

ISO SPICE EN SISTEMAS HIPERMEDIA EDUCATIVA

DANIEL JOSÉ SALAS ÁLVAREZ
Grupo de Investigación CIDLIS
Universidad Industrial de Santander
dsalas@cidlisuis.org

IVÁN ALFONSO GUARÍN VILLAMIZAR
Grupo de Investigación CIDLIS
Universidad Industrial de Santander
ivanalgu@cidlisuis.org
GUIA

RICARDO LLAMOSA VILLALBA
Profesor Escuela de Ingeniería Eléctrica, Electrónica y
Telecomunicaciones
Grupo de Investigación CIDLIS
Universidad Industrial de Santander
nrllamos@cidlisuis.org

RESUMEN

La investigación busca el mejoramiento continuo del proceso de desarrollo de sistemas Hipermedia educativa a través de la interpretación, adaptación y aplicación del estándar ISO SPICE ("Software Process Improvement and Capability Determination") [6]

El modelo que se describe considera todas las fases [12] del proceso de desarrollo de software y su ciclo de vida está orientado por el enfoque de prototipado, e incremental, enriquecido con técnicas rigurosas de construcción de software soportadas en UML (Unified Model Language) [11] y sus extensiones. El modelo se apoya en la teoría pedagógicas constructivistas de la flexibilidad cognitiva de Spiro [21] y Feltovich [8] y las ventajas del hipertexto e hipermedia en educación

PALABRAS CLAVES: Calidad, ISO- SPICE, Sistemas Hipermedia Educativa, Ingeniería del Software, Modelos de ciclo de vida.

ABSTRACT

This research is aiming for the continuous improvement of development process on Educational Hipermedia Systems by interpreting, adapting and the application of the ISO SPICE standard ("Software Process Improvement and Capability Determination") [6]

The model described considers all software development process phases [12] and its life cycle is using the prototyping and incremental approaches, besides it is using rigorous techniques for software construction supported by UML (Unified Model Language) [11] and its extensions. This model is supported by the constructivists pedagogical theories of cognitive flexibility from Spiro [21] and Feltovich [8], and the advantages of hypermedia and hypertext in education.

KEYWORDS: Quality, ISO- SPICE, Educational Hypermedia Systems, Software Engineering, Life cycle Models.

INTRODUCCIÓN

Cualquier tipo de software, Convencional, Hipertexto, Multimedia o Hipermedia, que se desarrolle, siempre convivirá con un término común: "la calidad". La

calidad en software se garantiza con la utilización de procesos y recursos apropiados, y se establece, para lograr el mejoramiento continuo, buscando siempre la plena satisfacción del usuario, y en lo posible, el logro de la máxima productividad.

La calidad del software está estrechamente relacionada con la calidad de los procesos. En este sentido, se requiere disponer de mecanismos concretos para observar, especificar, estimar, planificar, evaluar, analizar, sintetizar, verificar y predecir el comportamiento de los procesos en todo el ciclo de vida de construcción de un sistema, como es el caso de los Sistemas Hipermedia Educativa (SHE).

Se puede afirmar que la calidad en los SHE se consigue mediante el establecimiento de un conjunto de características deseables o ideales, con una mezcla equilibrada de procesos y recursos de producción con los que se logra productividad y satisfacción del usuario.

A nuestro juicio, la calidad de los procesos para la construcción en SHE, debe garantizarse durante todo el ciclo de vida de desarrollo, no al final. De acuerdo a esta razón el trabajo de investigación que sustenta este artículo, pretende, a través del uso de estándares de producción y los niveles de calidad definidos en ISO-SPICE, mejorar y asegurar una óptima producción de los SHE.

Desde la década de los 90, cuando aparecieron las primeras metodologías de Sistemas Hipermedia, tales como: HDM (Hipermedia Design Methodology)[9], RMM (Relationship Management Methodology)[13], OOHDM (Object Oriented Design Methodology [19], EORM (Enhanced Object Relationship Methodology)[15], entre otras, dichas metodologías han carecido dos grandes factores: la calidad de procesos de desarrollo y el uso de métodos y herramientas de ingeniería del software con las que se establezcan las mejoras prácticas de producción software.

Esta propuesta a diferencia de los trabajos anteriores [9][13][19][15], permite el aseguramiento de la calidad del proceso de producción software de SHE, a través de la aplicación de ISO SPICE, lográndose:

- 1.) Conocer el estado actual y establecer guías de mejora al proceso
- 2.) Establecer mejores prácticas de producción.
- 3.) Establecer estrategias para facilitar el manejo de la complejidad del proceso

El modelo propuesto, soportado en los paradigmas más actuales de ingeniería de software, puede contribuir a mejorar significativamente el desarrollo de SHE, mediante el adecuado uso de:

- 1.) Modelos de procesos[12];
- 2.) Técnicas de especificación de requisitos, diseño, implementación, verificación, validación, mantenimiento y retiro [20];
- 3.) Métodos de planificación y control del proyectos [18].

Este artículo presenta en un primer capítulo esta introducción, en una segunda parte el estado del arte de la

investigación, en tercer lugar la estructura general de modelo de ciclo de vida MCV-SHE. Luego en una cuarta parte se describe la forma de verificación del modelo, después se presenta la estrategia que se está siguiendo para la validación a través de los procesos y actividades de desarrollo y aplicación en SHE. Y finalmente en la quinta, sexta y séptima parte se presentan las conclusiones, el trabajo futuro, agradecimientos y la bibliografía que sustenta este artículo.

ANTECEDENTES

A. Metodologías de diseño de sistemas hipermedia

A principios de la década de los 90, se gestaron las primeras metodologías de diseño de sistemas hipermedia, algunas basadas en enfoques relacionales y otras agrupadas en enfoques orientadas a objetos, combinación de ambos enfoques y basadas en escenarios, las cuales ofrecen un conjunto de características que no poseen los modelos de ciclo de vida tradicionales, tales como: estructura y presentación del contenido y la navegación, entre otros.

La tabla No.1. muestra un resumen de las más relevantes metodologías de diseño de sistemas hipermedia, basadas en enfoques: relacional, objetos, relacional y objetos y escenarios.

Tabla 1. Metodologías de diseño de sistemas hipermedia

Metodología	Técnica de Modelado	Notación	Proceso Completo del Software	Aseguramiento de Calidad del proceso del Software
HDM (Hypermedia Desing Methodology)	Relacional	Entidad - Relación	No	No
RMM (Relationship Managament Methodology)	Relacional	Entidad-Relación	No	No
EORM (Enhanced Object Relationship Methodology)	Orientada a Objetos	OMT	No	No
OOHDM (Object Oriented Hypermedia Design Methodology)	Orientada a Objetos	OMT/UML propia y Adv's	No	No
SOHDM[16] (Scenary- based Object Hypermedia Design Methodology)	Escenarios y Vistas Orientadas a Objetos	Escena-rios, diagra-mas de clases y vistas orientas a Objetos	No	No
WSDM[5] (Web site Desgin Method)	Relacional y Orientada a Objetos		No	No
RNA[1] (Relationship Navegational Analysis)			No	No

Cada una de las metodologías aporta elementos importantes al desarrollo de software de tipo hipermedial, sin embargo se observa dos grandes debilidades: la aplicación de normas de calidad para las mejores prácticas en el desarrollo de software y la ausencia de fases del proceso del software. El modelo MCV-SHE, sustentándose en OOHDM, pretende proponer una solución a las debilidades expuestas anteriormente.

B. Estándares de calidad del software

Existe una amplia literatura de normas de calidad para las mejores prácticas en la producción de software, entre las que se destacan: ISO-9000, ISO-1207, CMM (Capability Maturity Model) e ISO-SPICE. ISO SPICE, está emergiendo como la norma más sólida para el mejoramiento de procesos software y de manera directa o indirecta recoge la filosofía de sus antecesores. ISO SPICE, es un estándar internacional orientado a la valoración del proceso del software, esta valoración puede llevarse a cabo en organizaciones que: "planeen, administren, monitoreen, controlen y mejoren la adquisición, suministro, desarrollo operación, evolución y soporte del software"[6]

El estándar está formado por nueve partes: 1.) conceptos y guía introductoria, 2.) un modelo para la administración de procesos, 3.) procesos de medición, 4.)guía para conducir la valoración, 5.)construcción, selección y uso de instrumentos y herramientas de valoración, 6.)calificación y entrenamiento de asesores, 7.)guía para usar en procesos de mejora, 8.)guía para usar en la determinación de capacidad del proceso y 9.) vocabulario.

Cada proyecto de ingeniería de software puede verse como un conjunto de actividades (prácticas básicas), que ISO SPICE, las divide en cinco grupos funcionales: 1).Organización: Corresponden a todas las actividades encaminadas a brindar soporte en los procesos de software. 2).Proyecto: son las actividades se deben llevar a cabo para ejecutar y controlar un proyecto 3).Ingeniería: Corresponde a las actividades a garantizar la calidad de los procesos. 4).Soporte: Actividades encaminadas a la documentación y ofrecer ayuda a todo el proceso del software. 5).Cliente - Proveedor: controla la relación entre un cliente y el vendedor o en terminología de la norma entre comprador y vendedor.

ISO SPICE tiene varios niveles de cumplimiento para cada una de las actividades. Los niveles de cumplimiento se llaman prácticas genéricas, las cuales están basadas en

cinco escalas: informal, planeado y revisado, bien definido, control cuantitativo y mejoramiento continuo.

Cada práctica básica de ISO-SPICE, se aplica al modelo MCV-SHE, según el caso, teniendo en cuenta que algunas prácticas se vuelven no aplicables, sin embargo la norma provee instrumentos para remplazarlas, pero debe ser de obligatorio cumplimiento las prácticas genéricas.

Tabla 2. Prácticas básicas aplicadas a MCV-SHE

Grupo Funcional	Prácticas Básicas	Sub - prácticas básicas
CUS Cliente- Proveedor	Identificar Necesidades de Usuarios	Obtener requisitos del usuario
		Entender las expectativas del usuario
		Mantener Informado al Usuario
ENG Ingeniería	Especificación de Requisitos	Determinar requisitos de software
		Analizar requisitos software
		Determinar Impacto Operativo
	Implementación	Evaluar requisitos con el cliente.
Actualizar requisitos		
Pruebas	Construcción de Unidades de Software, procedimientos de verificación.	
	Integración	Pruebas a unidades de software, pruebas conjuntas
PRO Proyecto	Planificación	Estrategias de adición de unidades de software.
		Obtener recursos
		Monitorear Progreso
		Ejecutar revisiones de manejo
SUP Soporte	Desarrollar Documentación	Ejecutar revisiones técnicas
		Manejar acuerdos
		Determinar requisitos de documentación
		Desarrollar Documentación
		Verificación Documentación
ORG Organiza-ción	Definir y mejorar el proceso	Distribuir Documentación
		Mantenimiento de Documentación
		Definir el Proceso
		Mejorar el proceso

C. Modelos pedagógicos

Los modelos pedagógicos más relevantes son: conductistas, cognitivistas y constructivistas, en cada uno de ellos subyacen un conjunto de importantes teorías, que ofrecen elementos que pueden aplicarse a la construcción de software con fines educativos, de tal forma que cada uno de estos enfoques permiten disímiles tipos de diseños, la cual orientan la estructuración del contenido y los tipos de interacciones del aprendiz con el software, entre otros aspectos destacables.

Los sistemas hipertexto e hipermedia desde la perspectiva constructivista permiten cierta libertad al aprendiz a la hora de construir sus propios aprendizajes. El modelo constructivista, apoyado en los trabajos de Spiro[21] y Feltoovich[8], son las principales teorías que sustentan el modelo MCV-SHE, desde el punto de vista pedagógico.

CONFORMIDAD DEL MCV-SHE CON ISO SPICE

A. Modelo de ciclo de vida SHE.

El modelo que se propone en este artículo está estructurado en cuatro fases[12]: análisis y especificación de requisitos, diseño, desarrollo y soporte. Cada fase tiene asociados procesos, y éstos a su vez, se relacionan con un conjunto de actividades [12]. En paralelo a las tres fases hay dos procesos que se aplican a todo el modelo: el aseguramiento de calidad y la documentación.

utilización de otros medios (videos educativos) que facilitan el proceso de enseñanza-aprendizaje.

La identificación de necesidades educativas se logran a través de la clasificación que plantea Burton[2] y Merrill[2]: necesidades normativas, necesidades sentidas, necesidades expresadas o demandas, necesidad comparativa y necesidad futura o anticipada. Toda necesidad se convierte un elemento de suma importancia para identificar la problemática educativa que se quiere resolver con un SHE. Todas las necesidades se constituyen en fuentes de datos o información que debe ser organizada y tabulada para conocer las verdaderas dimensiones del problema a resolver. En esta fase es útil diseñar y realizar encuestas a profesores, estudiantes, coordinadores académicos, con las cuales se reconozcan las debilidades susceptibles de fabricar con un SHE.

Otro factor que toma especial importancia en el contexto de la problemática educativa, es el relacionado con el aspecto socio cultural (etnias, costumbres, dialectos, marginalidad, entre otros) de la población objetivo, debido

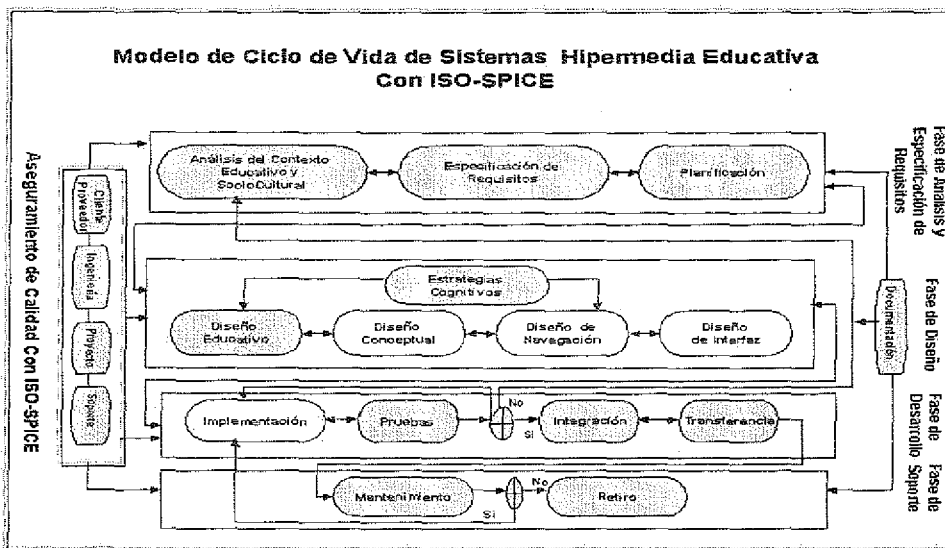


Figura 1. Modelo MCV-SHE en conformidad con ISO - SPICE

B. Análisis y especificación de requisitos

Esta fase consta de los siguientes procesos: análisis del contexto educativo y socio cultural, especificación de requisitos y planificación.

El análisis del contexto educativo está formado por las siguientes actividades: necesidades educativas, estudio de factibilidad y perfiles de usuario.

Indiscutiblemente, el desarrollo de un software con fines educativos debe corresponder a subsanar necesidades educativas que no se resuelven con la

a que es posible incluir mecanismos adicionales que desarrollen una comunicación fluida con la que se posibilite la identificación de las necesidades educativas.

Las necesidades educativas identificadas se corresponden con el dominio del problema a resolver. Hecho que da base para realizar la actividad de factibilidad, para conocer si es factible que la problemática educativa pueda ser resuelta con la utilización de un software educativo; es conveniente, saber qué tipo de software educativo es requerido: tutorial, ejercitación y práctica, simulación, hipertexto e hipermedia educativa.

Si es factible que la problemática educativa pueda solucionarse con la construcción de un sistema hipermedia educativa, es importante tener en cuenta que este tipo de sistemas son para los constructivistas[8][22], los más adecuados para los aprendizajes complejos o avanzados, debido a que los aprendices pueden construir sus propios conocimientos en función de sus necesidades e intereses, ayuda a la estructura cognitiva y posibilita la adquisición de un conocimiento asociativo.

Cabero[4], en las investigaciones sobre la hipermedia aplicada a la educación, argumenta que: los sistemas hipermedia ofrecen la posibilidad de que el aprendiz se convierta en procesador activo y constructor de su conocimiento, rompe con la concepción bancaria de la educación, donde la totalidad se encontraba en el profesor, permite la creación de un entorno más rico desde una perspectiva semiológica, donde los aprendices podrán comprender e interactuar con la información de la forma más adecuada a sus intereses.

Otro aspecto de suma importancia es la factibilidad técnica y financiera del proyecto SHE, debido a la complejidad de desarrollo según sea el tipo de software a fabricar.

Conocidas las necesidades educativas de la población y la magnitud del proyecto a través del análisis de factibilidad, la actividad a desarrollar es conocer el perfil de los usuarios o aprendices potenciales del SHE a construir. La Figura 2 resume las actividades del proceso análisis del contexto educativo y socio cultural.

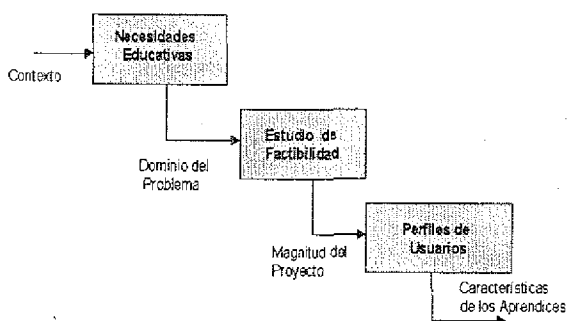


Figura 2. Actividades del contexto educativo y socio cultural

Si se conoce el perfil de los usuarios potenciales del software a construir se tendrá asegurado gran parte del éxito del proyecto. Para identificar el perfil de los usuarios se tiene en cuenta los estilos de aprendizaje de Richard Felder[7], y también se caracterizan a los aprendices desde la perspectiva de: "edad, desarrollo cognoscitivo (nivel de madurez), capacidad intelectual, conocimientos previos, actitudes, hábitos de estudio, discapacidades o deficiencias, nivel de conocimiento, intereses y nivel de experiencia"[17].

Identificado el perfil de los usuarios, se procede a determinar la funcionalidad del sistema hipermedia educativa a través de la utilización de casos de usos de UML, la cual permite establecer las diferentes interacciones entre los aprendices y demás actores con el SHE.

El siguiente proceso, después de haber determinado la funcionalidad del sistema, es lograr una adecuada planificación del proyecto (Ver Figura 3) estableciendo los costos, estimando los recursos y controlando las actividades.

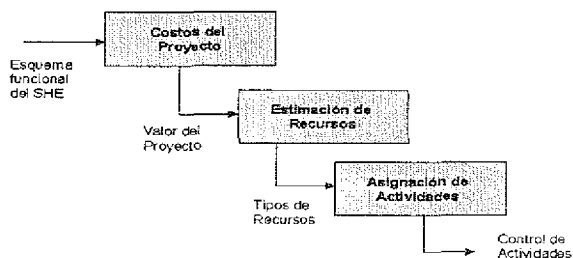


Figura 3. Actividades del proceso de planificación

C. Diseño

La fase de diseño está estructurado en cinco procesos: diseño educativo, diseño conceptual, diseño de navegación, diseño de interfaz abstracta y estrategias cognitivas.

El propósito de una representación del conocimiento en SHE es, en primer lugar, proporcionar una descripción del contenido desde el punto de vista educativo. Esta representación, basada en elementos de instrucción, constituye un nivel de abstracción que permite gestionar con mayor flexibilidad el conocimiento. Las unidades de instrucción o unidades de aprendizaje, que deben incluir objetivos educativos, contenidos, actividades de aprendizaje, evaluación y basadas en un conjunto de secuencia de etapas. El resultado de este proceso es la estructura del conocimiento del SHE y la definición de planes y estrategias instruccionales que se aplican a cada aprendiz para que logre los objetivos educativos propuestos.

El diseño conceptual, el diseño de navegación y la interfaz abstracta de este modelo están basado en la filosofía OOHDM, no obstante, se han aplicado en el diseño de navegación e interfaz abstracta, el uso de extensiones UML planteadas por Nora Koch.

El diseño conceptual está basado en el modelado de clases utilizando UML. El diseño de navegación que plantea Nora Koch, se fundamenta en dos aspectos claves: modelado del espacio y la estructura de navegación. En el modelado

del espacio de navegación se especifican clases de navegación (los nodos en una red navegación que se visitan por el usuario durante la navegación, tienen el mismo nombre asignado en el modelo conceptual, aunque se define con el estereotipo UML <clase de navegación> en el diseño de navegación). El modelo de la estructura de navegación, describe como desarrollar de manera estructurada la navegación a través de la utilización de elementos de accesos tales como: índices, tures guiados, consultas y menús. El resultado del diseño de navegación es la estructura de acceso y los contextos de navegación del SHE.

El diseño de interfaz abstracta se basa en tres aspectos claves: la primera hace referencia a los principios básicos que deben guiar una interfaz hombre-máquina, tal como los que propone Schneiderman[21], uso didáctico de imágenes, sonido, video, texto, animaciones y diseño de interfaz abstracta en sí, donde se definen los elementos del modelo de presentación.

Los elementos del diseño de interfaz abstracta que definen Nora Koch para la presentación está basada en marcos ("frames"), porque ofrece mayor claridad para presentar la navegación de los SHE, debido a que la presentación se divide en dos grandes partes una que provee la presentación de la navegación, mostrando los patrones de navegación de los usuarios y por otra parte mostrando el contenido correspondiente.

Además de los marcos, Nora Koch, utiliza extensiones de UML, para la interfaz de SHE y define un conjunto de estereotipos para identificar cada elemento, entre los que se destacan: clases de presentación, textos, botones, imagen, sonido y video entre otros. El resultado de este proceso es el modelo de presentación del SHE.

Paralelo al diseño educativo, diseño de navegación y diseño de interfaz, se desarrollan un conjunto de estrategias definidas por la ciencia cognitiva y apoyado en los trabajos de Thuring[23], donde la comprensión de un documento por parte de del usuario, se caracteriza por la construcción de un modelo mental el cual representa a los objetos y a sus relaciones estructurales cognitivas, lo que en consecuencia significa que si queremos incrementar la legibilidad de un documento debemos apoyar al lector en la construcción de su modelo mental, fortaleciendo aquellos factores que soportan el proceso y debilitando aquellos que la impiden.

Las estrategias cognitivas son instrumentos que permiten: incrementar la coherencia a nivel local y global de un hiperdocumento, ayudan a reducir la demora cognitiva del lector y por ende a mejorar la orientación y facilitar la navegación.

D. Desarrollo

La fase de desarrollo del modelo de SHE consta de cinco procesos: implementación, pruebas, integración y transferencia

El proceso de implementación consta de las siguientes actividades: selección de herramientas de programación, estrategia de prototipado, uso de patrones, desarrollo de unidades de software y procedimientos de verificación de unidades de software de SHE. La selección de herramientas de programación se constituye en un conjunto de elementos o criterios para una adecuada selección de herramientas de programación, sin que ésta sacrifique el diseño conceptual. La estrategia de prototipado, está formada por un conjunto de criterios basados en los principios que propone Goldeberg[10] y Rubin[10] para la construcción de prototipos.

Se plantea el uso de patrones, específicamente el MVC (Model View Controller) utilizando la arquitectura J2EE (Java 2 Enterprise Edition)[3], la cual permite representar el modelo que es donde reside la lógica del negocio y se implementa a través de JavaBean, las vistas, son generadas a través de páginas JSP (Java Server Application)[3], y el Controlador, que es un servlet o colección de servlets.

El desarrollo de unidades de software SHE hace referencia a la codificación en la herramienta de programación seleccionada de acuerdo a criterios previamente definidos. Cada unidad de software independiente de la plataforma de desarrollo debe incluir: código, estructura de los datos y base de datos, sumado a esto debe tener: documentación y compilación en el lenguaje seleccionado.

Para cada unidad de software de sistemas hipermedia educativa desarrollada debe documentarse a través de procedimientos para verificar que satisface los requisitos de diseños previamente definidos.

Las pruebas son en parte responsables de revisar la calidad del SHE que está en desarrollo y se constituyen en un poderoso instrumento que nos permiten conocer el cumplimiento de los requisitos especificados y si estos fueron implementados de manera adecuada, de igual forma permiten conocer las debilidades a nivel de unidades de software que están en desarrollo.

Las pruebas tienen asociadas dos actividades claves: verificación y la validación. La verificación consta de la siguientes sub-actividades: plan de pruebas del software, pruebas de unidades de unidades de software y pruebas conjuntas). La validación lleva asociada el plan de pruebas a la población objetivo, la cual consiste en definir y establecer cuando, donde, quienes y a que parte

representativa de la población interesada en el SHE, se van realizar las pruebas con el fin de determinar si los resultados corresponden a lo que la población requería.

El propósito de integrar y probar el proceso del software es integrar las unidades del software entre sí, que debe satisfacer los requisitos del software. Este proceso es cumplido a través de la agregación de unidades de software debidamente probadas de manera individual y de forma conjunta e integrando los resultados para asegurar satisfacer los requisitos del sistema hipermedia educativa.

La fase de desarrollo culmina con el proceso de transferencia, que se encarga de establecer los lineamientos para la entrega del software hipermedia al cliente.

E. Soporte

La fase de soporte consta de dos procesos: mantenimiento y retiro, el primero consiste en definir un plan que permite el establecimiento de las modificaciones a realizar al software después de un tiempo considerable de vida.

El proceso de retiro hace referencia a un conjunto de actividades que permiten determinar cuándo y bajo qué condiciones el SHE debe retirarse del uso, debido a que son demasiadas las modificaciones a realizar que se considera pertinente la construcción de otro software.

F. Aplicación de ISO SPICE a SHE.

ISO SPICE en la fase de análisis y especificación de requisitos de SHE

Definidas las prácticas básicas del modelo y las del estándar ISO SPICE aplicadas al modelo MCV-SHE (Ver Tabla 2), se procede a la aplicación de la norma a la fase de análisis y especificación de requisitos, tal como se muestra en la siguiente tabla.

Para cada proyecto SHE que se emprenda, tal como se muestra en la tabla anterior se relacionan las practicas básicas con las practicas genéricas o niveles de cumplimiento, basada en cinco escalas, informal, que se refiere a cuando cualquier actividad solamente se ejecuta o ejecuta de manera informal, su desempeño depende del conocimiento y esfuerzo individual; en este nivel los miembros del proyecto reconocen que una acción debe realizarse y hay acuerdo general para realizarla cuando sea requerida, además hay productos de trabajo identificables.

El segundo nivel indica que la actividad se está llevando a cabo mediante un proceso debidamente planeado y controlado y se verifica el desempeño según los procedimientos especificados y los productos de trabajo se ajustan a las normas y requisitos especificados.

Tabla 3. Prácticas básicas relacionadas con las prácticas genéricas en Análisis del Contexto Educativo.

ENG- Ingeniería Análisis del Contexto Educativo y Socio Cultural	Necesidades educativas				Informal
	Estudio de Factibilidad	P			Planeado y Revisado
	Perfiles de usuarios		F		Bien Definido
					Control Cuantitativo
					Mejoramiento Continuo

El tercer nivel hace referencia a una definición total y completa de la actividad, es decir la actividad obedece a la definición de un proceso estándar; el nivel cuatro indica se definen metas concretas de calidad y se debe tener un control cuantitativo del logro de dichas metas, el quinto nivel indica que la actividad está incluida en un proceso de mejoramiento continuo, para la cual es útil la utilización de varias actividades entre la que se destacan el análisis causal de defectos y la aplicación de modelos estadísticos, entre otras.

Para el logro de cada nivel se han elaborado un conjunto de indicadores, junto a una lista de verificación que plantea la norma para establecer el cumplimiento de cada práctica genérica, la valoración para cada nivel tal como se muestra en la tabla anterior se da teniendo en cuenta cuatro puntos, como lo plantea la norma: Not Adequate (N), Partially (P), Largely (L) y Fully (F), la primera valoración indica que el cumplimiento es inadecuado y no contribuye a la satisfacción del proceso, en el segundo que se cumple de manera parcial y contribuye en pequeña parte a la satisfacción del proceso, en el tercero indica que la actividad se ha implementado de manera amplia y contribuye a la satisfacción del propósito del proceso; la cuarta valoración expresa que la actividad es implementada completamente y contribuye ampliamente a la satisfacción del proceso.

Es importante resaltar que para el proceso de planificación se han definido un conjunto de prácticas básicas, además de las que indica la norma (ver Tabla 2) que deben relacionarse con las prácticas genéricas, tal como se muestra en la siguiente figura. La aplicación de ISO-SPICE, al proceso de especificación de requisitos, se realiza con la prácticas básicas de la norma (ver Tabla 2).

Tabla 4. Prácticas básicas relacionadas con las prácticas genéricas en proceso de planificación Informal

PRO- Proyecto	Planificación Costos del Proyecto	Costos del Proyecto	Informal			
			Planeado y Revisado	Bien Definido	Control Cuantitativo	Mejoramiento Continuo
		Estimación de Recursos	P			
		Asignación de Actividades	P			

ISO SPICE en la fase de diseño de sistemas hipermedia educativa

La tabla siguiente (ver Tabla No. 5) relaciona las prácticas básicas con las prácticas genéricas para la fase de diseño.

Tabla 5. Aplicación de ISO SPICE a la fase de diseño

ENG- Ingeniería	Desarrollar diseño del software	Diseño Educativo	Informal			
			Planeado y Revisado	Bien Definido	Control Cuantitativo	Mejoramiento Continuo
		Diseño Conceptual	F	L	P	N
		Diseño de Navegación	F	F	L	N
		Diseño de Interfaz Abstracta	P	L	L	N

Esta prácticas son aplicadas al grupo funcional ingeniería, de igual manera se realiza para cada una de las actividades de cada proceso y los demás grupos funcionales.

ISO SPICE en la fase de desarrollo

La aplicación de ISO SPICE a la fase de desarrollo de sistemas hipermedia educativa, se realiza para cada proceso, es decir para el proceso de: implementación,

pruebas e integración; y para el conjunto de actividades de cada proceso asociado al grupo funcional correspondiente.

Tabla 6. Aplicación de ISO SPICE a la fase de desarrollo de SHE

ENG- Ingeniería	Implementación de SHE	Pruebas en SHE	Integración de SHE	Mantenimiento de SHE	Retiro de SHE	Informal			
						Planeado y Revisado	Bien Definido	Control Cuantitativo	Mejoramiento Continuo
						L	P	P	P
						F	F	L	P
						P	P	P	L
						L	F	F	N
						N	N	N	N

El conjunto de valoraciones de todas las actividades, reflejan el estado y el mejoramiento de cada proceso. Igual que en las fases anteriores se aplican indicadores que muestran el logro alcanzado por cada actividad de manera que contribuyan a la satisfacción del propósito del proceso.

ISO SPICE en el proceso de documentación de SHE

El proceso de documentación de sistemas hipermedia educativa está compuesta por cinco actividades que deben aplicarse en cada uno de los procesos y las fases del modelo, es decir, la aplicación del proceso de documentación tal como lo muestra la Tabla 2 se aplica a todo el modelo de ciclo de vida.

Las actividades del proceso de documentación son: determinación de requerimiento de documentación SHE, desarrollar documentación de SHE, verificar documentación SHE y mantenimiento de documentación de SHE.

La determinación de requisitos de documentación, Identifique los requisitos para el documento a ser construido, incluyendo: título, público ó tipos de usuarios, propósito, objetivos a ser logrado, guía del contenido y medios de comunicación y los requisitos de la distribución. Para el diseño y desarrollo de documentación hipermedia es útil tener en cuenta, entre otros los siguientes aspectos: formato, descripción de contenido, numeración de páginas, colocación de figuras y tablas. A cada documento diseñado debe revisarse su contenido técnico y el estilo de presentación asociado a los mismos.

- La verificación consiste en la revisión de la documentación contra sus requisitos.
- La distribución de documentación de sistemas hipermedia educativa hace referencia a los diferentes medios de distribución de documentación, tal como: medios de comunicación electrónicos y papel.
- El mantenimiento de documentación de SHE es de vital importancia para realizar modificaciones a la documentación de un proceso o producto, para cual se sugiere un plan de mantenimiento que permita una organización adecuada de las modificaciones pertinente de la documentación del sistema hipermedia educativa construido.

La aplicación al proceso de documentación se realiza a cada actividad (ver Tabla 2), de tal manera que se reitera que la aplicación debe realizarse a cada actividad, se realizó en la fase de diseño y desarrollo solamente a los procesos para que se observara como es posible obtener diferentes valoraciones en los distintos niveles, pero es importante anotar, que no se puede llegar al siguiente nivel si no se han cumplido los indicadores de logros y las actividades que propone la norma.

VERIFICACIÓN DEL MODELO MCV-SHE

Con el fin de hacer visible la comprobación del modelo MCV-SHE, se ha desarrollado un sistema hipermedia educativa que apoya a los investigadores aprendices (SHEILA) en el proceso de su formación. SHEILA fue desarrollado bajo los lineamientos del modelo MCV-SHE, es decir cumpliendo las fases, procesos, actividades y valorando el proceso del software con ISO SPICE.

ESTRATEGIAS DE VALIDACIÓN DEL MODELO

Para realizar una validación inicial del modelo se construyó un SHE llamado SHEILA. SHEILA busca apoyar el proceso entrenamiento de nuevos investigadores en el CIDLIS a través del planteamiento del desarrollo de las principales secciones de un plan de proyecto, junto con los materiales educativos de apoyo, sistema de evaluación; todo esto orientado por la experiencia del centro de investigación y la serie Aprender a Investigar del ICFES. La validación del modelo MCV-SHE, se continuará realizando en el programa de Informática Educativa de la Universidad de Córdoba, la cual un conjunto de profesores del área de desarrollo de software educativo, lo utilizarán para la construcción de sistemas hipermedia educativa, con la participación activa de estudiantes.

El grupo de investigación GUIA (Gnosis Unificada en Ingeniería del Aprendizaje), a través de la aplicación de procesos y actividades de MCV-SHE, construirá tres SHE en la UIS que apoyen a los docentes del CIDLIS en el proceso de enseñanza de sus cursos.

CONCLUSIONES

El modelo MCV-SHE permite:

- 1) Apoyar de manera significativa al mejoramiento continuo del proceso del software hipermedia educativa a través de la interpretación, adaptación y aplicación del estándar ISO SPICE.
- 2) Facilitar el proceso de desarrollo de sistemas hipermedia educativa
- 3) Contribuir a que el desarrollo de software con fines educativo se realice a través de la aplicación de métodos, procesos y técnicas de la ingeniería del software. Actualmente apoyando el proceso de formación de investigadores en el CIDLIS.
- 4) Contribuir a la aplicación de las teorías del enfoque constructivista para el mejoramiento de procesos de enseñanza - aprendizaje.

TRABAJOS FUTURO

Incluir en el diseño de interfaz abstracta del Modelo MCV-SHE, diagramas de estado de UML, para mejorar el comportamiento de los objetos y desarrollar un conjunto de estrategias para su adecuado uso.

AGRADECIMIENTOS

Los autores expresan sus agradecimientos al grupo de investigación GUIA (Gnosis Unificada en Ingeniería del Aprendizaje) de la Universidad Industrial de Santander, por sus valiosos aportes y a la Universidad de Córdoba por patrocinar esta investigación.

REFERENCIAS

- [1] Bieber M. Galnares R. Web engineering and flexible hypermedia. In Proceedings of the Second Workshop on Adaptive Hypertext and Hypermedia, Hypertext '98, pp.252. 1998.
- [2] Burton, J.K. and Merrill, P.F. Needs Assessment: Goals Needs and priorities, Briggs, Chapter 2, p. 17-43. In

- Briggs, Gustafson, K. & Tillman (Eds.) *Instructional Design: Principles & Applications* (2nd. Ed.). 1977.
- [3] Brown, S, Robert B, Jayson. F. *Professional JSP*. Wrox Press, pp. 900. 2001.
- [4] Cabero, J. "Evaluación de medios y materiales de enseñanza en soporte multimedia". *Pixel-Bit. Revista de Medios y Educación*, 13, pp.23-45. Madrid. 1999.
- [5] De Troyer, O. and Leune, C. *WSDM: A User-Centered Design Method for Web Sites*, In: *Computer Networks and ISDN Sytems*, Proceedings of the 7th International World Wide Web Conference, Elsevier, pp. 85-94. 1998.
- [6] Eman, K., Drouin, J.N. *SPICE, Theory and Practice of software process Improvement and Capability Determination*. IEEE Computer Society. Wiley-IEEE Press. 502 pp. 1998.
- [7] Felder, R. *Meet Your Students: 1. Stan and Nathan*. *Chemical Engineering Education*, Spring, p. 68. number 46. 1989.
- [8] Feltovich, P.J, Spiro, R.J., Jacobson, M.J., & Coulson, R.L. Knowledge representation, content specification, and the development of skill in situation-specific knowledge assembly: Some constructivist issues as they relate to cognitive flexibility theory and hypertext. *Educational Technology*, pp. 22-25. 1991.
- [9] Garzotto, P. Paolini, Schwabe, D. "HDM, A model-based approach to hypertext application design", *F., ACM Transactions on Information Systems*, vol. 11, no. 1, p. 1-26. 1993.
- [10] Goldberg A., Rubin K. "Succeeding with Objects: Decisión Frameworks for projects management", Addison-Wesley Pub Co; 1st edition. 542 pages. 1995.
- [11] Grady B, James R, Ivar J. *El Lenguaje Unificado de Modelado*. Addison Wesley. pp.526. Madrid, 2000.
- [12] Humphrey, W.S., Kellner, M.I. "Software Process Modelling : Principles of Entity Process Models", *Proceed. of the 11th Int. Conference on software Engineering (ICSE)*, IEEE Comp. Society Press, pp. 331-342. 1989.
- [13] Isakowitz , T, Stohr, Edward y Balasurbramanian, P. "RMM: A methodology for structured hypermedia design", 38(3). pp. 34-44, 1995.
- [14] Koch, N. *A comparative Study of Methods for Hypermedia Development*, Technical Report 9905, Ludwig-Maximilians-Universität München, Institute of Computer Science. Pp.19. November 1999.
- [15] Lange, D. "An Object-Oriented Design Approach for Developing Hypermedia Information Systems", *Journal of Organizational Computing and Electronic Commerce*, v.6 n.3, p.269-293, 1996.
- [16] Lee H., Lee C. and Yoo C. A scenario-based object-oriented methodology for developing hypermedia information systems. In *Proceedings of 31st Annual Conference on Systems Science*, Eds. -Volume 2. p. 47. 1998.
- [17] Marqués, P. *Software Educativo: Guía de Uso y Metodología de Diseño*, Imprenta Bartola, Barcelona: Estel. pp. 225. 1995.
- [18] Pressman, R. *Ingeniería del Software: Un Enfoque practico*, Editorial Mc Graw Hill. pp 400. 2002. Madrid..
- [19] Rossi, G, Rossi, G. *The Object-Oriented Hypermedia Design Model*. *Communications of the ACM*, vol 38, num 8, pp 45-46. 1995.
- [20] Schach, S. *Classical and Object Oriented Software Engineering*, Fourth Edition, WCB/Mcgraw Hill, EE.UU. pp 616. 1999.
- [21] Schneiderman, B. *Designing the user interface: Strategies for Efecctive Human-Computer Interaction*, Third Edition, Addison-Wesley. pp. 639. 1997.
- [22] Spiro, R.J., Jehng, J.Ch. *Cognitive Flexibility and Hypertext: Theory and Technology for the Nonlinear and Multidimensional Transversal of Complex Matter*, Lawrence Erlbaum Associates, Hillsdale: New Jersey. D. Nix & R. Spiro (eds.), *Cognition, Education, and Multimedia*. p 30-40. 1990.
- [23] Thuring, M.; Hannemann, J., Haaake, J. "Hypermedia and Cognition: Designing for Comprehension", *Communications of the ACM*. Vol. 38. Num 8. Pp 57-66. 1995.