

DISEÑO DE SOFTWARE EDUCATIVO BASADO EN COMPETENCIAS

DESIGN OF EDUCATIONAL SOFTWARE BASED ON SKILLS

Manuel Fernando, Caro Piñeres.

*Coordinador Grupo de Investigación Edupmedia. Departamento de Informática.
Universidad de Córdoba. Colombia. E-mail: loshigos@yahoo.es*

Raúl Emiro, Toscazo Miranda.

*Coordinador de Desarrollo. Grupo de Investigación Edupmedia. Departamento de Informática.
Universidad de Córdoba. Colombia. E-mail: rtscmrn@yahoo.es*

Filadelfia María, Hernández Rozo.

*Joven Investigadora. Grupo de Investigación Edupmedia. Departamento de Informática.
Universidad de Córdoba. Colombia. E-mail: fimajisher@yahoo.es*

María Elena, David Lobo.

*Joven Investigadora. Grupo de Investigación Edupmedia. Departamento de Informática.
Universidad de Córdoba. Colombia. E-mail: mariaedavid@gmail.com*

Fecha de recepción: 20 de febrero de 2009

Fecha de aprobación: 11 de junio de 2009.

RESUMEN

El presente artículo describe un modelo de diseño de software educativo basado en competencias, el cual presenta una visión integral del desarrollo de estas aplicaciones mediante la combinación de componentes pedagógicos, didácticos, multimediales y de ingeniería de software. El modelo sugerido se compone de cinco fases que detallan paso a paso los aspectos que se deben tener en cuenta para la creación de software educativo. La fase inicial constituye la descripción del *diseño educativo*, en la cual se analiza la necesidad educativa, se plantean los objetivos de aprendizaje y se describen las competencias que se pretenden desarrollar con la aplicación; del diseño de éstas resultan las siguientes subfases: diseño de contenidos, diseño pedagógico y diseño de aprendizaje. Las competencias son el aspecto fundamental que abarca el desarrollo de este modelo, las cuales son primordiales para la realización de las fases a seguir conformadas por el diseño computacional y el diseño multimedial, estas se encargan del análisis y modelado del software, y del sistema de comunicación hombre-máquina. En la fase de *producción* se ensamblan los componentes elaborados o recolectados, según el caso. La última fase es la de *aplicación*, donde se hacen las pruebas de rigor para evaluar el desempeño del software en los contextos para los que fue desarrollado.

Palabras clave: Software educativo, ingeniería de software, educación, informática, competencias.

ABSTRACT

This article describes a designed model for an educative software based on competences, which presents a vision of developing these applications by combining teaching, learning, pedagogic, didactic and multimedia and software engineering components. The suggested model consists of five phases that describes step by step the process to be taken into account for the creation of an educative software. The initial phase, is the description of an educative design, which analyzes the educational needs, here are presented the learning objectives and outlines the competences that seek to develop the application. The design of these sub-phases are the following: content design, pedagogic design and learning design. The competences are the key point that involve the development of this model, which are essential for carrying out the steps to follow up the computational design and multimedia design, these are essential for the analysis and modeling software, and for the system man-machine communication. In the production stage the produced components are assembled or collected, according to the case. The last stage of the model is the implementation stage, where testings are done too rigorously assessing the performance of the software, in the contexts for which it was designed.

Keywords: Educational software, software engineering, education, computing, skills.

1. INTRODUCCIÓN

Como concepto de trabajo, desde la década del 90 el tema de competencias se ha convertido en una corriente de pensamiento dirigida al mejoramiento de la calidad de la educación, luego que el MEN (Ministerio de Educación Nacional) con el sistema Nacional de evaluación de la calidad (SABER) lo estableciera con el decreto 1860 de 1994 y el decreto 230 de 2002, especialmente para el desarrollo de propuestas educativas y pedagógicas [15], muchas de las cuales tienen la tecnología como mediación.

Gracias a los adelantos que la modernidad y la innovación tecnológica han traído a todos los campos, los software educativos han tomado gran importancia en los procesos de enseñanza, siendo herramientas que aportan tanto al docente como al estudiante espacios dinámicos de aprendizaje; por lo cual, y haciendo eco a lo mencionado anteriormente, es necesario que el diseño de los mismos tengan como eje central de su *diseño educativo* el sistema de competencias que se vaya a desarrollar, procurando hacer énfasis en capacitar a los alumnos para enfrentar los retos que se le puedan presentar en la sociedad actual, contribuyendo desde estas aplicaciones a la formación de personas con pensamiento crítico y analítico.

No obstante lo anterior, en general los modelos de desarrollo de software educativos no están contextualizados a la realidad de la educación en Colombia, al no contener directrices claras acerca del manejo de competencias en su diseño, dándole más importancia a la ingeniería de software o al diseño multimedial. El modelo que Edupmedia

presenta, prioriza el desarrollo de competencias, sin dejar de lado la parte estructural. El modelo de diseño de software educativo desarrollado por el grupo de Investigación Edupmedia, surge de la sistematización de la experiencia que los docentes y jóvenes investigadores del grupo han tenido durante tres años en el desarrollo de software adaptado a la realidad educativa y basado en el desarrollo de competencias.

2. ESTRUCTURA DEL MODELO

A continuación se describen los aspectos que hacen especialmente particular este modelo que presenta el grupo de investigación Edupmedia para el diseño de software educativo, cuyas características frente a los que en la actualidad se utilizan, brindan ventajas para su implementación en el contexto colombiano.

2.1. COMPETENCIAS COMO EJE CENTRAL DEL MODELO EDUPMEDIA.

Siendo las competencias el eje central del modelo, es necesario indicar las razones que lo justifican, con lo cual se brinda el sustento básico de su diferenciación frente a los modelos existentes así:

1. Los modelos utilizados en la actualidad no presentan una sección o apartado que brinde toda la información relacionada con las competencias requeridas según las necesidades educativas detectadas en la investigación previa, como sí lo contempla el modelo Edupmedia, pues en éste no sólo se enuncian las —competencias, sino que también se establece un formato de presentación (por objetivos y normas) que agrupa todos los componentes de las misma, lo que permite al diseñador recoger los aspectos necesarios para realizar una documentación completa, lo cual guiará con mayor certeza el proceso de desarrollo.
2. El diseño del sistema de competencias actúa como eje integrador del proceso de diseño y producción del software educativo, pues cada una de sus secciones brinda información de entrada para otros procesos del desarrollo, convirtiéndose así en el centro de partida y principal fuente de información para las fases siguientes. Así pues, la sección de *Recursos* dentro de la Norma da nociones precisas acerca de las herramientas complementarias que requiere el estudiante para realizar las actividades propuestas, las cuales debe proveer el software educativo.
 - La sección de los *Elementos* brinda las bases de las actividades que el usuario debe desarrollar con el software, por lo cual el diseñador debe tomar la decisión de hacer una ventana para cada *elemento* o agrupar varios elementos en una ventana; cabe destacar que a mayor cantidad de elementos por ventana, ésta se torna más compleja.
 - El listado de conceptos que se obtienen de las competencias y que se consignan en la sección de los *Conceptos* da origen al *diseño de contenidos*, en el cual se determina el nivel de profundidad y el tipo de vocabulario que se utilizará en el software.

- La sección del *Contexto* brinda al diseñador información que le permite determinar las características del ambiente que debe rodear las actividades de aprendizaje que se van a realizar con el software. Éstas pueden ser de tipo físico (laboratorios u otros lugares), o de tiempo (actualidad, siglo XIX).
- La sección de los *Indicadores* servirá para el diseño del sistema de evaluación del software educativo, pues en ellos se evidencian las habilidades y conocimientos que debe demostrar una persona para la realización de una tarea (elemento) en particular dentro de la competencia.

Con lo dicho anteriormente se evidencia que el eje central de todo el modelo son las competencias. Por otra parte, desde lo pedagógico se puede aducir que el modelo Edupmedia no se centra con ninguna teoría en particular, hecho que lo hace flexible y le permite amoldarse a las condiciones propias de la población objeto y a las características propias del entorno de la necesidad estudiada. También es factible que el modelo se adapte a las directrices pedagógicas propia de cada institución educativa, las cuales se encuentran consignadas en su Proyecto Educativo Institucional (PEI), así no se entraría a imponer un software que se trabaje en forma diferente a la realidad escolar.

2.2. DESCRIPCIÓN GENERAL

El modelo de desarrollo de software educativo que se plantea en este escrito tiene como base fundamental el sistema de competencias, e integra cinco fases (figura 1): *Diseño educativo, diseño multimedial, diseño computacional, producción y aplicación*. Este modelo es sencillo de aplicar y sus fases están bien documentadas, lo que lo hace idóneo para su utilización en procesos de desarrollo de software educativo.

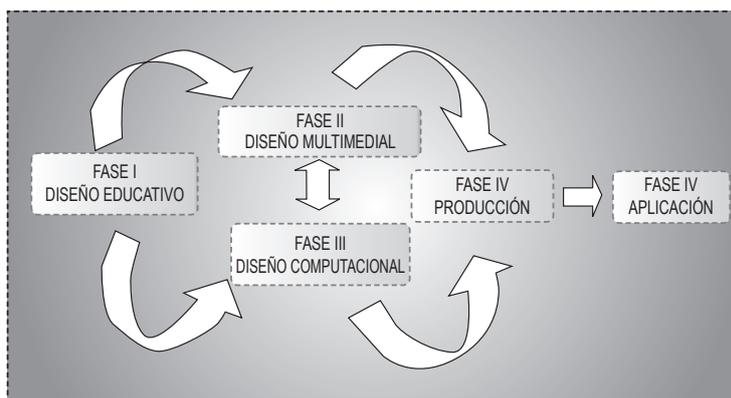


Figura 1. Modelo para el desarrollo de software educativo basado en competencias (Edupmedia-2008)

Cada fase del modelo está bien delimitada y se encarga de un trabajo en particular así: en la *Fase I* se hace el estudio de los factores educativos que sustentarán el diseño del software en cuanto a la didáctica, la pedagogía y la ética. La *Fase II*, trata acerca de los aspectos estéticos y del sistema de comunicación, según la población, sistema de conteni-

dos y estrategias planeadas en la fase anterior. En la *Fase III* se trabajan los fundamentos de ingeniería de software que requiere el desarrollo de una aplicación robusta y amigable. En la *Fase IV* se desarrollan y ensamblan los componentes del software, finalmente en la *Fase V* se utiliza el software en el aula de clases, evaluando su pertinencia e impacto. A continuación se explica cada fase en forma detallada.

2.3. FASE I: DISEÑO EDUCATIVO

Comprendiendo las necesidades que se presentan en la educación con relación al proceso de enseñanza –aprendizaje, es necesario realizar un estudio exhaustivo sobre el aprendizaje educativo que conlleve al establecimiento de un plan de trabajo en donde se especifiquen los aspectos que comprendan las distintas etapas que hacen parte de la fase del diseño educativo. Estas etapas se definen a continuación:



Figura 2. Fase I. Diseño educativo. (Eduppedia-2008)

2.3.1 Análisis de la Necesidad Educativa

Al iniciar un proceso de diseño y desarrollo de software educativo es importante tener presente los problemas que se evidencian en una población determinada para saber exactamente el tipo de necesidad educativa en la que se va a hacer énfasis, identificando las posibles causas que permitan plantear las diversas alternativas de solución a la problemática encontrada.

Entre las posibles fuentes de información podríamos contar los documentos de las instituciones educativas, folletos, boletines informativos, planes de estudio, estadísticas sobre pruebas de estado, libros de apuntes, entrevistas y cuestionarios.

2.3.2. Diseño de Fines Educativo

En el proceso de enseñanza y aprendizaje intervienen diversos factores que propenden la calidad educativa, dentro de estos factores sobresalen los fines educativos que integran los objetivos de aprendizaje, los valores y las dimensiones humanas, estos constituyen los propósitos fundamentales de la enseñanza que serán el punto de partida del desarrollo de las actividades escolares.

El diseño con fines educativos comprende el *nivel cognitivo*, el cual hace referencia al tipo de conocimiento que será trabajado por los estudiantes y se redacta normalmente en forma de objetivos, y *el nivel valorativo* hace referencia a los valores, normas y conductas que serán privilegiadas [6].

2.3.3. Diseño del Sistema de Competencias

En el contexto educativo el término competencias se extiende al conjunto de actividades que se deben realizar para enfatizar el desarrollo de potencialidades del sujeto a partir de lo que aprende en la escuela [9].

Ahora bien, la educación en Colombia se rige por un sistema de competencias en donde se educa al estudiante para que éste sea activo en la sociedad, es decir que sea capaz de enfrentar las distintas situaciones que se le presenten de una forma crítica y responsable, permitiendo que se desenvuelva en un espacio social, orientado por un pensamiento de carácter sistémico y tecnológico. A partir de esto es necesario incluir en la creación de software educativo especificaciones claras acerca de cómo organizar las competencias que se pretenden desarrollar con el software. En este caso en la primera fase del modelo, se introduce una sección para gestionar las competencias que se han identificado, luego de realizar el análisis de la necesidad educativa. Estas se estructuran en un formato que comprende dos grandes ámbitos que son los objetivos y las normas, los cuales se describen a continuación:

2.3.4. Objetivos

Esta dimensión abarca el enunciado y los elementos de la competencia.

Enunciado: Los objetivos son elementos fundamentales en todo proceso debido a que estos constituyen el derrotero a seguir y admiten a su vez la evaluación del estudio abordado. En esta sección se establecerán los objetivos de las competencias para ampliar las nociones sobre la forma de definir un ambiente de aprendizaje. Aquí se indican las finalidades que se pretenden lograr o adquirir por parte de los sujetos involucrados en el proceso.

Elementos: Para el tratamiento de las competencias se requieren una serie de actividades que dependen de cada objetivo, las cuales son enunciadas específicamente determinando lo que el estudiante debe ser capaz de realizar para cumplir el objetivo general.

2.3.5 Norma

Es la dimensión de la competencia que trata acerca del contexto, los recursos y las evidencias de conocimientos.

Contexto: Aquí se especifica el contexto donde se hace evidente la competencia, este puede ser un lugar físico o una situación determinada por un problema o reto que se debe solucionar.

Recursos: Son los recursos tangibles e intangibles que se requieren para realizar la competencia en el contexto determinado. Estos recursos se deben proveer en lo posible con el software. Así por ejemplo si se requiere un diccionario o una libreta de apuntes, el software debe proveer herramientas de diccionario y de libreta o agenda según el caso. Por tanto esta sección es el punto de partida para los elementos que debe llevar la barra de herramientas.

Indicadores: También llamados evidencias, son los conocimientos o actitudes que se deben ver o evaluar para saber si el estudiante o persona es competente. Por cada *elemento* de la competencia se pueden tener uno o más indicadores, por tanto esta sección será la base para la elaboración del sistema de evaluación del aprendizaje del software educativo.

Conceptos. *Esta sección contiene una lista detallada de los conceptos que aparecen en los apartados anteriores del formato de competencias y que se relacionan directa o indirectamente con el ámbito de la temática del software educativo. Aquí no es necesario definir o explicar cada concepto, pues más adelante con esta lista se desarrolla el diseño de contenidos donde se tratan con mayor profundidad sus significados y relaciones.*

Habilidades Requeridas. *La realización de cualquier actividad implica ciertos conocimientos, destrezas que son necesarias para lograr su cumplimiento, esto conlleva a que los estudiantes deban poseer ciertas aptitudes que se requieren para desarrollar cada una de las competencias así:*

- **Intelectuales:** Las habilidades intelectuales son aquellas que se necesitan en la realización de actividades mentales. Los test de Coeficiente intelectual (CI), los test de admisión, los test de admisión para el postgrado en negocios, están diseñados para asegurarse de la habilidad intelectual de los individuos [1]. Las habilidades intelectuales como la destreza numérica, la comprensión verbal, la velocidad perceptual, el razonamiento inductivo, el razonamiento deductivo, la visualización espacial y la memoria, son las aptitudes que deben poseer los estudiantes como requisito para desarrollar las actividades propuestas en cada competencia.
- **Físicas:** Las habilidades físicas son aquellas que se necesitan para realizar tareas que demandan vigor, destreza manual, fortaleza y características parecidas [1].

Las habilidades físicas comprenden las destrezas motoras que son fundamentales para el manejo de herramientas computacionales, que permiten a los estudiantes realizar con menor dificultad distintos tipos de actividades, en este caso los que intervienen en las competencias planteadas.

En el numeral 4.2 se presenta un ejemplo del formato de cómo se diseña una competencia (Tabla 8).

2.3.6 Diseño de Contenidos

Para la realización de un software educativo es necesario describir cada contenido que comprenda el problema estudiado, indicando la definición de cada categoría conceptual utilizada en dicho material educativo, haciendo énfasis en las características principales del conjunto de bases conceptuales.

El contenido se obtiene del listado de la sección de *conceptos* del formato de las competencias (ver tabla 1). Si existe más de una competencia, se hace un único listado con la sumatoria de todos los conceptos y se procede a elaborar sus definiciones según las características de la población y el nivel de profundidad deseado.

Tabla 1. Vista parcial de la matriz del diseño de contenidos.
(Software Lenguaje Maya - Edupmedia-2008).

Concepto	Características	Definición/Descripción
Sentarse-chum		Asumir el trono
Bailar-ak'taj		Él baila

2.3.7. Diseño Pedagógico

La pedagogía se refiere al saber o discurso sobre la educación como proceso de socialización, de adaptación. En sentido estricto, por pedagogía entendemos el saber riguroso sobre la enseñanza, que se ha venido validando y sistematizando en el siglo XX como una disciplina científica en construcción, con su campo intelectual de objetos y metodologías de investigación propios, según cada paradigma pedagógico [7].

En esta sección se especifica en primera instancia el modelo pedagógico que se ha seleccionado, recordando que el modelo Edupmedia no trabaja con un modelo estándar, sino a partir de la elección hecha por los expertos teniendo en cuenta diversos factores como las competencias a desarrollar, las necesidades educativas específicas y el contexto donde se aplicará.

El modelo pedagógico seleccionado se tomará como base para el diseño de las actividades de aprendizaje que tendrá el software educativo, esto se hace en forma detallada para tener una visión amplia del modelo en cuestión. Una vez hecho eso, se procede a especificar las características o aspectos de dicho modelo que serán materializadas en el software, pues generalmente debido a la gran variedad de aspectos que abarca un modelo en particular, no todos son viables de aplicar a determinadas necesidades educativas.

2.3.8. Diseño de Aprendizaje

Una vez realizado el *diseño pedagógico*, se inicia el diseño de una *estrategia de aprendizaje*, cuyo propósito es otorgar a los estudiantes estrategias efectivas para el mejor desempeño en áreas y contenidos específicos. La utilización de distintas formas de aprendizaje dependerá de las actividades efectuadas y de las características cognitivas de los estudiantes.

Para realizar el diseño de aprendizaje (tabla 2) se lleva a cabo el siguiente proceso: Inicialmente se toman los elementos de la competencia seleccionados y sus indicadores (trabajado en la sección *diseño de competencias*), posteriormente se especifican las características del modelo pedagógico que se verán reflejadas en el software (trabajado en la sección de *diseño pedagógico*); con esos tres datos se diseña una secuencia de aprendizaje cuya finalidad es que los estudiantes realicen las actividades descritas en los elementos bajo las directrices especificadas en las características del modelo pedagógico. La secuencia debe evidenciar los aspectos relacionados en los indicadores, si dicha secuencia cumple con los aspectos mencionados, puede ser considerada como adecuada para continuar el proceso de diseño.

2.4. FASE II: DISEÑO MULTIMEDIAL

A partir de los resultados obtenidos de la fase del diseño educativo se establece la fase número II denominada diseño multimedial (figura 3). Esta fase se encarga de la descripción de las etapas que permitirán conocer la planeación de actividades que se van a desarrollar, utilizando representaciones gráficas y recursos multimedia que en conjunto mostrarán el diseño de cada interfaz y la estructura que se va a utilizar en el software.

2.4.1. Planeación del proceso de producción

En todo proceso de desarrollo de software educativo se necesita planificar como quedará constituido u organizado el desarrollo de las actividades, esto ayuda a que durante el proceso se tenga un control que permita cumplir a cabalidad lo establecido en la planificación de la producción. El conjunto de procesos integra las metas u objetivos que se pretenden alcanzar, la distribución de las funciones que cada persona tendrá que asumir para lograr los objetivos que han sido planteados y por último un cronograma de actividades.

Tabla 2. Ejemplo resumido del diseño de la secuencia de aprendizaje basado en competencias. (Software Lenguaje Maya - Edupmedia-2008)

COMPETENCIA # 1		
Elementos	Aplicación modelo pedagogía	Indicadores
1. Identificar los verbos y la numeración del lenguaje maya	<p>Conoce los verbos más significativos de una nueva cultura.</p> <p>Los verbos escogidos son los principales de la cultura maya.</p>	1.1. Reconoce la estructura de los verbos del lenguaje maya (chum; ak'taj; tzak; chuk; jatz; puluuy; wol; uhti; pas; lok')
Secuencia de aprendizaje		
<p>Se ilustran los verbos más importantes del lenguaje maya según la representación y pronunciación en maya y la representación de estos en español. El usuario escogerá libremente que verbo desea visualizar. Para determinar el indicador 1.1 se mostrarán botones que representan los verbos y cuando el usuario da clic, se muestra la representación y pronunciación por medio de imágenes.</p>		



Figura 3. Fase II. Diseño multimedial. (Edupmedia-2008).

2.4.2. Diagrama de contenidos

La funcionalidad de los diagramas es representar gráficamente la información contenida en un estudio temático de tal forma que se hace necesario complementar el diseño de un software con un esquema que represente los contenidos que se encuentran en la fase del diseño de contenidos. Esta representación se puede esquematizar de distintas formas ya sea por medio de mentefactos, mapas conceptuales y mapas mentales.

2.4.3. Diseño del ambiente de aprendizaje

Esta etapa comprende el conjunto de directrices que permitirán una comunicación directa entre el usuario y la computadora, ya que se presentará visiblemente el diseño de cada una de las interfaces gráficas de usuario. Estas directrices se identifican secuencialmente iniciando con el diseño de la ventana estándar para seguir con cada ventana o interfaz y para finalizar con el diseño del mapa de navegación.

2.4.4. Mapa de navegación

El mapa de navegación de un software es una guía gráfica que se elabora para brindarle al usuario un fácil manejo del material computacional. Este esquema muestra la forma como están organizadas las interfaces gráficas de usuario, dentro de este esquema se incluye la imagen de la ventana principal y el despliegue de las demás cuando hay un hipervínculo que la interconecta.

2.4.5. Guión técnico

El guión técnico multimedial es un escrito que contiene detalladamente que es lo que queremos mostrar en la pantalla de nuestro computador. En el guión se escribe cómo va a ser cada “pantallazo”, eso es lo que vamos a ver cada vez que interactuemos con el entorno. Debemos escribir todo, cómo van a ser los fondos, botones, sonidos, fotografías, colores, tipo y color de letra y dónde va a estar situado todo [5].

La fase del guión técnico multimedial es muy importante, debido a que permite visualizar de una forma detallada la conformación de los diferentes componentes de las interfaces gráficas de usuario. En esta fase es donde se recolectan, describen y producen todos los recursos multimedia, los textos que se van a utilizar en el diseño de cada una de las ventanas.

2.4.6. Diseño de las interfaces

Las ventanas o interfaces gráficas del usuario (IGU), son la representación gráfica que concederá al usuario la visualización de las herramientas multimediales, los contenidos, las actividades y las evaluaciones que pueda tener el software educativo. Para diseñar las ventanas es necesario tener en cuenta el tipo de usuario a quien va dirigido el material computacional, además para su diseño y creación es importante tomar como base el tipo de software seleccionado en la *fase del diseño educativo*.

Ventana estándar: Es la ventana principal que servirá de modelo para el diseño de las demás interfaces. La ventana estándar se diseña por secciones que deben ser explicadas detalladamente para precisar la división de la ventana.

Descripción de las ventanas: Todas las ventanas se deben describir teniendo en cuenta el diseño y la organización dispuesta para su presentación, identificándolas en distintas categorías tales como: la imagen, el nombre de cada ventana, el texto, el audio, videos, animación y por último, las acciones. Estas categorías darán una mejor explicación del contenido que posee cada una de las interfaces. Dirigirse al apartado 4.5. (Ver figura 11 en adelante).

2.4.7. Índice de recursos

Los materiales multimedia deben utilizarse cuando éstos hagan alguna aportación relevante a los procesos de enseñanza y aprendizaje. Su uso eficiente siempre estará supeditado a la existencia de una necesidad educativa que razonablemente pueda satisfacer [12].

En la realización de un software educativo es vital seleccionar o producir recursos multimedia como el audio y video para lograr la interactividad del material educativo y captar la atención de los usuarios, permitiendo que estos obtengan mayor comprensión de los contenidos utilizados en el software.

2.5. FASE III: DISEÑO COMPUTACIONAL

La fase del diseño computacional comprende la descripción de los elementos que permitirán que el sistema funcione a cabalidad con los procesos que fueron definidos en el transcurso del diseño educativo y multimedial. Asimismo se hace una representación de la interacción que tiene el usuario con el sistema en desarrollo, permitiendo que los actores detallen si el software complementa y ayuda al proceso de enseñanza y aprendizaje.

Tabla 3. Ejemplo de Índice de recursos para una ventana.
(Software Lenguaje Maya - Edupmedia-2008)

Ventana Nro. 0014 – Construcción de frases simples					
Texto	Construye la frase que aparece en el recuadro				
Imagen	Glifos agrupados por categorías Sustantivos				
	img027.jpg	img028.jpg	img029.jpg	img030.jpg	img031.jpg
					
Audio	No aplica				
Video	Video de ayuda para ubicar los glifos en los bloques Video0010.avi				
Acciones	Barra de Glifos: Contará con botones de desplazamiento para colocar los glifos en el área de visualización. Se debe arrastrar cada Glifo desde la Barra hasta el área de trabajo y ubicarlo en un cuadro.				



Figura 4. Fase III. Diseño computacional. (Edupmedia-2008)

Descripción del proceso de desarrollo. En el proceso de desarrollo se ponen de manifiesto las etapas que harán parte del diseño del software, tales como la selección de una metodología o ciclo de vida, la cual se incluirá en el material computacional, así como las descripciones funcionales del sistema. El ciclo de vida permite organizar los procesos de tal forma que las actividades puedan ser definidas de una manera más ordenada y con calidad.

Análisis de requerimientos. Al planificar el diseño de un software educativo se debe tener presente la planta física y la población a quien va dirigida refiriendo a docentes, estudiantes y usuarios; es importante establecer los requerimientos a partir de la necesidad que se presentó en el proceso de enseñanza y aprendizaje, de esta forma se conocerá si el software cumplió a cabalidad con los objetivos dispuestos durante el proceso de diseño. Para desarrollar los requerimientos (tabla 4) se deben dividir éstos en secciones denominadas subsistemas y requerimientos funcionales.

Tabla 4. Ejemplo de un requerimiento formalizado.
(Software Lenguaje Maya - Edupmedia-2008)

REQUERIMIENTO Nro. 00046		
El sistema debe contar con un diccionario de Glifos Mayas		
Usuario	X	El usuario podrá escribir la palabra en español o inglés, y el diccionario le mostrará el glifo correspondiente.
Sistema		
Funcional	X	
No funcional		
Pre-condición		Post-condición
La palabra introducida debe existir en el diccionario		Semuestra el glifo, la pronunciación y escritura del mismo.
Fecha creación	20-10-2008	Responsable
Fecha procesamiento	01-12-2008	Raúl Toscano-Manuel Caro
Modificación		Causa/Responsable

Diagrama de casos de uso. Un caso de uso especifica el comportamiento de un sistema o de una parte del mismo, y es una descripción de un conjunto de secuencias de acciones, incluyendo variantes, que ejecuta un sistema para producir un resultado observable de valor para un actor.

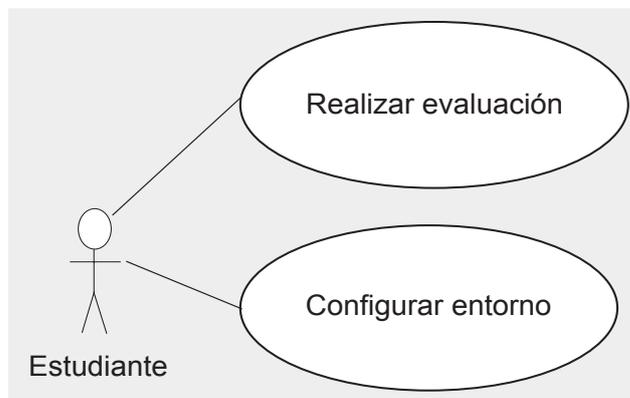


Figura 5. Ejemplo de un diagrama de casos de uso. (Software Lenguaje Maya - Edupmedia-2008)

Los casos de uso (figura 5) se utilizan para describir la forma como el usuario interactúa con el sistema y las acciones que presenta el sistema como respuesta a dicha interacción. Estos se dividen en elementos, actores, diagramas de casos de uso y su objetivo es permitir la captura de requisitos potenciales para el nuevo sistema. Los casos de uso deben cumplir dos objetivos: Definir los requisitos funcionales del sistema y proporcionar una base para la validación de las pruebas [2].

Diagrama de clases (figura 6). Un diagrama de clase muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estático de un sistema. Principalmente, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas [2].

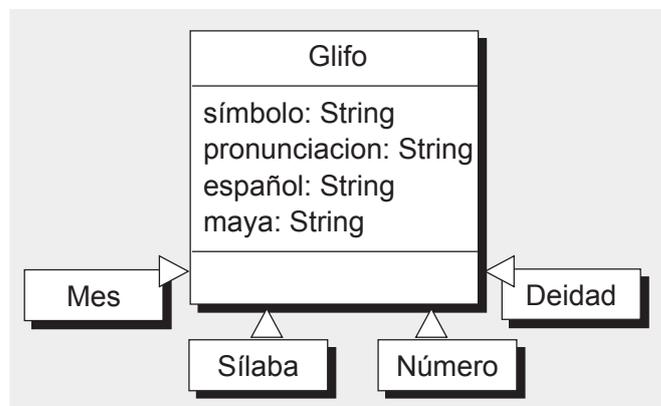


Figura 6. Ejemplo de un diagrama de clases. (Software Lenguaje Maya - Edupmedia-2008).

El diagrama de clases (figura 6) está compuesto por elementos identificados como clase y relaciones, que al unirlos conforman los esquemas o diagramas que representarán un modelo de clases. Estos diagramas resultan de la composición de los casos de uso.

El diagrama de clases debe expresar detalles de los objetos que existen en el área de enfoque, en él se incluyen atributos y métodos de cada una de las clases. A continuación se muestra un ejemplo del diagrama de clases.

Diagrama de objetos. Un diagrama de objetos es un diagrama que representa un conjunto de objetos y sus relaciones en un momento concreto. Gráficamente, un diagrama de objetos es una colección de nodos y arcos. Un diagrama de objetos es un tipo especial de diagramas y comparte las propiedades comunes al resto de los diagramas (un nombre y un contenido gráfico que es una proyección de un modelo) [2].

Diagrama de secuencia. Los Diagramas de Secuencia (figura 7) representan una interacción entre objetos de manera secuencial en el tiempo. Muestra la participación de objetos en la interacción entre sus "líneas de vida" (desde que se instancia el objeto) y los mensajes que ellos organizadamente intercambian en el tiempo.

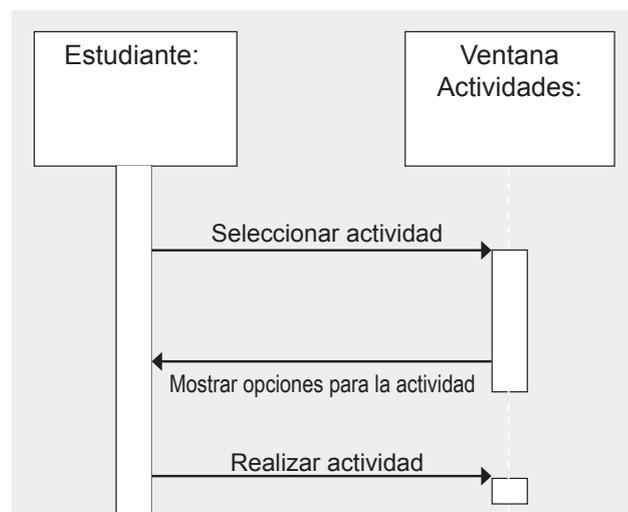


Figura 7. Ejemplo de un diagrama de secuencia. (Software Lenguaje Maya - Edupmedia-2008).

El responsable o actor es quien inicia el ciclo interactuando inicialmente con la interfaz de usuario (IGU); en seguida se inician todos los objetos que intervienen en el funcionamiento del aplicativo. En este diagrama se comienza a observar el comportamiento del sistema a partir de los eventos generados por los actores [2]. Veamos un ejemplo del diagrama de secuencias:

Modelado del sistema de conocimiento/bases de datos. En esta etapa se recomienda la utilización de bases de datos relacionales o archivos de texto con estructura

XML. Si la opción es la primera, se deberá crear un MER (Modelo Entidad Relación), un MR (Modelo relacional) basado en el MER (figura 8) y un diccionario de datos para los metadatos (como tipos, tamaños, y otras cualidades de los campos de datos). Estos modelos son de uso generalizado y cuenta con abundante bibliografía. Si la opción es el archivo de XML, bastará entonces con el diccionario de datos.

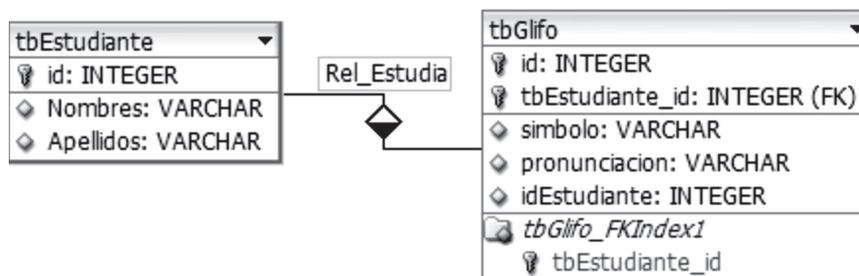


Figura 8. Vista parcial de un MER. (Software Lenguaje Maya - Edupmedia-2008 [3])

2.6. FASE IV: PRODUCCIÓN

En la fase de producción se agrupan todos los materiales obtenidos en el desarrollo de las fases anteriores, por tal motivo es aquí donde se realiza la implementación de todos los documentos obtenidos en las fases de diseños que serán codificados por un especialista en programación permitiendo el desarrollo final del software, que conllevará a la presentación de un prototipo el cual será evaluado para determinar si cumplió con los objetivos propuestos. Además como último paso se realizará un manual de usuario para que los usuarios puedan navegar fácilmente por el software.



Figura 9. Fase IV. Producción. (Edupmedia-2008).

Selección de herramienta de desarrollo. Para llevar a cabo la codificación es necesario seleccionar una herramienta o lenguaje de programación que sea acorde con las

especificaciones del software desarrollado, es importante tener en cuenta que la persona encargada de la codificación domine la herramienta o lenguaje seleccionado para que se facilite la programación de las actividades y contenidos del software, permitiendo el cumplimiento de las metas propuestas en términos de eficacia y eficiencia.

Codificación. Esta etapa ayuda a que el desarrollo del software con respecto a la programación tenga una presentación correcta, entendible y con buen estilo, la aplicación de estos parámetros permitirá fácilmente el mantenimiento del software para poder corregir los errores y modificar algún requisito cuando sea necesario.

Presentación del prototipo. El prototipo es la primera versión que se obtiene al culminar el diseño y desarrollo del software, esta presentación se realiza para verificar si el software cumplió con las especificaciones planteadas en la necesidad educativa, para esto se selecciona un grupo representativo de la población objeto de estudio a quienes se les mostrará y aplicará el prototipo.

Evaluación del prototipo. Al conocer los resultados de las pruebas desarrolladas al prototipo, se reúne el grupo de trabajo para realizar un análisis de las evaluaciones efectuadas, para determinar si es necesario realizar ajustes al prototipo dependiendo de los errores.

Manual de usuario. Pequeño libro que recibe el usuario junto con el sistema de composición, ya sea con referencia al hardware o al software, que le permite aprender las particularidades del equipo o del sistema y la forma de utilizarlo. Una vez familiarizado con el mismo, pasa a ser un instrumento de consulta para cuando se ha olvidado algún aspecto o se presenta alguna duda [11].

2.7. FASE V: APLICACIÓN

La fase de aplicación es la etapa final donde se mostrará el software terminado y funcionando a cabalidad. Esta comprende la aplicación del software a toda la población objeto de estudio en condiciones normales del aprendizaje para comprobar si el programa en realidad cumplió con los requisitos establecidos en los objetivos propuesto en la necesidad educativa.

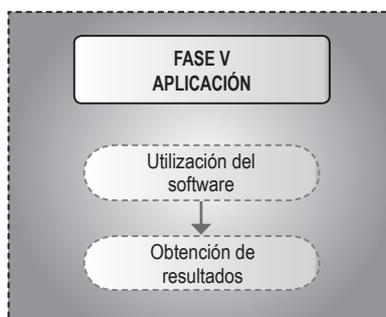


Figura 10. Fase V. Aplicación. (Eduppedia-2008).

Utilización del software. Para utilizar un software es necesario conocer cuales son las condiciones que se establecen para que los usuarios puedan interactuar sin mayor dificultad con el software, esto permitirá que la aplicación tenga el efecto esperado para lo cual fue desarrollada.

Obtención de resultados. En esta fase se realiza el análisis de los resultados obtenidos en la etapa de utilización del software. A partir de este estudio se logrará determinar si el software aplicado fue eficaz, eficiente y cuenta con los requisitos necesarios para seguir con su utilización en el aula de clases o por lo contrario, éste requiere que se realicen ajustes necesarios para cumplir con la necesidad para la cual fue diseñado.

3. COMPARACIÓN CON OTROS MODELOS

En este apartado primero se procederá a realizar una breve descripción de los modelos existentes en la actualidad para el desarrollo de software educativo y luego se presentarán las diferencias entre el modelo propuesto y los ya existentes a modo de cuadro comparativo.

3.1. MODELOS PARA EL DESARROLLO DE SOFTWARE EDUCATIVO

Ingeniería de software educativo con modelaje orientado por objetos: un medio para desarrollar micromundos interactivos [8]. La integración de Ingeniería de Software Educativo (ISE), con el modelaje 0.0, permite hacer uso efectivo de la información recolectada en la fase de análisis y diseño educativo; por está razón es necesario su inclusión en todas las etapas del ciclo de desarrollo para unificar los términos en los que se habla en cada etapa, estableciendo un modelo del mundo del problema y de su comportamiento; de este modo se hace referencia a objetos presentes en el modelo, extendiendo así su funcionalidad. Al llegar a la implementación, los resultados obtenidos se transcriben al lenguaje de programación escogido, cambiando la sintaxis en que se expresa el modelo, más no la semántica.

Metodología Dinámica para el Desarrollo de Software Educativo [14]. La metodología está compuesta por cuatro fases (Diseño Educativo, Producción, Realización e Implementación.) y un eje transversal que es la Evaluación. 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 correcciones, de ser requeridas.

Metodología THALES [10]. Representa un modelo híbrido no lineal y consta de seis fases: planeación, diseño, producción, prueba piloto, evaluación y mejoramiento. Para la versión 1 se estableció un ciclo en las fases diseño-producción-prueba piloto y se culmina con la evaluación-mejoramiento. En futuras evaluaciones puede tomarse la decisión de revisar los documentos desde el principio, es decir retomar la fase de planeación.

MOSCA - Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistémica [13]. Este modelo mide la calidad sistémica de una organización que desarrolla software partiendo de la calidad de su producto y de la calidad del proceso de desarrollo del mismo. El modelo MOSCA especifica los procesos que se deben mejorar en la empresa y las características que no son satisfechas por el producto de software desarrollado.

3.2. ANÁLISIS COMPARATIVO

Se puede apreciar en la tabla 5 que los otros modelos estudiados no cuentan con indicaciones precisas acerca de cómo involucrar las competencias que se esperan que el estudiante desarrolle con el proceso de diseño del software educativo. En el caso de los modelos extranjeros, se deduce que no soportan esta capacidad debido a estar inmersos en el contexto educativo local de cada país en el que se han propuesto.

Tabla 5. Tabla comparativa entre los modelos de diseño de software educativo estudiados.

Indicadores de comparación	Modelo propuesto	Modelos existentes		
	MODESEC (Modelo para el desarrollo de software educativo basado en competencias)	Metodología Dinámica para el Desarrollo de Software Educativo	Metodología THALES	Ingeniería de software educativo con modelaje orientado por objetos: un medio para desarrollar micromundos interactivos
Adaptado al sistema de competencias colombiano				
Personas que intervienen en el proyecto (menos de 10 personas)				
Fases e implementación de los distintos modelos	Cinco	Cuatro	Seis	Cinco
Norma o Estándar				
Disponibles software hechos con cada modelos				
Origen	Colombia	Argentina	Venezuela	Colombia
Tipos de modelos	Lineal solapado	Lineal solapado	Cíclico	Cíclico

Otros modelos como el Mosca y el 0.0 se basan más en actividades ingenieriles que le van a dificultar a los docentes de las instituciones de educación colombianas utilizarlos para implementar soluciones a las necesidades educativas que en ellas se puedan presentar.

4. APLICACIÓN Y VALIDACIÓN DEL MODELO

El modelo propuesto en este artículo se ha validado con éxito en los trabajos de grado de la línea de “Desarrollo de software educativo” que han sido orientados y/o dirigidos por el grupo de investigación Edupmedia. Se encuentran disponibles en CD los programas Edulógica, para refuerzo de la lógica matemática; Brújula, para tratar el programa de la disgrafía en niños y Zenulandia para la enseñanza de la cultura Zenú en Colombia.

Otra validación ha sido la realizada en la asignatura de Diseño de Software Educativo, de la cual se disponen en CD las aplicaciones: Lenguaje Wayuunaiki, La conquista de América, Los Mayas, Learning mandarín y Los Aztecas. Este último será el utilizado en el presente apartado como caso de estudio de la aplicación del modelo [4]. En la tabla 6 se presenta la descripción de un software educativo realizado con el modelo propuesto.

Tabla 6. Ficha de presentación general del software.

TÍTULO	SOFTWARE AZTECA
Objetivo General	
Desarrollar un software educativo que facilite, a los estudiantes de noveno grado de las instituciones educativas de Córdoba, conocer y aprender de las grandes culturas Americanas en especial la civilización Azteca.	
Objetivos Específicos	
Despertar el interés y motivación de los estudiantes para aprender sobre la civilización Azteca. Mejorar el grado de comprensión de las temáticas referentes a la civilización Azteca. Lograr que los estudiantes conozcan la historia y evolución del imperio Azteca, desde su fundación en 1325 hasta su caída en 1521.	
CREADORES: LUIS FERNANDO ÁLVARES FAJARDO Y LILIANA OME MORELO	
ASESORIA: MANUEL CARO PIÑERES. UNIVERSIDAD DE CÓRDOBA. 2008	

4.1. ESTUDIO DE LA NECESIDAD EDUCATIVA

Tabla 7. Ficha para diligenciar los datos de la necesidad educativa.

NECESIDAD EDUCATIVA
Tipo de necesidad
Según Burton y Merryl, la necesidad que se requiere tratar con el diseño de nuestro producto es de tipología sentida, ya que, esta temática fue sugerida por la institución educativa, con base en las falencias que presentan los estudiantes de noveno grado en las diferentes instituciones del departamento de Córdoba.

POBLACIÓN	
Rango de edad	
Las edades de los estudiantes de noveno grado de las instituciones educativas del departamento de Córdoba oscilan entre los 13 y 17 años de edad.	
Escolaridad	Estudiantes de noveno grado de las instituciones educativas del Departamento de Córdoba.
Conocimientos que poseen.	Los estudiantes deben tener conocimiento de las temáticas que en los colegios se dan antes de ésta.
Intereses y expectativas (población)	Desarrollar actitudes que le permitan manejar el software. Que el software les facilite la comprensión de la temática.
Intereses y expectativas (creadores)	
Desarrollar actitudes en los estudiantes que le permitan manejar el software. Que los estudiantes a través del software conozcan parte de la cultura latinoamericana, como la civilización azteca, de una manera más interactiva.	
ÁREA DE FORMACIÓN	
Área del saber	Sociales e Historia
Área de contenido	Grandes culturas Americanas (los Aztecas)
Unidades temáticas	Ubicación geográfica de la civilización azteca Historia de la fundación y consolidación del imperio Azteca Aspectos culturales de los Aztecas (economía y religión) Lenguaje Azteca (diccionario) Conquista y caída del imperio Azteca.
ESTADO ACTUAL - DIAGNÓSTICO	
A partir de observaciones y encuestas a docentes y estudiantes, de las distintas instituciones educativas del departamento de Córdoba, queda al descubierto que en estas se vienen presentando problemas de aprendizaje en las distintas áreas del saber, uno de los problemas más grandes se da en el área de sociales e historia, donde los estudiantes se muestran cansados, fatigados o simplemente desinteresados, durante el desarrollo de las actividades académicas en las que trata, sobre las grandes culturas Americanas, en especial la civilización Azteca.	
NECESIDAD EDUCATIVA - FORMULACIÓN	
Mediante el diagnóstico realizado nos podemos dar cuenta del desinterés de los estudiantes por el estudio de las grandes culturas Pre-Hispánicas Americanas y en especial la civilización Azteca.	
POSIBLES CAUSAS	
La falta de material didáctico para el desarrollo de estas clases. Los grandes volúmenes de contenido que tienen estas temáticas. La aplicación por parte del docente de una metodología muy poco atractiva para los estudiantes. La falta de creatividad y recursividad de los maestros para el desarrollo de su actividad académica.	
POSIBLES SOLUCIONES	
Diseñar y crear herramientas didácticas, ya sean software educativo, videos, materiales multimediales u otros, con el fin de que las clases sean más amenas. Incluir en este software educativo o material multimedial el contenido de esta temática. El uso de una metodología que sea del agrado de los estudiantes. Capacitar a los docentes con el fin de que sean más creativos al momento de planear sus clases.	

CONOCIMIENTOS Y HABILIDADES QUE DEBE TENER EL ESTUDIANTE	
1. PRECONCEPTOS	Tener claro los contenidos previos a esta temática.
2. PRECONDICIONES	Manejo básico del computador Tener habilidades psicomotrices.

4.2. DISEÑO DEL SISTEMA DE COMPETENCIAS

En este apartado se muestra una vista parcial de las competencias tratadas en el software.

Tabla 8. Ficha para diligenciar los aspectos de las competencias.

DISEÑO DE COMPETENCIAS	
Objetivos	Normas
Enunciados 1: Reconoce la ubicación geográfica del imperio azteca a lo largo de su historia.	Contexto: cuando se le solicite la ubicación geográfica del imperio Azteca. Recursos: Mapa de centro América. Imágenes del imperio azteca. Videos del imperio azteca. Software de la cultura Azteca.
Elementos	Indicadores
Identifica la ubicación geográfica de los aztecas en el siglo XIV y principios de siglo XV.	1.1. Relata la ubicación geográfica de los aztecas en esta época. 1.2. Reconoce en el mapa de centro América la ubicación geográfica de los aztecas. 1.3. Conocer la ciudad principal donde se ubicaron los aztecas mediante imágenes.
Identifica la ubicación geográfica de los aztecas a mediados del siglo XV.	1.1. Relata la ubicación geográfica de los aztecas en esta época. 1.2. Reconoce en el mapa de centro América la ubicación geográfica de los aztecas. 1.3. Conoce la ciudad principal donde se ubicaron los aztecas mediante imágenes.
Identifica la ubicación geográfica de los aztecas a finales del siglo XV.	1.1. Relata la ubicación geográfica de los aztecas en esta época. 1.2. Reconoce en el mapa de centro América la ubicación geográfica de los aztecas. 1.3. Conoce la ciudad principal donde se ubicaron los aztecas mediante imágenes.
1. Identifica la ubicación geográfica de los aztecas a principios del siglo XVI.	1.1. Relata la ubicación geográfica de los aztecas en esta época. 1.2. Reconoce en el mapa de centro América la ubicación geográfica de los aztecas. 1.3. Conoce la ciudad principal donde se ubicaron los aztecas mediante imágenes.
CONCEPTOS	Generalidades. Historia. Civilización. Aztecas. Ciudades. Texcoco. Tenochtitlan. <u>Azcapotzalco</u> . Tepanecas.

4.3. DISEÑO DE CONTENIDOS

En este caso sólo se muestran los contenidos que harán parte de las estrategias de la competencia que se ha descrito en el párrafo anterior. Recordemos que la fuente de los términos e ideas que aquí se explican es la sección de Conceptos del formato para la formalización de competencias. Ver apartado 4.2.

Tabla 9. Ficha para diligenciar el diseño de los contenidos.

CONCEPTOS DE LAS COMPETENCIAS		
CONCEPTOS	CARACTERÍSTICAS	DEFINICIÓN
AZTECAS		
- Tribus guerreras de filiación nahua. - Fundó a México-Tenochtitlan. - Estados más extensos de Mesoamérica. - Cazadores y recolectores.		Fueron un tribus guerreras de filiación nahua que fundó a México-Tenochtitlan convirtiéndose en el centro de uno de los Estados más extensos de Mesoamérica eran cazadores y recolectores también tenían tierras de producción para cubrir todos los gastos de su población.
TEPANECAS		
- Pueblo prehispánico de origen chichimeca. - Se instaló en la Cuenca de México a mediados del siglo XII de nuestra Era. - Se ubicaron en el año 1012 en la región poniente del lago Texcoco.		Es el nombre de un pueblo prehispánico de origen chichimeca,[]que se instaló en la Cuenca de México a mediados del siglo XII de nuestra era y se ubicaron en el año 1012 en la región poniente del lago de Texcoco.

4.4. DISEÑO PEDAGÓGICO

De acuerdo con lo especificado en el modelo de desarrollo, primero se selecciona un modelo pedagógico con base en las necesidades educativas y luego se verifican las características del modelo que son susceptibles para incluirlas en el software educativo.

- **Teoría del Modelo seleccionado para el software Azteca:** Constructivismo social y aprendizaje significativo: Es un modelo de enseñanza el cual nos dice que el individuo aprende mientras convive o interactúa con el medio en donde habita y los sujetos que lo rodean los cuales ayudan a que él construya un nuevo conocimiento.
- **Principios de esta teoría aplicables al software Azteca:**
 - El aprendizaje es responsabilidad del estudiante.
 - Se debe presentar un choque cognitivo para retar al estudiante y en cierta forma hacerlo desaprender ciertos conceptos que no son los correctos.
 - Los nuevos conceptos deben ser significativos esto es relacionado con su experiencia con otros. Que ya posea que le sirvan de ancla para el nuevo aprendizaje.

- El individuo aprende cuando interactúa con otros: Los alumnos se reúnen en grupos para explorar e interactuar con el software educativo de la cultura azteca, en donde los educandos se pueden ayudar mutuamente en caso de que si uno de ellos no entiende o presenta problemas al momento de interactuar con el Software, los demás miembros del grupo le pueden explicar o resolver la inquietud que tenga acerca de la temática o el uso de éste con el fin de adquirir nuevos conocimientos.

4.5. DISEÑO DE LAS ACTIVIDADES DE APRENDIZAJE

Recordemos que para diseñar las actividades de aprendizaje se deben tener en cuenta las competencias especificadas y el modelo pedagógico seleccionado.

Problema. Esta ventana materializa la premisa estipulada en el diseño pedagógico donde se presenta una situación problémica que genere un choque cognitivo para retar al estudiante y en cierta forma hacerlo desaprender algunos conceptos que no son los correctos. En esta ventana a modo de historia se le introduce en la temática y el problema que se va a resolver se le presenta en un video ambientado con animación.



Figura 11. Ventana de presentación de la situación problémica. (Software Azteca – Universidad de Córdoba. 2007).

Estrategias. En este caso el modelo pedagógico determina la utilización de estrategias que le darán al estudiante la información que necesita para resolver el problema que se le plantea al inicio del programa. Estas estrategias son de tipo exploratorio y no implican secuencialidad ni obligatoriedad de visita, esto se hace de este modo para ir en concordancia con el modelo seleccionado. Cada estrategia cuenta con una sección para su evaluación. Esto se hace con el fin de dividir el problema presentado al inicio, en varios problemas pequeños más fáciles de solucionar por parte del estudiante.

Estrategia 1

Nombre:	Ubicación Geográfica
Objetivo:	Dar a conocer el lugar en donde se ubicaron los aztecas a lo largo de su evolución.



Figura 12. Ventana de la estrategia para la ubicación geográfica.
(Software Azteca – Universidad de Córdoba. 2007).

Al desplegarse la ventana, aparece un mapa interactivo que contiene la ubicación de los aztecas, éste presenta una línea de tiempo que muestra la ubicación del imperio a lo largo de la historia y cuando se haga clic en la línea de tiempo aparece la imagen de la ciudad y al lado de la imagen un cuadro de texto con la explicación de la ciudad y ubicación del imperio azteca. A su vez se acompaña de audio y recursos extras como archivos de texto con extensión PDF para quienes deseen profundizar más en el tema.

Estrategia 2

Nombre:	Emperadores Aztecas Ingles /español.
Objetivo:	Conocer el perfil de cada uno de los emperadores y su periodo al mando en este pueblo azteca.

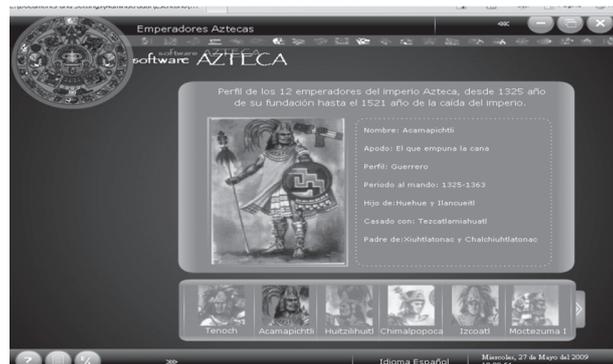


Figura 13. Ventana de la estrategia para los emperadores
(Software Azteca – Universidad de Córdoba. 2007).

La ventana cuenta con las imágenes de cada uno de los emperadores en plano medio con su texto debajo con su periodo al mando del imperio. Cuando se dé clic en las imágenes de alguno de ellos aparece su imagen en plano general y se muestra una ficha

con su perfil. La ventana está diseñada a modo de galería fotográfica y cuenta con el audio para ayudar a la correcta pronunciación del nombre de cada emperador. A su vez se acompaña de audio y recursos extras como archivos de texto con extensión PDF para quienes deseen profundizar más en el tema.

Estrategia 3

Nombre:	Calendario Ingles /español.
Objetivo:	Conocer el significado de cada una de las partes del calendario y su representación.

En esta ventana el usuario se encontrará con la imagen del calendario azteca, y cuando da clic en una de las partes del mismo, este despliega un texto con su significado y la ampliación del símbolo.

Evaluación. El usuario deberá leer las instrucciones del juego, en donde se indica cómo funciona, este deberá hacer clic sobre el personaje de Hernán Cortés para empezar a caminar por el escenario, cuando llegue a uno de los guardias debe hacer clic sobre éste, quien le formulará una pregunta que el usuario debe responder con las opciones que le den y si acierta continua caminando y seguirán las preguntas hasta llegar al emperador Moctezuma y ganar la partida, pero si se equivoca en una de las preguntas aparece un cuadro donde se informa el aspecto en que falló y al lado de éste hay un botón que lo lleva a las ventanas donde se encuentra la información para que repase todo lo referente a la pregunta y vuelva a intentarlo.



Figura 15. Ventana de la estrategia para la evaluación general (Software Azteca – Universidad de Córdoba. 2007).

Recordemos que la evaluación general debe estar ligada a la problemática planteada al inicio del software y debe abarcar todos los elementos contenidos en las estrategias de estudio.

5. CONCLUSIONES

En este artículo se ha demostrado que el modelo presentado es fácil de implementar por parte de los docentes y practicantes de las instituciones de educación básica y secundaria en Colombia, que en el fondo es el objetivo principal del modelo.

El modelo propuesto se convierte en una guía de fácil uso para implementar software educativo a la medida de las necesidades de una institución educativa, las cuales cuentan con profesionales en el área del problema, con expertos en pedagogía y didáctica, y con docentes expertos en informática. Con este personal mínimo se puede desarrollar un software educativo de calidad siguiendo el modelo propuesto, sin incurrir en gastos extra para la institución.

Con respecto a los otros modelos analizados, el modelo propuesto cuenta con grandes ventajas como por ejemplo ser el único que presenta una metodología para abordar de forma clara el diseño de las competencias que se van a tratar en el software educativo.

Otra ventaja que presenta es que está diseñado para desarrollar software educativo con bajo presupuesto y con un equipo de trabajo que no supera las cinco personas. Estos son factores decisivos en la realidad colombiana, pues las instituciones educativas de las regiones cuentan con presupuestos limitados que no les permiten hacer grandes inversiones en este tipo de materiales.

Aunque el modelo esté planteado para el desarrollo de software educativo a la medida, esto no es impedimento para poder utilizarlo a nivel industrial en la producción en masa, ya que los principios se mantienen igual.

TRABAJOS FUTUROS

En la línea de diseño de materiales educativos y metodologías para el diseño de herramientas de software buscaremos profundizar en la aplicación precisa de cada una de las teorías pedagógicas contemporáneas en el desarrollo de software educativo con el fin de crear estrategias claras que le permitan a los docentes y diseñadores seleccionar la mas adecuada para cada necesidad educativa y luego poder aplicarla de la mejor forma posible en el software.

REFERENCIAS

- [1] AMORÓS, E. 2007. Comportamiento organizacional: En busca del desarrollo de Ventajas Competitivas. Escuela de Economía de USAT de Lambayeque. Perú www.usat.edu.pe/carreras1/economía272.

- [2] BOOCH, G. 2001. Lenguaje de Modelado Unificado. Editorial Adisson Wesley. Madrid.
- [3] CARO, M., GALARCIO, E., RUIZ, S. 2008. Software Lenguaje Maya. EdupMedia. Universidad de Córdoba. Montería.
- [4] CARO, M.; ALVAREZ, L.; OME, L. 2008. Software Azteca. EdupMedia. Universidad de Córdoba. Montería.
- [5] Cómo elaborar un guión multimedial parte I. <http://nodocreativo.blogspot.com/2008/04/cmo-realizar-un-guin-multimedia-parte-i.html>
- [6] DE ZUBIRÍA, J. 2001. De la escuela nueva al constructivismo. Un análisis crítico, Cooperativa Editorial Magisterio. Bogotá, Colombia.
- [7] FLÓREZ, R. 2005. Pedagogía del Conocimiento, segunda edición. Editorial Nomos. Bogotá D.C.
- [8] GOMEZ, R.; GALVIS, A.; MARIÑO, O. 2000. Ingeniería de software educativo y modelaje orientado a objetos. Un medio para desarrollar micro mundos interactivos. Universidad de Los Andes. Bogotá. Colombia.
- [9] LOSADA, A., MORENO, H. 2003. Competencias básicas aplicadas al aula. Ediciones SEM (Servicio educativo del magisterio). Bogota D.C.
- [10] MADUEÑO, L. 2003. Modelo THALES de desarrollo de software educativo. Universidad de Zulia. Venezuela.
- [11] Manual de usuario. <http://www.proyectosfindecarrera.com/definicion/manualdeusuario.htm>
- [12] MARQUÉS, P. 1999. Selección y Uso de Recursos Multimedia. Diseño de Actividades. Usos en el Aula Informática. <http://dewey.uab.es/pmarques/interven.htm>.
- [13] MENDOZA, L.; PEREZ, M.; GRIMÁN, A. 2004. Prototipo de modelo sistémico de calidad (MOSCA) del software. Universidad Simón Bolívar. Caracas Venezuela.
- [14] MOLINA, R. 2006. Modelo de desarrollo de software educativo. Argentina.
- [15] TORRADO, M. 1999. El desarrollo de competencias: una propuesta para la educación colombiana. Mimeo. Bogotá D.C.