técnica. Tal vez en el futuro los desarrolladores traten de elaborar materiales cuidadosamente planeados para que puedan tener la flexibilidad de adaptarse a diferentes temas generales, y con ligeras modificaciones en su uso a las circunstancias locales. Pero siempre quedará una puerta abierta para la producción propia de material complementario con propósitos y aplicaciones concretas.

Los medios tecnológicos son más que simples herramientas auxiliares del profesor, entre ellos se pueden mencionar los medios audiovisuales, los medios de comunicación en general y los que se basan en Tecnología Informática. Todos estos medios constituyen frecuentemente la vanguardia del progreso educativo y cuando se usan con propiedad afectan la estructura de planes y programas y lo que es más, llegan a modificar la estructura íntima del proceso de aprendizaje. En teoría, estos medios tecnológicos son esenciales para lograr una comunicación efectiva en la enseñanza a grandes grupos y son el único medio de comunicación didáctica en muchos programas para enseñanza individual o estudio independiente.

Por lo regular muchos de los profesores piensan que lo mejor es enemigo de lo bueno, por utilizar lo mejor, olvidamos lo bueno, es decir, por querer el último grito tecnológico no usamos didácticamente multitud de posibilidades que creativamente usadas pueden ser fuente idónea de aprendizaje.

Teniendo en cuenta todo lo anterior y aunado a que los costos económicos para la adquisición de equipo tecnológico se han limitado en nuestro país, además de que en la actualidad los países en vías de desarrollo, como México, deben estar actualizados en el uso de medios tecnológicos para

crear un nuevo proceso de enseñanza-aprendizaje de calidad; es necesario proponer herramientas cuyo objetivo sea el de mejorar la calidad de la información que se le proporciona al alumno, así como las condiciones de enseñanza del profesor. Pero el desarrollo de software educativo tiene que contemplar las carencias que se tienen en el proceso de enseñanza-aprendizaje tradicional, es por eso que para poder desarrollar un método que se aplique a la creación de software educativo se tiene que analizar las metodologías y procesos existentes, esto con el fin de cumplir con todos los requerimientos de un nuevo proceso.

Con el objetivo de proponer un método alternativo para el desarrollo de SE, a continuación se presentan un análisis sobre las principales metodologías encontradas en la literatura. El estudio comparativo aquí presentado se enfoca en efectuar un análisis empírico sobre los modelos y metodologías existentes, para formular así un método basado en el concepto de prácticas efectivas.

2.3. Ingeniería de Software Educativo con Modelado Orientado a Objetos

2.3.1. Propósito

La metodología propuesta por Galvis [Galvis93] propone cuatro fases, todas ellas destinadas al único objetivo de crear un micromundo de tipo juego para poder aprender; permitiendo así la generación de experiencias que sirven de base para que el aprendiz genere o se apropie de nuevo conocimiento, ya sea de forma algorítmica o heurística.

De acuerdo a Galvis "un micromundo es un ambiente de trabajo reducido tan simple o tan complejo como amerite lo que se aprende, donde suceden o pueden suceder cosas relevantes a lo que interesa aprender, dependiendo de lo que el usuario realice".

2.3.2. Enfoque

La propuesta sigue la metodología de Ingeniería de Software Educativo (ISE) y se complementa con el paradigma Orientado a Objetos (OO). Su objetivo es crear micromundos altamente interactivos tomando en cuenta el potencial tecnológico y los recursos disponibles en la actualidad, teniendo una sólida base educativa y de comunicación con el usuario.

2.3.3. Estructura

Las fases con las que cuenta la metodología propuesta por Galvis se muestran en la Figura 2.1 y son descritas brevemente a continuación:

2.3.3.1. Análisis

Su objetivo principal es determinar el contexto en el cual se va a crear la aplicación y derivar de allí los requerimientos que deberá atender la solución interactiva, como complemento a otras soluciones basadas en uso de otros medios (personales, impresos, audio-visuales, de experiencia), teniendo claro el rol de cada uno de los medios educativos seleccionados y la viabilidad de usarlos. Dicha fase debe presentar, al menos, información como la que se describe a continuación:

- Característica de la población y objetivos. Como la edad, sexo, características físicas y mentales, experiencias previas, expectativas, actitudes, aptitudes, intereses o motivadores por aprender.
- Conducta de entrada y campo vital. En donde se tiene el nivel escolar, desarrollo mental, físico o psicológico, entorno familiar y escolar, etc.

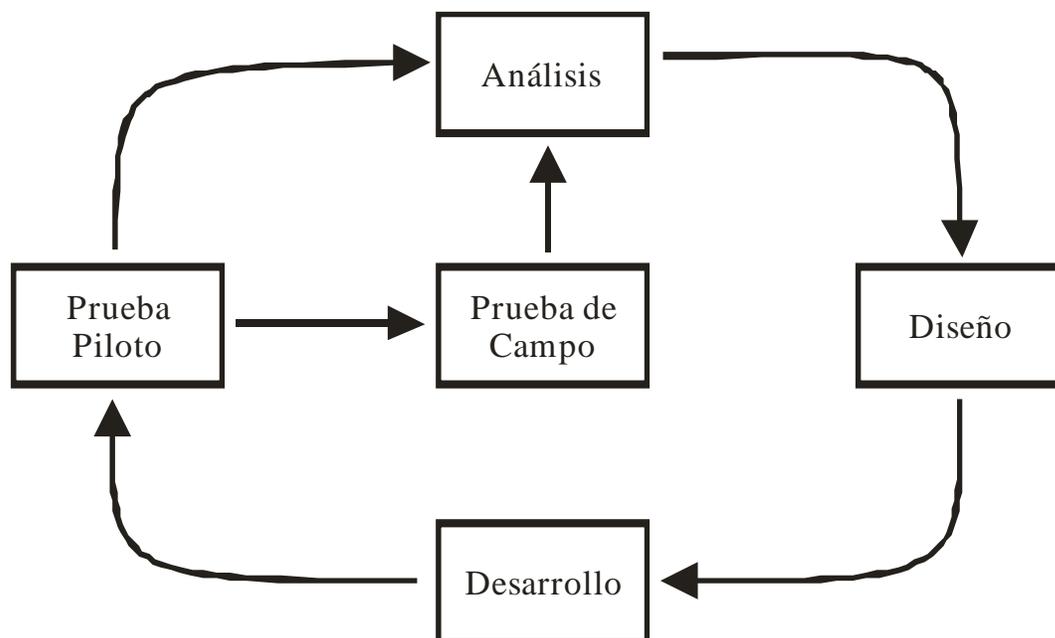


Figura 2.1. Metodología ISE propuesta por Galvis.

- Problema o necesidad a atender. Aquí se necesita recurrir a los mecanismos de análisis de necesidades educativas, como entrevistas, análisis de resultados académicos y demás. para detectar lo problemas o posibles necesidades que deben ser atendidas.
- Principios pedagógicos y didácticos aplicables. En esta parte se lleva a cabo el análisis del proceso de enseñanza-aprendizaje para establecer cómo debe enfocarse el ambiente, qué factores debe tomar en cuenta y qué objetivos debe cumplir.
- Justificación de uso de los medios interactivos. Es importante que el apoyo informático sea tomado en cuenta siempre y cuando no exista un mecanismo mejor para resolver el problema: soluciones administrativas, soluciones académicas, cambios en las metodologías de clase, mejoras a los medios y materiales de enseñanza contemplando el uso de los medios informáticos.

Al final de esta etapa se debe formular los requerimientos que deberá atender el material interactivo que se desea obtener. Esta especificación de requerimiento debe contener lo siguiente:

- Descripción de la aplicación. Es un documento en donde se tienen las características particulares del SE dentro del área de aplicación.
- Restricciones. Es un documento con las limitaciones que debe tener la aplicación, abarcando los aspectos tales como: población objetivo y sus características, áreas de contenido y sus características, principios pedagógicos aplicables, modos de uso de la aplicación, conductas de entrada y demás.
- Los escenarios de interacción. Son diagramas que permiten ver secuencias de interacción entre el usuario y la aplicación, es una forma de determinar la secuencia de acciones entre diferentes partes de la aplicación involucrada al llevar a cabo una determinada actividad. Con base en estos diagramas se pueden ver cuales pueden ser las necesidades de información en cada escenario de interacción y se puede ir pensando en cuales pueden ser los algoritmos que serán usados.

Una vez terminada la primera fase de análisis se procede a la fase de diseño que a continuación se describe.

2.3.3.2. *Diseño*

En esta etapa el autor (o programador) desarrolla el diseño del micromundo interactivo el cual se realiza en tres niveles diferentes: educativo, comunicacional y computacional. Al diseñar el ambiente en el que se desarrollará la acción se deben definir los elementos que se determinaron como necesarios en todos los micromundos interactivos y aquellos deseables que convenga para el caso. La identificación de estos elementos en esta etapa permite crear un vínculo eficiente con la etapa de desarrollo. A continuación se describe cada una de los 3 diferentes tipos de diseño:

- **Diseño educativo.** En esta parte se establece lo que hay que enseñar o reforzar para cubrir las necesidades encontradas con apoyo de los Materiales Educativos Computarizados (MEC). Como resultado de la fase de diseño educativo se debe tener lo siguiente: el contenido y su estructura; el micromundo; el sistema de motivación y el sistema de evaluación. De acuerdo con Galvis en esta etapa del diseño se debe de responder a las siguientes preguntas “¿Qué aprender con los MEC?, ¿En que micromundo aprenderlo?, ¿Cómo motivar y mantener motivados a los usuarios?, ¿Cómo saber que el aprendizaje se está logrando?” para comprender algunas de estas preguntas Galvis cita a Piaget diciendo que “Una de sus contribuciones principales es mostrar que la gente posee diferentes teorías acerca del mundo, los niños aprenden mejor cuando son alentados a apoyarse sobre su propia intuición y a emplear lo que ya saben para desarrollar sus ideas”. Para esta etapa se pueden utilizar diferentes tipos de evaluaciones como por ejemplo: Evaluación sumativa, averiguar cuánto logró el aprendiz; Evaluación diagnóstica, aplicada antes de apoyar la interacción con los MEC, para saber el punto de partida y Evaluación formativa, en donde se representan situaciones para ayudar a descubrir o practicar, transferir y afianzar destrezas, conceptos o habilidades.
- **Diseño comunicacional.** En esta fase del proceso de diseño se define la interfaz (forma de comunicación usuario-programa) de la aplicación. Es importante conseguir que la interfaz sea amigable, flexible y agradable de usar; también debe de ser consistente, y es necesario que sea altamente interactiva. Al diseñar la interfaz también se debe tener en cuenta las restricciones tecnológicas, características de la población y aspectos psicológicos de la percepción.
- **Diseño computacional.** Por último en esta etapa del diseño se tiene como resultado, cada una de las diferentes clases de objetos, incluyendo sus atributos (indicando si serán públicos o privados), el conjunto de métodos y la invariante de cada clase que corresponde al conjunto de restricciones o de requisitos que debe siempre cumplir una determinada clase. También se debe ilustrar la lógica acerca de cómo se desarrollan cada una de las actividades en el modelo, para ello se deben refinar los casos de uso. Para hacer esto se pueden usar diagramas de interacción que pueden ser de 2 tipos: diagramas de secuencia o diagramas de colaboración. En estos diagramas se puede ver la secuencia de mensajes entre los diferentes objetos involucrados en cada caso de uso y se pueden modelar todas las alternativas que puedan presentarse en cada caso.

Toda esta información ayuda a redefinir el modelo antes de iniciar la fase de desarrollo. Además permite validar si el modelo es completo y permite satisfacer todos los requerimientos de la aplicación. Una vez cumplidos los puntos del diseño se procede a la parte del desarrollo.

2.3.3.3. *Desarrollo*

Se implementa la aplicación usando toda la información obtenida anteriormente, se toma la definición de clases y se implementa en el lenguaje escogido tomando en cuenta las restricciones computacionales que se tengan. Se tiene que establecer la herramienta de desarrollo sobre la cual se va a implementar la aplicación.

Para seleccionar la herramienta de desarrollo Galvis propone tomar en cuenta algunos criterios como: costo, disponibilidad en el mercado, portabilidad de la aplicación desarrollada, facilidades al desarrollador (ambientes gráficos de desarrollo, mecanismos de depuración, manejo de versiones, etc.). Además en esta etapa se busca que el modelo del mundo sea independiente de la interfaz, con el objetivo de facilitar el trabajo y permitir trabajar en paralelo.

La interfaz se implementa utilizando la especificación que arrojó la etapa del diseño computacional. Además, esta etapa también cuenta con una parte en donde se desarrollan pruebas a lo largo y al final del desarrollo para depurar los componentes del modelo generado, haciendo validación de los prototipos con expertos durante la etapa de diseño y prueba cada uno de los módulos desarrollados.

2.3.3.4. *Prueba*

La etapa de prueba del sistema ya desarrollado se cumple poniendo a disposición una versión beta del micromundo interactivo. Esto se realiza con una muestra de la población, y se verifica que efectivamente la aplicación satisface las necesidades y cumple con la funcionalidad requerida.

2.4. Metodología para desarrollo de Software Educativo de Pere Marqués

2.4.1. Propósito

En 1995 Pere Marqués propone una metodología para facilitar el proceso de diseño y desarrollo de software educativo [Marques95], la cual contempla once etapas generales que son divididas en fases más específicas.

La importancia de la propuesta de Marqués se puede resumir con la siguiente cita: *“El proceso de elaboración del software educativo no es un proceso lineal, sino iterativo: en determinados momentos de la realización se comprueba el funcionamiento, el resultado, se evalúa el producto y frecuentemente se detecta la conveniencia de introducir cambios”*.

2.4.2. Enfoque

Contrario a la metodología anterior, la definición de las once etapas que propone Pere, se orienta más a la parte pedagógica y deja un poco de lado la parte técnica que corresponde a la implementación del software.

2.4.3. Estructura

La estructura de la metodología de Pere se describe a continuación:

2.4.3.1. *Génesis de la idea*

La elaboración de un software educativo siempre parte de una idea inicial que parece potencialmente poderosa para favorecer los procesos de enseñanza-aprendizaje y que toma forma poco a poco. Dicha idea tiene principalmente como autores a los profesores, pedagogos y diseñadores de software educativo.

La idea principal de un programa constituye una forma global de lo que se quiere crear, tratándose de responder a dos preguntas importantes: el QUÉ (materia y nivel en que se quiere trabajar) y el CÓMO (estrategia didáctica) y poco a poco se completa y se concreta hasta llegar al primer diseño del programa: el *diseño funcional*.

De acuerdo a Pere, existen dos formas para realizar la creación de la idea: por libre iniciativa, en donde las ideas son generadas por los profesores, pedagogos o diseñadores de software educativo y pueden surgir por diferentes circunstancias (por ejemplo, hablando con los alumnos de sus problemas en la escuela, comentando con otros profesores experiencias educativas, buscando nuevas formas de ejercitar técnicas que exigen mucha práctica, detectando deficiencias del sistema, visualizando programas educativos o utilizando otros medios didácticos). Y la segunda forma es por encargo, en donde las ideas son generadas por una editorial de software educativo o de una administración pública, los clientes que hacen el encargo acostumbran proporcionar a los diseñadores un marco, unas especificaciones centradas en aspectos pedagógicos y político-comerciales, que la idea resultante deberá respetar.

Alguna práctica que se realiza para la generación de las ideas es formar dos listas de objetivos curriculares: una con objetivos que se piensen alcanzar más fácilmente mediante el uso de determinados programas didácticos conocidos, y otra con objetivos que consideren que podrían conseguirse mejor si existiesen determinados programas que, en este caso, se deberán inventar.

2.4.3.2. Prediseño o diseño funcional

Una vez concebida la idea, la metodología procede a realizar un diseño funcional, el cual constituye el primer guión del programa que pondrá el énfasis en los aspectos pedagógicos del proyecto tales como: contenidos, objetivos, estrategia didáctica, y demás. Si el software educativo se elabora por encargo, este primer diseño servirá para presentarlo al jefe del proyecto y a los clientes para que se someta a una prueba de aceptación y determinen su conformidad o inconformidad con el diseño.

Este diseño se puede distribuir a los profesores para que aporten su opinión y sus sugerencias. En el diseño funcional resulta recomendable que intervenga un equipo de especialistas, el equipo de diseñadores pedagógicos, que estará integrado por profesores con amplia experiencia didáctica en el tema en cuestión y que puedan proporcionar conocimientos sobre la materia, pedagogos o psicopedagogos, que proporcionen instrumentos de análisis y de diseño pedagógico, y algún especialista en tecnología educativa.

En la elaboración del diseño funcional se pueden utilizar algunas técnicas para el desarrollo de la creatividad, como por ejemplo la técnica del “*brainstorming*”, bibliografía sobre diseño de software educativo, bibliografía sobre el tema en específico que se piensa tratar en el software, software educativo cercano al que se quiere hacer, plantillas de diseño que ayudaran en el proceso para concretar el proyecto. Al final se debe tener una presentación la cual consiste en desarrollar una exposición general del programa que se piensa desarrollar tomando en cuenta algunos aspectos como, la descripción sintética del programa y de sus objetivos, rasgos más característicos, motivación, guión general, hardware y software necesarios para la realización del proyecto. Esta presentación del prediseño debe proporcionar a los lectores una primera idea global del material que se pretende elaborar.

También se debe tener la concertación de los aspectos pedagógicos en donde se definen los objetivos educativos como por ejemplo, evaluar la eficacia del programa y racionalizar la organización de los contenidos. También se definen los alumnos a los que está destinado el programa y se evalúan características como edad, nivel de desarrollo cognoscitivo (nivel de

madurez), conocimientos previos y capacidades generales que ha de tener: nivel educativo, conocimientos relacionados con la temática del programa, estructura cognoscitiva, capacidad intelectual (nivel de inteligencia general y factorial), actitudes, intereses, hábitos de estudio y organización y discapacidades o deficiencias; y por último las estrategias didácticas que se piensan utilizar, debe comprender aspectos como las actividades que hay que proponer a los alumnos, el tratamiento de los errores, los elementos motivadores, los posibles caminos pedagógicos, etc.

También se deben tomar en cuenta los contenidos organizándolos de forma que vayan de los aspectos más fáciles y concretos a los más complejos y abstractos, de los elementos conocidos por los alumnos a los que les son desconocidos, de las presentaciones globales o sintéticas a las visiones analíticas, de las visiones episódicas a las sistemáticas, de los que requieren el uso de habilidades globales a los que implican el uso de habilidades específicas. Tomando en cuenta las relaciones interdisciplinarias y contemplando niveles de dificultad.

Dependiendo del tipo de software educativo se deben contemplar otros aspectos, por ejemplo, si es un programa de tipo base de datos: incluir la estructura de las bases de datos, las interrelaciones entre ellas, las formas de acceso a los datos (búsqueda, ordenación, clasificación, captura...), si es un simulador: los modelos que presentará y la organización de los conceptos (que deberán resultar claros y adecuados al nivel de abstracción de los alumnos), las variables con que se trabajarán y las interrelaciones entre las variables que se podrán representar interrumpidamente por medio de fórmulas, con tablas de comportamiento, mediante grafos, etc., si es un constructor: los elementos que contemplará y las propiedades o los comportamientos que tendrá cada uno.

También se deben tener en cuenta las actividades mentales que los alumnos desarrollarán delante de la computadora. La identificación previa de estas operaciones mentales que interesa que realicen los alumnos contribuirá a aumentar la calidad didáctica de las actividades interactivas que se diseñen. Algunas actividades mentales que los alumnos pueden desarrollar al interactuar con los programas, pueden ser el ejercitar habilidades psicomotrices, observar, percibir el espacio y el tiempo y orientarse en ellos, reconocer, identificar, señalar, recordar, explicar, describir, reconstruir, memorizar (hechos, datos, conceptos, teorías...), comparar, discriminar, clasificar, conceptualizar (conceptos concretos y abstractos), manipular conceptos, relacionar, ordenar, comprender, interpretar, representar, traducir, transformar, hacer cálculos mecánicos, resolver problemas de rutina, aplicar reglas, leyes, procedimientos, métodos, inferir, prever, buscar selectivamente información, sintetizar, globalizar, resumir, analizar (pensamiento analítico), elaborar hipótesis, deducir (razonamiento deductivo), inducir, generalizar, razonar lógicamente, estructurar, analizar la información críticamente, evaluar, experimentar (mediante prueba y error), construir, crear (expresión creativa, pensamiento divergente), transformar, imaginar (asociaciones, cambios de entorno), expresar, comunicar, exponer estructuralmente, negociar, discutir, decidir, resolver problemas inéditos, que implican la comprensión de nuevas situaciones, planificar proyectos, seleccionar métodos de trabajo, organizar, investigar, desarrollar, evaluar necesidades, procesos y resultados, reflexionar sobre los mismos procesos mentales (metacognición), intuir.

Se deben tener en cuenta algunas actividades interactivas que debe proponer el programa mediante las cuales se realiza el intercambio de informaciones entre los alumnos y la máquina que permite que las acciones de los estudiantes puedan ser valoradas y tratadas por el programa. Considerando aspectos como la naturaleza de las actividades educativas, estructura, acciones y respuestas permitidas al alumno, duración, tipo de control de la situación de aprendizaje que tendrá el alumno.

Se deben contemplar diferentes caminos pedagógicos ya que el software tiene que prever opciones que permitan seguir diferentes caminos pedagógicos a los alumnos y que faciliten: la

elección de los temas y de las actividades, la reformulación de los conceptos, el cambio de la secuencia de los contenidos, el repaso de los puntos mal comprendidos, la selección del nivel de dificultad, repasar, profundizar, ver ejemplos, etc. La determinación de estos recorridos se puede determinar de dos maneras, primeramente de manera explícita, es decir por decisión de los alumnos y la segunda de manera implícita, es decir en función de las respuestas de los alumnos.

Se deben detectar los errores y se deben tener cuenta los tipos de tratamiento como: tratar los errores según el tipo de refuerzo o de corrección, según la valoración que haga el error, según la naturaleza del error, ya que pueden existir diferentes tipos de errores como errores de conocimiento, de comprensión, de análisis, de procedimiento y errores de ejecución y para cada uno de ellos corresponderá su acción respectiva.

Pero considera que es necesario contemplar elementos motivadores ya que la motivación es uno de los grandes motores del aprendizaje y un buen antídoto contra el fracaso escolar. Los elementos motivadores más utilizados en los programas didácticos son: elementos que presentan un reto, elementos que estimulan la curiosidad o la fantasía, elementos que representan un estímulo o una penalización social y el ritmo variado y progresivo del programa.

Los aspectos algorítmicos y estructurales del software reflejan una primera aproximación a la estructura del programa, y se concretan en diversos gráficos y diagramas comentados como por ejemplo un diagrama general del programa en donde se reproduce la estructura básica del algoritmo por lo regular se representa en forma de diagrama de flujo. Y se representan todos sus módulos entre ellos el módulo de presentación y gestión de menús, el módulo de actividades interactivas, el módulo de ayuda, el módulo de evaluación y los diferentes módulos auxiliares.

La organización de los menús determina cómo se presentan los menús, por ejemplo, según su entorno tradicional o según el entorno Windows®. Contemplar también los parámetros de configuración del programa tales como conectar o desconectar los efectos sonoros, cambiar el color de algunos elementos en la pantalla, ajustar el tiempo de respuesta y fijar el nivel de dificultad de las actividades.

Contempla también la definición de las formas de interacción entre los alumnos y el programa también conocido como el entorno de comunicación, este deberá ser lo más ergonómico posible, aquí se realiza el diálogo entre los estudiantes y el programa. Para llevar a cabo este punto se consideran tres características, la primera es el diseño de las pantallas, aquí se determinarán zonas que realizarán funciones específicas y que se repetirán (si es posible) en todas las pantallas del programa. Por ejemplo, zona de comentarios, zona de órdenes, caja de herramientas, zona de trabajo, la segunda es el uso del teclado y del ratón, que es importante ya que se interesa crear un entorno de comunicación con el programa que resulte muy fácil de usar y agradable para el alumno. Se debe establecer una sintaxis sencilla e intuitiva y prever un sistema de ayuda para el manejo del programa. Algunas de las acciones y respuestas que el alumno puede llegar a tomar se pueden clasificar en tres acciones, la primera es por selección de entre las opciones que ofrece el programa por la pantalla, por ejemplo: preguntas del tipo si/no, cuestionarios de respuesta múltiple, menús de opciones. La segunda es con producción de respuesta por ejemplo: mover algún elemento por la pantalla, cambiar un objeto de lugar, trazar una trayectoria, establecer correspondencias entre listas, asociaciones, ordenar palabras, y demás; y por último la acción de elaborar una respuesta libre. La tercera es contemplar otros periféricos, en donde se describe la función de los diferentes periféricos complementarios que se utilicen como impresora, teclado conceptual, lector de tarjetas, micrófono, reconocedor de voz, video, CD-ROM, lápiz óptico, pantalla táctil, MODEM, convertidores analógico-digitales, etc.

Para finalizar se tiene también un primer guión sobre la documentación del programa, que es en donde se incluye un esquema con una primera aproximación al formato y al contenido de la documentación que acompañará al programa, en donde se debe incluir una ficha resumen que recoge las principales características del programa. Un manual de usuario, donde se explica todo lo que necesita saber el usuario del programa para utilizarlo sin problemas y una guía práctica, que es un documento dirigido al personal docente en donde se ofrecen sugerencias sobre la integración curricular del programa, sus formas de uso, actividades complementarias, estrategias para evaluar el rendimiento de las situaciones educativas que genera el programa, etc.

Las siguientes etapas de la metodología de Pere se consideran parte de la descripción técnica del desarrollo del software y son el estudio de viabilidad y el marco del proyecto, dossier completo de diseño o diseño orgánico, programación y elaboración del prototipo alfa-test, redacción de la documentación del programa, evaluación interna, ajustes y elaboración de la versión 1.0 y por último la publicación y mantenimiento del producto.

2.5. Metodología Extendida para la Creación de Software Educativo desde una Visión Integradora

2.5.1. Propósito

La metodología que propone Cataldi [Cataldi03], está basada en dos partes principales, que son las teorías del aprendizaje y las metodologías propias de la ingeniería del software a las que Cataldi denomina *Ingeniería de Software Educativo*. Ambas partes se articulan en “*la matriz de actividades del modelo del ciclo de vida elegido*”, que desde nuestro punto de vista fue el comienzo de la definición de una metodología que unió los aspectos técnicos y pedagógicos, sin embargo no abarcó por completo ambos aspectos.

Después de analizar las dos partes principales, Cataldi describe una problemática de la siguiente manera “*las metodologías propias de la ingeniería del software no se preocupan de los aspectos pedagógicos-didácticos del producto software educativo a desarrollar*”, y propone como solución a este problema la formulación de la extensión de una metodología para que estos aspectos sean contemplados; punto de partida de esta propuesta de tesis. Para Cataldi, esta extensión se logró redefiniendo la matriz de actividades correspondientes al ciclo de vida del software elegido, siendo éste el documento en el que quedarían definidos todos los procesos y las actividades a efectuar en cada etapa del desarrollo del software.

2.5.2. Enfoque

Para desarrollar su solución Cataldi exploró los diferentes ciclos de vida utilizados con mayor frecuencia en el marco de las metodologías de ingeniería de software, tales como el modelo en cascada, el modelo incremental o de refinamientos sucesivos, el prototipo evolutivo, el modelo en espiral y los modelos orientados a objetos. Cataldi eligió el modelo de prototipo evolutivo en base a tres razones:

- Cuando se trata del desarrollo de un software por encargo, es deseable obtener un primer esbozo de lo que será el programa tan pronto como sea posible a fin de satisfacer la curiosidad del usuario y para saber si en realidad es lo que se quiere.
- Es necesario saber lo antes posible si los desarrolladores han interpretado correctamente las especificaciones y las necesidades del usuario.

- Y por último los usuarios no tienen una idea acabada de lo que desean, por lo tanto los desarrolladores deben tomar decisiones y suponer que es lo que el usuario quiere. Por ello la emisión de los prototipos brinda la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado.

La decisión del modelo que Cataldi toma, se apoya en la ventaja de la realización de los cambios en etapas tempranas y la posibilidad de emisión de varios prototipos evaluables durante el desarrollo, él obtiene de este modo paralelamente una metodología integral también para el proceso de evaluación del programa. Aspecto que hasta el surgimiento de esta metodología no había sido considerado de manera explícita.

2.5.3. Estructura

La estructura de la metodología propuesta por Cataldi se descompone en doce fases las cuales contemplan cada una de las actividades que se deberán implementar para el desarrollo de un software educativo:

- Factibilidad (FAC)
- Definición de requisitos del sistema (RES)
- Especificación de los requisitos (REP)
- Diseño del prototipo (DPR)
- Diseño detallado del prototipo (DDP)
- Desarrollo del prototipo (codificación) (DEP)
- Implementación y prueba del prototipo (IPP)
- Refinamiento Iterativo de las especificaciones del prototipo (RIT)
- Diseño del sistema final (DSF)
- Implementación del sistema (ISF)
- Operación y mantenimiento (OPM)
- Retiro (RET)

Una vez definidas las actividades correspondientes a estas etapas se procede a la construcción del documento denominado matriz de actividades, donde se definen las actividades a desarrollar en cada fase, y los procesos asociados a cada una de ellas. Cataldi identificó que las diez primeras fases debían extenderse. Por lo tanto, se incorporaron procesos nuevos que contemplaron las necesidades pedagógicas-didácticas y se definieron nuevas actividades que cubrieron cuestiones pedagógicas-didácticas en los procesos existentes.

Los procesos nuevos que se incorporaron así como sus actividades se describen a continuación:

- El proceso de identificación de la necesidad educativa. Corresponde a identificar la necesidad del programa educativo y seleccionar la teoría educativa a utilizar.
- El proceso de análisis de los requisitos educativos. Donde se definen los objetivos educativos, las características del grupo destinatario, los contenidos y el recorte de contenidos, las estrategias didácticas, las actividades mentales a desarrollar, el nivel de

integración curricular, el tipo de uso del programa y el nivel de interactividad, los efectos motivadores, los posibles caminos pedagógicos, el tiempo y modo de uso del programa y el hardware asociado.

- El proceso de evaluación de los prototipos de software. Este proceso realiza el instrumento de evaluación, se evalúan los prototipos del programa, se elaboran los resultados, se identifican cambios y ajustes a realizar, se llevan a cabo las modificaciones pertinentes y se archivan los resultados.
- El proceso de evaluación interna y externa del software. Donde se crea el instrumento de evaluación, se evalúa interna y externamente el programa, se elaboran los resultados y se identifican los cambios y ajustes a realizar, se llevan a cabo las modificaciones pertinentes y se archivan los resultados.
- El proceso de evaluación contextualizada. Donde se diseña la evaluación, por ejemplo se definen los grupos de control y experimental, el grupo docente, el tiempo y el modo. Se aplican las pruebas, se identifican los posibles problemas y se realizan las posibles modificaciones y ajustes de la versión.
- El proceso de documentación didáctica. Donde se planifica la documentación didáctica, se elabora la guía didáctica, se agrega la información didáctica pertinente, se produce la documentación y se adjunta al programa.

Cataldi agregó nuevas actividades que se incorporan a los procesos existentes:

- El proceso de selección del modelo del ciclo de vida. Se selecciona un modelo de ciclo de vida acorde con la teoría educativa elegida.
- El proceso de iniciación, planificación y estimación del proyecto. Se establece la matriz de actividades considerando la teoría educativa elegida.
- El proceso de exploración de conceptos. Se identifican las necesidades educativas, se formulan posibles soluciones potenciales y se formulan soluciones potenciales compatibles.
- El proceso de asignación del sistema. Se definen las funcionalidades del programa y se desarrolla la arquitectura del programa en base a la teoría educativa elegida.
- El proceso de análisis de los requisitos de software. Se define el tipo de programa a desarrollar, se define el tipo de interactividad y se integran requisitos educativos y de software.
- El proceso de diseño. Se definen la organización de los menús y los tipos de iconos a usar, se seleccionan los efectos y textos a usar, se asegura la facilidad de la lectura, se realizan los diseños de las pantallas y el diseño de los menús, se realizan los guiones técnicos (*storyboard*), se definen los criterios de navegación, se definen las actividades, el tipo de módulos y las ayudas didácticas.
- El proceso de documentación técnica. Se incluyen los resultados de las evoluciones.

Desde el punto de vista metodológico, la definición de Cataldi para cada uno de los pasos a seguir determinando las herramientas a usar, significa disponer de una manera clara y precisa de una hoja de ruta para continuar con el proyecto. Las Tablas 2, 3 y 4 presentan un listado de los todos los procesos que se toman en cuenta en la metodología de Cataldi.

Por otro lado, al completar la matriz para desarrollar el software experimental, Cataldi recomienda que se deben tener en cuenta tres puntos críticos, el primero es la necesidad de usar un lenguaje común entre los miembros del equipo de desarrollo; esto se debe a las diferentes disciplinas que incluyen profesionales de diferentes ramas de las ciencias de la computación y de las ciencias de la educación. El segundo es definir el significado de interacción con el software y el establecimiento de diferentes niveles de interacción cuando sea requerido; y por último, abarcar el concepto de calidad, ya que los programas educativos tienen características muy específicas que deben ser tomadas en cuenta, entre otras cosas, las necesidades del grupo destinatario y los objetivos educativos. Esta calidad esta relacionada con su grado de adaptación a un contexto en particular donde convergen una serie de variables, tales como las características curriculares, las edades del grupo destinatario, el estilo de aprendizaje y de enseñanza que se requiere para un análisis correcto.

Cataldi nos deja una aportación invaluable: *“Un software educativo debe permitir evaluar cuatro aspectos básicos: los aspectos educativos pedagógicos y didácticos, aspectos técnicos-económicos, aspectos comunicacionales y organizativos-estructurales del programa”*.

Tabla 2. Procesos a tomar en cuenta dentro del uso de la metodología propuesta por Cataldi.

Procesos	Documento de salida	Métodos / Técnicas / Herramientas a emplear
Proceso de identificación de la necesidad educativa	Definición del marco educativo y comunicacional	Encuesta, entrevista
Proceso de selección del modelo de ciclo de vida	Ciclo de vida adoptado	--
Proceso de iniciación, planificación y estimación del proyecto	Plan de gestión del proyecto	Diagrama de Gantt o CPM (método del camino crítico) Modelos empíricos de estimación
Proceso de seguimiento y control del proyecto (programa)	Análisis de riesgos y plan de contingencias Registro histórico del proyecto	Modelación, Prototipos Revisiones Auditorias Análisis CPM
Proceso de gestión de calidad del software	Plan de garantía de calidad Recomendaciones de mejora de calidad	Técnicas de planificación Métricas de calidad del software
Proceso de exploración de conceptos	Informe de necesidades Posibles soluciones factibles	Análisis Costo-Beneficio DFD, Prototipos
Proceso de asignación del programa (sistema)	Especificación de requisitos funcionales de hardware y software Especificación de interfaces del sistema o programa. Descripción funcional Arquitectura	DFD Módulos
Proceso de análisis de los requisitos educativos	Especificación de los objetivos y estructuración de conceptos Selección de contenidos y pertinencia	Enfoques cognitivistas Enfoques constructivistas Estrategias cognitivas

Tabla 3. Procesos a tomar en cuenta dentro del uso de la metodología propuesta por Cataldi (continuación).

Procesos	Documento de salida	Métodos / Técnicas / Herramientas a emplear
Proceso de análisis de los requisitos software	Especificación de requisitos del software, de interfaces de usuario y otros tipos de software. Interfaz de hardware con el sistema físico	Análisis estructurado DFD, Diagramas E/R Técnicas de prototipado
Proceso de diseño de los contenidos	Identificación de los procesos mentales a estimular Definición de las actividades a realizar por los alumnos Jerarquización de los conceptos	Uso de estrategias cognitivas Teoría de Ausubel y Novak ¹ Uso de mapas conceptuales
Proceso de diseño del software	Descripción del diseño del software y de la arquitectura Descripción del flujo de información, bases, interfases y algoritmos	Programación estructurada Programación Orientada a Objetos Técnicas de prototipado
Proceso de implementación e integración de los módulos	Datos para las pruebas Documentos del sistema o programa y del usuario. Plan de integración	Lenguajes de programación
Proceso de instalación	Plan de instalación Informe de instalación	Lenguajes de programación
Proceso de operación y soporte	Histórico de pedidos de soporte	Análisis estadístico
Proceso de mantenimiento	Recomendaciones de mantenimiento	Reaplicar el ciclo de vida
Proceso de retiro	Plan de retiro	- -
Proceso de verificación y validación	Plan de verificación y validación Plan de pruebas. Especificación y resumen de la prueba Software probado	Pruebas de caja negra y pruebas de caja blanca
Proceso de evaluación de los prototipos software	Diseño del instrumento de evaluación. Resumen de la prueba. Selección de la muestra	Cuestionario estructurado, semi y abierto
Proceso de evaluación interna y externa del software	Diseño del instrumento de evaluación Resumen de la prueba Selección de la muestra	Cuestionario estructurado, semi y abierto

¹ Para mayor información consultar: Ausubel, D. D P, Novak, J. D. & Hanesian, H. Psicología Educativa, Ed. Trillas, México,1983.

Tabla 4. Procesos a tomar en cuenta dentro del uso de la metodología propuesta por Cataldi (continuación).

Procesos	Documento de salida	Métodos / Técnicas / Herramientas a emplear
Proceso de evaluación contextualizada	Diseño de la experiencia. Definición de los grupos de control y experimental	Técnicas de análisis pre-post Test de Raven ² Prueba de Wilcoxon ³
Proceso de configuración	Plan de gestión de la configuración	Base de datos Diagramas de Gantt
Proceso de documentación técnica	Plan de documentación técnica	
Proceso de la documentación didáctica	Plan de confección de la documentación didáctica	
Proceso de formación y capacitación del personal	Plan de formación y capacitación	

2.6. Metodología Dinámica para el Desarrollo de Software Educativo

2.6.1. Propósito

La metodología propuesta por Arias [Arias02] tiene como propósito establecer una metodología dinámica que sirva de modelo para el desarrollo de SE.

2.6.2. Enfoque

La metodología se basa en la necesidad de concebir el medio de instrucción, es decir, la computadora, como un medio dinámico, aprovechando la potencia de diseño a través del desarrollo multimedia.

2.6.3. Estructura

La metodología esta compuesta por cuatro fases (Diseño educativo, Producción, Realización e Implementación) y un eje transversal que es la Evaluación. La metodología tiene como característica que no se requiere la culminación de una fase para pasar a la otra, es posible obtener rápidamente un prototipo que permita hacer validaciones parciales y proporcionar las correcciones requeridas.

2.6.3.1. Diseño Educativo.

Esta etapa se comprende de once subetapas, cada una de estas desempeña una función específica que a continuación se describe:

- El estudio de las necesidades. Este estudio especifica una situación de aprendizaje determinada. Si se habla de una situación de aprendizaje es fácil determinar las necesidades,

² Para mayor información consultar: Raven, J. C., Court, J.H. & Raven, J. Test de Matrices Progresivas. Escalas Coloreada, General y Avanzada, Ed. Paidos iberica, 1993.

³ Para mayor información consultar: Wilcoxon, F. Individual Comparisons by Ranking Methods. Biometrics Bull 1: 80-83. 1995.

tales como: tiempo a emplear en una actividad o clase, mucho contenido, poco contenido, muchos alumnos, generar actividades de refuerzo, etc.

- La descripción del aprendiz. Es necesario saber cual es la potencia de audiencia para seleccionar aspectos relacionados con la cultura, costumbres, edades, estilos de aprendizaje, etc.
- Propósito y objetivos referidos al proyecto. Se define lo que se quiere hacer desde el punto de vista del medio y para qué se quiere hacer.
- Formulación de objetivos terminales de aprendizaje. Se redactan los objetivos generales y específicos que se quieran alcanzar con el uso del material.
- Análisis estructural. Se especifican las sub-habilidades a desarrollar, tomando en cuenta los atributos básicos de los conceptos que se quieran trabajar.
- Especificación de los conocimientos previos. Las competencias, habilidades y destrezas que debe tener el usuario son los que finalmente van a determinar el éxito o no del material educativo computarizado.
- Formulación de objetivos específicos. Dichos objetivos deben ser lo más simples posible.
- Selección de estrategias del medio de instrucción. Se definen los eventos de aprendizaje que el diseñador ha considerado para lograr los objetivos propuestos. Cuando se diseñan las estrategias del medio de instrucción, el diseñador tiene que pensar que está desarrollándolas para implementarlas en un medio que no es estático, sino dinámico.
- Contenido. Se selecciona y organiza con cuidado el contenido temático que se desea. Se hace una lista de temas o puntos de interés.
- Selección de estrategias de evaluación. Se refieren a la selección y/o diseño de estrategias de evaluación de los aprendizajes.
- Determinación de variables técnicas. Se especifican aspectos relacionados con metáforas, principio de orientación, uso de íconos, botones, fondos, textos, planos, sonidos, videos, animaciones, simulaciones, etc. Una vez terminada esta etapa se procede a la etapa de producción del software.

2.6.3.2. Producción

Esta etapa se encuentra dividida en tres subetapas, la primera corresponde al guión de contenido, el cual es un esquema de la descripción del usuario en el cual se anota el propósito, se señala el tema y los objetivos específicos de aprendizaje, se decide cuál es la línea de producción, se establece el esquema de navegación y se realiza la página Web o diagrama de contenido. La segunda subetapa es el desarrollo del guión didáctico el cual es redactado con un lenguaje sencillo y claro. Se utiliza un vocabulario que sea familiar para el usuario. Se presenta el contenido ya desarrollado utilizando como soporte las estrategias elaboradas -puede ser asociado a un guión literario. Y por último se encuentra el *storyboard*, el cual es el resultado de la visualización del guión didáctico o libreto y se nutre de la determinación de las variables técnicas especificadas en la fase anterior. Es importante tomar en cuenta las teorías referidas a la percepción, la importancia del uso del color, sonido, las zonas de comunicación en pantalla, etc.

2.6.3.3. Realización

Esta etapa corresponde a la etapa de realización donde se plasman las ideas y requisitos que se determinaron en las dos etapas anteriores. Esta nueva etapa se encuentra dividida en dos subetapas; la primera corresponde al desarrollo del prototipo, donde el primer prototipo es el *storyboard*, luego, a partir de este, se diseña cada una de las pantallas que conformarán el material educativo computarizado. Se dispondrá de una red de pantallas que permitan verificar si el producto tiene sentido para satisfacer la necesidad educativa. La segunda etapa corresponde a la corrección del prototipo, aquí se debe dejar abierta la posibilidad de realizar ajustes y revisiones a forma de ir logrando, por aproximaciones sucesivas, mejoras hasta obtener el producto deseado.

2.6.3.4. Implementación

Esta etapa corresponde a la etapa en donde se desarrolla y se implementa en código el software definido. Una vez que se dispone de un diseño debidamente documentado se lleva a cabo el diseño computacional en el cual se especifica el tipo de software y hardware a emplear.

2.6.3.5. Eje Transversal de Evaluación

Esta etapa es ejecutada constantemente y corresponde a la evaluación. Existe una evaluación continua independientemente de la fase, esta evaluación se hace en función de los resultados que se van obteniendo durante todo el proceso.

Arias menciona que “en el desarrollo de materiales educativos computarizados, es importante, el tiempo y el esfuerzo dedicado en el desarrollo del diseño instruccional. El plan instruccional (diseño educativo) representa la base que orientará la calidad del programa educativo a crear. Contar con una metodología que permita realizar ajustes permanentes durante todo el proceso de desarrollo del software educativo, facilita la actividad de desarrollo de la herramienta”.

2.7. Propuestas de Sistemas de Educación basados en WEB

En el campo de los SEBW se muestran diferentes líneas de investigación y desarrollo que trabajan con diferentes componentes que especifican al sistema, por ejemplo:

- Componentes que utilizan el contenido de materiales.
- Componentes que aplican las prácticas cognitivas y pedagógicas, también las dedicadas a la enseñanza colaborativa.
- El uso de facilitadores de conocimiento.
- Componentes que utilizan el “secuenciamiento” [Peredo03].
- Componentes que utilizan el desarrollo de arquitecturas tipo Sistemas Multi-Agente.
- El uso de las Ontologías [Ayala04].

2.7.1. Propósito

Existen diferentes propósitos de los SEBW dependiendo de los componentes utilizados en el desarrollo del sistema, por ejemplo:

- Los que manejan el contenido de los materiales tienen como propósito la globalización de los materiales para emplearse en diversos programas y sesiones de aprendizaje

proporcionados por los autores y por aquellos otros interesados en reusarlos para sus propios fines, como es el Modelo de Objetos de Aprendizaje (LOM, *Learning Object Model*) concebido por IEEE-LTSC [URL13].

- Los que manejan las prácticas cognitivas y pedagógicas tienen como propósito explorar diversas teorías y métodos que implementan habilidades adaptativas en los sistemas las cuales habilitan la supervisión y valoración del proceso de aprendizaje y alcanzar así una mejora en la enseñanza personalizada, tal y como la propuesta de Héctor García [Garcia03].
- Las aplicaciones dedicadas a la enseñanza colaborativa tienen el propósito de impulsar la enseñanza personalizada del educando con el concurso de sus compañeros en el logro de un aprendizaje común, como la propuesta de Toshio Okamoto [Okamoto03].
- Los que manejan el uso de facilitadores de conocimiento tienen como propósito personalizar el proceso de enseñanza-aprendizaje de cada integrante del equipo, como es la propuesta de Gerardo Ayala y Yoneo Yano [Ayala98].
- La línea que maneja el secuenciamiento tiene como propósito el implementar ambientes inteligentes en la planeación y control de experiencias de aprendizaje, como la iniciativa de Leonid Sheremetov [Sheremetov02].
- La línea que aplica el desarrollo de arquitecturas de tipo Sistemas Multi-Agente tiene como propósito incorporar las herramientas de la Inteligencia Artificial Distribuida (IAD), como por ejemplo la propuesta de Patricia A. Jaques [Jaques03].
- La línea que utiliza el uso de las Ontologías tiene como propósito el proporcionar un medio para compartir y rehusar los recursos de contenido al proveer un vocabulario asociado a un conjunto de elementos de conocimiento relacionados semánticamente con el cual se establece el significado de los términos más importantes de un dominio específico, tal y como se percibe en los proyectos de Ayala y Sossa [Ayala04].

2.7.2. Enfoque

Actualmente la nueva generación de los SEBW se caracteriza por el diseño de sistemas organizados en varios niveles funcionales de servicio, el trabajo cooperativo dedicado a compartir acervos de contenido de los materiales, la educación orientada a través del paradigma de aprendizaje adaptativo, la construcción colaborativa de conocimiento entre los educandos, la implementación de ambientes inteligentes en la planeación y control de aprendizaje, el aprovechamiento de las aplicaciones disponibles y el material publicado en Internet mediante servicios Web, el desarrollo de la Web semántica, la aplicación de herramientas de la IAD por medio del uso de Agentes y Ontologías [Ayala04].

2.7.3. Estructura

En la estructura típica de los componentes que generalmente emplean los SEBW se presenta un patrón de módulos y funciones que bien pueden ser representados a través de la Arquitectura de Sistemas de Tecnologías de Aprendizaje propuesta por IEEE-LTSC [URL11] la cual se divide en cinco niveles especializados (ver Figura 2.2). Otro modelo es el empleado por la Iniciativa de Conocimiento Abierto (OKI, *Open Knowledge Initiative*) [URL12] integrada por universidades como Stanford y el Instituto Tecnológico de Massachussets. Este modelo está inspirado en el concepto multicapas basado en Interfaz de Programación de Aplicaciones (API, *Application Programming Interface*). También se puede ver el esquema diseñado por la Alianza para la Autoría

y Distribución de Instrucción Remota vía Redes para Europa (ARIADNE) el cual emplea herramientas especializadas en la simulación de sistemas, generación de cuestionarios, producción de formas de evaluación, edición de multimedia, aplicación de métodos pedagógicos, manejo de material de aprendizaje e interfaz del usuario organizadas en una arquitectura de cuatro capas.

También se tienen estructuras en lo que respecta a la presentación del material de aprendizaje, en donde sobresale la Estructura de Contenido perteneciente al Sharable Content Object Reference Model (SCORM) propuesta por la Iniciativa de Aprendizaje Distribuido Avanzado (ADL, *Advanced Distributed Learning*) [URL10], en el que se diseña un esquema para identificar y organizar los recursos de aprendizaje con base en un taxonomía y plan de exposición, en el que los elementos denominados activos (objetos básicos de texto, sonido e imagen), SCA (SCA, *Sharable Content Assets*) que son el conjunto de activos sin manejo de parámetros con el sistema y SCO (SCO, *Sharable Content Object*) que son el grupo de activos con transferencia de parámetros del sistema, ambos son organizados por una estructura jerárquica [URL10].

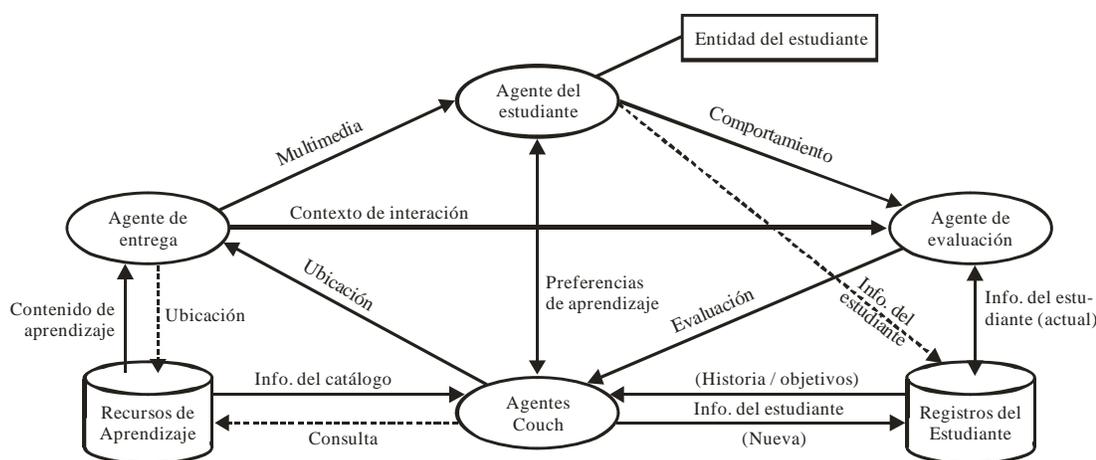


Figura 2.2. Arquitectura del sistema LTSA-IEEE-LTSC 184.1. [URL11]

2.8. Desarrollo de Objetos de Aprendizaje Reutilizables Inteligentes para los Sistemas Educativos Basados en Web

2.8.1. Propósito

Uno de los problemas básicos que enfrenta la comunidad de aprendizaje es el cómo producir y entregar contenidos de calidad y experiencias de aprendizaje en línea, mediante la composición, revisión y actualización de manera eficiente del contenido de los mismos. Así, el propósito de Sheremetov es desarrollar la estructura de un modelo dinámico para la composición del desarrollo de SEBW, centrando su investigación en la reutilización, la accesibilidad, la durabilidad y la interoperabilidad de la didáctica de los materiales y entornos de educación virtual [Peredo05].

2.8.2. Enfoque

El enfoque que presenta la propuesta de Sheremetov es el de producir materiales de aprendizaje con interfaz y funcionalidad estandarizada, ricos en contenido multimedia, interactividad y comentarios. Esta iniciativa se enfoca en el uso del paradigma Orientado a Objetos y la descripción de una arquitectura multi-agente.

2.8.3. Estructura

La estructuración del modelo dinámico para la composición de estos componentes se basa en el concepto gráfico del modelo de representación del conocimiento mediante una estructura multi-agente, como *middleware*, para sistemas de educación basados en Web, además de la descripción del material que se ha denominado Intelligent Reusable Learning Components Object Oriented (IRLCOO por sus siglas en inglés) [Sheremetov05].

Sheremetov examina un tipo especial de material llamado Componente de Software Inteligente Reutilizable y Programable ampliando la capacidad de Interacciones de Aprendizaje (*LI, Learning Interactions*) de Flash Macromedia. Los niveles genéricos de la arquitectura de los agentes para los Sistemas de Administración de Aprendizaje (LMS, *Learning Management System*) son definidos e integrados en los IRLCOO que son un tipo especial de Objetos de Aprendizaje Reutilizables (RLO, *Reusable Learning Object*) que producen el contenido del aprendizaje multimedia, interactividad y retroalimentación. Por otro lado, se describe la arquitectura multi-agente para IRLCOO, seguimiento, recuperación y la entrega, utilizando la arquitectura genérica y la Plataforma de Agentes de Componentes (CAP, *Component Agent Platform*).

El contenido del aprendizaje en línea regularmente contiene: materiales para explicar los conceptos, un sistema de navegación, evaluaciones y herramientas de colaboración, así como otros elementos gráficos e interactivos. Los contenidos de aprendizaje deben considerar también a los alumnos. Como mínimo, el contenido de aprendizaje debe reconocer quien es el alumno y guardar un registro de información sobre la experiencia de este.

Un RLO es cualquier recurso digital que puede ser reutilizado para apoyar el aprendizaje basado en Web. Los ejemplos de RLO's puede incluir video y audio pregrabados, un módulo del curso, animaciones, gráficos, aplicaciones basadas en la Web, páginas Web, PDF's, documentos de Office, y otras piezas que están destinadas a entregar por completo experiencias como una lección. Desde el punto de vista pedagógico, cada RLO podría desempeñar un determinado papel dentro de una metodología de diseño instruccional. Los objetos de aprendizaje pueden ser re-ensamblados para crear nuevos cursos o secuenciado para formar vías de formación.

En la ingeniería de software, un componente es un bloque reutilizable de un programa construido que se puede combinar con otros componentes en la misma u otras computadoras dentro de una red distribuida para formar una aplicación. Es decir, de acuerdo a Sheremetov "*un componente es una pieza de software reutilizable en forma binaria que puede conectarse a otros componentes con una cantidad relativamente pequeña de esfuerzo*" [Sheremetov05].

Aplicando estas ideas al desarrollo de los RLO, los componentes que se utilizaron como base de los elementos contenidos en el formato de los Componentes Reutilizables de Aprendizaje (RLC, *Reusable Learning Components*) tienen pequeños *clips* para el aprendizaje de los elementos, así, la idea original fue generar una biblioteca de RLC para SEBW que contuviera diferentes plantillas para RLC, botones programables con pequeños clips, interfaces para los parámetros de los clips, por ejemplo, botones genéricos para audio, vídeo, exámenes, gráficos, etc. Las interfaces se almacenan como archivos en una forma estática. Asimismo, las plantillas de aprendizaje se crean para añadir la funcionalidad o mejorar las plantillas ya existentes. La liberación del ActionScript versión 2,0 dentro de Flash MX 2004 permitió la redefinición de la RLC como un Componente Orientado a Objetos de Aprendizaje Reutilizable (RLCOO, *Reusable Learning Components Object Oriented*), habilitando así la aplicación del paradigma orientado a objetos, además de permitir la adición de algunas funcionalidades comunes dentro de la API tal y como la comunicación con el LMS, la comunicación con los agentes y la carga dinámica de los activos en tiempo de ejecución.

Es así que se generó un nuevo tipo de componentes, los IRLCOO's que son desarrollados para las plantillas avanzadas, añadiendo comentarios de los usuarios y la funcionalidad de la generación dinámica de componentes para la auto-configuración de los materiales didácticos.

2.9. Sistemas de Educación Adaptativos e Inteligentes Basados en Web Usando Programación Orientada a Componentes

2.9.1. Propósito

La educación centrada en el estudiante indica que los procesos de secuenciación, autoría de contenido y evaluación, determinan los objetivos de aprendizaje de los estudiantes basándose en las salidas de los SEBW. Además, el repositorio de contenido y evaluación deberá ser el adecuado para las necesidades en particular de cada individuo. Pero, al mismo tiempo, tienen que ser flexibles y disponibles para ser adaptados y utilizados por una amplia comunidad de desarrolladores y estudiantes, respectivamente. Canales et. al determinaron que era necesario el desarrollo de una arquitectura SEBW que considerara toda una diversidad de necesidades y proporcionara la funcionalidad necesaria sobre la base de las habilidades de la Web.

De acuerdo a Canales, la Ingeniería de Software tiene que considerar las especificaciones particulares de los SEBW con el fin de adaptar las soluciones de acuerdo a los tiempos. Por otra parte, los autores de los contenidos con componentes reutilizables y las tareas de evaluación tienen que ser completadas automáticamente lo mejor posible. Los componentes ayudan a reducir la complejidad, la gestión, y reutilización de los contenidos. Por lo tanto, con el fin de entregar experiencias inteligentes y de adaptación de enseñanza-aprendizaje, es necesaria la previsión de resultados del modelo de estudiante, además de la evaluación automática del comportamiento de los estudiantes [Canales07].

El propósito es contribuir con un modelo que permita llevar a cabo Sistemas Educativos Adaptables e Inteligentes basados en la Web que toma como fundamento la especificación IEEE 1484 LTSA, que es la norma para la Arquitectura del Sistema de Tecnologías de Aprendizaje (LTSA, *Learning Technology System Architecture*) [URL11] [URL15], que considera las distintas necesidades de aprendizaje de los estudiantes por medio de una arquitectura global y el marco de trabajo para el desarrollo SEBW.

2.9.2. Enfoque

La especificación IEEE 1484 LTSA, lanzada por el grupo de trabajo del Instituto de Ingenieros en Electricidad y Electrónica (IEEE, *Institute of Electrical and Electronics Engineers*) a través del Comité de Estándares de Tecnologías de Aprendizaje (LTSC, *Learning Technology Standards Committee*), cubre un amplio rango de sistemas comúnmente conocidos como tecnologías de aprendizaje, entrenamiento basado en computadora, instrucción asistida por computadora, tutoría inteligente, tecnología de educación y entrenamiento, y demás. Siendo el propósito de este modelo mostrar una nueva Arquitectura Orientada a Componentes y Agentes (AOCA), para el desarrollo de SEBW basados en la especificación IEEE 1484 LTSA [Peredo08].

La solución se enfoca en la especificación IEEE 1484 LTSA y en SCORM perteneciente a la ADL, utilizando los IRLCOO desarrollados por Peredo [Peredo05].

El comité de estándares propuso una arquitectura que se pretende sea un estándar para todos los sistemas de tecnologías de aprendizaje. A continuación se describirán brevemente las capas que lo conforman (Figura 2.1):

- Capa 1. Las interacciones de los ambientes y de los estudiantes: que consiste en la adquisición, transferencia, intercambio, y formulación del conocimiento o la información a través de la interacción con el ambiente.
- Capa 2. Características de diseño relacionadas al estudiante: concernientes a los efectos que tienen sobre el diseño de los sistemas de tecnología de aprendizaje.
- Capa 3. Sistema de componentes: que describe la arquitectura basada en componentes, identificando los procesos que intervienen en el sistema y se centran en el estudiante.
- Capa 4. Implementación de perspectivas y prioridades: se describen los sistemas tecnológicos de aprendizaje desde una amplia variedad de perspectivas por referencia a subconjuntos de la capa de componentes del sistema.
- Capa 5. Interoperabilidad y funcionalidad de los componentes –codificación, APIs, protocolos: se describen los componentes genéricos plug and play (interoperables) e interfaces de una arquitectura tecnológica de aprendizaje basada en las tecnologías de la información.

2.9.3. Estructura

Básicamente la AOCA, propone mejoras en la capa 3 de la especificación IEEE 1484 LTSA.

La arquitectura modificada consiste de cuatro procesos: Entidad del estudiante, Evaluación, Coach y Entrega. Se establece la necesidad de dos bases de datos: registros del estudiante y recursos de aprendizaje, además de catorce flujos de información.

El proceso del Coach está dividido dentro de dos subprocesos: Coach y Coach virtual. Esta decisión se debe a las necesidades individuales de los estudiantes, algunas deben de ser atendidas por el Coach de manera manual, mientras que otras como: secuencia de navegación de los contenidos, secuenciación de las evaluaciones, estatus de actividades, estatus de tareas, etc., deben ser capturadas de manera automática, siendo responsabilidad del Coach virtual la captura de las métricas mediante los componentes IRLCOO de los contenidos, evaluaciones, preguntas Frecuentes o FAQ (*Frequently Asked Questions*), foros de discusión, chats y herramientas colaborativas.

2.9.3.1. Plataforma IRLCOO

Los IRLCOO fueron desarrollados con un integrador de medios llamado Flash de Macromedia, este integrador de medios tiene un lenguaje de OOP denominado ActionScript 3 (MACROMEDIA, 2005). Una característica fundamental de los componentes es la carga en tiempo de ejecución de los medios (Archivos Flash Video, Archivos MP3, Archivos SWF y Archivos JPG), además de ofrecer un ambiente adaptivo y programable que se adapta a las necesidades de los estudiantes. El objetivo es generar una librería multimedia de componentes IRLCOO para sistemas de SEBW teniendo en mente la separación del contenido de la navegación, para su posterior reutilización. Mientras que el Run Time Environment de la ADL permite separar los contenidos de la navegación, el costo que hay que pagar con este esquema es alto, las instituciones educativas requieren de un LMS compatible, además de que los materiales solo son accesibles vía Web. Mientras que con el esquema basado en componentes es posible producir materiales educativos duales, que pueden ser accedidos vía Web, y pudiendo ser utilizados los componentes IRLCOO como unidades de composición para aplicaciones de escritorio, usándolos como componentes ActiveX, distribuibles vía CD o DVD. Los componentes usan diferentes niveles dentro del Flash Player, permitiendo conformar una estructura jerárquica. Esta estructura permite la reutilización binaria de componentes, creando componentes especializados más pequeños, reusables, y adecuados

para agregarse dentro de componentes más complejos en tiempo de ejecución. Los componentes IRLCOOs pueden contar con una API para la comunicación con el sistema LMS.

2.9.3.2. *Arquitectura orientada a servicio*

La AOCA esta basada en la Arquitectura Basada en Servicio (SOA, *Service Oriented Architecture*). SOA provee servicios y contenidos gracias al uso de la segunda generación para SW, SOAP, XML, Lenguaje de Definición de Servicios Web (*Web Services Definition Language*, WSDL por sus siglas en inglés) y UDDI (UDDI, *Universal Description, Discovery, and Integration*). La ventaja de usar SOA reside en el desarrollo de servicios, que proveen interoperabilidad a nivel de aplicación. Esto se lleva a cabo debido a que los servicios ocultan los detalles del ambiente de ejecución, a través de interfaces que tienen todas las aplicaciones. Estos servicios son llamados software y tienen la habilidad de emitir y consumir mensajes de datos XML, sin importar la plataforma, *middleware*, sistema operativo o hardware [Peredo08].

2.10. Estudio Comparativo de las Metodologías y Modelos.

Como punto inicial del análisis de las metodologías que se identificaron como aportaciones relevantes y que determinaron un aporte especial en la época en que se publicaron; se pretende realizar un análisis comparativo sobre las principales contribuciones de cada trabajo. Una vez realizado este análisis, se plantea la creación de una metodología alternativa para el desarrollo de software educativo basada en prácticas efectivas que llamamos MeSoFT. Para iniciar el análisis de las metodologías presentadas en este capítulo, se establece un conjunto de criterios a priori que, de acuerdo a nuestra experiencia, debería tener cualquier metodología que trate de representar una alternativa eficiente en el campo. A continuación se presenta y describe la importancia de los criterios propuestos:

- **Orientada a SE**

Descripción: Permite determinar si la metodología o método está destinada específicamente al desarrollo del SE o se puede emplear para el desarrollo de cualquier aplicación SW. El objetivo es determinar la relevancia de la metodología dentro del contexto de búsqueda.

Justificación: Desde el punto de vista del usuario, es importante que el proceso metodológico para desarrollar un SE se encuentre destinado, exclusivamente, a la labor educativa.

- **Comprensible**

Descripción: Permite determinar que el proceso de la realización del SE sea lo más claro y preciso posible, esto con el objetivo de proporcionar la información necesaria para el profesional que implementa la metodología.

Justificación: Es importante que los procesos que conforman la metodología se encuentren descritos de forma comprensible, porque en el proceso de enseñanza-aprendizaje participa un conjunto de profesionales con diferentes capacidades, algunos expertos en pedagogía que no son expertos en el desarrollo de aplicaciones y otros por el contrario expertos desarrolladores pero que no son capaces de identificar las partes pedagógicas del desarrollo.

- **Estructura en procesos y actividades**

Descripción: Permite determinar si la metodología o el modelo define ciertos criterios que se encuentren estructurados y que además defina una secuencia de actividades que sirvan como guía en el desarrollo de un SE.

Justificación: Es importante que la secuencia entre los procesos y las actividades se encuentren correctamente estructurados para evitar que exista una pérdida de información y para lograr que un proyecto sea lo más coherentemente posible, de acuerdo a nuestra teoría esto terminará en un proceso eficiente para el desarrollo de un SE.

- **Enfoque a la mejora de procesos**

Descripción: Permite determinar si la metodología o el modelo se encuentran enfocadas a la mejoras de los procesos que intervienen en el proceso de enseñanza-aprendizaje, o bien si durante la ejecución de la metodología existe un proceso que verifique la mejora continua de los procesos con la finalidad de mejorar un proceso que se realiza de forma común.

Justificación: La importancia de verificar que los procesos que realizan las metodologías o modelos se encuentren enfocados a la mejora, es ofrecer una mejor calidad en los materiales presentados, así como establecer una mejora continua de los procesos.

- **Incorporación de un mecanismo pedagógico**

Descripción: Permite identificar si la metodología o modelos, se encuentran apoyadas en los aspectos pedagógicos del proceso de enseñanza-aprendizaje.

Justificación: Este aspecto es de suma importancia puesto que se pretende el desarrollo de SE, y de acuerdo a las propuestas revisadas es importante no descuidar el aspecto pedagógico de los materiales que se utilizan, así como la forma en que a los alumnos se les brinda esa información.

- **Fácil implementación**

Descripción: Esta destinado a determinar la dificultad de implementar la metodología, de acuerdo a la facilidad con la cual, una persona experta o no experta (en ambos aspectos, técnico y pedagógico) se puede permitir el uso de la metodología para el desarrollo de un SE.

Justificación: El objetivo que se propone al evaluar este aspecto es tener un parámetro que indique que tan complicado será la implementación de la metodología o el modelo para el desarrollo de un SE. Además, nos permite determinar la necesidad de personas expertas o no expertas para los aspectos técnicos y pedagógicos.

- **Servicios Web**

Descripción: Permite determinar si la metodología o el modelo es aplicable para desarrollo de proyectos en donde se cree SEBW, esta característica indica si la metodología o el modelo aprovecha las ventajas del uso de la tecnología en aplicaciones Web.

Justificación: Actualmente el uso de las aplicaciones Web y cada día la ampliación de los recursos de Internet para toda la población y en especial para el sector educativo, tiende a desarrollar sistemas de educación que interactúen o que se puedan utilizar a través del Web, a distancia.

- **Procesos de evaluación**

Descripción: Permite definir si la metodología o el modelo contempla módulos o procesos en donde se permita la evaluación de los materiales presentados a los alumnos, con el fin de obtener una forma de estimar el aprovechamiento de los alumnos hacia los temas presentados.

Justificación: La evaluación de los alumnos hacia los materiales presentados en las sesiones que se tienen frente al SE permite evaluar los conocimientos adquiridos y obtener una relación del nivel de aprendizaje presentado, a su vez también permite presentar a los alumnos el material indicado de acuerdo a su nivel y evitar que el usuario repita los materiales y se presenten pérdidas de tiempo o distracciones.

- **Repositorio de activos**

Descripción: Permite definir si la metodología o el modelo dispone de un repositorio de activos en donde a través de las diferentes ejecuciones del SE se puedan almacenar resultados, practicas efectivas, experiencias, nuevos materiales, etc. y así realizar un análisis para generar nueva información o poder presentar mejor calidad en los procesos.

Justificación: Este aspecto es introducido con la finalidad de valorar el uso de un repositorio de activos basado en la experiencia o prácticas efectivas que faciliten el proceso de la creación de los materiales educativos. Tales prácticas efectivas deben ser objetivas y su obtención debe evitar un comportamiento erróneo del SE.

- **Plantillas y documentos propios**

Descripción: Permite definir si la metodología o modelo establece plantillas o documentos que permitan al usuario entender el proceso de desarrollo, y facilitar la implementación del mismo, así como tener la facilidad de adquirir los conocimientos más relevantes por parte de los docentes y de los alumnos.

Justificación: Es necesario que se contemplen documentos definidos y estructurados orientados correctamente para permitir la correcta obtención de información relevante al desarrollo del SE.

- **Validación práctica**

Descripción: Permite definir si la metodología o modelo fue probada con éxito, es decir si existe un SE que haya sido creado mediante la aplicación de la metodología o el modelo.

Justificación: Es importante definir si la metodología o modelo ha sido probada con éxito ya que esto permite evaluar el desarrollo del SE y también permite evaluar los niveles de aprendizaje alcanzados.

- **Reutiliza materiales aprendidos**

Descripción: Permite determinar si la metodología o el modelo define y utiliza un repositorio de experiencias de materiales educativos utilizados como recurso para planificar nuevos esfuerzos en otro proyecto similar.

Justificación: El objetivo de utilizar materiales ya aprendidos no es cuestión de intentar unificar los materiales y criterios de evaluación, si no entender e identificar las circunstancias por las cuales la metodología es la apropiada para el desarrollo del SE.

- **Define el tipo de lenguaje de programación del sistema**

Descripción: Permite determinar si la metodología o el modelo define y utiliza un tipo específico de lenguaje de programación para el desarrollo del SE.

Justificación: El objetivo de definir una metodología es el hecho de permitir que esta pueda implementarse en cualquier tipo de lenguaje de programación, con el único objetivo de que

cualquier desarrollador experto tenga la posibilidad de utilizar el lenguaje de programación que más le favorezca.

- **Actividades para el aspecto técnico del proyecto**

Descripción: Permite determinar si la metodología o el modelo define las actividades principales sobre el desarrollo de los aspectos técnicos, como por ejemplo: plataforma de desarrollo, plataforma del Sistema Operativo (OS, Operating System), Interfaces gráficas de usuario (GUI, Graphics User Inteface), requerimientos mínimos del sistema.

Justificación: Este aspecto es introducido porque es importante identificar los requerimientos técnicos mínimos que se necesita tener para poder llevar a cabo tanto el desarrollo e implementación del SE, como la ejecución del mismo, con el objetivo de no tener fallas en la evolución de su desarrollo.

A continuación la Tabla 5 presenta un análisis empírico sobre las características de los modelos y metodologías revisadas, indicando aquellas que consideramos de vital importancia, de acuerdo a nuestros criterios, para definir nuestro método.

Tabla 5. Cuadro comparativo entre metodologías y modelos para el desarrollo de SW Educativo.

Características	Metodologías y Modelos					
	Pere	Galvis	Arias	Cataldi	SEBW	Peredo
Orientada a SW Educativo	Si	Si	Si	Si	Si	Si
Comprensible	No	No	Si	Si	No	No
Estructura en procesos y actividades	No	No	Si	Si	Si	Si
Enfoque a la mejora de procesos	No	No	No	No	Si	Si
Incorporación de un mecanismo pedagógico	Si	Si	Si	Si	Si	Si
Fácil implementación	No	No	No	No	No	No
Servicios Web	No	No	No	No	Si	Si
Procesos de evaluación	No	No	No	Si	Si	Si
Repositorio de activos	No	No	No	No	Si	Si
Plantillas y documentos propios	No	No	No	Si	Si	Si
Validación práctica	Si	-	-	Si	Si	Si
Reutiliza materiales aprendidos	No	No	No	No	Si	Si
Define el tipo de lenguaje de programación del sistema	No	No	No	No	No	Si
Define actividades para el aspecto técnico del proyecto	No	No	No	Si	Si	Si

Como resultado de evaluar los criterios previamente establecidos en cada una de las metodologías presentadas con anterioridad se pudo identificar prácticas efectivas que han contribuido al éxito de proyectos de SE, así como prácticas deficientes que es necesario omitir si se busca generar productos educativos con altos niveles de rendimiento. Las principales aportaciones de este análisis se pueden resumir de la siguiente forma:

- La metodología de Galvis fue uno de los primeros intentos para formalizar el desarrollo de SE. Sin embargo, la literatura encontrada no muestra suficiente información como para considerarla como un conjunto de prácticas “efectivas”. La estructura de la metodología es difícil de entender y la ambigüedad de las actividades mostradas evita que el proceso de desarrollo pueda ser repetitivo y mejore los niveles de aprendizaje. Una de las principales aportaciones de Galvis es la iniciativa de incluir el Modelado Orientado a Objetos para identificar diagramas de interacción que permitan representar el SE entre tres niveles importantes: educativo, comunicacional y computacional.
- Pere Marqués propone una metodología que facilita el diseño y desarrollo de SE por su estructura dividida en etapas. Sin embargo, la contribución de Marqués se orienta más a satisfacer la parte pedagógica y deja a un lado la parte técnica del SE, la aportación de

Marqués vislumbra tempranamente la construcción de un guión preliminar que indique las principales características del SE; lo que más tarde será denominado por Cataldi *storyboard*

- La propuesta de Cataldi incrementa el nivel de detalle en el desarrollo de SE, al separar las actividades en dos dominios: pedagógico y técnico. El refinamiento de Cataldi consiste en un ciclo de vida que define todos los procesos y actividades a efectuar en cada etapa del desarrollo. La aportación de Cataldi introduce la posibilidad de realizar cambios en etapas tempranas y de realizar prototipos evaluables durante el desarrollo: la evaluación del programa.
- La propuesta de Arias conserva la estructura en fases y fortalece el concepto de *evaluación* mediante un eje transversal. La aportación de Arias se basa en la construcción de guiones que apoyen al desarrollador en la identificación de objetivos técnicos y pedagógicos. La metodología agrega la herramienta del *storyboard* para plasmar las ideas y requisitos que son determinados a lo largo del desarrollo. Arias propone el introducir un mecanismo que realice ajustes permanentes durante todo el proceso de desarrollo de SE para facilitar la interacción con la herramienta construida.
- La propuesta de los SEBW se caracteriza por el diseño de contenidos organizados en varios niveles de complejidad, el trabajo cooperativo para compartir experiencias, el paradigma de aprendizaje adaptativo, y el trabajo colaborativo entre los educandos. La contribución de esta nueva generación de SEBW recalca la importancia de dividir claramente los aspectos pedagógicos y técnicos del producto. Se incluye la reutilización de componentes, contenidos, y lecciones aprendidas. Se fortalece el concepto de evaluación en línea y adecuación del material educativo de acuerdo al nivel de dominio del estudiante.

Dado lo anterior se intentará desarrollar un método que cubra los aspectos resaltados en el análisis desarrollado, pero que además incluya el concepto de mejora de procesos para refinar las actividades de acuerdo a las experiencias obtenidas con los proyectos piloto desarrollados. Se propone la iniciativa de desarrollar un mecanismo pedagógico propio del método propuesto. Se incluirá un área de evaluación que determine fallas en el progreso de los estudiantes y permita corregir en caso de presentarse desviaciones severas. Se fortalecerá la reutilización de material mediante la definición de un repositorio de activos que contenga las plantillas y documentos que garanticen la correcta aplicación del método.

3. Diseño de MeSoFT

3.1. Prácticas efectivas

Tal y como se mencionó, las prácticas efectivas son las prácticas que las personas con experiencia reconocida en un campo específico han identificado como contribuciones significativas para el éxito de un proyecto [Adams04]. Según Jones [Walker03], “*las prácticas efectivas son la implementación de metodologías y herramientas que mejoran la productividad y la calidad de un proyecto y de un producto software final*”.

Siguiendo con estas definiciones, este trabajo propone el desarrollo de un método basado en prácticas y modelos de calidad efectivos (CMMI-DEV 1.2 [Software06] y TSP [Humphrey00]) para definir MeSoFT e intentar contribuir con la mejora de la calidad del aprendizaje a distancia. Con este método, la introducción de las prácticas efectivas no obligará a los estudiantes a seguir rigurosamente un orden de actividades para aprender un tema. En lugar de esto, se propone un repositorio de datos para guiar a los estudiantes dentro de un ciclo de aprendizaje (véase Figura 3.1).

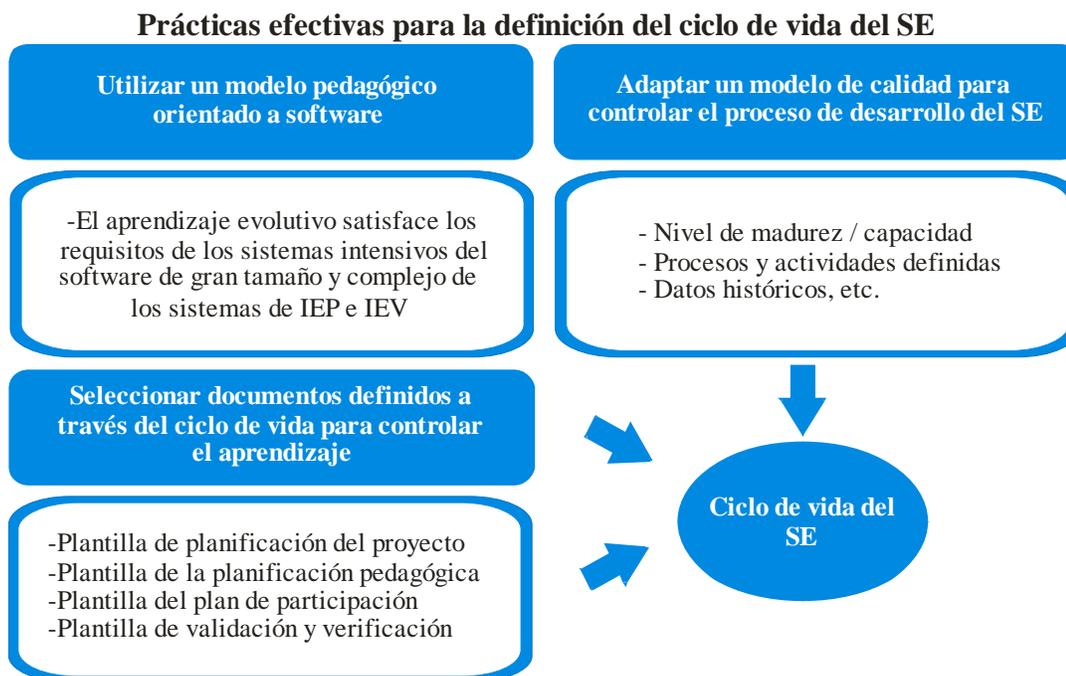


Figura 3.1. Definición del ciclo de vida del SE

El repositorio de prácticas pedagógicas efectivas resume el conocimiento de especialistas en el campo de la pedagogía. MeSoFT ofrece un número amplio de plantillas para guiar a los estudiantes en todas las prácticas, desde las más fáciles hasta los niveles más complejos.

El mecanismo de prácticas efectivas que aquí se propone se deriva de dos modelos de calidad, el Modelo de Madurez y Capacidad Integrado para el Desarrollo v1.2 (CMMI-DEV 1.2, *Capability Maturity Model Integration for Development v1.2*) [Software06] y del Proceso de Software en Equipo (TSP, *Team Software Process*) [Humphrey00]. Suponemos que el CMMI-DEV v1.2 cubrirá todas las prácticas necesarias para establecer y mejorar el ciclo de vida del SE. Se plantea que MeSoFT proporcione un ciclo de mejora para identificar y almacenar nuevas prácticas efectivas en cada proyecto de SE que se desarrolle. Las prácticas del TSP ayudan a que los estudiantes trabajen como un equipo y MeSoFT ofrece todas las plantillas necesarias para supervisar y controlar el proyecto de SE. Con la estructura propuesta se intenta proporcionar un método para desarrollar SE que cubra tanto los aspectos pedagógicos como los técnicos al mismo tiempo. El mecanismo de mejora, las plantillas y las prácticas constituyen un método alternativo para desarrollar SE enfocándose en las áreas de IEP e IEV (véase Figura 3.2).

Hasta ahora, los programadores habían confiado solamente en su propio conocimiento y no habían utilizado información válida para construir su SE; la única información usada eran manuales y libros de consulta.

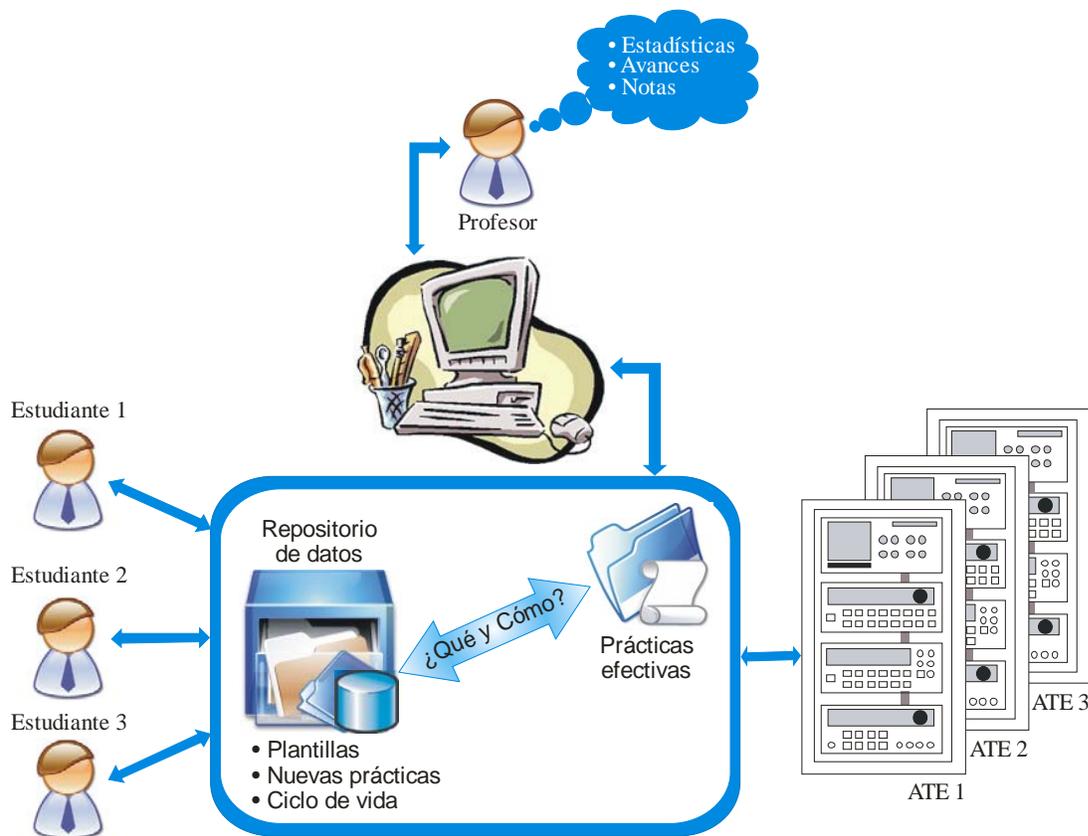


Figura 3.2. Un mecanismo para mejorar el proceso de aprendizaje continuo.

En base a los resultados del análisis realizado en el Capítulo 2, se definen las siguientes consideraciones:

- La solución propuesta estará destinada y orientada al desarrollo de SE, será fácil de comprender porque todos los procesos que se ejecuten se encontrarán descritos de tal forma que un usuario, con al menos conceptos básicos de ingeniería de software y pedagogía, pueda implementarla sin complicaciones.
- La estructura del método propuesto se organizará en una arquitectura que estará dividida en Áreas de Proceso (AP) genéricas que cubrirán los aspectos técnicos y pedagógicos, y una más que cubrirá la información gestionada por ambas partes. A su vez cada AP contendrá actividades específicas que permitirán proporcionar la especificación de cada proceso que se realice en la implementación de un SE.
- La solución implementará una orientación a la mejora de procesos, puesto que se propone introducir una AP que supervise, mida, verifique y tome las medidas necesarias durante todo el desarrollo del SE.
- También se intentará incorporar un mecanismo pedagógico que almacene prácticas efectivas de expertos en el área educativa mediante un repositorio de activos que contendrá dichas prácticas, este repositorio de activos contendrá también plantillas y documentos propios que faciliten a los usuarios la definición de los requerimientos pedagógicos y técnicos para el desarrollo del SE.
- Se propone que el método reutilizará materiales aprendidos, gracias al repositorio de activos que contendrá, y en donde se almacenarán todos estos datos con la finalidad de mejorar los procesos de desarrollo del SE fortaleciendo así la mejora de los procesos.
- Se pretenderá reflejar la facilidad de implementación del método utilizando una estructura de fácil seguimiento que cubra el aspecto de implementación. Aprovechando la nueva tendencia del desarrollo de SE, se pretenderá implementar servicios través de la Web para abarcar un campo más amplio de alcance educativo del sistema desarrollado.
- Se incluirá una AP que permita la evaluación de los resultados obtenidos por los alumnos para que con estos datos se pueda generar una opinión para el personal docente acerca del aprovechamiento de los alumnos al utilizar este tipo de aplicaciones.
- Se intentará no definir dentro del aspecto técnico el tipo de lenguaje de programación del SE, es decir, el encargado de desarrollar el SE lo podrá implementar en el lenguaje de programación que más se le facilite, teniendo la ventaja de no tener que aprender un nuevo lenguaje de programación, tampoco de la necesidad de tener un experto en un lenguaje de programación específico para el método.
- Por último, dado que el método estará basado conforme a los modelos de calidad como el CMMI-DEV v1.2 y el TSP se tendrá mayor cobertura hacia la definición correcta de las actividades para el aspecto técnico del proyecto y tendrá un ciclo de vida del SE de acuerdo a los estándares actuales.

A continuación se describen cada una de los componentes de la solución propuesta.

3.2. Áreas de proceso

Desde un punto de vista diferente a la literatura revisada en el Capítulo 2, MeSoFT está compuesto de tres categorías de Áreas de Proceso que aseguran la implementación de las prácticas efectivas: **Gestión del Proyecto, Gestión Pedagógica, y Gestión de la Información.**

Las AP de la Gestión del Proyecto cubren las actividades relacionadas con la planificación, la supervisión y el control del proyecto del SE. Las áreas de proceso de la Gestión del Proyecto son las siguientes: *Planificación del Proyecto*, *Supervisión del Proyecto*, y la *Medición y Análisis*. Como se muestra en la Figura 3.3.

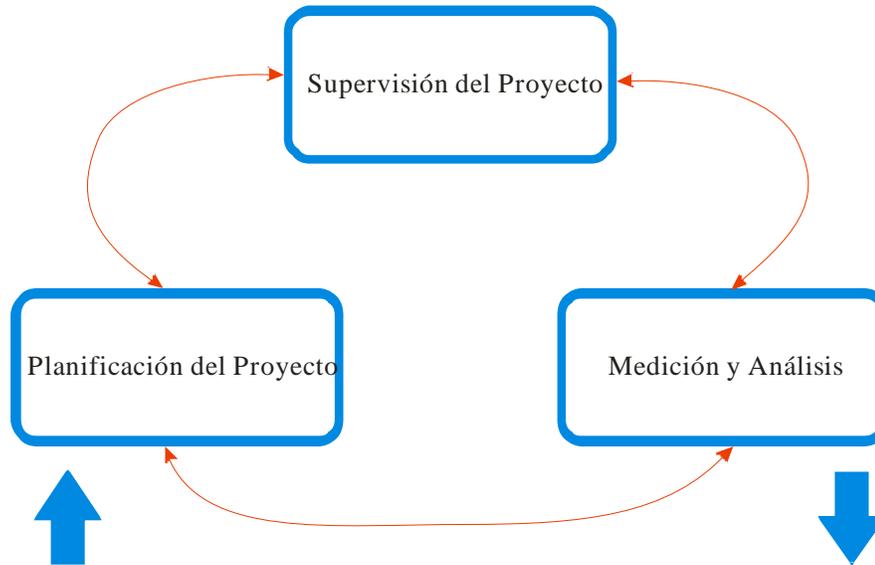


Figura 3.3. Área de Proceso de la Gestión del Proyecto

Las AP de la Gestión Pedagógica cubren las actividades relacionadas con los requisitos del proyecto del SE. Las AP de la Gestión Pedagógica son las siguientes: *Gestión de los Requisitos*, *Validación*, y *Verificación*. En la Figura 3.4 se muestran las áreas de proceso de la Gestión Pedagógica.

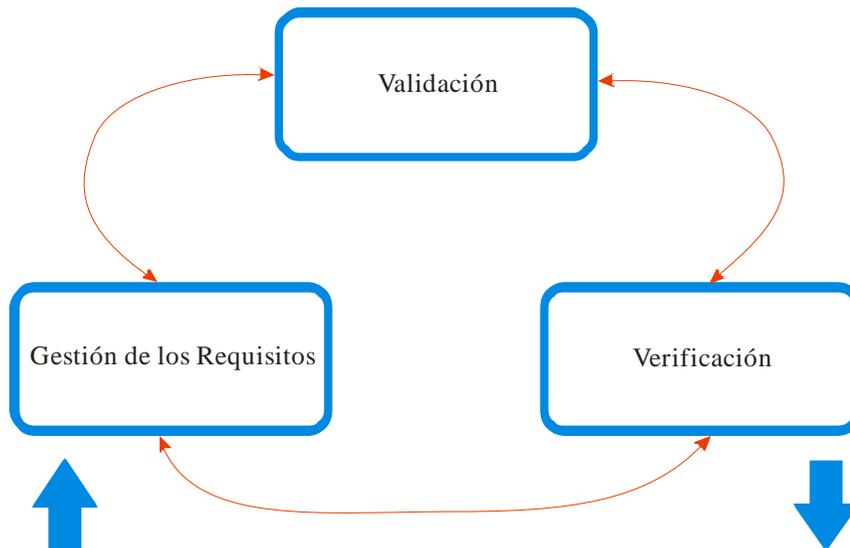


Figura 3.4. Áreas de Proceso de la Gestión Pedagógica.

La AP de la Gestión de la Información cubre las actividades de la gestión de datos relacionados con el proyecto del SE. Esta área de proceso proporciona actividades para la definición de un proceso estándar para el desarrollo del SE y ofrece documentos y plantillas estándares para controlarlo (ver Figura 3.5).

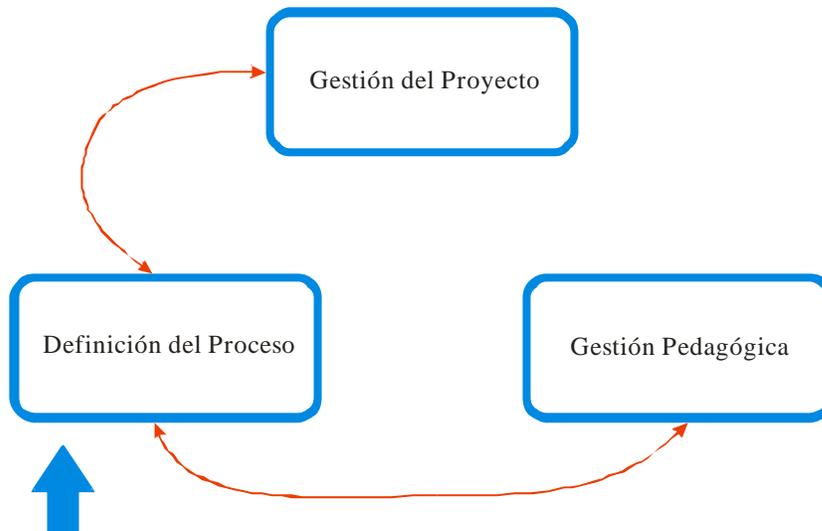


Figura 3.5. Definición de Proceso de la Gestión de la Información.

Todas las categorías de MeSoFT establecen **prácticas genéricas y específicas** que se logran con las AP (véase Figura 3.6). Una AP de MeSoFT es un conjunto de prácticas efectivas relacionadas en un área que, cuando son implementadas colectivamente, satisfacen un conjunto de metas consideradas importantes para llevar a cabo mejoras en esa área.



Figura 3.6. Categorías de áreas de procesos de MeSoFT.

Una práctica específica de MeSoFT es la descripción de una actividad que se considera importante en la realización de la meta específica asociada. Por ejemplo, una práctica específica del proceso de Supervisión del Proyecto es "Supervisar el plan de participación".

El título de una práctica específica (precedida por el número de la práctica) y cualquier nota asociada a esa práctica específica se consideran los componentes informativos del método.

Las prácticas genéricas de MeSoFT se llaman así porque la misma práctica se aplica no sólo a una AP sino a múltiples AP. Una práctica genérica es la descripción de una actividad que se considera importante en la realización de la meta genérica asociada. El título de una práctica genérica (precedida por el número de la práctica) y cualquier nota asociada a esa práctica se consideran también componentes informativos del método.

Las tres categorías de MeSoFT estarán compuestas de siete AP: 3 para la Gestión del Proyecto, 3 para la Gestión Pedagógica, y 1 para la Gestión de la Información, tal y como se describe a continuación.

3.2.1. Áreas de proceso de la Gestión del Proyecto

Las AP que se encargan de realizar la Gestión del Proyecto son:

- **Planificación del Proyecto (PP):** El propósito de la PP es establecer y mantener los planes que definen las actividades del proyecto del SE.
- **Supervisión del Proyecto (SP):** El propósito de la SP es proporcionar el entendimiento del progreso del proyecto del SE de tal forma que sea posible tomar acciones correctivas apropiadas cuando el desempeño del proyecto se desvía significativamente del plan.
- **Medición y Análisis (MA):** El propósito de la MA es desarrollar y tener una capacidad de medición que se utilice para apoyar necesidades de la Gestión de Información del SE.

3.2.2. Áreas de proceso de la Gestión Pedagógica

Las AP que se encargan de la Gestión Pedagógica son:

- **Gestión de Requisitos (GR):** El propósito de la GR es gestionar los requisitos del proyecto del SE e identificar inconsistencias entre estos requisitos, los planes del proyecto del SE y los productos del trabajo (todo este contenido es parte del repositorio de prácticas pedagógicas efectivas).
- **Validación (VAL):** El propósito de la VAL es demostrar que un SE satisface el uso previsto cuando es puesto en su entorno previsto (todo este contenido es parte del repositorio de prácticas pedagógicas efectivas).
- **Verificación (VER):** El propósito de la VER es asegurarse de que los componentes seleccionados del SE resuelven sus requisitos especificados.

3.2.3. Áreas de proceso de la Gestión de la Información

La AP que se encarga de la Gestión de la Información son:

- **Definición del Proceso (DP):** El propósito de la DP es establecer y mantener un conjunto viable de activos del proceso y estándares de trabajo, que son el repositorio de prácticas efectivas de MeSoFT.

A pesar de que las AP estén agrupadas de esta manera, éstas a menudo interactuarán y tendrán efecto en otra sin importar el grupo en que haya sido definida. Por ejemplo, el área de proceso de Planificación del Proyecto proporciona las prácticas específicas que se utilizan en el área de proceso de la Gestión de Requisitos para seleccionar una solución técnica de entre varias soluciones

alternativas. La Gestión de Requisitos es un área de proceso de la Gestión Pedagógica y la Planificación del Proyecto es un área de proceso de la Gestión del Proyecto.

La Figura 3.7 describe las interacciones de las AP dentro de su categoría y las interacciones entre las AP de otras categorías. Las interacciones entre las AP que pertenecen a diversas categorías se describen en las referencias de MeSoFT dentro de la sección Relaciones con otras Áreas de Proceso.

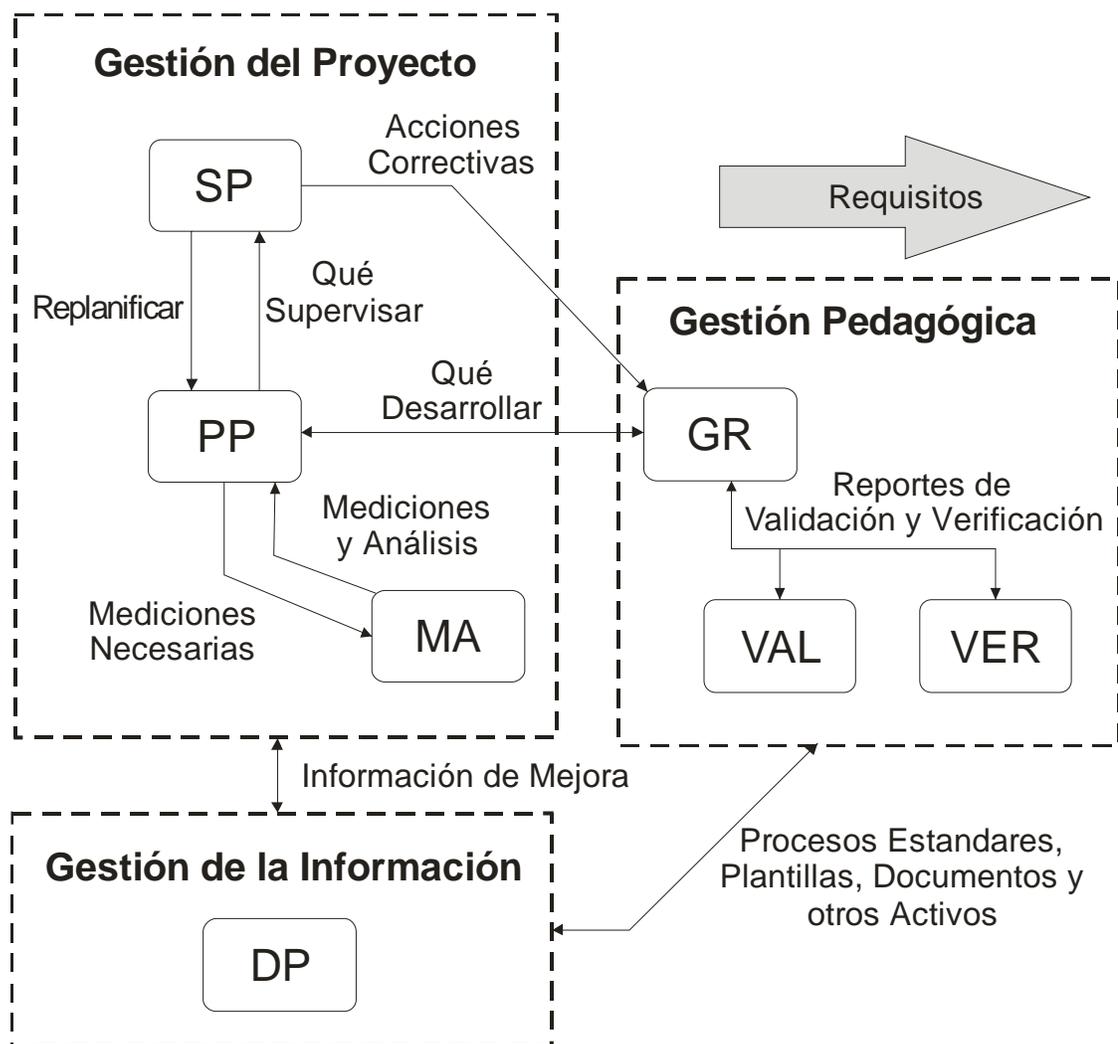


Figura 3.7. Áreas de proceso de MeSoFT.

3.3. Relaciones entre Áreas de Proceso

Las relaciones existentes entre las diferentes áreas de proceso que existen en MeSoFT, permiten saber qué información es necesaria entre las áreas para que estas a su vez tengan un buen desempeño y la operabilidad y los resultados del método sean los correctos. A continuación, y de acuerdo a la Figura 3.7, se describe el funcionamiento de cada una de las relaciones.

- **Acciones correctivas.** Las acciones correctivas son acciones tomadas para eliminar las causas de un problema existente, con el objetivo de evitar su recurrencia y tomar las medidas pertinentes para eliminar la causa que generó el problema.

Algunas de las acciones correctivas pueden ser:

- Seguimiento y medición de los procesos.
- Verificar que todo lo planeado originalmente se encuentre dentro de los límites establecidos.
- Determinar e implementar las acciones necesarias.
- Registrar los resultados de las acciones tomadas.
- Evaluar la necesidad de adoptar acciones para asegurar que no vuelva a ocurrir.
- Seguimiento y medición del proyecto.

Es necesario saber en qué momento se deben tomar las acciones correctivas, por ejemplo: se deben tomar cuando se presenta inconsistencia en la operación y/o aplicación de las AP, cuando el incumplimiento con las características del SE es constante o cuando haya un incumplimiento con lo planificado.

- **Replanificar.** La replanificación se lleva a cabo cuando en el AP de la Planificación del Proyecto se toma una acción correctiva apropiada y se hace el cambio dentro del plan del proyecto, teniendo esto como un cambio en la planificación de lo que ya se había planteado originalmente.
- **¿Qué supervisar?.** El AP de la Planificación del Proyecto proporciona a la Supervisión del Proyecto los aspectos que se tienen que supervisar y evaluar constantemente, esto con el fin de tomar las acciones correctivas pertinentes y así ejercer una replanificación sobre el proyecto a su debido tiempo, evitando gastos en tiempo y en esfuerzo.
- **¿Qué desarrollar?.** El AP de la Gestión de los Requisitos proporciona a la AP de la Planificación del Proyecto, los aspectos básicos, las características que debe contener y los objetivos que debe cumplir el proyecto. Dependiendo de si la Supervisión del Proyecto arroja una acción correctiva la Gestión de Requisitos cambia los aspectos y características del proyecto.
- **Mediciones Necesarias.** El AP de la Planificación del Proyecto le indica al AP de Medición y Análisis, los parámetros y características necesarias que se deben tomar en cuenta para que el proyecto cumpla con los objetivos planteados originalmente.
- **Mediciones y Análisis.** El AP de Medición y Análisis devuelve las mediciones y análisis realizados al AP de la Planificación del Proyecto de acuerdo con la información que le fue proporcionada y comparada con los parámetros y niveles que cumplen con los objetivos originales.
- **Reportes de Validación y Verificación.** El AP de la Gestión de Requisitos envía los requisitos que se obtuvieron originalmente para la realización del proyecto del SE a las AP de la Validación y Verificación, dichas áreas regresan reportes indicando si el SE satisface y cumple los objetivos y requisitos especificados para los que fue desarrollado.
- **Información de Mejora.** Esta información se encuentra en circulación entre las AP de Gestión de la Información y la Gestión del Proyecto, y corresponde a las mejoras en los diferentes procesos existentes en el método, estas mejoras se crean y se proponen dentro del desarrollo de un proyecto.

- Procesos Estándares, Plantillas, Documentos y otros Activos.** Es la información que esta ligada entre la Gestión Pedagógica, Gestión del Proyecto y la Gestión de la Información, esta información corresponde a los activos viables (plantillas, cuestionarios, documentos, etc.) y los estándares de trabajo con los cuales se forman el repositorio de de las prácticas efectivas de MeSoFT.

3.4. Definición de las Áreas de Proceso de la Gestión del Proyecto

3.4.1. Planificación del Proyecto

Procesos de MeSoFT

La primera área de proceso de MeSoFT es la *Planificación del Proyecto*. La planificación comienza con los requisitos que definen al producto y al proyecto. Incluye la estimación de las tareas de trabajo, la determinación de los recursos necesarios, el establecimiento del calendario, y la identificación y el análisis de los riesgos del proyecto. Durante el establecimiento del plan del proyecto puede ser necesaria la iteración de estas actividades, dicho plan proporcionará las bases para realizar y controlar las actividades del proyecto de acuerdo a las especificaciones de los alumnos o usuarios finales. El plan del proyecto requerirá ser revisado a medida que el proyecto avance con el objetivo de efectuar los cambios en los requisitos, estimaciones inexactas, acciones correctivas, y cambios en los procesos.

Relaciones con otras áreas de MeSoFT

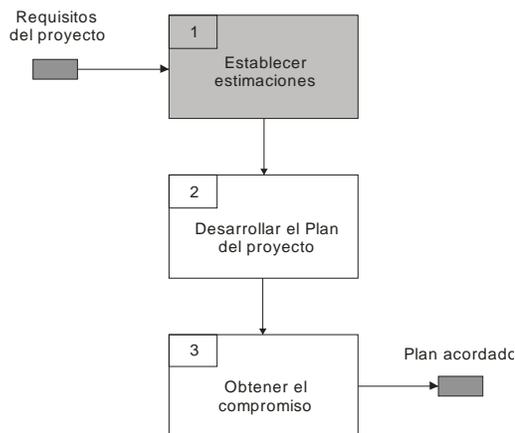
Remitirse a la Gestión de Requisitos para mayor información sobre los requisitos utilizados en la planificación.

En esta sección

En esta sección se describe información detallada de las actividades del proceso de Planificación del Proyecto.

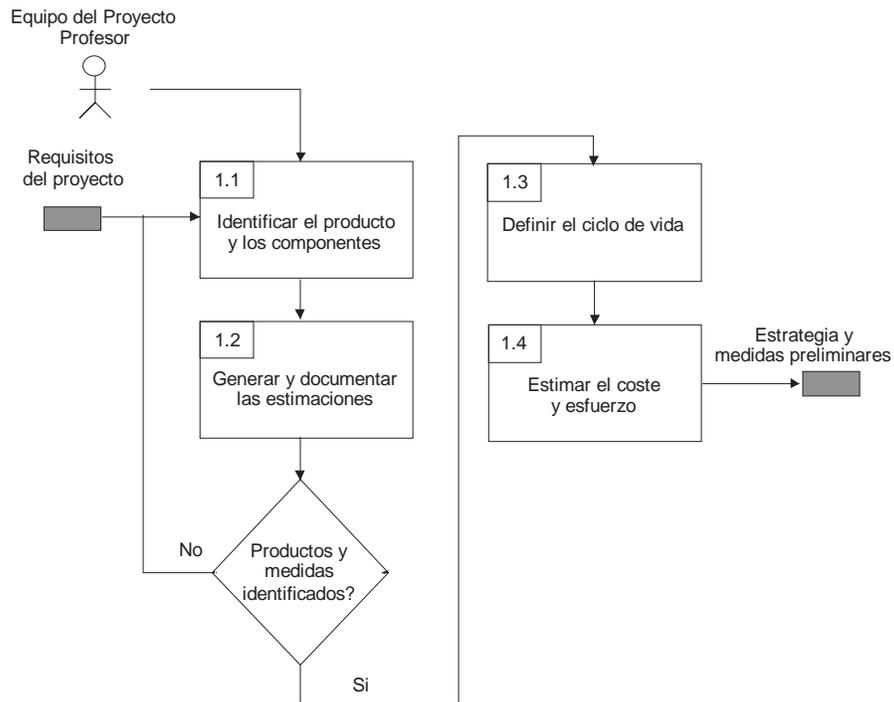
Actividad	Ver página
1 Establecer estimaciones	52
2 Desarrollar el plan del proyecto	56
3 Obtener el compromiso	62

Diagrama general de actividades



3.4.1.1. Establecer estimaciones

Diagrama de actividades



Identificar el producto y los componentes [1.1]

Propósito

El propósito de la actividad es guiar al equipo de desarrollo en la identificación del producto que será generado para lograr el objetivo pedagógico actual y los posteriores.

Roles de los participantes

Profesor de área.
Líder de equipo.

Entradas

Entrada	Fuente
Requisitos del proyecto	Externa
Resumen de tamaño (TAM)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Diseño conceptual (DIS)	La lista de productos y actividades es necesaria para la revisión de los participantes y el mantenimiento del plan mientras que el proyecto progresa.

Criterios de entrada Requisitos del proyecto, alcance del proyecto, actividades y productos identificados, descripciones de los productos y actividades.

Criterios de salida Organización de los productos del proyecto.

Criterios de evaluación Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Generar y documentar las estimaciones [1.2]

Propósito El propósito de la actividad es guiar al equipo en la generación de una estrategia de desarrollo y las estimaciones preliminares de tiempo y tamaño.

Roles de los participantes Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Requisitos del proyecto	Externa
Diseño conceptual (DIS)	Interna (Repositorio de activos)
Resumen de tamaño (TAM)	Interna (Repositorio de activos)
Resumen de calidad (CAL)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Estimaciones de tiempo y presupuesto en relación a la planificación (TAM)	El objetivo de las estimaciones es calcular el esfuerzo a emplearse en el proyecto a través de la identificación de tareas, roles y responsabilidades en el entorno educativo.

Criterios de entrada Procedimientos para estimar (repositorio de procesos estándar), lista de productos y actividades.

Criterios de salida Repositorio de medidas.

Criterios de evaluación Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Definir el ciclo de vida [1.3]

Propósito El propósito de la actividad es definir las fases del proyecto sobre las cuales se limitará el esfuerzo de la planificación.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Requisitos del proyecto	Externa
Diseño conceptual (DIS)	Interna (Repositorio de activos)
Plantilla de actividades (ACTI)	Interna (Repositorio de activos)
Catálogo de ciclos de vida	Externa

Salidas

Salida	Descripción
Ciclo de vida del proyecto (ACTI)	Un ciclo de vida es un marco de trabajo que contiene los procesos, las actividades y tareas relacionadas con el desarrollo, uso y mantenimiento de un producto software educativo desde la definición de sus requisitos hasta la terminación de su uso.

Criterios de entrada

Repositorio de activos, estimaciones de tiempo y presupuesto.

Criterios de salida

Repositorio de medidas, estimaciones de la planificación.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Estimar el coste y esfuerzo [1.4]**Propósito**

El propósito de la actividad es estimar el coste y esfuerzo del proyecto para las listas de productos y actividades creadas.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Estimaciones de tiempo y presupuesto (TAM)	Interna (Estimaciones de la planificación)
Ciclo de vida del proyecto (ACTI)	Interna (Repositorio de activos)
Diseño conceptual (DIS)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
ACTI con estimaciones de coste y esfuerzo	Las estimaciones de coste y esfuerzo están basadas en los resultados del análisis usando modelos o datos históricos aplicados al tamaño, actividades y otros parámetros de la planificación.

Criterios de entrada

Procedimientos para medir coste y esfuerzo, modelos históricos de esfuerzo y coste.

Criterios de salida

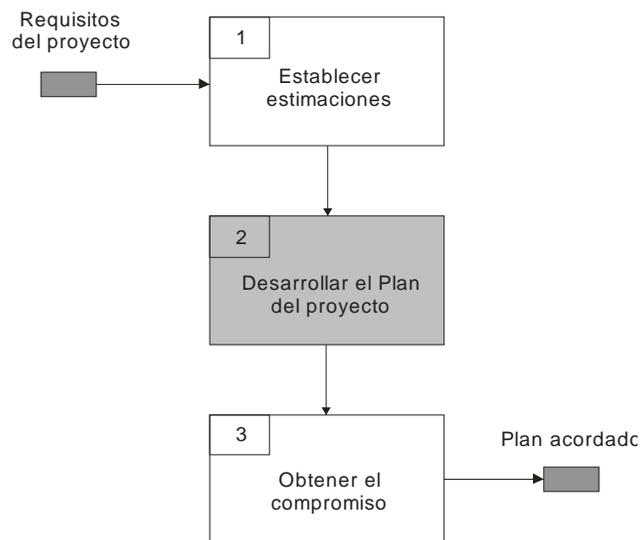
Estimaciones de la planificación.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

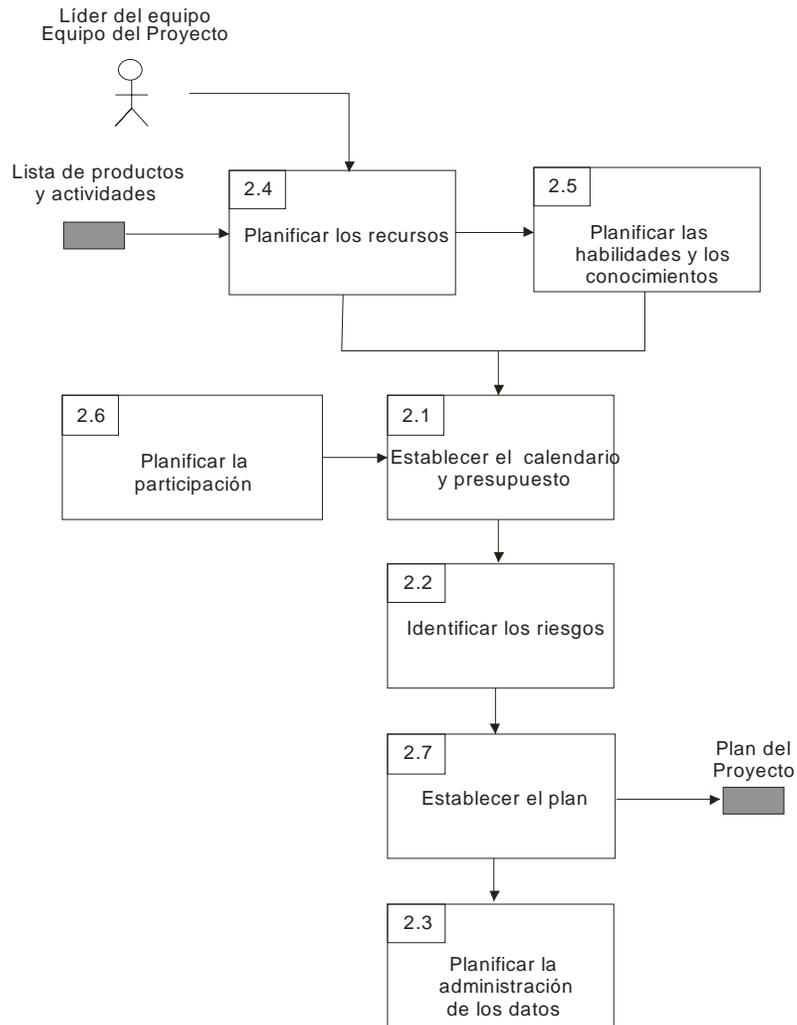
Diagrama general de actividades

La planificación del proyecto está compuesta de tres actividades. A continuación se describe la segunda de estas:



3.4.1.2. Desarrollar el plan del proyecto

Diagrama de actividades



Establecer el calendario y presupuesto [2.1]

Propósito

El propósito de la actividad es guiar al equipo en el desarrollo de un calendario y un presupuesto basado en las estimaciones desarrolladas anteriormente.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Estimaciones de tiempo y esfuerzo (ya reflejadas en ACTI)	Interna
Plantilla de calendario (CALEN)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Calendario del proyecto (CALEN)	El calendario determina el tiempo asignado a cada una de las actividades identificadas en el proyecto educativo.
Presupuesto del proyecto (en ACTI y CALEN)	El presupuesto del proyecto distribuye de forma apropiada los recursos disponibles.

Criterios de entrada

Estimaciones de la planificación (tiempo y coste).

Criterios de salida

Estimaciones de la planificación.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Identificar los riesgos [2.2]**Propósito**

El propósito de la actividad es guiar al equipo en el desarrollo de un análisis preliminar sobre los riesgos del proyecto.

Roles de los participantes

Líder de equipo.

Entradas

Entrada	Fuente
Taxonomía de riesgos	Interna (Repositorio de activos)
Registro para seguimiento de los riesgos (RIE)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Riesgos identificados (RIE)	Los riesgos identificados permitirán identificar situaciones, peligros, amenazas o vulnerabilidades potenciales que puedan afectar negativamente los esfuerzos y planes de trabajo.

Criterios de entrada

Lista de productos y actividades, estimaciones de la planificación.

Criterios de salida

Registro de seguimiento.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Planificar la administración de los datos [2.3]**Propósito**

El propósito de la actividad es utilizar y mantener el repositorio de activos del proyecto en donde serán manejados, actualizados y homogeneizados los datos desde el lanzamiento hasta el avance del mismo.

Roles de los participantes

Líder de equipo.
Responsable de soporte (parte del Equipo de proyecto)

Entradas

Entrada	Fuente
Requisitos del proyecto	Externa
Datos del proyecto	Interna

Salidas

Salida	Descripción
Formatos de contenido	Los formatos que harán uniforme el manejo y la distribución de los datos del proyecto deberán ser almacenados en el repositorio de activos.
Plan de la administración de los datos (DAT)	El plan para la administración de los datos asegura el acceso a los datos, establece los mecanismos para almacenar los datos y determina qué datos serán identificados, recogidos y distribuidos.

Criterios de entrada

Repositorio de activos

Criterios de salida

Plan de la administración de los datos, proceso estándar de gestión de versiones

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Planificar los recursos [2.4]**Propósito**

El propósito de la actividad es determinar los recursos necesarios para la ejecución del proyecto.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Requisitos del equipo (basados en el alcance y tamaño del proyecto)	Externa
Lista de productos y actividades (ACTI)	Interna (Repositorio de activos)
Estimaciones de tiempo y presupuesto (en ACTI)	Interna (Repositorio de medidas)
Estimaciones de coste y esfuerzo	Interna (Repositorio de medidas)

Salidas

Salida	Descripción
Plan de la administración de los recursos (ACTI con lo que hace el plan)	El plan asegura la gestión adecuada de los recursos para procesos, equipo/instalaciones y componentes.

Criterios de entrada

Lista de instalaciones/equipo, necesidad de habilidades del proyecto (derivadas de la lista de productos).

Criterios de salida

Definiciones y diagramas de flujo de los procesos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Planificar las habilidades y los conocimientos [2.5]**Propósito**

El propósito de la actividad es determinar las habilidades y los conocimientos necesarios para la ejecución del proyecto.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Habilidades disponibles del equipo	Externa
Lista de productos y actividades (ACTI)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Inventario de habilidades y formación (HAB por fase y con habilidades)	El inventario debe ser parte de una base de datos. Las habilidades y formación requeridas dependerán directamente de las habilidades disponibles en el equipo para apoyar la ejecución del proyecto.

Criterios de entrada

Mecanismos de provisión de conocimiento y habilidades (p.e. manuales, uso de gente externa al equipo, referencias de lenguajes).

Criterios de salida

Bases de datos (habilidades y formación).

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Planificar la participación [2.6]**Propósito**

El propósito de la actividad es identificar el tipo de personas y funciones que requieren de representación en el proyecto y describir su relevancia y el grado de interacción con las demás tareas del proyecto.

Roles de los participantes

Profesor de área.
Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Inventario de habilidades y formación (HAB)	Interna (Repositorio de activos)
Ciclo de vida del proyecto (ACTI)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plan de participación (INV)	El plan define la entrega de reportes de estado, desde el lanzamiento, tanto para el profesor como para el alumno, incluyendo los resultados de los relanzamientos.

Criterios de entrada

Lista de los participantes relevantes, roles y responsabilidades.

Criterios de salida Mecanismos para la identificación de roles; utilizar la matriz participantes/actividades (PARTI).

Criterios de evaluación Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Establecer el plan [2.7]

Propósito El propósito de la actividad es establecer y mantener el contenido general del plan del proyecto.

Roles de los participantes Líder de equipo.

Entradas

Entrada	Fuente
Estimaciones de la planificación	Interna (Repositorio de medidas)
Riesgos identificados (RIE)	Interna
Ciclo de vida del proyecto (ACTI)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plan del proyecto (PROY)	Un documento que trata todos los artefactos relevantes de la planificación para alcanzar el entendimiento, compromiso y rendimiento mutuo de todos aquellos relacionados con brindar apoyo al objetivo educativo reflejado.

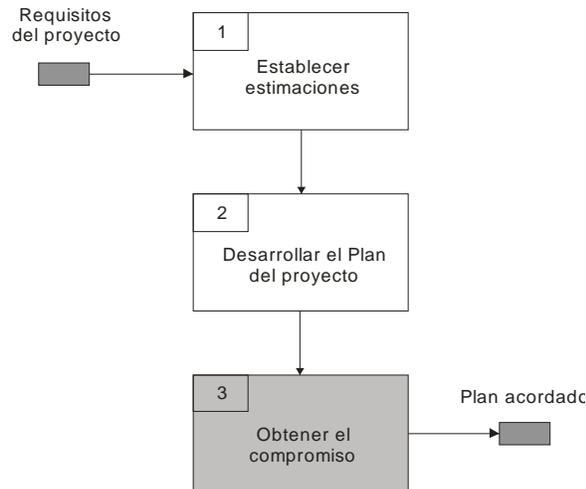
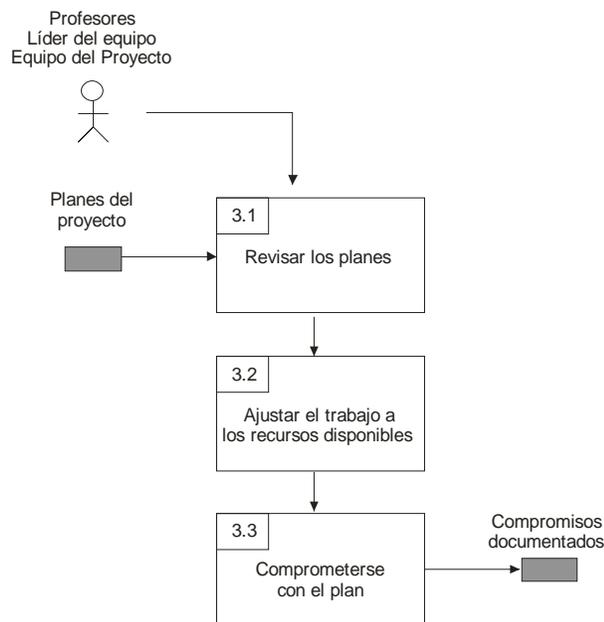
Criterios de entrada Planes de la administración de los datos, de la administración de los recursos y de la participación, documentos de inventario de habilidades.

Criterios de salida Repositorio de activos.

Criterios de evaluación Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Diagrama general de actividades

La planificación del proyecto está compuesta de tres actividades. A continuación se presenta la tercera de estas:

**3.4.1.3. Obtener el compromiso****Diagrama de actividades****Revisar los planes [3.1]****Propósito**

El propósito de la actividad es lograr un ajuste de todos los planes que afectan al proyecto para asegurar la comprensión generalizada del alcance, los objetivos, los roles, y las relaciones que se requieran para el éxito del mismo.

Roles de los participantes

Profesor.
Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Plan del proyecto (PROY)	Interna (Repositorio de activos)
Plan de participación (INV)	Interna (Repositorio de activos)
Plan de administración de los recursos	Interna (Repositorio de activos)
Inventario de habilidades y formación (HAB)	Interna (Repositorio de activos)
Resumen de calidad (CAL)	Interna (Repositorio de activos)
Resumen del tamaño del producto (TAM)	Interna (Repositorio de activos)
Plantilla de actividades (ACTI)	Interna (Repositorio de activos)
Plantilla de calendario (CALEN)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Registros de las revisiones sobre los planes	Las revisiones identifican información de los planes individuales repetida o similar a la presentada en el plan del proyecto, y apoyan a determinar quién tiene la autoridad, la responsabilidad y el control del proyecto.

Criterios de entrada

Todos los planes que afecten al proyecto.

Criterios de salida

Repositorio de activos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Ajustar el trabajo a los recursos disponibles [3.2]**Propósito**

El propósito de la actividad es comparar y ajustar los planes en función de los recursos existentes y/o potenciales durante el desarrollo.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Requisitos del proyecto	Externa
Presupuesto del proyecto	Externa
Calendario del proyecto (CALEN)	Interna (Repositorio de activos)
Inventario de habilidades y formación (HAB)	Interna (Repositorio de activos)
Plan de participación (INV)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Disparidades revisadas	Las disparidades tratadas permiten desarrollar planes razonables del proyecto dados los costes y los objetivos previstos dentro de las expectativas.

Criterios de entrada

Métodos y parámetros para estimar

Criterios de salida

Repositorio de activos

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Comprometerse con el plan [3.3]**Propósito**

El propósito de la actividad es obtener el compromiso de todos los participantes responsables de realizar y apoyar la ejecución del plan.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Plan de participación (INV)	Interna (Repositorio de activos)
Plan del proyecto (PROY)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Compromisos documentados	El documento elicit los compromisos del equipo del proyecto con el plan que resuelve las metas fijadas por el objetivo educativo, y los compromisos hacia una de las alternativas presentadas por el equipo.

Criterios de entrada

Disparidades revisadas, calendario del proyecto, presupuesto del proyecto, requisitos del proyecto.

Criterios de salida

Repositorio de activos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

3.4.2. Supervisión del Proyecto**Procesos de MeSoFT**

Otro proceso incluido en MeSoFT es la *Supervisión del Proyecto*. El propósito de esta área es proporcionar un entendimiento del progreso de los proyectos de tal forma que puedan tomarse las acciones correctivas apropiadas cuando el rendimiento de los mismos se desvíe significativamente del plan.

Un plan documentado del proyecto es la base para supervisar las actividades, comunicar el estado y tomar acciones correctivas. El progreso del proyecto se determina comparando el producto actual y los atributos de la tarea, esfuerzo, coste, y calendario con el plan en hitos preestablecidos o niveles de control dentro del calendario del proyecto o la lista de actividades desarrollada. La visibilidad apropiada permite tomar acciones correctivas en el momento oportuno cuando el rendimiento se desvía significativamente del plan. Una desviación es significativa si, cuando se deja sin resolver, impide que el proyecto alcance sus objetivos. Cuando el estado actual se desvía significativamente de los valores esperados, las acciones correctivas son tomadas según resulte apropiado. Estas acciones pueden requerir la replanificación, la cual puede incluir la revisión del plan original, el establecimiento de nuevos acuerdos, o la inclusión de actividades de corrección.

Relaciones con otras áreas de MeSoFT

Remitirse a la Planificación del Proyecto para mayor información sobre el plan del proyecto, incluyendo cómo especificar el nivel apropiado de supervisión del proyecto, las medidas usadas en la supervisión, y los riesgos conocidos.

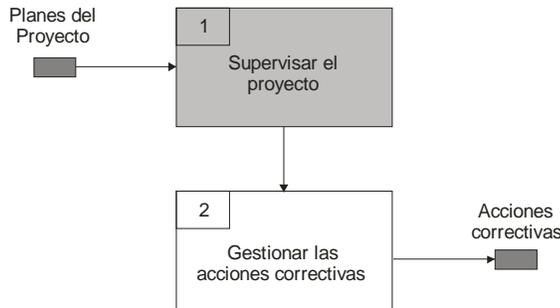
Remitirse a la Medición y Análisis para mayor información sobre los procesos de medición, análisis y almacenamiento de información.

En esta sección

En esta sección del documento se describe información detallada de las actividades del proceso de Supervisión del Proyecto.

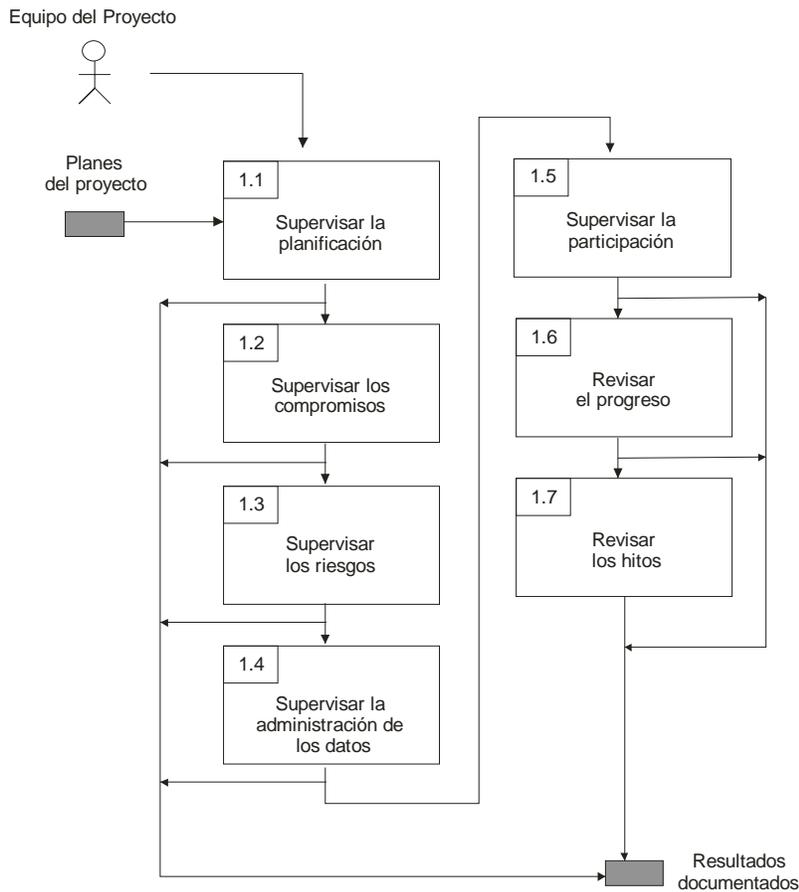
Actividad	Ver página
1 Supervisar el proyecto	66
2 Gestionar las acciones correctivas	72

Diagrama general de actividades



3.4.2.1. Supervisar el proyecto

Diagrama de actividades



Supervisar la planificación [1.1]

Propósito El propósito de la actividad es supervisar el rendimiento actual y el progreso del proyecto contra el plan del proyecto formulado.

Roles de los participantes Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Calendario (CALEN)	Interna (Repositorio de activos)
Diseño conceptual (DIS)	Interna (Repositorio de activos)
ACTI con estimaciones de coste y esfuerzo	Interna (Repositorio de activos)
Plan de la gestión de recursos (ACTI con lo que hace el plan)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plantillas con valores actualizados	Los valores actuales son utilizados como “datos históricos” para proporcionar una base para las estimaciones de la planificación del proyecto.

Criterios de entrada Alcance del proyecto, actividades y productos identificados, estimaciones descripciones de los productos y actividades, planes del proyecto.

Criterios de salida Valores actuales vs. Valores estimados.

Criterios de evaluación Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Supervisar los compromisos [1.2]

Propósito El propósito de la actividad es supervisar los compromisos contra aquellos identificados en el plan del proyecto.

Roles de los participantes Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Compromisos documentados	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Compromisos revisados y actualizados	Es necesario supervisar los compromisos y tomar acciones correctivas cuando estos cambian.

Criterios de entrada

Compromisos externos e internos.

Criterios de salida

Compromisos no cumplidos o en riesgo de no cumplirse.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Supervisar los riesgos [1.3]**Propósito**

El propósito de la actividad es verificar el estado de los riesgos identificados en el plan del proyecto.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Riesgos identificados (RIE)	Interna (Repositorio de activos)
Plan de participación (INV)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Riesgos identificados (RIE) actualizados	La revisión a los riesgos permite seguir de cerca el estado de los mismos, si existiese algún cambio en la probabilidad de ocurrencia o en la prioridad del riesgo esta debe ser comunicada inmediatamente a los participantes.

Criterios de entrada

Estado actual de los riesgos.

Criterios de salida

Riesgos revisados.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Supervisar la administración de los datos [1.4]

Propósito

El propósito de la actividad es supervisar el plan formulado para utilizar y mantener el repositorio de activos del proyecto en donde serian manejados, actualizados y homogeneizados los datos desde el inicio hasta el avance del proyecto.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Formatos de contenido	Externa
Plan de la administración de los datos (DAT)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plan actualizado de la administración de los datos (DAT)	Una vez que se ha terminado el plan para la administración de los datos del proyecto, la gestión debe ser supervisada para asegurar que los planes son logrados.

Criterios de entrada

Descripciones de las actividades para la administración de los datos.

Criterios de salida

Problemas e impactos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Supervisar la participación [1.5]

Propósito

El propósito de la actividad es supervisar la participación de los involucrados en las actividades establecidas en el plan del proyecto.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Plan de participación (INV)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plan actualizado de participación (INV)	La revisión de los planes permitirá asegurar que se están realizando las interacciones apropiadas entre los participantes identificados previamente.

Criterios de entrada

Descripciones de las actividades de los participantes.

Criterios de salida

Problemas e impactos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Revisar el progreso [1.6]**Propósito**

El propósito de la actividad es revisar periódicamente el progreso, rendimiento y los problemas del proyecto.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Ciclo de vida del proyecto (ACTI)	Interna (Repositorio de activos)
Plan de participación (INV)	Interna (Repositorio de activos)
Plan de la administración de recursos (TASK con lo que hace el plan)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plantillas actualizadas	Las plantillas mantendrán informados a los implicados sobre el avance del proyecto.

Criterios de entrada

Lista de participantes, medidas para controlar el proyecto.

Criterios de salida

Problemas e impactos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Revisar los hitos [1.7]

Propósito

El propósito de la actividad es supervisar los logros y resultados del proyecto en hitos seleccionados previamente.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Calendario (CALEN)	Interna (Repositorio de activos)
Plan del proyecto (PROY)	Interna (Repositorio de activos)
Compromisos documentados	Interna (Repositorio de activos)
Riesgos identificados (RIE)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Resultados de las revisiones	Los hitos son eventos significativos dentro del proyecto que ayudarán a determinar el cumplimiento de las actividades dentro de los planes.

Criterios de entrada

Hitos predeterminados en el Ciclo de Vida seleccionado.

Criterios de salida

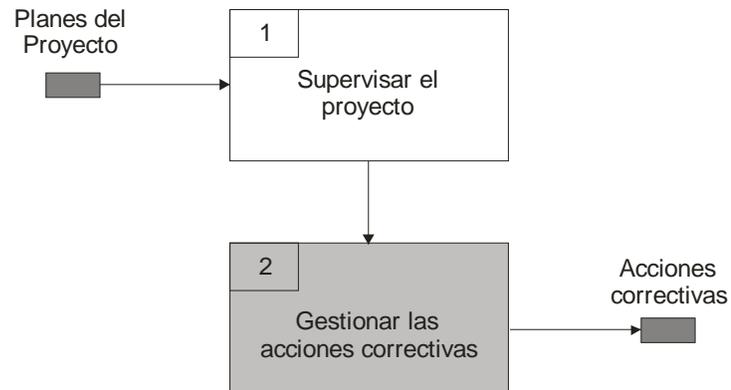
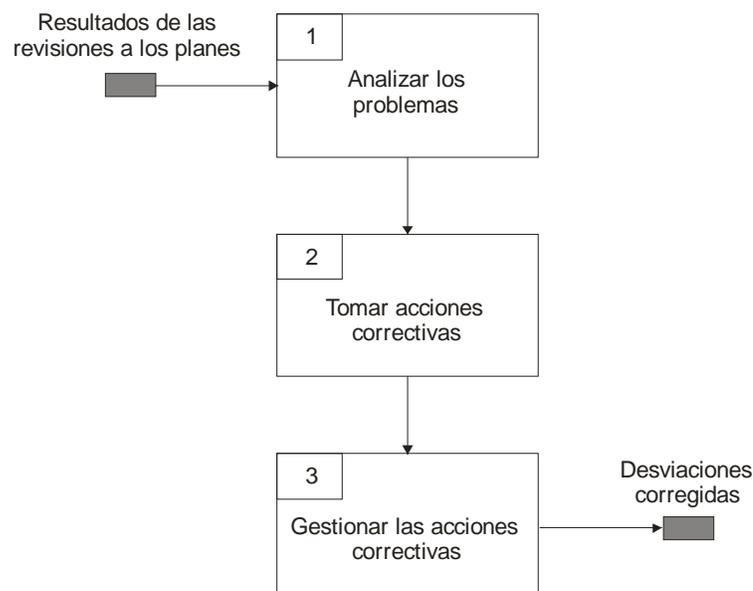
Problemas e impactos.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Diagrama general de actividades

El seguimiento y control del proyecto está compuesta de dos actividades. La segunda de éstas se muestra a continuación:

**3.4.2.2. Gestionar las acciones correctivas****Diagrama de actividades****Analizar los problemas [2.1]****Propósito**

El propósito de la actividad es recoger y analizar los problemas y determinar las acciones correctivas necesarias para tratarlos.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Resumen del tamaño del producto (TAM)	Interna (Repositorio de activos)
Resumen de calidad (CAL)	Interna (Repositorio de activos)
Calendario (CALEN)	Interna (Repositorio de activos)
Compromisos documentados	Interna (Repositorio de activos)
Plan del proyecto (PROY)	Interna (Repositorio de activos)
Riesgos identificados (RIE)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Acciones correctivas	La gestión de las acciones correctivas hasta el cierre es un punto crítico. No es suficiente identificar las tareas, es necesario confirmar que han sido completadas.

Criterios de entrada

Plantillas del repositorio.

Criterios de salida

Problemas identificados en las plantillas.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Tomar acciones correctivas [2.2]**Propósito**

El propósito de la actividad es tomar acciones correctivas sobre los problemas identificados.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Diseño conceptual (DIS)	Interna (Repositorio de activos)
Resumen de calidad (CAL)	Interna (Repositorio de activos)
Calendario (CALEN)	Interna (Repositorio de activos)
Compromisos documentados	Interna (Repositorio de activos)
Plan del proyecto (PROY)	Interna (Repositorio de activos)
Riesgos identificados (RIE)	Interna (Repositorio de activos)
Plan de participación (INV)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Plan de acciones correctivas (CORRE)	En algunos casos, las acciones correctivas pueden ser supervisar la situación.

Criterios de entrada

Plantillas del repositorio.

Criterios de salida

Problemas identificados, acciones correctivas identificadas (en plantillas).

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Gestionar las acciones correctivas [2.3]**Propósito**

El propósito de la actividad es gestionar las acciones correctivas planificadas hasta el cierre de las mismas.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Plan de acciones correctivas (CORRE)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Resultados de las acciones correctivas	Los resultados deben ser analizados para determinar la efectividad de las acciones correctivas.
Lecciones aprendidas	Las lecciones aprendidas al aplicar las acciones correctivas pueden servir de entradas a los procesos de Planificación y Gestión del Riesgo.

Criterios de entrada

Plan de acciones correctivas, acciones a mitigar.

Criterios de salida

Plan de acciones correctivas, acciones establecidas.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

3.4.3. Medición y Análisis**Procesos de MeSoFT**

La tercera área de proceso de MeSoFT es *Medición y Análisis*. El uso de esta área se enfoca a satisfacer las necesidades de medir el progreso del proyecto, el tamaño del producto o la calidad, o el rendimiento del proceso al apoyar la toma de decisiones y acciones correctivas. El objetivo de la *Medición y Análisis* es desarrollar una capacidad de medición entre el equipo de desarrollo que sea utilizada para apoyar sus problemas en la gestión de la información del proyecto. Los objetivos de realizar la medición en un proyecto de software educativo se enfocan en reducir el tiempo de entrega, entregar la funcionalidad completa, mejorar los niveles anteriores de calidad, y demás.

Relaciones con otras áreas de MeSoFT

Remitirse a la Planificación del Proyecto para mayor información sobre la estimación de parámetros y otras necesidades de la planificación.

Remitirse a la Supervisión del Proyecto para mayor información sobre la determinación de variaciones en el rendimiento del proyecto.

Remitirse a la Gestión de Requisitos para mayor información sobre las necesidades de los alumnos y cómo mantener la trazabilidad de éstas a lo largo del proyecto.

Remitirse a la Definición del Proceso para mayor información sobre cómo establecer el repositorio de información del producto educativo.

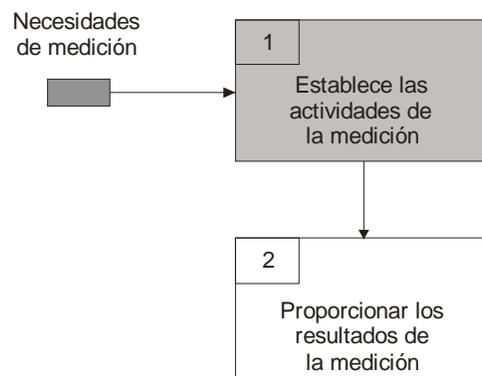
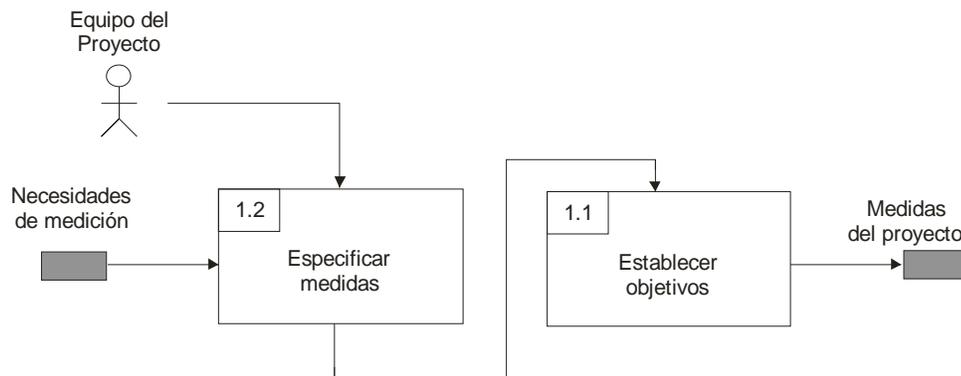
En esta sección

En esta sección se describe información detallada de las actividades del proceso de Medición y Análisis.

Actividad	Ver página
1 Establecer las actividades de la medición	76
2 Proporcionar los resultados de la medición	78

Diagrama general de actividades

El proceso de medición y el análisis está compuesto de dos actividades que se presentan en el siguiente diagrama.

**3.4.3.1. Establecer las actividades de la medición****Diagrama de actividades****Establecer objetivos [1.1]****Propósito**

El propósito de la actividad es establecer objetivos que documenten las razones por las cuales se están realizando la medición y el análisis. Las fuentes de estos objetivos pueden ser la gestión técnica, o las necesidades del proyecto, producto, o implementación del proceso.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Compromisos documentados	Interna (Repositorio de activos)
Plan del proyecto (PROY)	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Documento de necesidades y objetivos (NEC)	Los objetivos de la medición son documentados y revisados por todos los participantes en el proyecto, y actualizados si fuera necesario.

Criterios de entrada

Plan del proyecto, registro de la supervisión del proyecto, entrevistas con participantes inconformes.

Criterios de salida

Necesidades y objetivos de la medición documentados, base de datos actualizada.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Especificar medidas [1.2]**Propósito**

El propósito de la actividad es refinar los objetivos establecidos anteriormente en medidas cuantificables. Las medidas podrán ser “base” o “derivadas”.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

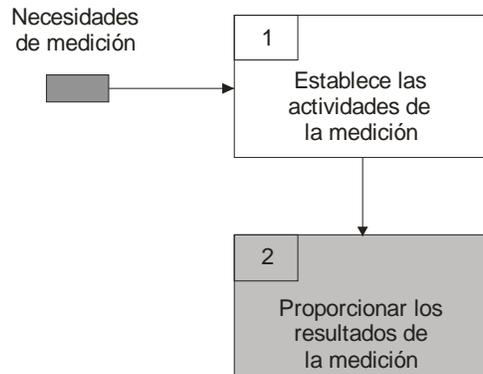
Entradas

Entrada	Fuente
Documento de necesidades y objetivos (NEC)	Interna (Repositorio de activos)

Salidas

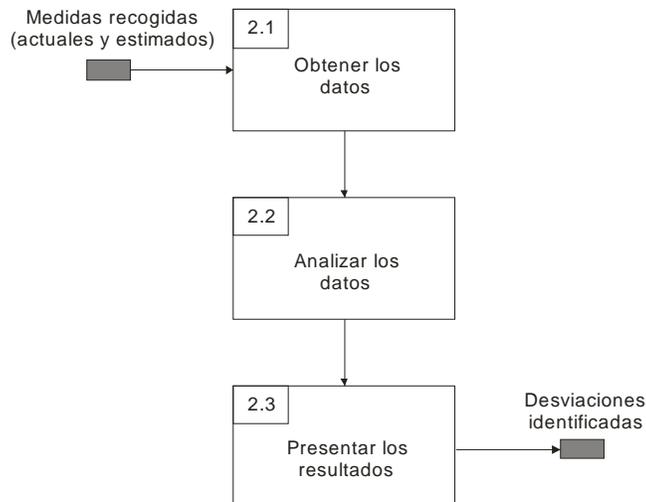
Salida	Descripción
Lista de medidas base y derivadas	La lista obtenida especifica únicamente las medidas que es necesario recoger para los productos de tipo educativo.

Criterios de entrada	Repositorio de medidas.
Criterios de salida	Medidas ubicadas en plantillas de control, base de datos actualizada.
Criterios de evaluación	Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).
Diagrama de actividades	El proceso de medición y el análisis está compuesto de dos actividades. La segunda de éstas se muestra a continuación:



3.4.3.2. Proporcionar los resultados de la medición

Diagrama de actividades



Obtener los datos [2.1]

Propósito

El propósito de la actividad es obtener información que ayude en la supervisión del rendimiento, el cumplimiento de las actividades acordadas, facilitar la toma de decisiones sobre las gestiones pedagógica y técnica, y tomar acciones correctivas.

Roles de los participantes

Equipo de proyecto.

Entradas

Entrada	Fuente
Lista de medidas base y derivadas	Interna (Repositorio de activos)
Plantillas que requieran toma de medidas	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Documento de resultados	El objetivo de este documento es registrar los valores recalculados para las medidas reflejadas en las plantillas de MeSoFT.

Criterios de entrada

Repositorio de medidas.

Criterios de salida

Medidas ubicadas en plantillas de control, base de datos actualizada.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Analizar los datos [2.2]**Propósito**

Los propósitos de la actividad son analizar los datos, conducir análisis adicionales si es necesario, y revisar los resultados con todos los participantes técnicos en el proyecto. En caso de que se requiera una revisión futura, la actividad lo registrará.

Roles de los participantes

Líder de equipo.
Equipo de proyecto.

Entradas

Entrada	Fuente
Documento de resultados	Interna (Repositorio de activos)
Lista de medidas base y derivadas	Interna (Repositorio de activos)
Plantillas que requieran toma de medidas	Interna (Repositorio de activos)

Salidas

Salida	Descripción
Reporte de análisis	Después de varios ciclos, los resultados de los análisis planeados pueden sugerir (o requerir) la refinación de las medidas.

Criterios de entrada

Repositorio de medidas.

Criterios de salida

Medidas ubicadas en plantillas de control, base de datos actualizada.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

Presentar los resultados [2.3]**Propósito**

El propósito de la actividad es informar a tiempo a los participantes del proyecto sobre las posibles desviaciones en los parámetros técnicos del proyecto y asistir al equipo en la toma de decisiones y acciones correctivas.

Roles de los participantes

Equipo de proyecto.

Entradas

Entrada	Fuente
Documento de resultados	Externa
Reporte de análisis	Externa

Salidas

Salida	Descripción
Reporte de entrega (incluye información de guía para la interpretación de los resultados)	Los resultados de la medición son comunicados a tiempo para que sean utilizados de acuerdo a los criterios establecidos al inicio del proyecto.

Criterios de entrada

Repositorio de medidas.

Criterios de salida

Medidas ubicadas en plantillas de control, base de datos actualizada.

Criterios de evaluación

Utilizar la medida de tiempo (diferencia entre tiempo real y estimado).

3.5. IADIAL

Para el área de proceso de la Gestión Pedagógica hemos propuesto un modelo de prácticas efectivas para reflejar los aspectos pedagógicos necesarios para el desarrollo del SE. El mecanismo pedagógico de MeSoFT se basa en la Resolución Técnica del Problema con modificaciones orientadas a mejorar los resultados de aprendizaje a distancia; hemos llamado a este modelo IADIAL (*IADIAL, Initiating, Analyze, Design, Implementing, Assessment, and Learning*) y tiene las etapas de Inicialización, Análisis, Diseño, Implementación, Evaluación y Aprendizaje (véase Figura 3.8). IADIAL se compone de las siguientes etapas:

- **Inicialización:** MeSoFT utiliza una variación de la Técnica de Resolución de Problemas como una técnica didáctica para identificar una idea y resolver un problema específico. Esta solución se establece por los profesores y los estudiantes mediante el establecimiento de un objetivo de aprendizaje basado en valoraciones.

El establecimiento de una estrategia pedagógica consiste primero en identificar los problemas en el proceso enseñanza-aprendizaje actual; después se identifica la forma en que los alumnos suelen resolver estos problemas y el especialista propone una serie de soluciones alternativas; las consecuencias de las soluciones propuestas (ventajas y desventajas) son ponderadas de acuerdo al factor personal, social, a corto plazo o largo plazo, si la consecuencia que se obtiene es personal, se darán 2 puntos; si es social, se dará 1 punto; si es a largo plazo, se puntuará con 2 puntos; si es a corto plazo, 1 punto; es necesario tener en cuenta estas combinaciones: si la consecuencia es al mismo tiempo personal y a largo plazo, se darán 4 puntos; si la consecuencia es al mismo tiempo personal y a corto plazo, la puntuación es de 3 puntos; si es social y a largo plazo, se puntuará con 3; y si es social y a corto plazo, con 2 puntos; otras combinaciones quedan fuera de la ponderación. El último paso consiste en probar las soluciones con la valoración más alta (en ventajas).

- **Análisis:** El AP de la Gestión de Requisitos proporciona una guía para obtener las necesidades específicas de software. Esta etapa se centra en las siguientes cuestiones:
 - *Objetivos y características de los estudiantes:* Los datos como edad, sexo, condición social, lecciones aprendidas, las expectativas, intereses o motivaciones para aprender.
 - *Medio ambiente del estudiante:* nivel académico, situación psicológica, ambiente familiar y de estudio.
 - *Problema a resolver:* MeSoFT proporciona plantillas para identificar los problemas a través de entrevistas y análisis de los resultados académicos.
 - *Principios pedagógicos de medición:* El AP de Medición y Análisis se utiliza por los estudiantes para reunir datos sobre los actuales principios pedagógicos aplicados para resolver el problema identificado. Se utilizan estos datos en la etapa de Evaluación para garantizar que el SE genera un impacto positivo en el proceso de aprendizaje.

El producto final de esta etapa es el conjunto de funciones y requisitos específicos. Esta especificación recomienda el estándar IEEE Std. 830-1998 para establecer un modelo de plantilla que incluya los conceptos anteriores.

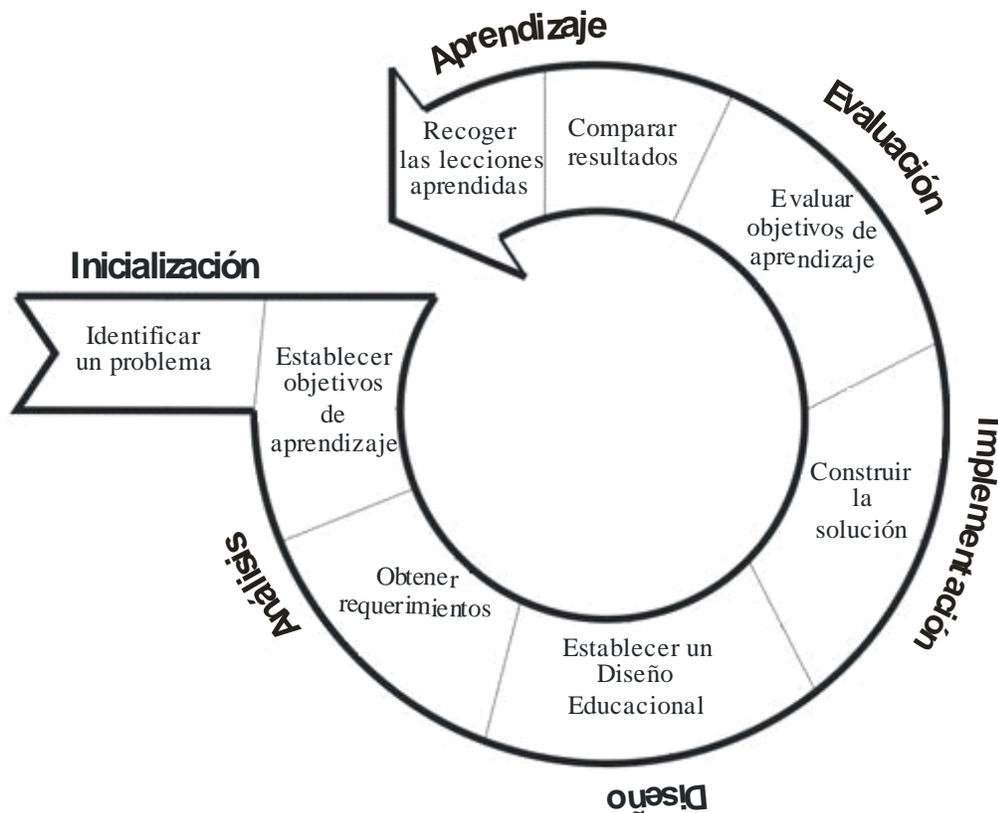


Figura 3.8. Modelo IADIAL.

- **Diseño:** Esta etapa establece un objetivo de aprendizaje a través de tres niveles:
 - *Diseño Educativo:* Este nivel proporciona un apoyo para obtener los contenidos educativos y temas de estructura, y establecer una motivación y sistema de evaluación.
 - *Diseño de la interfaz:* Este nivel se recomienda una estructura básica para la GUI de acuerdo a las necesidades identificadas.
 - *Diseño Computacional:* Este nivel define la composición interna de programas informáticos educativos. La etapa de diseño se basa en la Técnica del Modelado de Objetos (OMT, *Object-Modeling-Technique*) [Rumbaugh97] para proporcionar un conocimiento sólido de diseño de software. La Gestión de la Información está compuesta de muchos activos para obtener objetos, funcional y dinámica de modelos a fin de reflejar todos los requisitos definidos.
- **Implementación:** El diseñador de software puede desarrollar los programas educativos sobre la base de todos los conocimientos previos, seleccionando cualquier lenguaje de programación.
- **Evaluación:** En este punto, el diseñador del software debe demostrar dos cosas: primeramente que el software es funcional y que el objetivo de aprendizaje se ha cumplido. Las AP de Validación y Verificación proporcionan todas las prácticas para probar el producto de educación obtenido. La segunda parte trata de probar el software educativo en un entorno real y comparar los datos de medición de cursos anteriores con los resultados obtenidos mediante la nueva herramienta.

- **Aprendizaje:** MeSoFT recoge todas las prácticas utilizadas en problemas específicos, no sólo para el éxito de los proyectos. Un proyecto fallido establece los conocimientos necesarios para evitar los mismos problemas similares en un contexto pedagógico.

Las prácticas que reflejan la estructura interna de IADIAL son las que se resumen en el área de proceso de la Gestión Pedagógica.

3.6. Definición de las Áreas de Proceso de la Gestión Pedagógica

3.6.1. Gestión de Requisitos

Procesos de MeSoFT

La primera área de proceso de la Gestión Pedagógica de MeSoFT es la *Gestión de Requisitos*. La obtención de los requisitos comienza con el plan acordado que define la idea de lo que se va a realizar de proyecto. Incluye la recopilación de los requisitos de los contenidos educativos, y obtiene las características de los alumnos o usuarios finales del proyecto, también se relaciona con los reportes de validación y verificación en determinado momento del proyecto. Los cambios en los requisitos pueden irse generando de acuerdo vaya avanzando el proyecto, deberá tener una constante comunicación con la Planificación del Proyecto ya que si existiese un cambio en los requisitos deberá ser informado para cambiar la planificación original.

Relaciones con otras áreas de MeSoFT

Remitirse a la Supervisión del Proyecto para mayor información sobre las acciones correctivas generadas en la supervisión.

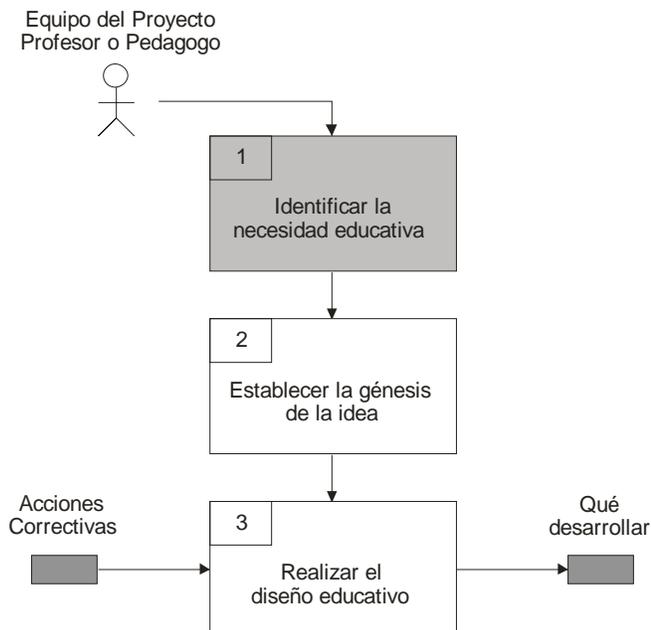
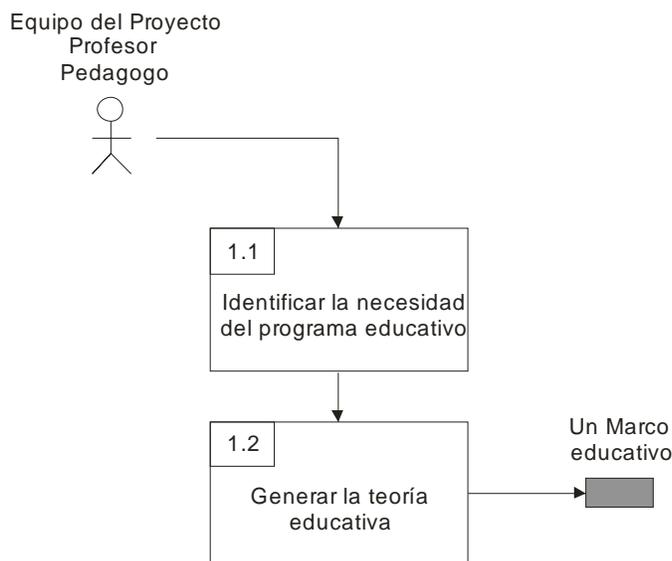
Remitirse a la Planificación del Proyecto para tener mayor información sobre el Plan acordado del proyecto.

Remitirse a la Validación y Verificación para la generación y obtención de reportes.

En esta sección

En esta sección se describe información detallada de las actividades del proceso de Gestión de Requisitos.

Actividad	Ver página
1 Identificar la necesidad educativa	84
2 Establecer la génesis de la idea	86
3 Realizar el diseño educativo	88

Diagrama general de actividades**3.6.1.1. Identificar de la necesidad educativa****Diagrama de actividades****Identificar la necesidad del programa educativo [1.1]****Propósito**

El propósito de la actividad es identificar si existe una necesidad para crear un programa o software destinado a la educación para lograr el objetivo pedagógico actual y los posteriores.

Roles de los participantes

Profesor de área.

Pedagogo.

Líder de equipo.

Entradas

Entrada	Fuente
Entrevista a los alumnos (ALU)	Interna (repositorio de activos)
Entrevista a los profesores (PROF)	Interna (repositorio de activos)

Salidas

Salida	Descripción
Problemas detectados (PROB)	El documento permite identificar el problema o la falta de recursos educativos para lograr una mejor comprensión de los conocimientos.

Criterios de entrada

Fallas detectadas en el proceso de enseñanza-aprendizaje del alumno.

Criterios de salida

El problema identificado, modificación de la base de información.

Criterios de evaluación

Utilizar como medida la facilidad de los alumnos/profesores a trabajar con la tecnología.

Generar la teoría educativa [1.2]**Propósito**

El propósito de la actividad es obtener los fundamentos pedagógicos acerca de cómo subsanar el problema identificado.

Roles de los participantes

Profesor.
Pedagogo.
Líder de proyecto.

Entradas

Entrada	Fuente
Bibliografía sobre el tema en específico	Externa
Análisis de los contenidos educativos	Externa
Problemas detectados (PROB)	Interna (repositorio de activos)

Salidas

Salida	Descripción
Marco educativo y comunicacional (MAR)	El documento permite saber cual es el entorno educativo, define las técnicas utilizadas en el proceso de enseñanza–aprendizaje y define cuales son las mejores formas de presentar el tema ante los alumnos.

Criterios de entrada

Identificación del problema, libros acerca del tema en específico, formas de solucionarlo en la forma tradicional

Criterios de salida

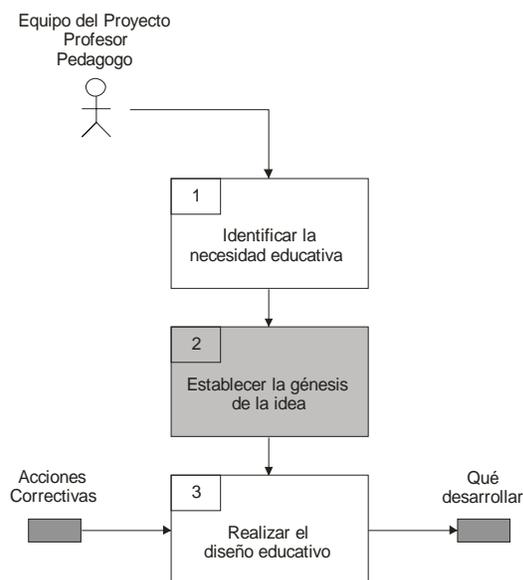
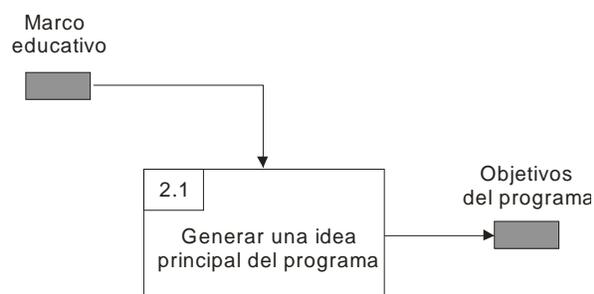
Información de los contenidos educativos.

Criterios de evaluación

Utilizar la medida de comparar el plan de estudios con el documento generado.

Diagrama general de actividades

La Gestión de Requisitos está compuesta de tres actividades. A continuación se describe la segunda de éstas:

**3.6.1.2. Establecer la génesis de la idea****Diagrama de actividades**

Generar una idea principal del programa [2.1]

Propósito El propósito de la actividad es partir de una idea inicial que parece potencialmente poderosa para favorecer los procesos de enseñanza/aprendizaje.

Roles de los participantes
 Profesor
 Pedagogo
 Líder de equipo

Entradas

Entrada	Fuente
Marco educativo y comunicacional (MAR)	Interna (repositorio de activos)

Salidas

Salida	Descripción
Lista de objetivos curriculares (CUR)	Se debe generar una lista con objetivos alcanzables mediante el uso de determinados programas didácticos conocidos, también debe generar otra lista con objetivos que consideren que podrían conseguirse mejor si existiesen determinados programas que, en este caso, se deberán inventar.

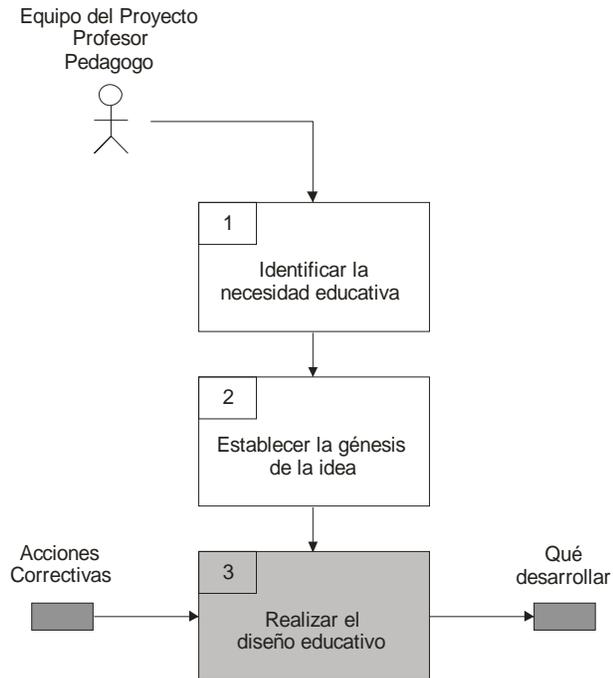
Criterios de entrada Programas didácticos conocidos (si existiesen).

Criterios de salida Lista de objetivos curriculares almacenados en el repositorio de datos.

Criterios de evaluación Utilizar la medida de comparación entre la lista de objetivos curriculares generada y la lista de objetivos curriculares planteados en el programa general de estudio.

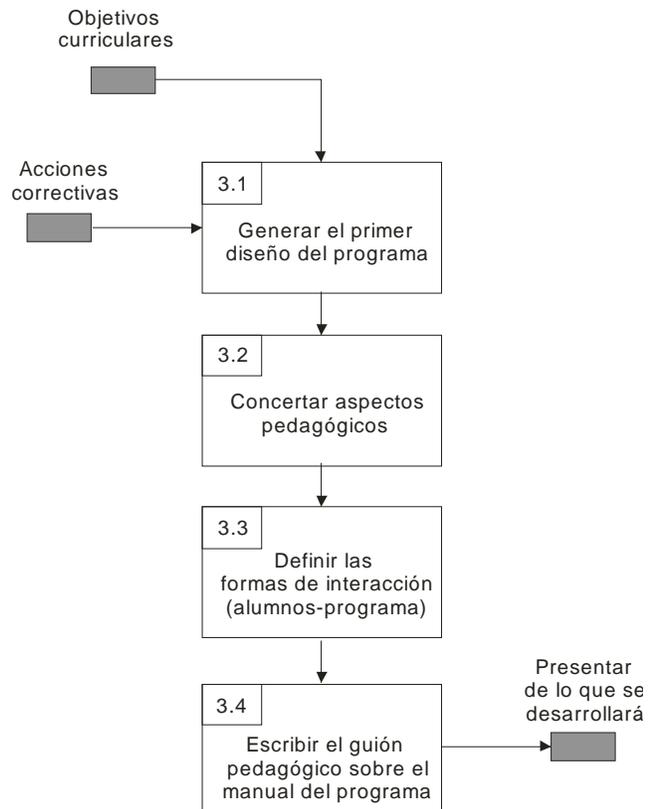
Diagrama general de actividades

La gestión de requisitos está compuesta de tres actividades. A continuación se describe la tercera de estas:



3.6.1.3. Realizar el Diseño educativo

Diagrama de actividades



Generar el primer diseño del programa [3.1]**Propósito**

El propósito de la actividad es aplicar la técnica del “brainstorming” en donde se pueden considerar aspectos pedagógicos y funcionales como las motivaciones, las primeras reflexiones sobre los contenidos y los objetivos, las posibles actividades interactivas y el primer borrador de las pantallas y del entorno de comunicación en general, si es necesario intervienen las modificaciones sobre el diseño inicial del programa.

Roles de los participantes

Profesor.
Pedagogo.
Líder de equipo.

Entradas

Entrada	Fuente
Lista de objetivos curriculares (CUR)	Interna (repositorio de activos)
Diseño conceptual (DIS)	Interna (repositorio de activos)

Salidas

Salida	Descripción
Diseño computacional (MOD)	Se debe generar un primer bosquejo del diseño del programa que se quiere representar, o bien si existen cambios sobre el prediseño del programa (Ver Actividad de “Diseño Computacional” en el modelo IADIAL de MeSoFT).

Criterios de entrada

Software educativo cercano al que se quiere hacer, bibliografía sobre diseño de software educativo, bibliografía sobre el tema en específico que se piensa tratar en el programa, diagramas de modelado, arquitectural de sistema y demás.

Criterios de salida

Plantillas de diseño, que ayudan en el proceso para concretar el proyecto y almacenar en el repositorio de datos (Ver plantillas de la Planificación del Proyecto).

Criterios de evaluación

Dependen del tema específico.

Concertar aspectos pedagógicos [3.2]**Propósito**

El propósito de la actividad es retomar las listas de objetivos curriculares y definir los objetivos reales, los contenidos válidos, los alumnos usuarios del programa y la estrategia didáctica que se piensa utilizar.

Roles de los participantes

Profesor.
Pedagogo.
Líder de equipo.

Entradas

Entrada	Fuente
Lista de objetivos curriculares (CUR)	Interna (repositorio de activos)

Salidas

Salida	Descripción
Objetivos educativos (EDU)	Determinar los objetivos educativos mediante la evaluación de la eficacia del programa y la organización de los contenidos.
Definición de perfiles	Determinar a los alumnos usuarios del programa, esto es definir aspectos como edad, nivel de desarrollo cognoscitivo (nivel de madurez), conocimientos previos y capacidades generales así como el nivel educativo, los conocimientos relacionados con la temática del programa, la estructura cognoscitiva, la capacidad intelectual (nivel de inteligencia general y factorial), las actitudes, intereses, hábitos de estudio y organización y las discapacidades o deficiencias.
Definición de contenidos (utilizar los modelos de objetos y dinámico de MOD)	Fijar los contenidos considerando los aspectos más fáciles y concretos hasta los más complejos y abstractos. O partir de los elementos conocidos por los alumnos a los que les son desconocidos. También se puede ir de las presentaciones globales o sintéticas a las visiones analíticas. O ir de las visiones episódicas a las sistemáticas. Ir de los que requieren el uso de habilidades globales a los que implican el uso de habilidades específicas. Contenidos que vayan destacando las relaciones interdisciplinarias, contemplando niveles de dificultad. Revisando la estructura del programa por ejemplo, si es un programa de tipo base de datos: incluir la estructura de las bases de datos, las interrelaciones entre ellas, las formas de acceso a los datos (búsqueda, ordenación, clasificación, captura, etc.). Si es un simulador: los modelos que presentará y la organización de los conceptos (que deberán resultar claros y

	<p>adecuados al nivel de abstracción de los alumnos), las variables con que se trabajará y las interrelaciones que se podrán representar ininterrumpidamente por medio de fórmulas, con tablas de comportamiento, mediante grafos, etc. Por último si es un constructor: los elementos que contemplará y las propiedades o los comportamientos que tendrá cada uno.</p>
<p>Actividades a realizar con los alumnos y la computadora (reflejadas en CUR)</p>	<p>Definir actividades mentales que los alumnos desarrollarán delante de la computadora para aumentar la calidad didáctica de las actividades interactivas que se diseñen como son el ejercitar habilidades psicomotrices, observar, percibir el espacio y el tiempo y orientarse en ellos, reconocer, identificar, señalar, recordar, explicar, describir, reconstruir, memorizar (hechos, datos, conceptos, teorías, etc.).</p>
<p>Actividades interactivas que debe proponer el programa (reflejadas en CUR)</p>	<p>Definir actividades interactivas que debe proponer el programa para propiciar el intercambio de información entre los alumnos y la máquina y que permitan que las acciones de los estudiantes puedan ser valoradas y tratadas por el programa, definiendo aspectos como la naturaleza de las actividades educativas, la estructura de las actividades, las acciones y respuestas permitidas al alumno, la duración y el tipo de control del aprendizaje.</p>
<p>Estrategia pedagógica (PED)</p>	<p>Proponer caminos pedagógicos dando a los alumnos bifurcaciones que permitan seguir diferentes itinerarios pedagógicos y que faciliten: la elección de los temas y de las actividades, la reformulación de los conceptos, el cambio de la secuencia de los contenidos, el repaso de puntos mal comprendidos, la selección del nivel de dificultad, repasar y profundizar. La determinación del recorrido puede ser de manera explícita, es decir por decisión de los alumnos o de manera implícita, es decir en función de las respuestas de los alumnos.</p>
<p>Definir el tratamiento de los errores (RIE)</p>	<p>En caso de optar por la determinación del recorrido de manera implícita, los errores se pueden tratar dependiendo del tipo de corrección (corrección sin ayuda y corrección con ayuda). Según la valoración que haga el error pueden ser de valoración mediante mensajes, valoración por</p>

	medio de elementos cuantitativos o valoración por medio de efectos musicales y visuales y según la naturaleza del error pueden ser errores de conocimiento, errores de comprensión, errores de análisis, errores de procedimiento o errores de ejecución.
Elementos motivadores (PED)	Definir elementos motivadores que son los motores del aprendizaje y son un buen antídoto contra el fracaso escolar, se pueden utilizar elementos que presentan un reto, elementos que estimulan la curiosidad o la fantasía, elementos que representan un estímulo o una penalización social o manipular un ritmo variado y progresivo del programa.
Integración curricular	Analizar la integración curricular tomando en cuenta las características de los alumnos y los objetivos curriculares.

Criterios de entrada

Lista de objetivos curriculares.

Criterios de salida

Objetivos educativos, definición de perfiles, definición de contenidos, la estrategia pedagógica.

Criterios de evaluación

Por determinar.

Definir las formas de interacción (alumnos-programa) [3.3]**Propósito**

El propósito de la actividad es crear el entorno de comunicación y realizar el dialogo entre los estudiantes y el programa.

Roles de los participantes

Profesor.
Pedagogo.
Líder de equipo.

Entradas

Entrada	Fuente
Objetivos educativos (EDU)	Interna (repositorio de activos)
Diseño computacional (MOD)	Interna (repositorio de activos)
Estrategia pedagógica (PED)	Interna (repositorio de activos)
Integración curricular	Externa

Salidas

Salida	Descripción
Diseño del storyboard (HIS)	Realizar un primer diseño de las pantallas determinando zonas que realizarán funciones específicas y que se repetirán (si es posible) en todas las pantallas del programa, por ejemplo definir la zona de comentarios, la zona de órdenes, la caja de herramientas y la zona de trabajo
Importancia de los accesorios hardware (utilizar el modelo funcional de MOD)	Uso del teclado y del ratón para crear un entorno de comunicación con el programa el cual resulte muy fácil de usar y agradable al alumno. Se debe establecer una sintaxis sencilla e intuitiva y prever un sistema de ayuda para el manejo del programa tomando en cuenta que las acciones y respuestas por parte de los alumnos pueden ser por selección de entre las opciones que ofrece el programa por la pantalla, por preguntas del tipo si/no, por cuestionarios de respuesta múltiple, por menús de opciones, también pueden ser con producción de respuesta como mover algún elemento por la pantalla, cambiar un objeto de lugar, trazar una trayectoria o establecer correspondencias entre listas, asociaciones, ordenar palabras, etc.
Importancia de periféricos complementarios (utilizar el modelo funcional de MOD)	Describir la función de los diferentes periféricos complementarios que se utilicen como por ejemplo la impresora, el teclado conceptual, el lector de tarjetas, el micrófono, el reconocedor de voz, video, CD-ROM, lápiz óptico, pantalla táctil, MODEM, convertidores analógico-digitales, etc.
Criterios de entrada	Listas de objetivos curriculares.
Criterios de salida	Definición de los elementos externos a utilizar con el programa, así como el primer diseño de las pantallas a utilizar.
Criterios de evaluación	La medida debe ser tomada de acuerdo a la necesidad de ayuda de dispositivos auxiliares hacia a los alumnos y a la forma de representar el programa.

Escribir el guión pedagógico sobre el manual del programa [3.4]**Propósito**

El propósito de la actividad es realizar un esquema con una primera aproximación al formato y al contenido de la documentación que acompañará al programa.

Roles de los participantes

Profesor.
Pedagogo.
Líder de equipo.

Entradas

Entrada	Fuente
Diseño computacional (MOD)	Interna (repositorio de activos)
Objetivos educativos (EDU)	Interna (repositorio de activos)

Salidas

Salida	Descripción
Manual del programa	Es un esquema que incluye una ficha resumen que recoge las principales características del programa, el manual de usuario donde se debe explicar todo lo que necesita saber el usuario del programa para utilizarlo sin problemas y la guía práctica que es un documento dirigido al personal docente en donde se ofrecen sugerencias sobre la integración curricular del programa, sus formas de uso, actividades complementarias, estrategias para evaluar el rendimiento de las situaciones educativas que genera el programa, etc.
Presentación	El autor desarrollará una exposición general del programa que se piensa desarrollar y se tomarán en cuenta aspectos como la descripción sintética del programa y de sus objetivos, rasgos más característicos, la motivación del proyecto, el guión general y por último el hardware y software necesario.

Criterios de entrada

Se debe contar con todos los datos almacenados en el repositorio de datos.

Criterios de salida

El documento de la presentación, así como las versiones de manual de usuario del programa.

Criterios de evaluación

La medida de evaluación se tomará de acuerdo al contenido de la presentación comparándolo con lo planteado originalmente.

3.6.2. Verificación

Procesos de MeSoFT

La segunda área de proceso de la Gestión Pedagógica de MeSoFT es la de *Verificación*. La *Verificación* se lleva a cabo una vez que se tenga realizado el proceso de la Gestión de Requisitos y tiene como propósito el asegurar que el trabajo o producto cumple con los requisitos especificados

Relaciones con otras áreas de MeSoFT

Remitirse a la Gestión de Requisitos para obtener más información sobre los requisitos del proyecto.

Remitirse a la Validación para mayor información sobre la forma de confirmar que un producto o componente de producto cumple con su uso y con los contenidos educativos planteados originalmente.

Remitirse a la Gestión de la Información para obtener las plantillas, procesos, documentos y otros activos.

Nota especial

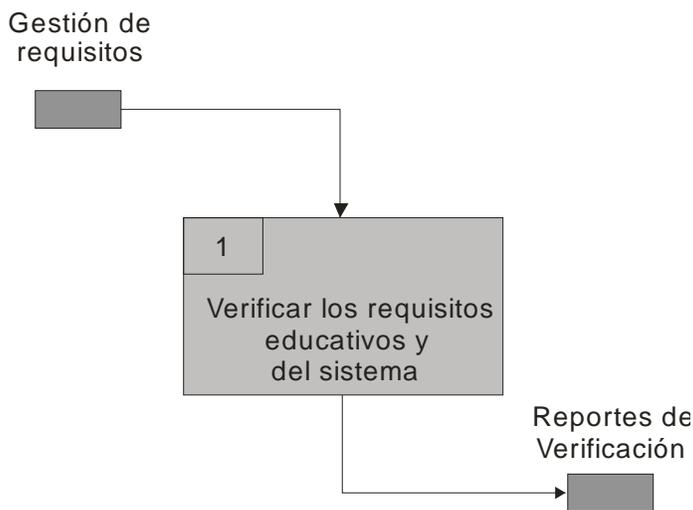
Para esta área de trabajo la descripción se divide en dos partes (pedagógica y técnica) por lo que la estructura se modifica sensiblemente para detallar el grado de aplicación en ambos aspectos.

En esta sección

En esta sección se describe información detallada de las actividades del proceso de Validación.

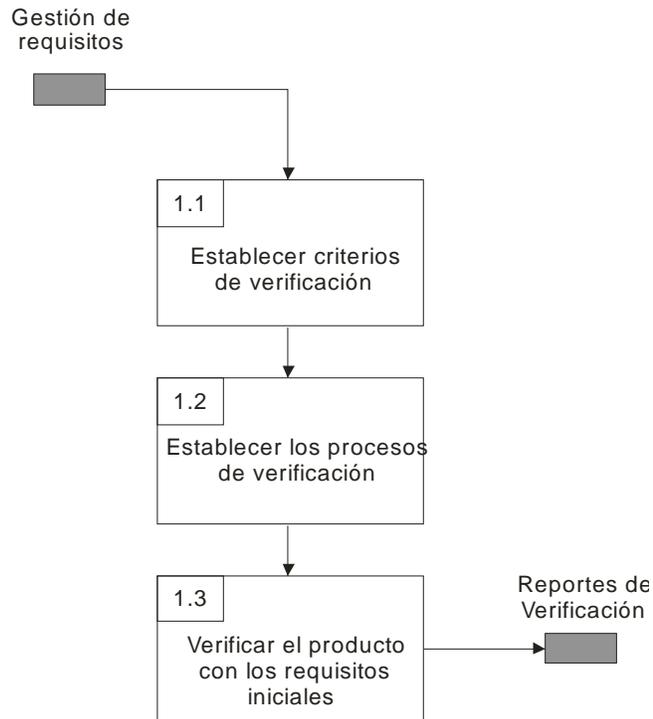
Actividad	Ver página
1 Verificar los requisitos educativos y del sistema	96

Diagrama general de actividades



3.6.2.1. Verificar los requisitos educativos y del sistema

Diagrama de actividades



Establecer criterios de verificación [1.1]

Propósito

El propósito de la actividad es establecer los criterios para asegurar que el objetivo de su producto cumple con los requisitos pedagógicos y del sistema.

Roles de los participantes

Aspecto pedagógico	Profesor de área. Pedagogo.
Aspecto técnico	Líder de equipo.

Entradas

Entrada Pedagógica	Fuente
Listas de objetivos curriculares (CUR)	Interna (repositorio de activos)
Diseño computacional (MOD)	Interna (repositorio de activos)
Estrategia pedagógica (PED)	Interna (repositorio de activos)
Entrada Técnica	Fuente
Requisitos del Proyecto	Externa

Salidas

Salida	Descripción
Criterios de verificación (Elementos en DAT)	El documento permite identificar los criterios de verificación dependiendo del proceso y del producto a desarrollar

Criterios de entrada

Tipo de prueba, parámetro de la prueba, componente a verificar.

Criterios de salida

Procedimientos y criterios de verificación.

Criterios de evaluación

Utilizar la medida de valor ganado, porcentaje de alumnos aprobados y porcentaje de defectos en el programa.

Establecer los procesos de verificación [1.2]**Propósito**

El propósito de la actividad es establecer los métodos a utilizar para la verificación de los requisitos pedagógicos y del sistema.

Roles de los participantes

Aspecto pedagógico	Profesor de área. Pedagogo.
Aspecto técnico	Líder de equipo.

Entradas

Entrada Pedagógica	Fuente
Criterios de Verificación (VERI)	Interna (repositorio de activos)
Resumen del Plan de Calidad (CAL)	Interna (repositorio de activos)
Entrada Técnica	Fuente
Requisitos del proyecto	Externa

Salidas

Salida	Descripción
Lista de métodos o formas para realizar la verificación	Se tendrá una lista de las formas o métodos que se pueden utilizar para llevar a cabo el proceso de verificación de los requisitos educativos.

Criterios de entrada

Aspecto pedagógico	Los objetivos curriculares.
Aspecto técnico	Requisitos del proyecto.

Criterios de salida Las formas de verificación más adecuadas de los procesos del programa como son los casos de uso, pruebas de aceptación, pruebas de cobertura, pruebas de estrés.

Criterios de evaluación Después de construir una versión beta: utilizar la medida de valor ganado, porcentaje de alumnos aprobados y porcentaje de defectos en el programa.

Verificar el producto con los requisitos iniciales [1.3]

Propósito El propósito de la actividad es realizar un examen minucioso de los componentes e identificar sus defectos para su eliminación y recomendar otros cambios que se necesitan.

Roles de los participantes	Aspecto pedagógico	Profesor de área. Pedagogo.
	Aspecto técnico	Líder de equipo.

Entradas

Entrada Pedagógica	Fuente
Listas de objetivos curriculares (CUR)	Interna (repositorio de activos)
Estrategia pedagógica (PED)	Interna (repositorio de activos)
Entrada Técnica	Fuente
Requisitos del proyecto	Externa

Salidas

Salida	Descripción
Reportes de verificación (actualizar valores en DAT)	Se tendrán los reportes generados de la evaluación de cada uno de los procesos sometidos al proceso de verificación, estos reportes serán el parámetro para realizar las acciones correctivas apropiadas.

Criterios de entrada	Pedagógica	Los objetivos curriculares.
	Técnica	Requisitos del proyecto.

Criterios de salida Los reportes de verificación.

Criterios de evaluación

Se pueden utilizar como medida algunas técnicas de análisis de datos pre-post, prueba de Wilcoxon, o algunos test como el de Raven.

3.6.3. Validación

Procesos de MeSoFT

La tercera área de proceso de la Gestión Pedagógica de MeSoFT es la de *Validación*. El propósito de Validación es demostrar que un producto o componente de producto cumple con su uso cuando se pongan en funcionamiento. La validación debe ser aplicada al aspecto pedagógico y técnico del producto.

Relaciones con otras áreas de MeSoFT

Remitirse a la Gestión de Requisitos para obtener más información sobre los requisitos del proyecto.

Remitirse a la Verificación para mayor información sobre como verificar que los requisitos del producto son los correctos y que cumplen con los contenidos educativos planteados originalmente.

Remitirse a la Gestión de la Información para obtener las plantillas, procesos, documentos y otros activos.

Remitirse a la Gestión del Proyecto para obtener más información acerca de la transformación de requisitos en las especificaciones del producto y para adoptar medidas correctivas cuando la validación identifica problemas que afectan al producto o productos educativos.

Nota especial

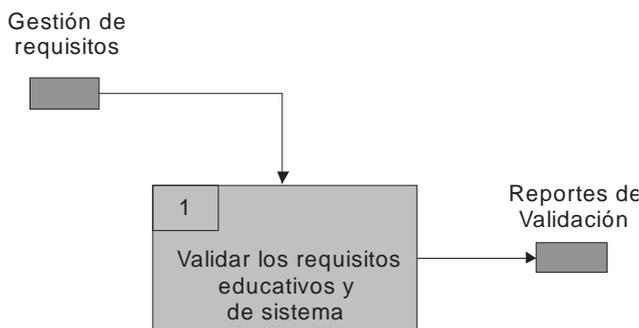
Para esta área de trabajo la descripción se divide en dos partes (pedagógica y técnica) por lo que la estructura se modifica sensiblemente para detallar el grado de aplicación en ambos aspectos.

En esta sección

En esta sección se describe información detallada de las actividades del proceso de Validación.

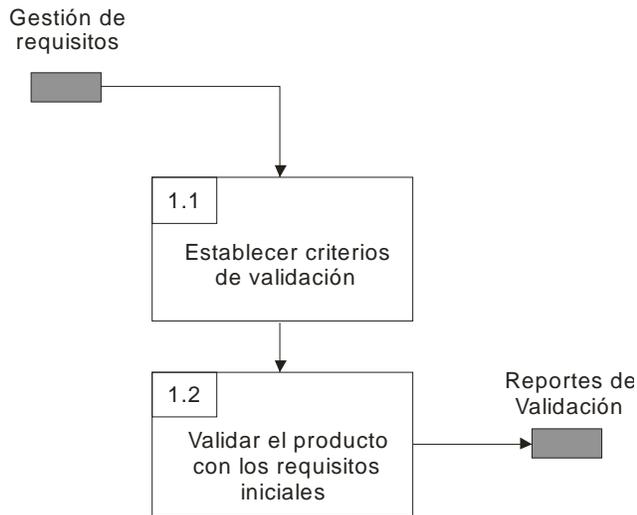
Actividad	Ver página
1 Validar los requisitos educativos y de sistema	100

Diagrama general de actividades



3.6.3.1. Validar los requisitos educativos y de sistema

Diagrama de actividades



Establecer criterios de validación [1.1]

Propósito El propósito de la actividad es definir los criterios para garantizar que el producto cumple la tarea para la cual fue realizado.

Roles de los participantes	Aspecto pedagógico	Profesor de área. Pedagogo.
	Aspecto técnico	Líder de equipo.

Entradas

Entrada Pedagógica	Fuente
Listas de objetivos curriculares (CUR)	Interna (repositorio de activos)
Diseño computacional (MOD)	Interna (repositorio de activos)
Entrada Técnica	Fuente
Requisitos del proyecto	Externa

Salidas

Salida	Descripción
Criterios de validación (VALI)	El documento permite identificar los criterios de validación dependiendo del proceso y del producto a desarrollar.

Criterios de entrada	Pedagógica	Los criterios curriculares.
	Técnica	Requisitos del proyecto.

Criterios de salida Los criterios de validación.

Criterios de evaluación Utilizar la medida de valor ganado, porcentaje de alumnos aprobados y porcentaje de defectos en el programa.

Validar el producto con los requisitos iniciales [1.2]

Propósito El propósito de la actividad es que el programa o los componentes del programa están validados para garantizar que cumplen con los requisitos educativos planificados y sean aptos para su uso en su entorno operativo.

Roles de los participantes	Aspecto pedagógico	Profesor de área. Pedagogo.
	Aspecto técnico	Líder de equipo.

Entradas

Entrada Pedagógica	Fuente
Criterios de validación (VALI)	Interna (repositorio de activos)
Entrada Técnica	Fuente
Requisitos del proyecto	Externa

Salidas

Salida	Descripción
Reportes de validación (REPVALI)	Se tendrán los reportes generados de la evaluación de cada uno de los procesos sometidos al proceso de validación, estos reportes serán el parámetro para realizar las debidas acciones correctivas apropiadas

Criterios de entrada	Pedagógica	Los objetivos curriculares.
	Técnica	Requisitos del proyecto.

Criterios de salida Los reportes de validación los cambios y ajustes a realizar.

Criterios de evaluación

Se toma como medida los resultados de aplicar cuestionarios estructurados, semi-estructurados y abiertos.

3.7. Definición del Área de Proceso de la Gestión de la Información

3.7.1. Definición del proceso

Procesos de MeSoFT

La única área de proceso de la Gestión de la Información de MeSoFT es la *Definición del Proceso*. El objetivo de esta área es establecer y mantener un conjunto de activos y estándares para el entorno de trabajo. MeSoFT ya cuenta con una serie de activos y procesos estándares para el desarrollo de software educativo (en forma de plantillas, documentos y prácticas), pero también permite la adición de activos nuevos para incrementar el repositorio del método. Esta área de proceso se encarga de guiar a los alumnos en el uso de los activos existentes o generación y adición de nuevos.

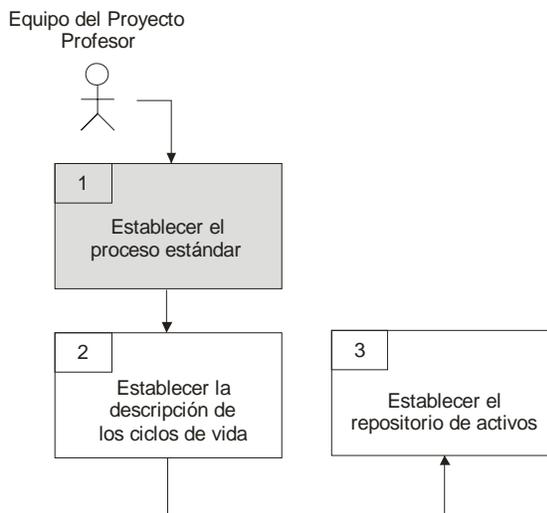
En esta sección

En esta sección se describe información detallada de las actividades del proceso de Definición del Proceso.

Actividad	Ver página
1 Establecer el proceso estándar	103
2 Establecer la descripción de los ciclos de vida	106
3 Establecer el repositorio de activos	108

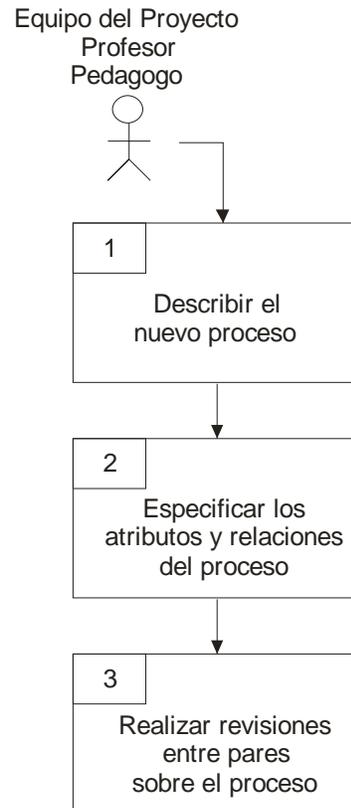
Notas

Para esta área de proceso no aplican los Criterios de Entrada, Salida y Evaluación puesto que son actividades que no son medidas (solo deben seguirse sistemáticamente) y no generan productos de trabajo.

Diagrama general de actividades

3.7.1.1. Establecer el proceso estándar

Diagrama de actividades



Describir el nuevo proceso [1.1]

Propósito

El propósito de la actividad es descomponer y definir el nuevo proceso con el nivel necesario de detalle como para que pueda ser realizado de manera consistente a través de los proyectos educativos.

Roles de los participantes

Profesor de área.

Pedagogo.

Líder de equipo.

Entradas

Entrada	Fuente
Estructura de actividades	Externa
Bosquejo del proceso	Externa

Salidas

Salida	Descripción
Plantillas para el nuevo proceso	Las nuevas plantillas deben seguir la estructura de las plantillas de MeSoFT y deberán ser integradas al Anexo A una vez que hayan terminado las revisiones por pares.
Actividades desglosadas del proceso	Cada una de las actividades propuestas deberá ser dividida en partes más específicas para cubrir un mayor nivel de detalle. La idea base será la parte genérica.
Descripciones de todos los productos a obtener	Cada actividad deberá ser asociada con todos los productos que serán generados mediante su ejecución.

Especificar los atributos y relaciones del proceso [1.2]**Propósito**

El propósito de la actividad es refinar el proceso generado mediante la identificación de atributos críticos y relaciones de trabajo entre todos los elementos que lo componen.

Roles de los participantes

Profesor de área.

Pedagogo.

Líder de equipo.

Entradas

Entrada	Fuente
Estructura de actividades	Externa
Bosquejo del proceso	Externa

Salidas

Salida	Descripción
Roles del proceso	Personas involucradas con la nueva actividad (agregar los nuevos roles a PARTI).
Métodos aplicables	MeSoFT recomienda técnicas de soporte para realizar algunas actividades que no son realizadas implícitamente por el método. Si algunas de las actividades nuevas requieren la especificación de este tipo de técnicas, el nuevo

	proceso deberá considerarlas en su descripción.
Entradas/salidas	Cada actividad deberá ser asociada con entradas para su ejecución y salidas como resultados de su ejecución. Si existiera alguna relación con otra área o actividad deberá mencionarse.
Puntos de verificación	Al igual que las áreas de MeSoFT, un nuevo proceso deberá ser verificado para asegurar que cumple con su descripción.
Criterios de salida	Elementos que se asume existen para ejecutar la actividad.
Criterios de medición	Si el nuevo proceso requiere de nuevas medidas, estas serán incorporadas a las plantillas de MeSoFT.

Realizar revisiones entre pares sobre el proceso [1.3]

Propósito

El propósito de la actividad es asegurar la consistencia y completitud del proceso. El proceso podrá ser revisado siempre que sea necesario.

Roles de los participantes

Profesor de área.

Líder de equipo.

Entradas

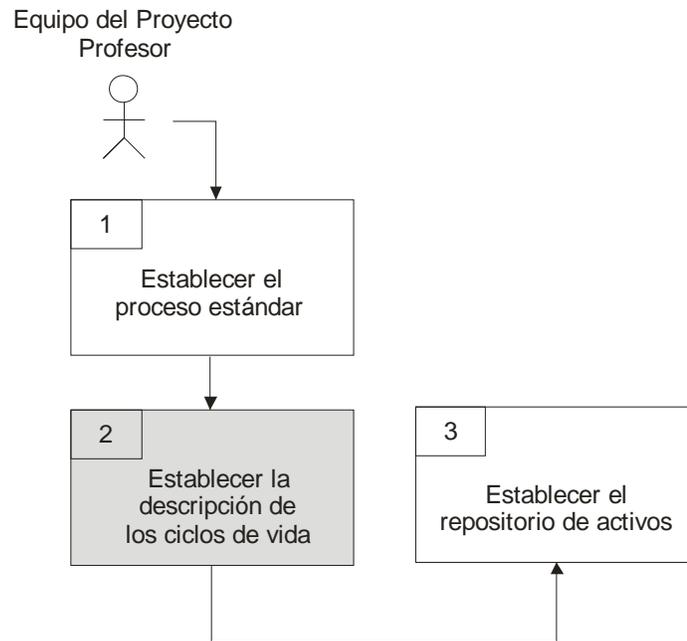
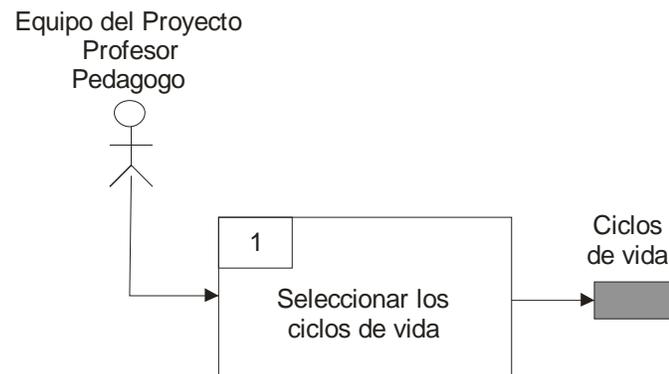
Entrada	Fuente
Descripción general del proceso	Externa

Salidas

Salida	Descripción
Descripción formal del proceso	La versión refinada resultado de las revisiones por pares es la que deberá ser agregada a MeSoFT.

Diagrama general de actividades

La definición del proceso está compuesta de tres actividades. A continuación se describe la segunda de estas:

**3.7.1.2. Establecer la descripción de los ciclos de vida****Diagrama de actividades****Seleccionar los ciclos de vida [2.1]****Propósito**

El propósito de la actividad es establecer y mantener descripciones de los modelos de ciclo de vida aprobados para ser utilizados en un proyecto específico.

Roles de los participantes

Profesor de área.
Líder de equipo.

Entradas

Entrada	Fuente
Descripción del proyecto	Externa
Fases del ciclo de vida	Externa (espiral, cascada, evolutivo, incremental, iterativo)

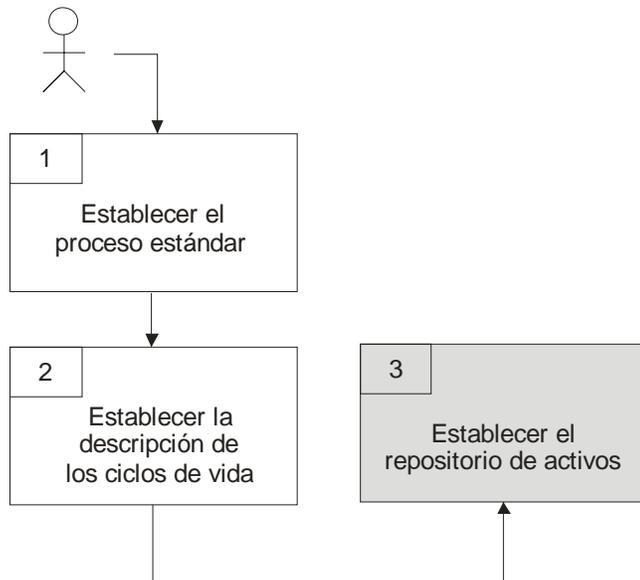
Salidas

Salida	Descripción
Descripción formal del ciclo de vida	MeSoFT incluye un ciclo de vida específico para el desarrollo de software educativo. Sin embargo, existe la posibilidad que se desee incorporar un ciclo alternativo de acuerdo al tipo de proyecto.

Diagrama general de actividades

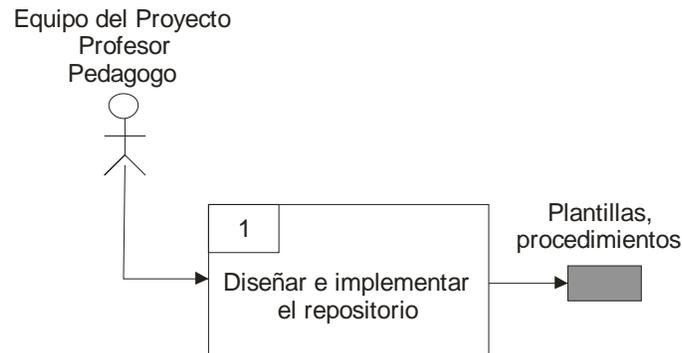
La definición del proceso está compuesta de tres actividades. A continuación se describe la tercera de estas:

Equipo del Proyecto
Profesor



3.7.1.3. Establecer el repositorio de activos

Diagrama de actividades



Diseñar e implementar el repositorio [3.1]

Propósito

El propósito de la actividad es establecer la implementación de una base de datos de donde se pueda obtener la información para cada proyecto.

Roles de los participantes

Profesor de área.
Líder de equipo.

Entradas

Entrada	Fuente
Plantillas de MeSoFT (Anexo A)	Interna

Salidas

Salida	Descripción
Repositorio del proyecto	Todos los documentos de MeSoFT representan en sí a la Base de Datos. Los proyectos de software educativo deberán almacenar todas las plantillas de MeSoFT en formato electrónico para su fácil edición.

4. Resultados obtenidos

Con el fin de validar la solución propuesta en esta tesis, se desarrolló un SE con MeSoFT, el Software del LabCD. En primer lugar, los alumnos procedieron a recopilar datos históricos sobre anteriores proyectos relacionados con el área de la IEPV. Los datos recabados son útiles para facilitar el proceso de la estimación de un proyecto piloto (a través del área de la Planificación del Proyecto) y compararlos con los resultados obtenidos con MeSoFT. El desarrollo del SE se inició con una reunión con los profesores y estudiantes, así como una breve explicación del proyecto. Todos los profesores tienen problemas con la entrega de los proyectos relacionados con el área de la IEPV. Utilizando la técnica educativa actual, los alumnos se desvían del calendario para entregar las prácticas relacionadas con un tema del programa académico, también se presenta una desviación del esfuerzo para terminar el trabajo; se excede el costo relacionado con la estimación del presupuesto para completar la práctica, existe un gran número de defectos en la entrega, y el problema más importante, un creciente número de estudiantes no aprobados debido a la falta de equipo que les permita terminar su trabajo.

Los profesores y los estudiantes están interesados en desarrollar una herramienta auxiliar basada en una computadora que proporciona un laboratorio virtual con equipo e instrumentos virtuales concretos para desarrollar la práctica como lo hacen en los laboratorios. Teniendo en cuenta estas entrevistas y los datos históricos de anteriores cursos, los profesores y los estudiantes definen objetivos de aprendizaje basados en su experiencia previa (véase la Tabla 6), ésta es la fase de inicialización de IADIAL.

4.1. Datos históricos recopilados

Los datos sobre los proyectos anteriores no fueron suficientes para completar el estudio. Los profesores sólo disponen de datos sobre horarios, costos, y calificaciones, pero no tienen datos sobre los defectos o los esfuerzos para aplicar la fase de Verificación de todos los objetivos.

Tabla 6. Objetivos del proyecto.

Objetivos	Valor del Objetivo
Entrega tardía	< 8% (1 semana)
Desviación del esfuerzo	< 15%
Costo de desviación	< 15%
% Defectos	< 5%
% De alumnos aprobados	> 70%

Los datos recogidos fueron un calendario de estimaciones y medidas, el tamaño, la fase de esfuerzo y defectos. Los estudiantes calculan medidas para hacer comparaciones con los resultados del proyecto piloto. De acuerdo con las características de los estudiantes el diseño del software analiza el problema y selecciona un ciclo pedagógico que incluyó las siguientes fases:

- *Fase de desarrollo:* Los profesores asignan las prácticas de acuerdo con el programa académico y revisan los resultados después de dos semanas y media.
- *Fase de prueba:* Los profesores revisan los resultados por alumno y hacen las correcciones a su trabajo, si es apropiado. Existe un circuito de retroalimentación para proporcionar nuevos conocimientos a los estudiantes.
- *Fase de entrega:* Los profesores evalúan todas las prácticas en los laboratorios físicamente y asignan una nueva práctica.

Las prácticas y ejercicios seleccionados fueron los siguientes: PRO-1 (Introducción a los instrumentos de estimulación y medida), PRO-2 (Implementación de un filtro paso bajo), PRO-3 (modulación por amplitud), PRO-4 (modulación por amplitud; versión virtual), y PRO-5 (Ley de Ohm).

4.2. Definición de procesos

MeSoFT selecciona el ciclo de vida de su repositorio de prácticas efectivas. El proceso de diseño para desarrollar la herramienta educativa se basa en principios básicos del TSP para gestionar todas las actividades a través de métricas y componentes del proceso. Documentos como la Especificación de Requisitos y Diseño de Alto Nivel son ahora controlados por la especificación activa de MeSoFT para administrar el proyecto. Las prácticas efectivas de MeSoFT proveen un plan para vigilar y controlar el proyecto (las áreas de PP y PS) y estableció el método del valor ganado para evaluar el progreso del proyecto. La Tabla 7 resume los principios básicos de MeSoFT que se utilizaron en este proyecto y las diferencias entre la método actual (seguido por los estudiantes).

Tabla 7. Principales procesos de desarrollo de MeSoFT implementados.

Procesos de MeSoFT	Técnicas de desarrollo tradicionales de SE
Proceso bien definido para hacer fácil la medición y estimación	Proceso ambiguo sin fases y los productos sin limitación
Filosofía de equipo	Proceso centrado en la demanda de experiencia de los alumnos
Calidad en el producto educativo	No hay plan, no hay calidad
Delimitación de contenidos especializados sobre las necesidades de los profesores y los estudiantes	Proyecto aceptado sin el conocimiento sobre la delimitación de temas
Seguimiento y control del proyecto a través del método del valor ganado	No existe un mecanismo real para determinar el progreso del proyecto
Evaluación del contenido de los temas incluidos con los especialistas (por semana)	No existe un mecanismo para evaluar la introducción de los contenidos en los instrumentos educativos

Los alumnos utilizaron el repositorio de prácticas pedagógicas efectivas para proponer una configuración interna de un Laboratorio Virtual y resolver el problema. Este Laboratorio Virtual se dividió en tres componentes:

- Su funcionalidad, la cual está relacionada con la capacidad del uso del laboratorio. El LabCD debe utilizarse para la parte académica y/o con fines de investigación.
- Su aspecto descriptivo, el cual está relacionado con la capacidad de los instrumentos y sistemas del laboratorio. Se puede utilizar, por ejemplo, como laboratorio de instrumentación electrónica programable o como un laboratorio de instrumentación virtual de electrónica.
- Su estructura, la cual se compone de una parte física o hardware, y una parte lógica, o software.

La parte física se compone de 10 estaciones de trabajo interconectadas por una red local, como se muestra en la Figura 4.1. Cada estación de trabajo tiene un sistema ATE con un PC, que lo forman un osciloscopio, una fuente DC, un generador de señales, un multímetro y un PICSTAR ® Plus programador de desarrollo de microcontrolador de la empresa Microchip.



Figura 4.1. Estructura física del LabCD.

La parte lógica, por otra parte, tiene el software necesario para el manejo de cada instrumento, la adecuada ejecución de aplicaciones, el sistema operativo y los lenguajes de programación. Todos estos dispositivos están conectados por un sistema de comunicaciones GPIB [Garcia04].

4.3. Implementación del software educativo

Como se ha dicho, los diseñadores de software han desarrollado el Software para el LabCD para validar el método alternativo que se presenta en este trabajo. Las Figuras 4.2, 4.3, 4.4 y 4.5 muestran algunas plantillas de MeSoFT utilizadas para el diseño del LabCD. La herramienta aplicada para la instrumentación electrónica proporciona servicio continuo a través de un sitio Web ubicado en el servidor de la Universidad. El acceso a todos los dispositivos depende de los horarios de trabajo.

Nombre

Pedro Hernández Soto

Fecha

10 Junio 2008

Nombre del Proyecto

LabCD

Tarea		Horas Planificadas		Tamaño/Valor Planificado			Actual	
Fase	Nombre Tarea	Total Horas	Horas Acumuladas	Unidades de Tamaño	Tamaño	Valor Planificado	Horas	Semanas
IADIAL	Identificar el problema	12	1	Hr		10		
IADIAL	Establecer objetivos de aprendizaje	3	2	Hr		2		
IADIAL	Obtener requisitos	5	1	Hr		3		
PP	Establecer estimaciones	5	1	Hr		7		
PP	Desarrollar el plan del proyecto	7	2	Hr		7		
IADIAL	Establecer un diseño educativo	4	1	Hr		5		
PP	Obtener el compromiso	10	1	Hr		15		
IADIAL	Construir la solución	35	1	LOC	500	350		
SP	Supervisar el proyecto	20	1	Hr		30		
SP	Gestionar las acciones correctivas	12	2	Hr		10		
MA	Establecer actividades de medición	4	1	Hr		5		
MA	Proporcionar los resultados	2	1	Hr		5		

Figura 4.2. Plantilla ACTI para la planificación de tareas.

Nombre	Hernán Valdivieso	Fecha	27 Junio 2008
Nombre del Proyecto	LabCD		

1. ¿Cuál es el proceso enseñanza-aprendizaje desarrollado en el curso presencial?									
Exposiciones con diapositivas y realización de prácticas extra-clase									
2. ¿Cuáles fueron las formas de comunicación o medios de interacción utilizados en el curso presencial?									
Proyectores de acetatos, cañones de proyección									
3. ¿Consideras que el proceso enseñanza-aprendizaje de un curso presencial puede mejorarse? Si tu respuesta es afirmativa menciona las formas.									
Equipo electrónico, computadoras, laboratorio de electrónica									
4. ¿Consideras que el desempeño de tu profesor fue adecuado o puede mejorarse aún más? Si tu respuesta es afirmativa, ¿cuáles serían tus recomendaciones para lograrlo?									
Podría mejorarse con una mayor cantidad de prácticas									
5. ¿Cuáles son las prácticas, formas de evaluación y formas de interacción que te agradaron de tu profesor?									
Desarrollo de sistemas mínimos, uso de protoboard									
6. ¿Cuáles de estas se deben conservar?	7. ¿Cuáles de estas no se deben conservar?								
Prácticas en laboratorio	Revisión de apuntes								
Configuración de software									
7. Si se te permitiera rediseñar el desempeño del proceso enseñanza-aprendizaje actual, ¿qué mejoras harías?									
Adquisición de software para apoyar la parte práctica									
8. En el caso de las mejoras, ¿estarías dispuesto a adaptarte a estas durante un nuevo curso?									
Claro que sí									
9. ¿Cuál es tu nivel de dominio en cuanto al uso de la computadora?									
Experto	<input type="checkbox"/>	Avanzado	<input checked="" type="checkbox"/>	Medio	<input type="checkbox"/>	Normal	<input type="checkbox"/>	Nulo	<input type="checkbox"/>
10. ¿Consideras la posibilidad de incluir el uso de la computadora como mejora al proceso enseñanza-aprendizaje? ¿En qué forma lo harías?									
Videojuego	<input checked="" type="checkbox"/>	Práctica virtual	<input checked="" type="checkbox"/>	Narrativa con personajes	<input type="checkbox"/>	Otras (indicar)			

Figura 4.3. Plantilla ALU para la entrevista de los alumnos.

Nombre	<u>José Luis Granados</u>	Fecha	<u>12 Sep 2008</u>
Nombre del Proyecto	<u>LabCD</u>	Responsable	<u>Josué G.</u>
Número de riesgo	<u>15</u>		

Fuente				
Fase en la que surge el riesgo	<u>PP</u>	Revisado por	<u>José Luis</u>	
Actividad en la que se presenta el riesgo	<u>Establecer estimaciones</u>	Durante la fase	<u>PP</u>	
Descripción del riesgo actual:				
<i>Se están recogiendo estimaciones inexactas por la falta de experiencia de los</i>				
<i>estudiantes en este proceso</i>				
Causas identificadas para el riesgo actual:				
<i>Inexperiencia</i>				
<i>Falta de herramientas para estimar</i>				
Acciones correctivas/preventivas				
Id	Nombre de la acción	Descripción de la acción	Responsable	Fecha
1	<i>Formular reporte</i>		<i>José Luis</i>	<i>6 sep</i>
2	<i>Solicitar habilidad</i>		<i>José Luis</i>	<i>6 sep</i>
3	<i>Agregar habilidad</i>		<i>José Luis</i>	<i>10 sep</i>
4	<i>Generar fuente</i>		<i>José Luis</i>	<i>10 sep</i>
Seguimiento				
Fecha	Id	Observaciones/evidencias	Conclusiones	
<i>11 sep</i>	<i>1</i>	<i>Se corrigieron las estimaciones</i>	<i>Ninguna</i>	
	<i>2</i>	<i>Revisar de nuevo en fase SP</i>	<i>Si es necesario habrá que planificar</i>	
Elaboró <u>José Luis Granados</u> Aprobó <u>Josué García</u>				

Figura 4.4. Plantilla CORRE para el Plan de Acciones Correctivas.

Nombre Hernán Valdivieso

Nombre del Proyecto LabCD

Fecha 07 julio 2008

Versión 0.1

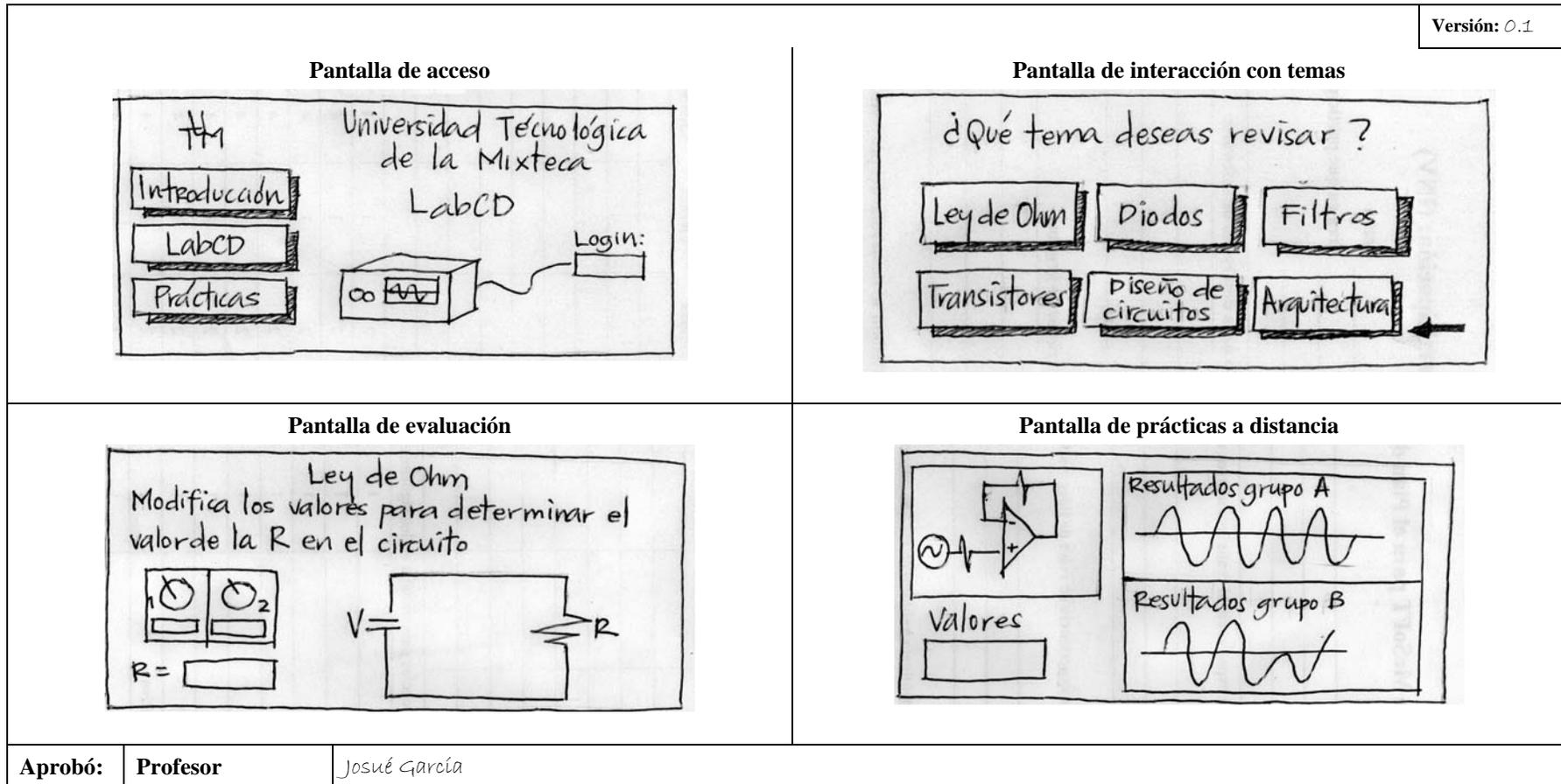


Figura 4.5. Plantilla HIS para el Storyboard.

Una de las ventajas es que cualquier profesor o estudiante (sin importar en donde se encuentren) tienen acceso directo mediante el siguiente enlace URL: <http://www.utm.mx/~labcd/LabCD.htm> tal y como se muestra en la Figura 4.6.



Figura 4.6. Software del LabCD.

La herramienta del LabCD utiliza el entorno de desarrollo como es Agilent VEE creado por Agilent Technologies, LabVIEW creado por National Instruments Corporation u otro lenguaje de programación que se adecue al diseñador del SW. El ejemplo aquí descrito está desarrollado en Agilent VEE porque se trata de un lenguaje de programación orientado a la instrumentación electrónica programable y virtual. Los cinco ejercicios seleccionados (PRO-1, PRO-2, PRO-3, PRO-4, PRO-5) ya se han aplicado en el LabCD (los cinco ejercicios en un período de sesiones del LabCD), uno para la instrumentación electrónica programable y otro para la instrumentación electrónica virtual. El ejercicio de instrumentación electrónica virtual permitió a los estudiantes simular sistemas de operaciones de instrumentación sin ningún tipo de manipulación física (disponible en el LabDC). En este sentido, una interfaz gráfica de usuario (GUI) que se utiliza directamente realiza la simulación de las funciones del dispositivo en el mismo equipo, al igual que se hace a mano. La figura 4.7 muestra la interfaz gráfica utilizada para la Ley de Ohm (PRO-5).

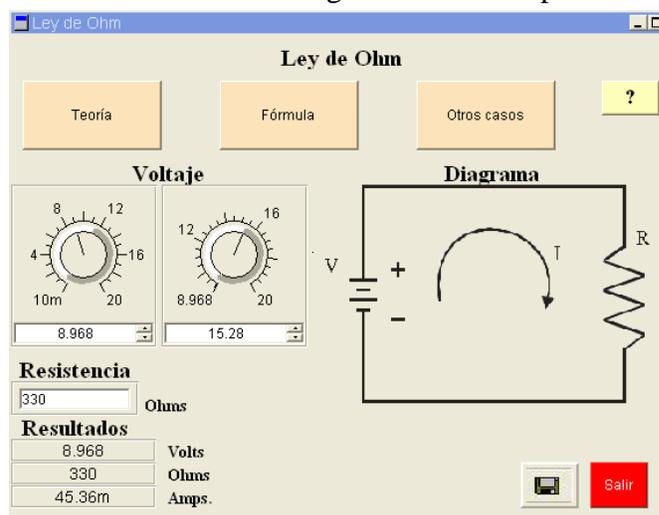


Figura 4.7. GUI para el ejercicio de la ley de Ohm (virtual).

El ejercicio de instrumentación electrónica programable permitió el control entre los diferentes dispositivos y sistemas de una PC. El estudiante fue, por lo tanto, capaz de llevar a cabo la automatización de una prueba en el sistema. La Figura 4.8 muestra un ejercicio de un filtro pasa-bajas (PRO-2).

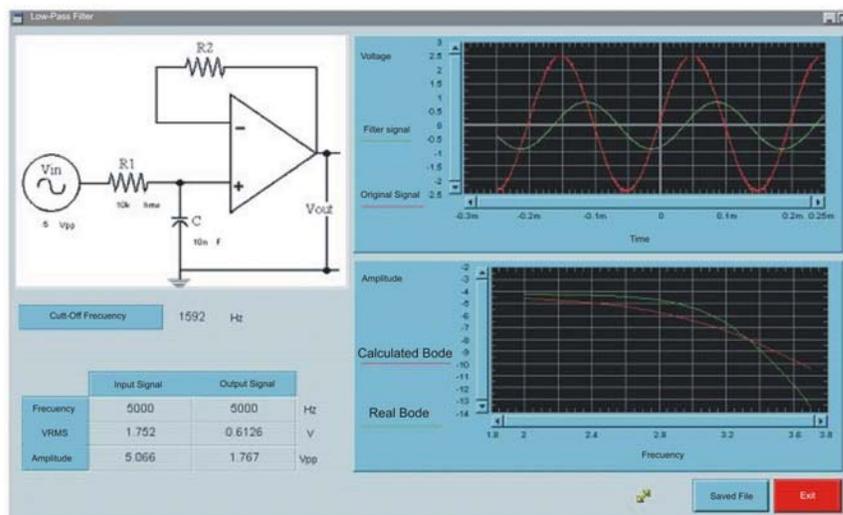


Figura 4.8. GUI para el ejercicio del filtro pasa-bajas (programable).

4.4. Datos recogidos

Con los datos históricos de cinco ejercicios se seleccionaron a veinticinco estudiantes relacionados con el curso IEPV y se procedió a evaluar los resultados obtenidos utilizando la Tabla 5 como medida de éxito sobre los objetivos definidos (ver Tabla 8):

- **Objetivo 1:** Práctica realizada dentro del tiempo, costo y presupuesto definido.

Tabla 8. Datos estimados contra datos reales.

Objetivos	Estimado	Real	Desviación
Programa [PROY]	13.0	14.0	7.7%
Esfuerzo [HORAS]	950.0	1121.0	18.0%
Tamaño [KLOC]	6.9	8.5	22.5

La Tabla 9 muestra que todos los ejercicios se terminaron dentro de los tiempos de calendarización definidos con sólo una semana de retraso (tal vez porque es la primera vez que los estudiantes utilizan el Software o los equipos del LabCD, además se incluye el tiempo de desarrollo del LabCD). El esfuerzo para hacer las prácticas es muy positivo porque la desviación es mínima.

Tabla 9. Buscando el Objetivo 1.

Objetivos	Valor del objetivo	Valor obtenido	Desviación
Entrega tardía	< 8% (1 semana)	7.7%	-3.8%
Desviación del esfuerzo	< 15%	18.00%	20.2%
Costo de la desviación	< 15%	18.00%	20.2%

Es importante mencionar que los estudiantes fueron capaces de detectar un error en la semana 7 de sus ejercicios y que se pudo detectar un pequeño retraso en la entrega (véase Figura 4.9). El máximo beneficio fue que sin MeSoFT la herramienta no hubiera podido evaluar el contenido de la práctica y generar mensajes de control para el profesor.

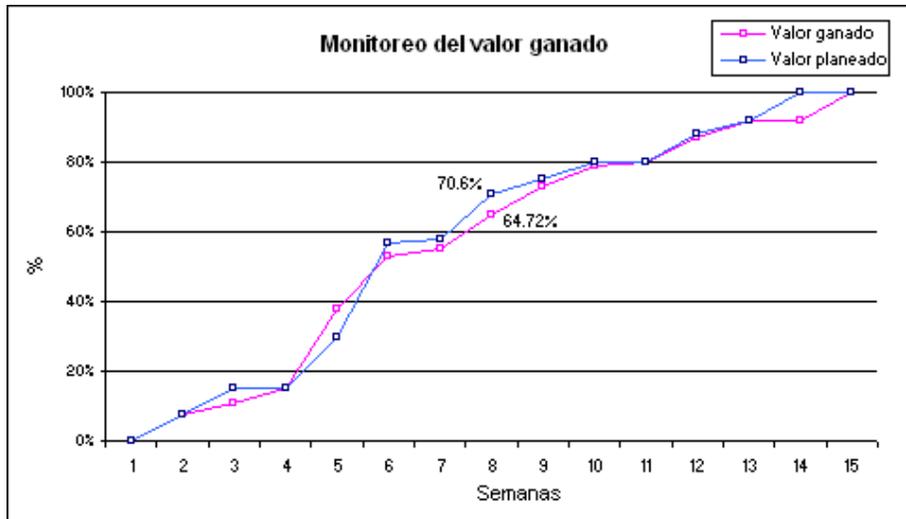


Figura 4.9. Monitoreando el valor ganado para los ejercicios seleccionados.

- **Objetivo 2:** Reducir el tiempo de pruebas.

La Tabla 10 muestra que el tiempo para ejecutar las pruebas en los ejercicios se redujo significativamente. El lector debe recordar que el valor objetivo se estableció a partir del dato histórico de la media de los ejercicios.

Tabla 10. Cumpliendo el Objetivo 2.

Objetivo	Estimado	Real	Desviación
Tiempo reducido en la prueba	< 24.4 %	10.0 %	-59.1%

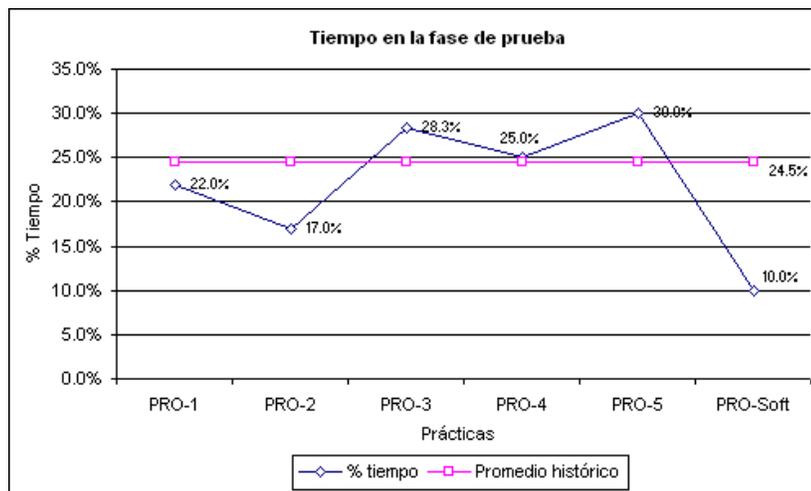


Figura 4.10. Reducción de tiempo en la fase de prueba.

La Figura 4.10 muestra que los estudiantes sólo tomaron un 10.0% del total de tiempo para probar las configuraciones de sus ejercicios, esto significa el 59.1% menos de la media histórica de los ejercicios.

- **Objetivo 3:** Mejorar la calidad de los ejercicios de una manera que se incremente el porcentaje de alumnos aprobados.

La Tabla 11 muestra que estas dos cuestiones previstas evidencian los beneficios de la aplicación de MeSoFT. Una de ellas fue la notoria reducción de defectos en los ejercicios, y la otra fue la mejora en el porcentaje de alumnos aprobados que del 65.7% se incrementó al 86.2% (ver Figura 4.11).

Tabla 11. Buscando el Objetivo 3.

Objetivo	Estimado	Real	Desviación
% defectos en la entrega	< 5.0%	3.8 %	-24.8 %
Porcentaje de alumnos aprobados	> 70.0 %	86.2 %	37.4%

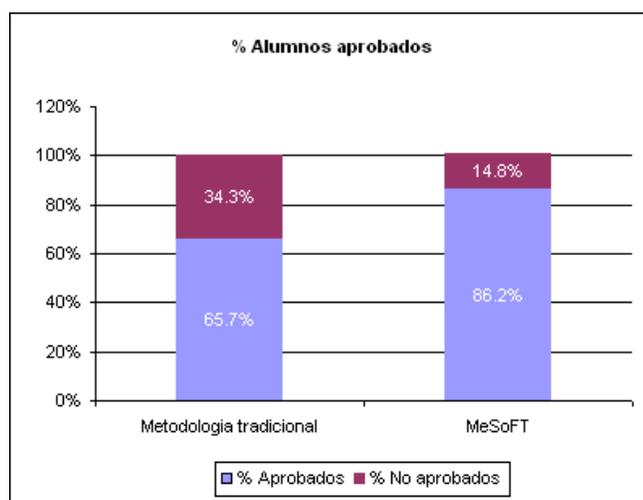


Figura 4.11. Mejora del % de alumnos aprobados.

5. Conclusiones

Un método para desarrollar SE, como MeSoFT, podría ser el comienzo de un cambio revolucionario en herramientas de aprendizaje. MeSoFT ha incluido una serie de actividades, denominadas "prácticas efectivas", a fin de establecer un "mapa de carreteras" para apoyar el desarrollo de SE. Los estudiantes y los investigadores ahora pueden utilizar no sólo libros y manuales para desarrollar este tipo de herramientas, sino también un método alternativo para desarrollar y controlar sus propios proyectos educativos con la gestión de un repositorio de prácticas efectivas.

Este trabajo ha introducido el nuevo elemento de un repositorio adaptado para prácticas efectivas. El enfoque del método propuesto se centra en: la parte pedagógica y la parte del aspecto técnico, las cuales se utilizaron para construir el software de LabCD, como el principal componente para incluir en la estructura del método. MeSoFT ofrece un conjunto de prácticas enfocadas en la vigilancia y el control de los aspectos técnicos del producto educativo y las prácticas de IADIAL a fin de reflejar los aspectos pedagógicos. Sin embargo, el experimento mostrado en el Capítulo 4 resalta poca consideración sobre las cuestiones relativas a la forma en que el proceso cognitivo se logra en la mente del estudiante, o la forma en que el individuo almacena los conocimientos adquiridos.

En base a los resultados obtenidos, se concluye que los métodos alternativos, tales como MeSoFT, proporcionan un entorno único que beneficia directamente a las dos partes involucradas: académicos y estudiantes. El trabajo de investigación en esta área define un repositorio de procesos de operación que se podría utilizar por los desarrolladores para intercambiar la información sobre las mejores prácticas en proyectos de SE. Un trabajo futuro se centrará en la mejora de MeSoFT con miras a la obtención de un modelo estándar para desarrollar cualquier tipo de SE.

La validación realizada en el Capítulo anterior sobre el método propuesto permitió identificar hallazgos importantes:

- Se logró desarrollar un método orientado exclusivamente al desarrollo de SE. De acuerdo a la validación realizada sobre el método los desarrolladores no mostraron problemas al entender la estructura de las actividades y de los procesos. Este resultado nos motiva a afirmar que MeSoFT es comprensible ; sin embargo, se requerirá de un mayor número de proyectos piloto para generalizar esta afirmación.
- El método propuesto tiene una arquitectura basada en procesos y actividades que cubre los aspectos técnicos y pedagógicos. Durante el desarrollo del LabCD se logró utilizar el enfoque de mejora del proceso esto mediante la adición del área de proceso de medición y análisis.

- MeSoFT incorporó un mecanismo pedagógico, llamado IADIAL, que almacenó prácticas efectivas de expertos en el área educativa. Este repositorio contiene plantillas y documentos propios que facilitan la definición de los requerimientos pedagógicos y técnicos mediante la reutilización.
- La solución propuesta incluyó un proceso de evaluación, reflejado en la fase de Evaluación del modelo IADIAL, que permite obtener una relación del nivel de aprendizaje y modificar el contenido educativo para evitar que el usuario repita los materiales y se presenten pérdidas de tiempo o distracciones.
- Desgraciadamente, la hipótesis establecida en el Capítulo 3 de esta tesis, sobre que los diseñadores deberían tener conocimiento “básico” para utilizar la solución propuesta; al trabajar con el grupo de validación se determinó que no es así. Cabe mencionar que se trabajó con estudiantes de tercer semestre (que no habían cursado la materia de Ingeniería de Software); por lo que se recomienda que los usuarios hayan tomado por lo menos un curso de Ingeniería de Software y que posean conocimientos medios-avanzados de pedagogía.
- Se aclara que se requiere de una mayor cantidad de proyectos piloto para justificar la funcionalidad de MeSoFT. Los resultados mostrados en el Capítulo 4 no demuestran de manera rotunda el beneficio de utilizar el método propuesto; sin embargo, son parámetros iniciales muy buenos que servirán para incrementar los datos históricos que permitan, en un futuro, asegurar la óptima funcionalidad de esta solución.

Como trabajo futuro se recomienda la mejora del método propuesto al proponer e incrementar los activos que cubren el aspecto pedagógico. La intención de la propuesta expuesta en esta tesis, es fomentar el interés por desarrollar métodos alternativos para el desarrollo de SE, que la comunidad estudiantil pueda utilizar y refinar mediante su aplicación en diversos contextos educativos. De acuerdo a cómo varíe la complejidad del SE a desarrollar, los desarrolladores podrían aumentar el número de plantillas de acuerdo al tipo de producto que quieran desarrollar. La línea actual de investigación está siendo enfocada a diseñar y desarrollar un mecanismo que permita identificar componentes reutilizables independientes del tema específico que cubra el SE. Esta herramienta sería incluida en la fase de análisis del modelo IADIAL.

En base a lo anterior, se considera que el problema de diseñar e implementar SE no ha sido resuelto del todo; sin embargo, estamos seguros que MeSoFT contribuye en la definición de métodos y modelos genéricos que buscan incrementar la calidad del aprendizaje en nuestro país.

6. Anexo A.- Activos de MeSoFT

6.2. Plantilla de MeSoFT para Entrevista a Alumnos: (ALU)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

1. ¿Cuál es el proceso enseñanza-aprendizaje desarrollado en el curso presencial?									
2. ¿Cuáles fueron las formas de comunicación o medios de interacción utilizados en el curso presencial?									
3. ¿Consideras que el proceso enseñanza-aprendizaje de un curso presencial puede mejorarse? Si tu respuesta es afirmativa menciona las formas.									
4. ¿Consideras que el desempeño de tu profesor fue adecuado o puede mejorarse aún más? Si tu respuesta es afirmativa, ¿cuáles serían tus recomendaciones para lograrlo?									
5. ¿Cuáles son las prácticas, formas de evaluación y formas de interacción que te agradaron de tu profesor?									
6. ¿Cuáles de estas se deben conservar?	7. ¿Cuáles de estas no se deben conservar?								
7. Si se te permitiera rediseñar el desempeño del proceso enseñanza-aprendizaje actual, ¿qué mejoras harías?									
8. En el caso de las mejoras, ¿estarías dispuesto a adaptarte a estas durante un nuevo curso?									
9. ¿Cuál es tu nivel de dominio en cuanto al uso de la computadora?									
Experto	<input type="checkbox"/>	Avanzado	<input type="checkbox"/>	Medio	<input type="checkbox"/>	Normal	<input type="checkbox"/>	Nulo	<input type="checkbox"/>
10. ¿Consideras la posibilidad de incluir el uso de la computadora como mejora al proceso enseñanza-aprendizaje? ¿En qué forma lo harías?									
Videojuego	<input type="checkbox"/>	Práctica virtual	<input type="checkbox"/>	Narrativa con personajes	<input type="checkbox"/>	Otras (indicar)			

6.3. Plantilla de MeSoFT para el Resumen del Plan de Calidad: (CAL)

Nombre _____ Fecha _____
 Nombre del Proyecto _____
 Producto _____

Porcentaje de Productos Libre de Defectos	Planificado	Actual
En Compilación		
En Pruebas Unitarias		
En Pruebas de Construcción e Integración		
En Pruebas del Sistema		
Defectos por Página	Planificado	Actual
Inspección de Requisitos		
Inspección de Diseño de Alto Nivel		
Defectos por KLOC		
Revisión de Código		
Compilación		
Inspección de Código		
Prueba Unitarias		
Pruebas de Construcción e Integración		
Prueba de Sistema		
Desarrollo Total		
Tasa de Defectos	Planificado	Actual
Revisión de Código / Compilación		
Revisión de Diseño de Alto Nivel / Pruebas Unitarias		
Tiempos de Desarrollo		
Inspección de Requisitos		
Inspección de Diseño de Alto Nivel		
Diseño de Bajo Nivel / Código		
Revisión de Código		

6.5. Plantilla de MeSoFT para el Plan de Acciones Correctivas: (CORRE)

Nombre _____ Fecha _____
 Nombre del Proyecto _____ Responsable _____
 Número de riesgo _____

Fuente				
Fase en la que surge el riesgo		_____	Revisado por _____	
Actividad en la que se presenta el riesgo		_____	Durante la fase _____	
Descripción del riesgo actual:				
Causas identificadas para el riesgo actual:				
Acciones correctivas/preventivas				
Id	Nombre de la acción	Descripción de la acción	Responsable	Fecha
Seguimiento				
Fecha	Id	Observaciones/evidencias	Conclusiones	
Elaboró		Aprobó		

6.6. Plantilla de MeSoFT para describir la Lista de Objetivos Curriculares: (CUR)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Nombre del Curso: _____		Número de Horas a la Semana: _____	
Porcentaje de calificación: Teoría _____		Práctica _____ Trabajo extraclase _____	
Objetivos alcanzados con asistencia de la computadora			
	Objetivos del curso	Programa didáctico conocido/utilizado	Descripción del programa
1			
2			
3			
4			
5			
6			
7			
8			
Objetivos alcanzados sin asistencia de la computadora			
	Objetivos del curso	¿Podría utilizarse un programa didáctico?	Descripción del programa
1			
2			
3			
4			
5			
6			
7			
8			
Observaciones:			

6.7. Plantilla de MeSoFT para el Plan de la Gestión de los Datos: (DAT)

Nombre _____ Fecha _____
 Nombre del Proyecto _____
 Producto _____

Tamaño del Programa	Planificado	Actual	
Páginas Diseño Alto Nivel (DBN)			
Páginas Diseño Bajo Nivel (DAN)			
Páginas Requisitos (REQ)			
Páginas de texto (TEXT)			
LOC Base (medido)			
LOC Borrado			
LOC Modificado			
LOC Añadido			
LOC Reutilizado			
LOC Total / Nuevo-cambiado			
LOC Total			
LOC Total / Nuevo-reutilizado			
Tiempo por Fases	Planificado	Actual	% Actual
Diseño conceptual			
Planificación			
Requisitos			
Plan de Pruebas del Sistema			
Inspección de Requisitos			
Plan de Pruebas de Integración			
Planificación de la Implementación			
Diseño detallado			
Revisión del Diseño detallado			
Desarrollo de Pruebas			
Inspección Diseño Detallado			
Código			
Revisión de Código			
Compilación			
Inspección de Código			
Prueba Unitarias			

**6.8. Plantilla de MeSoFT para el Plan de la Gestión de los Datos: (DAT)
(continuación)**

Nombre _____ **Fecha** _____
Nombre del Proyecto _____
Producto _____

Prueba de Sistema	_____
Documentación	_____
Postmortem	_____
Totales	_____

6.10. Plantilla de MeSoFT para describir la Lista de Objetivos Educativos: (EDU)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Nombre del Curso:		Número de Horas a la Semana:
Porcentaje de calificación:	Teoría _____	Práctica _____ Trabajo extraclase _____
Objetivos alcanzados con asistencia de la computadora		
	Programa didáctico conocido/utilizado	Descripción del programa
1		
2		
3		
4		
5		
6		
7		
8		
Objetivos alcanzados sin asistencia de la computadora		
	¿Podría utilizarse un programa didáctico?	Descripción del programa
1		
2		
3		
4		
5		
6		
7		
8		
Observaciones:		

6.12. Plantilla de MeSoFT para el Storyboard: (HIS)

Nombre _____

Nombre del Proyecto _____

Fecha _____

Versión _____

<p>Pantalla de acceso</p>		<p>Pantalla de interacción con temas</p>		<p>Versión:</p>
		<p>Pantalla de evaluación</p>		<p>Pantalla de prácticas a distancia</p>
<p>Aprobó:</p>	<p>Profesor</p>			

6.13. Plantilla de MeSoFT para el Plan de Participación: (INV)

Nombre	_____	Fecha	_____
Nombre del Proyecto	_____	Área en que participa	_____
Formación/Grado	_____		

Describe brevemente tu experiencia al trabajar en otros equipo para desarrollo de software:

--

Describe brevemente tu experiencia en roles de líder o administrador de grupos:

--

Lista las horas de tu calendario habitual en las que puedes contribuir al desarrollo del proyecto.

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
8:00-9:00							
9:10-10:10							
10:20-11:20							
11:30-12:30							
12:40-13:40							
13:50-14:50							

Ordena de 1 a 5 tus preferencias para trabajar en los siguientes roles:

Líder de equipo	1	2	3	4	5
Programador	1	2	3	4	5
Planificador	1	2	3	4	5
Analista	1	2	3	4	5
Auditor de calidad	1	2	3	4	5
Diseñador de interfaces	1	2	3	4	5
Soporte (incluidos testers)	1	2	3	4	5
Especialista en área educativa	1	2	3	4	5
Especialista técnico	1	2	3	4	5

6.14. Plantilla de MeSoFT para describir el Marco Educativo y Comunicacional: (MAR)

Nombre _____ Fecha _____
Nombre del Proyecto _____

Prácticas de interacción aceptables

- | | |
|----|----|
| 1. | 5. |
| 2. | 6. |
| 3. | 7. |
| 4. | 8. |

Prácticas de interacción inaceptables

- | | |
|----|----|
| 1. | 5. |
| 2. | 6. |
| 3. | 7. |
| 4. | 8. |

Técnicas educativas utilizadas

- 1.
- 2.
- 3.
- 4.

Presentación del material en clase

Pizarrón Rotafolio Cañón Acetatos Otras (Indicar)

Presentación de tareas y/o prácticas

Libreta Computadora Laboratorio Programa Otras (Indicar)

Observaciones:

6.15. Plantilla de MeSoFT para el Diseño Computacional: (MOD)

Nombre del Proyecto _____ Fecha _____ Versión _____

Modelo de objetos		Versión:
Aprobó:	Líder de Equipo	

6.16. Plantilla de MeSoFT para el Diseño Computacional: (MOD)

Nombre del Proyecto _____ Fecha _____ Versión _____

Modelo funcional		Versión:
Aprobó:	Líder de Equipo	

6.17. Plantilla de MeSoFT para el Diseño Computacional: (MOD)

Nombre del Proyecto _____ Fecha _____ Versión _____

Modelo dinámico		Versión:
Aprobó:	Líder de Equipo	

6.18. Plantilla de MeSoFT para las Necesidades y Objetivos de Medición: (NEC)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Causa que origina la medición					
Gestión técnica					
Necesidades del producto					
Necesidades del proyecto					
Retraso en el plan del proyecto					
Seguimiento del proyecto					
Variación en el rendimiento					
Problemas en implementación del proceso definido en DP					
Desacuerdo del equipo					
Otros (especificar):					
Fase en que inicia la medición		Especificar actividades			
Diseño conceptual					
Planificación					
Requisitos					
Plan de Pruebas del Sistema					
Inspección de Requisitos					
Plan de Pruebas de Integración					
Planificación de la Implementación					
Diseño detallado					
Revisión del Diseño detallado					
Desarrollo de Pruebas					
Inspección Diseño Detallado					
Código					
Revisión de Código					
Repositorio de medidas del equipo		De producto	De proceso	Aprovechamiento	Combinación
Tipo de medidas a utilizar					

6.20. Plantilla de MeSoFT para describir la Estrategia Pedagógica: (PED)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Pondera los siguientes problemas (alumno)					
	Valoración				Solución habitual al problema
1 Dificultad al comprender ciertos temas	0	1	2	3	
2 Dificultad al comprender el curso completo	0	1	2	3	
3 El proceso de enseñanza es meramente teórico	0	1	2	3	
4 Dificultad de comprensión si se cambia la secuencia de los contenidos	0	1	2	3	
5 Dificultad alta en el aprendizaje	0	1	2	3	
6 Dificultad para asistir al curso	0	1	2	3	
7 Carencia de mecanismos de repaso	0	1	2	3	
8 Falta de profundidad en los temas	0	1	2	3	
9 Dificultad al realizar las prácticas del curso	0	1	2	3	

Soluciones alternativas (especialista) (ponderaciones 2 y 3 de alumno)								
	Solución			Consecuencia				
1		Social		Personal		Corto plazo		Largo plazo
2		Social		Personal		Corto plazo		Largo plazo
3		Social		Personal		Corto plazo		Largo plazo
4		Social		Personal		Corto plazo		Largo plazo
5		Social		Personal		Corto plazo		Largo plazo
6		Social		Personal		Corto plazo		Largo plazo
7		Social		Personal		Corto plazo		Largo plazo
8		Social		Personal		Corto plazo		Largo plazo
9		Social		Personal		Corto plazo		Largo plazo

6.21. Plantilla de MeSoFT para describir la Estrategia Pedagógica: (PED) (continuación)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Valoración de consecuencias por solución alternativa			
Número de solución:			
Estrategia o solución propuesta:			
Ventajas	Puntuación	Desventajas	Puntuación
Total			

Número de solución:			
Estrategia o solución propuesta:			
Ventajas	Puntuación	Desventajas	Puntuación
Total			

Número de solución:			
Estrategia o solución propuesta:			
Ventajas	Puntuación	Desventajas	Puntuación
Total			

Resumen de soluciones escogidas (máximas puntuaciones)	
1	4
2	5
3	6
Aprobó Profesor	

6.22. Plantilla de MeSoFT para describir el Problema educativo Identificado: (PROB)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Prácticas educativas aceptables			
1.	5.		
2.	6.		
3.	7.		
4.	8.		
Prácticas educativas inaceptables			
1.	5.		
2.	6.		
3.	7.		
4.	8.		
Recursos educativos disponibles			
Excelentes	<input type="checkbox"/>	Buenos	<input type="checkbox"/>
		Regulares	<input type="checkbox"/>
		Nulos	<input type="checkbox"/>
Disponibilidad de equipo de cómputo			
Excelente	<input type="checkbox"/>	Buena	<input type="checkbox"/>
		Regular	<input type="checkbox"/>
		Nula	<input type="checkbox"/>
Problemas detectados			
Enseñanza		Aprendizaje	
Descripción		Descripción	
1	_____	1	_____
2	_____	2	_____
3	_____	3	_____
4	_____	4	_____
5	_____	5	_____
Observaciones:			

6.23. Plantilla de MeSoFT para Entrevista a Profesores: (PROF)

Nombre _____ Fecha _____

Nombre del Proyecto _____

1. ¿Cuál es el proceso enseñanza-aprendizaje que desarrolló en su curso presencial?									
2. ¿Cuáles fueron las formas de comunicación o medios de interacción que utilizó en su curso presencial?									
3. ¿Considera que el proceso enseñanza-aprendizaje de su curso presencial puede mejorarse? Si su respuesta es afirmativa mencione las formas.									
4. ¿Considera que el rendimiento de sus alumnos fue adecuado o puede mejorarse aún más? Si su respuesta es afirmativa, ¿cuáles serían sus recomendaciones para lograrlo?									
5. ¿Cuáles son las prácticas, formas de evaluación y formas de interacción que utilizó como profesor?									
6. ¿Cuáles de estas considera se deben conservar?	7. ¿Cuáles de estas considera no se deben conservar?								
7. Si se le permitiera rediseñar el desempeño del proceso enseñanza-aprendizaje actual, ¿qué mejoras haría?									
8. En el caso de las mejoras, ¿estaría dispuesto a utilizar estas durante un nuevo curso?									
9. ¿Cuál es su nivel de dominio en cuanto al uso de la computadora?									
Experto	<input type="checkbox"/>	Avanzado	<input type="checkbox"/>	Medio	<input type="checkbox"/>	Normal	<input type="checkbox"/>	Nulo	<input type="checkbox"/>
10. ¿Considera la posibilidad de incluir el uso de la computadora como mejora al proceso enseñanza-aprendizaje? ¿En qué forma lo haría?									
Videojuego	<input type="checkbox"/>	Práctica virtual	<input type="checkbox"/>	Narrativa con personajes	<input type="checkbox"/>	Otras (indicar)			

6.24. Plantilla de MeSoFT para el Plan del Proyecto: (PROY)

Nombre _____ Fecha _____
 Nombre del Proyecto _____ Semana No. _____

Datos semanales		Planificado	Actual	
Horas trabajadas en el proyecto		_____	_____	
Valor ganado		_____	_____	
Horas totales para la terminación de las tareas en la fase		_____	_____	
Datos semanales por miembro de equipo	Horas planificadas	Horas actuales	Valor planeado	Valor ganado
Líder de equipo	_____	_____	_____	_____
Programador	_____	_____	_____	_____
Analista	_____	_____	_____	_____
Auditor de calidad	_____	_____	_____	_____
Soporte	_____	_____	_____	_____
Totales	_____			
Tareas completadas	Horas planificadas	Horas actuales	Valor planeado	Valor ganado
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
Totales	_____			
Seguimiento de los riesgos				
Número			Estado	
_____	_____		_____	
_____	_____		_____	
Otros detalles importantes				

6.25. Plantilla de MeSoFT para el Reporte de la Validación: (REPVALI)

Nombre _____ Fecha _____

Nombre del Proyecto _____

Referencia del criterio

Nombre del criterio

Prioridad del criterio Alta/Eencial Media/Deseado Baja/Opcional

Entrada al software

Salida Esperada

Salida Obtenida

Problema identificado

Riesgo asociado al problema
identificado

Recomendación

Referencia del criterio

Nombre del criterio

Prioridad del criterio Alta/Eencial Media/Deseado Baja/Opcional

Entrada al software

Salida Esperada

Salida Obtenida

Problema identificado

Riesgo asociado al problema
identificado

Recomendación

Aprobó: | Líder de Equipo

| Profesor

6.26. Plantilla de MeSoFT para el Seguimiento de los Riesgos: (RIE)

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

Fecha	Riesgo/problema	Número	Prioridad	Propietario	Fecha de análisis	Fecha de resolución

Descripción: _____

6.28. Taxonomía de Riesgos de MeSoFT

Nombre _____ Fecha _____
 Nombre del Proyecto _____

Especificación del riesgo			
Origen	Proceso <input type="checkbox"/>	Producto <input type="checkbox"/>	Entorno <input type="checkbox"/> Todos <input type="checkbox"/>
Consecuencia asociada	Costo <input type="checkbox"/>	Calendario <input type="checkbox"/>	Rendimiento <input type="checkbox"/> Calidad <input type="checkbox"/>
Severidad	Crítica <input type="checkbox"/>	Seria <input type="checkbox"/>	Moderada <input type="checkbox"/> Menor <input type="checkbox"/>
Probabilidad de ocurrencia	Alta <input type="checkbox"/>	Media <input type="checkbox"/>	Baja <input type="checkbox"/> % _____
Impacto del riesgo	Efectos a corto plazo <input type="checkbox"/>		Efectos a largo plazo <input type="checkbox"/>
Estrategia de mitigación recomendada			
Miembro del equipo a quién se asigna el riesgo _____		Rol _____	
Disparadores del riesgo	Acciones recomendadas para el plan de acción	Responsable	
_____	_____	_____	
_____	_____	_____	
_____	_____	_____	
_____	_____	_____	
_____	_____	_____	

Aprobado _____ Rechazado _____

6.29. Plantilla de MeSoFT para los Criterios de la Validación: (VALI)

Nombre _____ Fecha _____

Nombre del Proyecto _____

Referencia del criterio

Nombre del criterio

Tipo Procedimiento Interfaz Entrada Salida

Seguridad Portabilidad Rendimiento Mantenibilidad

Fuente del criterio Profesor Alumno Líder de Equipo Otro usuario

Prioridad del criterio Alta/Eencial Media/Deseado Baja/Opcional

Referencia del criterio

Nombre del criterio

Tipo Procedimiento Interfaz Entrada Salida

Seguridad Portabilidad Rendimiento Mantenibilidad

Fuente del criterio Profesor Alumno Líder de Equipo Otro usuario

Prioridad del criterio Alta/Eencial Media/Deseado Baja/Opcional

Referencia del criterio

Nombre del criterio

Tipo Procedimiento Interfaz Entrada Salida

Seguridad Portabilidad Rendimiento Mantenibilidad

Fuente del criterio Profesor Alumno Líder de Equipo Otro usuario

Prioridad del criterio Alta/Eencial Media/Deseado Baja/Opcional

Referencia del criterio

Nombre del criterio

Tipo Procedimiento Interfaz Entrada Salida

Seguridad Portabilidad Rendimiento Mantenibilidad

Fuente del criterio Profesor Alumno Líder de Equipo Otro usuario

Prioridad del criterio Alta/Eencial Media/Deseado Baja/Opcional

7. Anexo B.- Acrónimos

SE	Software Educativo
TI	Tecnologías de Información
ISE	Ingeniería de Software Educativo
RUP	Rational Unified Process
LISI	Laboratorio de Información de Sistemas
MOSCA	Modelo Sistémico de Calidad
SEBW	Software Educativo Basado en WEB
PVEI	Instrumentación Electrónica Programable y Virtual
UTM	Universidad Tecnológica de la Mixteca
IEP	Instrumentación Electrónica Programable
IEV	Instrumentación Electrónica Virtual
NIELVIS	National Instruments Educational Laboratory Virtual Instrumentation Suite
LabCD	Laboratorio de Comunicaciones Digitales
ATE	Automated Test Equipment
INEGI	Instituto Nacional de Estadística Geografía e Informática
OO	Orientado a Objetos
MEC	Materiales Educativos Computarizados
LOM	Learning Object Model
IAD	Inteligencia Artificial Distribuida
OKI	Open Knowledge Initiative
API	Application Programming Interface
ARIADNE	Autoría y Distribución de Instrucción Remota vía Redes para Europa
SCORM	Sharable Content Object Reference Model
ADL	Advanced Distributed Learning
SCA	Sharable Content Assets
SCO	Sharable Content Object
IRLCOO	Intelligent Reusable Learning Components Object Oriented
LI	Learning Interactions

LMS	Learning Management System
RLO	Reusable Learning Object
CAP	Component Agent Platform
RLC	Reusable Learning Components
RLCOO	Reusable Learning Components Object Oriented
LTSA	Learning Technology System Architecture
IEEE	Institute of Electrical and Electronics Engineers
LTSC	Learning Technology Standards Committee
AOCA	Arquitectura Orientada a Componentes y Agentes
FAQ	Frequently Asked Questions
SOA	Service Oriented Architecture
WSDL	Web Services Definition Language,
UDDI	Universal Description, Discovery, and Integration
CMMI/DEV	Capability Maturity Model Integration for Development
TSP	Team Software Process
AP	Áreas de Proceso
PP	Planificación del Proyecto
SP	Supervisión del Proyecto
MA	Medición y Análisis
GR	Gestión de Requisitos
VAL	Validación
DP	Definición del Proceso
OMT	Object-Modeling-Technique
IADIAL	Initiating, Analyze, Design, Implementing, Assessment, and Learning

8. Anexo C.- Actas de publicaciones

Instrumentation and Measurement
Technology Conference – IMTC 2007
Warsaw, Poland, May 1-3, 2007

An alternative methodology for developing virtual instruments in virtual labs

Hernandez-Martinez, Heriberto & Garcia-Matias, Josue
Electronics and Informatics Institute, Mixtec University of Technology
Huaquapan de Leon, Oaxaca (Mexico) <http://www.utm.mx>
(hhdez, josue)@mixteco.utm.mx

Abstract - This paper describes the configuration and management processes of hardware, firmware, and software resources of a Digital Communication Laboratory, linked to instrumentation networks. A real implementation using programmable electronic and virtual electronic instrumentations networked to a Virtual Laboratory shows that these tools are really valuable in providing academic and research support.

Keywords - Electronic instrumentation, virtual instrumentation, virtual laboratory.

1. INTRODUCTION

In recent years, an increasing demand of academic work has been experienced due to changes in educational models. These new models need to be adapted to the ongoing evolution of new technologies such as Digital Communication Laboratory (DCLab). The DCLab provides advantages over traditional laboratories in that it has similar resources to those used in organizations and it could be integrated into three main areas: i) instrumentation networks, ii) wireless communications and iii) optic communications.

In addition, the DCLab's management groups different areas such as electronic instrumentation and microcontroller based systems and it also provides a platform for high-level software applications used in industry and educational environments [1, 2]. Thus the DCLab has the benefit to directly relate instrumentation networks making possible to design and develop devices and systems through a network environment.

This paper therefore shows an application of DCLab technology as an alternative methodology for developing virtual instrument in virtual labs. This alternative has showed to be excellent in supporting the academic and research work conducted at the Mixtec University of Technology (UTM). The DCLab structure used at the UTM and its description is detailed throughout this paper.

2. BACKGROUND OF ELECTRONIC INSTRUMENTATION

In 1965, Hewlett Packard (now known as Agilent Technologies) presented the Hewlett Packard-Interface Bus (HPIB) for electronic equipment intercommunica-

tion which later became the standard IEEE 488 or General Purpose Interface Bus (GPIB) [3, 4]. In 1987, the limitations of the GPIB in terms of the physical specifications led to an extension entitled IEEE 488.1. This extended version guaranteed the easy programming of instruments through the standardization of message exchange protocols, data formats, syntax formats, status reports and standard commands to different kinds of instruments. Nevertheless, the new version entitled IEEE 488.2 came after, although it did not provided the Automated Test Equipment (ATE) systems design [5, 6].

Later, in 1990, the ATE was accessible with the Standard Commands for Programmable Instruments (SCPI) norm that defined a individual conceptual model for programmable instrument design. This new model had the advantage to increase the GPIB instrument processing capability and data transfer rate over the GPIB bus (1 Mbps).

Simultaneously, in 1987, the VME eXtension for Instrumentation (VXI) architecture was introduced. This architecture was based on the VME bus, with modular instruments inserted in a chassis, having reached a data transfer rate over 40 Mbps.

More recent, in 1997, the industry adopted the PCI eXtensions for Instrumentation (PXI) standard [7], which compared to the VXI offered greater velocities and it was completely compatible with the personal computer's architecture.

As a consequence programmable electronic instrumentation, based on the GPIB and VXI busses, nowadays enables the design of ATE systems with different computer and operating system platforms using the concept of Virtual Instrument (VI) [1].

3. IMPLEMENTATION OF THE DCLAB TECHNOLOGY

In order to support academic and research at the UTM, the DCLab technology was introduced as detailed in this paper. Roughly speaking, the used DCLab comprises a Local Area Network (LAN) interface and a LAN/GPIB gateway to connect the electronics devices. The main DCLab structure is illustrated in Figure 1.

Implementing Virtual Practices using an Alternative Methodology to Develop Educational Software

Garcia Ivan¹, Garcia Josue²

¹Postgraduate Department, Technological University of the Mixtec Region

²Advanced Electronics Laboratory, Technological University of the Mixtec Region
Huajuapán de León, Oaxaca (Mexico) <http://www.utm.mx>
{ivan, josue}@myoo.utm.mx

Abstract

Educational Software is one of the pillars of distance learning and educational systems and has become the basic tool for future generations of students. However, recent methodologies used in this development have too many problems: a lack of common theoretical frameworks which can be used by anyone in the project, and excessive formality in both technical and pedagogical factors. Activities employed in the development of educational software are complex because the process depends on the developer's expertise, aspects of software engineering, and the acquisition and implementation of pedagogical knowledge. We propose the introduction of "effective practices" within an alternative methodology to develop this kind of software. The identification of effective practices is focused internally to ensure the effective realization of the development process, and externally to guide the pedagogical monitoring of a project. Our effective practices provide the basis of an alternative methodology for the development of educational software under rigorous conditions that enable us to achieve a highly successful and repeatable process in the field of electronic instrumentation.

1. Introduction

A traditional problem in educational instruments is to design them in such a way that they effectively accomplish the purpose for which they were created, which means that communication and interaction with the user should be effective. The designer, however, faces a different issue, namely, how to improve the learning process and store all information that creates a variety of learning environments [4].

Educational Software (ES), or courseware, is defined as a didactic instrument to facilitate effective teaching-learning processes in traditional, classroom-based as

well as distance learning. According to [2], [5], [8], [13], [14], [15] and [17] we know that it is possible to substantially improve academic performance and that results can vary according to the software used and the methodology applied in its development.

Nowadays, the term "educational" is added on to any product designed for teaching purposes. ES, however, has not been used as a formal process of teaching specifically designed to acquire the knowledge, skills and procedures to ensure that all students obtain just the necessary knowledge [10]. ES development requires specific conditions that include didactic and pedagogic issues within the analysis and design phases.

The evolution of Information Technologies (IT) has provided a new alternative in the field of education. This has given rise to important changes in the educational community and brought advantages to all of its members, even the modification of traditional models of teaching. Virtual Labs have arisen as an alternative improvement to the concept of "distance education". The term distance education originated in countries which were having problems meeting the demands for education. It therefore became necessary to develop new technologies to expand educational opportunities to more and more students.

The technological advantages of software and hardware make for better resource management through the development of interactive systems. These systems must be capable of providing just the necessary knowledge based on the student's level and field of interest.

However, some academic institutions lack the necessary infrastructure to do this (equipment, teachers, installations, etc.). The development of Virtual Labs or Virtual Instrumentation, for example, provides an alternative solution to these problems, specifically in the field of Electronic Sciences, [9].



MeSoft: Una Metodología Alternativa para el Desarrollo de Software Educativo basada en prácticas efectivas

Garcia Ivan¹, Garcia Josué²

¹Departamento de Postgrado, Universidad Tecnológica de la Mixteca

²Laboratorios Avanzados de Electrónica, Universidad Tecnológica de la Mixteca
Huajuapán de León, Oaxaca (Mexico) <http://www.utm.mx>
(ivan, josue}@nuyoo.utm.mx

Abstract

El software educativo es uno de los pilares de los sistemas de enseñanza-aprendizaje a distancia que es utilizado como una herramienta para las generaciones futuras de estudiantes. Sin embargo, las recientes metodologías para el desarrollo de software educativo tienen demasiados problemas como la carencia de marcos de trabajo comunes los cuales puedan ser utilizados para cualquier proyecto, y la excesiva formalidad de ambos factores, el técnico y el pedagógico. Las actividades para el desarrollo de software educativo son complejas porque el proceso está enfocado en la experiencia del desarrollador, como los aspectos técnicos de la Ingeniería de Software y la adquisición e implementación del conocimiento pedagógico. Este trabajo propone la introducción de las "prácticas efectivas" en una metodología alternativa para desarrollar software educativo. La identificación de prácticas efectivas está enfocada a asegurar que el desarrollo del proceso sea conducido con eficacia y orientado a la supervisión pedagógica del proyecto. Las prácticas efectivas que aquí se proponen proporcionan las bases de una metodología alternativa para desarrollar software educativo con el rigor necesario para desarrollar software comercial, eso nos permite obtener un proceso que se puede repetir con altos niveles de éxito en el área de la instrumentación electrónica, específicamente.

1. Introducción

Un problema tradicional de los instrumentos educativos es como diseñarlos para lograr con eficacia los objetivos para los cuales fueron creados, esto significa, que la comunicación e interacción con el usuario pueda ser efectiva. Sin embargo, la premisa del diseñador es otra, mejorar el proceso de aprendizaje y almacenar toda la información que permita una variedad de ambientes de aprendizaje [4].

El Software Educativo (SE), o también llamado *courseware*, se define como un instrumento didáctico efectivo que permite el proceso de enseñanza-aprendizaje en la forma tradicional, presencial o a larga distancia. De acuerdo con [2], [5], [8], [13], [14], [15] y [17] sabemos que es posible mejorar el funcionamiento académico en forma substancial y que estos resultados pueden variar por el software usado y la metodología aplicada para su desarrollo.

Actualmente, el término "educativo" se agrega a cualquier producto diseñado con alguna intención de enseñar un tema específico. Sin embargo, el SE no se ha utilizado como un proceso formal de enseñanza que posea un diseño específico para adquirir el conocimiento,

A Methodology Based on Effective Practices to Develop Educational Software *Una Metodología Basada en Prácticas Efectivas para Desarrollar Software Educativo*

Ivan García Pacheco¹ and Josué García Matías²

¹Postgraduate Department, Technological University of the Mixtec Region

²Department of Information Technology, University of the Sierra Juárez

¹Huajuapán de León, Oaxaca (Mexico) <http://www.utrn.mx>

²Ixtlan de Juárez, Oaxaca (Mexico) <http://www.unsij.edu.mx>

ivan@mixteco.utrn.mx, josue@juppa.unsij.edu.mx

Article received on May 09, 2007; accepted on December 11, 2007

Abstract

Educational Software is one of the pillars of distance learning and educational systems and has become the basic tool for current generations of students. However, recent methodologies used in this development have too many problems: a lack of common theoretical frameworks which can be used by anyone in the project, and excessive formality in both technical and pedagogical factors. Activities employed in the development of educational software are complex because the process depends on the developer's expertise, aspects of software engineering, and the acquisition and implementation of pedagogical knowledge. We propose the introduction of "effective practices" within an alternative methodology to develop this kind of software. The identification of effective practices is focused internally to ensure the effective realization of the development process, and externally to guide the pedagogical monitoring of a project. Our effective practices provide the basis of an alternative methodology for the development of educational software under rigorous conditions that enable us to achieve a highly successful and repeatable process in the field of electronic instrumentation.

Keywords: Effective practices, learning tools, educational software based on the Web, educational technologies, virtual instrumentation, distance learning.

Resumen

El software educativo es uno de los pilares de los sistemas de enseñanza-aprendizaje a distancia que es utilizado como una herramienta para las generaciones futuras de estudiantes. Sin embargo, las recientes metodologías para el desarrollo de software educativo tienen demasiados problemas como la carencia de marcos de trabajo comunes los cuales puedan ser utilizados para cualquier proyecto, y la excesiva formalidad de ambos factores, el técnico y el pedagógico. Las actividades para el desarrollo de software educativo son complejas porque el proceso está enfocado en la experiencia del desarrollador, como los aspectos técnicos de la Ingeniería de Software y la adquisición e implementación del conocimiento pedagógico. Este trabajo propone la introducción de las "prácticas efectivas" en una metodología alternativa para desarrollar software educativo. La identificación de prácticas efectivas está enfocada a asegurar que el desarrollo del proceso sea conducido con eficacia y orientado a la supervisión pedagógica del proyecto. Las prácticas efectivas que aquí se proponen proporcionan las bases de una metodología alternativa para desarrollar software educativo con el rigor necesario para desarrollar software comercial, esto nos permite obtener un proceso que se puede repetir con altos niveles de éxito en el área de la instrumentación electrónica, específicamente.

Palabras clave: Prácticas efectivas, herramientas educativas, software educativo basado en Web, tecnologías educativas, instrumentación virtual, educación a distancia.

1 Introduction

Education, as a fundamental function of society, is obligated not only to inculcate values, provide knowledge, develop abilities and educate people; but also it is obligated to generate and maintain knowledge, designing and implementing novel learning-teaching modalities, creation of tools that satisfy the student requirements; but, the most important aspect, it must be capable of responding to the continuous social transformations of a world in permanent evolution. Educational Software (ES), or courseware, is defined as a didactic instrument to facilitate effective teaching-learning processes in traditional, classroom-based as well as distance learning. [Adomavicius, 2004; Cataldi, 2003; Garcia, 2002; Kocijanci and O'sullivan, 2004; Lipeikine, 2003; Marqués, 1995; Squires and

9. Bibliografía

- [**Adams04**] Adams, Richard J.; Eslinger Suellen; Owens, Karen L; Rich, Mary A.: “Software Acquisition Best Practices 2004 edition”. 3° OSD Conference on the Acquisition of Software-Intensive Systems, The Aerospace Corporation, January, 2004.
- [**Arias02**] Arias, M., López, A. y Rosario H.: “Metodología Dinámica para el Desarrollo de Software Educativo”. Ponencia de la III Conferencia Internacional sobre Educación, Formación y Nuevas Tecnologías, VirtualEduca 2002, Valencia, España, Junio 2002.
- [**Ayala04**]. Ayala, A. y Azuela, J. “Educación Basada en WEB: Un Estado del Arte”, Unidad Profesional Interdisciplinaria de Ingeniería y Ciencias Sociales y Administrativas (UPIICSA) y Centro de Investigación en Computación del IPN (CIC). XX Simposio Internacional de Computación en la Educación SOMECE2004, Puebla, México, 2004.
- [**Ayala98**] Ayala, G. & Yano, Y. Collaborative learning environment based on intelligent agents, Elsevier, Expert Systems with Applications. 14. 1998, pp. 129-137
- [**Brusilovksy00**] Brusilovksy, P. Web Lectures Electronic Presentations in Web-Based Instruction, Syllabus, No.1, pp. 18-23. 2000.
- [**Cabero92**] Cabero Almenara Julio: “Diseño de software informático”. Revista Bordon, ISSN: 0210-5934 Volumen 44, Número 4, pp. 383-391, España, 1992.
- [**Cataldi99**] Cataldi, Z., Lage, F., Pessacq, R. y García Martínez, R.: “Ingeniería de software educativo”. VI Congreso Internacional de Ingeniería Informática, ICIE99, Universidad de Buenos Aires, Buenos Aires, Argentina, Agosto 1999.
- [**Cataldi03**] Cataldi, Z., Lage, F., Pessacq, R. y García Martínez, R.: “Metodología extendida para la creación de software educativo desde una visión integradora”. Revista Latinoamericana de Tecnología Educativa, RELATEC, Universidad de Extremadura, ISSN - 1695-288X, Volumen 2 Número 1, España, 2003.
- [**Canales07**] Canales, A., Peña, A., Peredo, R., Sossa, H. and Gutiérrez, A. “Adaptive and intelligent web based education system: Towards an integral architecture and framework”. Experts Systems with Applications No. 33, Elsevier Ltd., pp. 1076-1089. 2007.
- [**Cumming98**] Cumming, G., Okamoto, T. and Gomez, L. Advanced Research Computers in Education, IOS 1998.
- [**Diaz-Anton03**] Díaz-Antón, Maria G., Pérez Maria A., Grimman Anna C. y Mendoza Luís E.: “Propuesta de una Metodología de Desarrollo de Software Educativo Bajo un Enfoque de Calidad Sistémica”. IV Congreso Multimedia Interactivo, Universidad de Barcelona, Barcelona, España, 2003.

- [**Diaz02**] Díaz de Feijoo, María Gabriela.: Propuesta de una metodología de desarrollo y evaluación de software educativo bajo un enfoque de calidad sistémica. Trabajo especial de para obtener el grado de especialista en informática, Decanato de estudios de Postgrado, Universidad Simón Bolívar, Caracas, Venezuela. Junio 2002.
- [**Fernandez99**] Fernández Silano Mariano: “Modelos de Desarrollo de Software”. Revista Agenda Académica, Volumen 6, Número 2, pp. 135-139, Venezuela, 1999.
- [**Galvis98**] Gómez Castro Ricardo A., Galvis Panqueva Alvaro H., y Mariño Drews Olga: “Ingeniería de software educativos con modelaje orientado por objetos: Un medio para desarrollar micromundos interactivos”. Informática Educativa Vol 11, No. 1 pp. 9-30, UNIANDES-LIDIE, Universidad de los Andes, Bogota, Colombia, 1998.
- [**Garcia02A**] García Roselló, E., González Dacosta, J., , García Pérez-Schofield, J.B., Valdés Pardo V.G.: “¿Existe una situación de crisis del software?”. VI Congreso Iberoamericano de Informática Educativa, RIBIE02, España, Noviembre 2002.
- [**Garcia02B**] García Roselló, E., González Dacosta, J., Valdés Pardo V.G., Mandado, E., Baltasar García J., Pérez Cota, M.: “Una propuesta para la reutilización de componentes en el proceso de desarrollo de software educativo”. VI Congreso Iberoamericano de Informática Educativa, RIBIE02, España, Noviembre 2002.
- [**Garcia03**] García, H, Reyes, C.A. y Morales, R.. “Diseño e implementación de mapas cognitivos difusos para tutoriales inteligentes”. XVI Congreso Nacional y II Congreso Internacional de Informática y Computación de la ANIEI, Zacatecas, Octubre 22 al 24, 2003.
- [**Garcia04**] García Matías, Josué N.: Administración de los Recursos de un Laboratorio de Comunicaciones Digitales Mediante su Instalación, Configuración y Puesta en Funcionamiento. Tesis de licenciatura, Universidad Tecnológica de la Mixteca. Junio 2004.
- [**Garcia07**] Garcia, J. & Martinez, H. “An alternative methodology for developing virtual instruments in virtual labs”. IMTC2007, Instrumentation and Measurement Technology Conference Proceedings. IEEE, Warsaw, Poland, October 2007.
- [**Gonzalez98**] González Castañón Miguel Ángel: “Evaluación de Software Educativo: Orientaciones para su uso pedagógico”. Proyecto enlaces, Universidad del Bío-Bío Medellín, Colombia
- [**Humphrey00**] Humphrey, W. Introduction to Team Software Process. SEI Series in Software Engineering. Addison Wesley, 2000.
- [**Jaques03**] Patricia A. Jaques, Magda Bercht, Everton Bocca, Rosa M. Vicari. Cognitive Reasoning to Respond Affectively to the Student. Proceedings of the IASTED International Conference, Computers and Advanced Technology in Education; June 30 – July 2, 2003, Rhodes, Greece.
- [**Kimovski03**] Kimovski, G., Trajkovic, V. and Davcev, D. “Negotiation-based Multi-agent Resource Management in Distance Education”. Proceedings of the IASTED International Conference, Computers and Advanced Technology in Education; Rhodes, Greece, 2003.
- [**Lipeikine03**] Lipeikine Joana: “Virtual Learning Environments as a Supplement to Traditional Teaching”. Informatics in Education, Vol. 2 No. 1 pp. 53-64., Institute of Mathematics and Informatics, Vilnius, Lithuania, April 2003.
- [**Marques95**] Marqués Pere.: Metodología para elaboración de software educativo. Software Educativo., Guía de uso y metodología de diseño, Editorial Estel, Barcelona, España, 1995.
- [**Miltiadis03**] Miltiadis D. Lytras & Angeliki Tsilira. “Semantic Web-Challenges for E-Learning: a Discussion of the New Research Agenda in the E-Learning Field”. Proceedings of the IASTED

International Conference, Computers and Advanced Technology in Education; Rhodes, Greece, 2003.

[**Okamoto03**] Okamoto, T. “The Model of Collaborative Learning and Technological Environment for Evoking Interactivity-Building of Knowledge”. Proceedings of the IASTED International Conference, Computers and Advanced Technology in Education; June 30 – July 2, 2003, Rhodes, Greece

[**Peredo05**] Peredo, R., García, C., Fabela, O., Larios, E. y Ramos, E. “CSPR(i) versión 3.0 Orientados a Objetos para Materiales Multimedia en Educacion a distancia”, XVI Congreso Nacional y II Congreso Internacional de Informática y Computación de la ANIEI, México, 2003.

[**Peredo05**] Peredo, R., Balladares, L. & Sheremetov, L. “Development of Intelligent Reusable Learning Objects for Web-based Education Systems”, Experts Systems with Applications No. 28, Elsevier Ltd., pp. 273-283, 2005.

[**Peredo08**] Peredo, R., Canales, A., Balladares, L. y Menchaca, A. Arquitectura Para Sistemas De Educación Basada En Web Usando Programación Orientada A Componentes. Revista Diálogo Educ., Vol. 8, Num. 24, Págs. 485-502, Curitiba, Brasil, mayo/agosto 2008.

[**Ramirez04**] Ramirez, J. and Flores, M. “Change in the Educational Scheme from a Traditional to a Redesigned System: Case Study of the Management Department at the ITESM-Campus Monterrey”. Proceedings of Networked Learning Conference, 2004.

[**Rebak00**] Rebak, Daniel R., Blackmon, Bill, and Humphreys, Anne R.; Carnegie Mellon Research Institute, “The Virtual University: Developing a dynamic learning management portal”, Research profile, IEEE magazine, July – September 2000.

[**Sheremetov01**] Sheremetov, L. & Uskov, V. “Hacia la nueva generación de sistemas de aprendizaje basado en la Web (Towards the New Generation of Web-based Learning Environments)”. Computación y Sistemas, 2001.

[**Sheremetov02**] Sheremetov, L. & Peredo, R. “Development of Reusable Learning Materials for WBE using Intelligent Components & Agents”, Technical Report, Instituto Mexicano del Petróleo y Laboratorio de Agentes del Centro de Investigación en Computación, México, 2002.

[**Squires01**] Squires, D.; McDougall, A.: Como elegir y utilizar software educativo, Editorial Morata, Segunda edición, ISBN 84-7112-418-1, Madrid, España, 2001.

[**Software06**] Software Engineering Institute: CMMI for Development (CMMI-DEV, V1.2). CMU/SEI-2006 TR-008, Software Engineering Institute, Carnegie Mellon University. 2006.

[**UNAM00**] UNAM (Ed.). La educación superior en el siglo XXI. México: UNAM, 2000.

[**Walker03**] Walker, E: “Implementing Best Practices in the Join Battlespace Infosphere (JBI) program at AFRL”. Conference on the Acquisition of Software-Intensive System, Carnegie Mellon University, January 28-30, 2003.

9.1. Sitios de Internet

[**URL1**] <http://www.inegi.gob.mx> , “Página del Instituto Nacional de Estadística, Geografía e Informática”, INEGI 2007.

[**URL2**] http://discovery.chillan.plaza.cl/~uape/actividades/etapa2/software/doc/software_edu.htm “El software educativo”. Marqués Pere. Universidad Autónoma de Barcelona. 2007

[**URL3**] http://www.ni.com/academic/ni_elvis/esa/lfb.htm, “Página de NI ELVIS de la firma National Instruments”, NI Inc. 2007.

- [URL4] <http://estudios.universia.es/recursos/auladigital/educacion/software-educativo/index.htm> “Página del Portal Universia”, Diseño y Evaluación de Programas Educativos. Marqués Pere. Universidad Autónoma de Barcelona. 2003
- [URL5] <http://www.teddi.ua.es/> “Página Principal de TEDDi”, Tecnología, EDucación, Desarrollo, Innovación. Universidad de Alicante. 2003
- [URL6] <http://mondragon.angeltowns.net/paradiso/>, “Página de Paraíso LOGO”, 2006
- [URL7] http://es.wikipedia.org/wiki/Software_educativo, “Pagina principal del portal Wikipedia”, 2007
- [URL8] <http://www.monografias.com/trabajos14/taller/taller.shtml#cuat>, “Página del Curso taller Hacia una educación universitaria de calidad y mejora continua”, 2005
- [URL9] <http://www.uhu.es/cine.educacion/didactica/0015tecnologiaeducativa.htm>, “Página principal de Metodología y Didáctica en la educación”, Enrique Martínez-Salanova Sánchez, Feb, 2008
- [URL10] <http://www.adlnet.org>, “ADL. Advanced Distributed Learning”. 2007.
- [URL11] <http://ltsc.ieee.org/wg1/files/ltsc-400.html>. IEEE/LTSC. Institute of Electrical and Electronic Engineers, Inc. / Learning Technology Systems Architecture.
- [URL12] <http://web.mit.edu/oki>. Web page of “OKI. Open Knowledge Initiative”, MIT Massachusetts Institute of Technology. 2008
- [URL13] <http://www.ieeeltsc.org/working-groups/wg12LOM/> Web page of “Learning Object Metadata standard” 2007
- [URL14] <http://ariadne.unil.ch/main.navigation.html>ARIADNE. Web page of “Alliance of Remote Instructional Authoring and Distribution Networks for Europe”, 2007.
- [URL15] <http://ieeeltsc.org/>. Web page of “Committee IEEE of Learning Technology Systems Architecture”, 2008.
- [URL16] <http://www.adlnet.gov/> Web page of Advanced Distributed Learning, official Web site of the U.S. Government, 2007

9.2. Otras referencias

- [Adomavicius04] Adomavicius J., Bareisa, E., Kersiene, V., Sekliuckis, V.: “Modernization of Information Technologies Studies at University Level”. Informatics in Education, Vol. 3 No. 1 pp. 3-18., Institute of Mathematics and Informatics, Vilnius, Lithuania, 2004.
- [Britain99] Britain, S., Liber, O.: “A Framework for Pedagogical Evaluation of Virtual Learning Environments”. JISC Technologies Applications, University of Wales, Bangor, October 1999.
- [Chin03] Chin Paul.: Virtual Learning Environments. LTSN Physical Sciences Practice Guide, ISBN 1-903815-06-1, February 2003.
- [CINEA03] Centro en Investigaciones en Estadística Aplicada, “Estudio sobre la Educación a Distancia”. Universidad Tres de Febrero, Buenos Aires, Argentina, Agosto, 2003.
- [Giraldo02] Giraldo, H., Álvarez, M. y Aedo, J.: “Implementación de una tecnología para la construcción de laboratorios remotos para la enseñanza de electrónica usando Internet”. Vicerectoría de investigaciones de la Universidad de Antioquia, Medellín, Colombia, Junio 2002.
- [Gonzalez02] González Ramírez Byron: “Criterios y Métodos de Evaluación de Software Educativo”. Eje de Capacitación en Informática, Universidad de San Carlos de Guatemala, Guatemala, Noviembre 2002.

- [**Gomez01**] Gómez, Margarita Victoria: Paulo Freire y la agenda de la educación latinoamericana en el siglo XXI, Primera Edición, Editorial CLACSO, ISBN 950-9231-63-0, Septiembre 2001.
- [**Gros00**] Gros Salvat Begoña: “Del Software Educativo a Educar con Software”. Revista Quaderns Digital, Num. 24 Software Educativo, ISSN 1575-9393, España, 2000.
- [**Kocijancic04**] Kocijancic S. y O’sullivan C.: “Real or Virtual Laboratories in Science Teaching is This Actually Dilema?”. Informatics in Education, Vol. 3 No. 2 pp. 239-250, Institute of Mathematics and Informatics, Vilnius, Lithuania, August 2004.
- [**Kurilovas05**] Kurilovas Eugenijus: “Several Aspects of Technical and Pedagogical Evaluation of Virtual Learning Environments”. Informatics in Education, Vol. 4 No. 2 pp. 215-252, Institute of Mathematics and Informatics, Vilnius, Lithuania, August 2005.
- [**Mancinas99**] Mancinas Abelardo: “¿Qué Modelos Pedagógicos para qué Nuevas Tecnologías de la Información?”. XV Simposio, Sociedad Mexicana de Computación en Educación 1999, SOMECE, Guadalajara, México, 1999.
- [**Markuskaite02**] Markuskaite Lina: “Information Technology Implementation: Analysis Theoretical Methods and Frameworks”. Informatics in Education, Vol. 1 pp. 111-128, Institute of Mathematics and Informatics, Vilnius, Lithuania, June 2002.
- [**Pizarro05**] Pizarro, Rubén Adrián: “Software educativo para temas de calculo”. Tecnología de la Información y Comunicación en la Enseñanza de las Ciencias, TICEC05, Argentina, 2005.
- [**Rumbaugh97**] Rumbaugh, J., BlaHa, M., Premerlani, W., Frederick, E. and Lorensen, W. Modelado y Diseño Orientado a Objetos: Metodología OMT. Editorial Prentice Hall, 1997.
- [**Salazar04**] Salazar Roselis: “Un Modelo para el Uso de las TICS en Educación a Distancia”. VIII Congreso de Educación a Distancia CREAD MERCOSUR/SUL 2004, Córdoba, Argentina, 2004.
- [**Salcedo00**] Salcedo Lagos Pedro: “Ingeniería de Software Educativo, Teorías y metodologías que la sustentan”. Revista Ingeniería Informática, Edición número 6, Departamento de Ingeniería Informática y Ciencias de la Computación DIICC, Universidad de Concepción, Concepción, Chile, 15 Septiembre 2000.
- [**Sosa04**] Sosa Jost, Eliana B., Valesani, Maria E. y Mariño, Sonia I.: “Software Educativo para Apoyo a la Enseñanza de la Técnica de Modelado y Diseño Orientado a Objetos”. Comunicaciones Científicas y Tecnológicas 2004, Universidad Nacional del Nordeste, Corrientes, Argentina, 2004.

