

**Título: El proyecto de Software Educativo. Algunas consideraciones.**

**Autor: MSc. José Antonio Gutiérrez Lam. Profesor Auxiliar**

**Centro de trabajo: Universidad de Ciencias Pedagógicas "Rafael María de Mendive"  
de Pinar del Río**

**E-mail: [j.lam@ucp.pr.rimed.cu](mailto:j.lam@ucp.pr.rimed.cu)**

### **Resumen:**

Resulta importante conocer algunos elementos acerca del proceso de producción del software educativo, para de esta forma lograr, que el aprovechamiento de las computadoras en el proceso docente - educativo tenga un papel relevante. Para ello, se hace necesario dotarlas de un software educativo de calidad, lo que debe medirse en términos del conocimiento que sean capaces de representar y transmitir.

### **Abstract:**

It is important to understand some elements of the production process of educational software, to thus make the use of computers in the teaching process has an important role. To do this, it is necessary to give them a quality educational software, which is measured in terms of knowledge that can represent and convey.

### **La Ingeniería de Software**

Se considera que "la Ingeniería de Software es la rama de la ingeniería que aplica los principios de la Informática y las Matemáticas para lograr soluciones costo-efectivas (eficaces en costo o económicas) a los problemas de desarrollo de software, es decir, permite elaborar consistentemente productos correctos, utilizables y costo-efectivos"<sup>1</sup>

El proceso de ingeniería de software se define como "un conjunto de etapas parcialmente ordenadas con la intención de lograr un objetivo, en este caso, la obtención de un producto de software de calidad" <sup>2</sup>

El proceso de desarrollo de software "es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo. Concretamente define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo"<sup>3</sup>

Después de analizar estas definiciones, resulta importante destacar, que el proceso de desarrollo del software, requiere de un conjunto de conceptos, una metodología y un lenguaje propio. Este proceso, también llamado ciclo de vida del software, comprende cuatro grandes fases: concepción, elaboración, construcción y transición. La concepción precisa el alcance del proyecto y define los casos de uso del sistema. La elaboración conlleva a proyectar un plan, definir las características y cimentar la arquitectura. La construcción permite desarrollar el producto y la transición posibilita transferir el producto a sus usuarios.

### **¿Cómo clasificar la diversidad de software que existe?**

Existen diversos criterios para clasificar el software; según Carlos Expósito<sup>4</sup>, el software se clasifica en 4 categorías:

- Software de Sistema Operativo: El sistema operativo es el gestor y organizador de todas las actividades que realiza la computadora. Marca las pautas según las cuales se intercambia información entre la memoria central y la externa, y determina las operaciones elementales

<sup>1</sup> Cota A. Ingeniería de Software: soluciones avanzadas. 1994. -- pp. 5-13

<sup>2</sup> Jacobson, I. Applying UML in The Unified Process. Rational Software.-- Disponible en <http://www.rational.com/uml>

<sup>3</sup> Jacobson, I. Applying UML in The Unified Process. Rational Software.

<sup>4</sup> Expósito, Carlos. CD Maestría. Conceptos generales de software. - La Habana : Editorial Pueblo y Educación , 2005

que puede realizar el procesador.

- Lenguajes de Programación: Mediante los programas se indica a la computadora que tarea debe realizar y cómo efectuarla, pero para ello es preciso introducir estas órdenes en un lenguaje que el sistema pueda entender.
- Software de Uso General: ofrece la estructura para un gran número de aplicaciones empresariales, científicas y personales. La mayoría de software para uso general aparecen como paquetes; es decir, con software y documentación orientada al usuario.
- Software de aplicaciones: esta diseñado y escrito para realizar tareas específicas personales, empresariales o científicas como el procesamiento de nóminas, la administración de los recursos humanos o el control de inventarios. Todas estas aplicaciones procesan datos (recepción de materiales) y generan información (registros de nómina) para el usuario.

En Cuba, se ha logrado alcanzar una alta relevancia en la producción de software para la educación, aprovechando, el inmenso volumen de información de que dispone el hombre en los momentos actuales y los propios factores que han motivado una masividad en el uso de la tecnología, para la formación integral de las nuevas generaciones, lo que por supuesto ha posibilitado que se convierten en medios muy útiles para alcanzar este fin.

### **El software educativo.**

La introducción y utilización efectiva de las computadoras con fines docentes es un fenómeno complejo y de amplias perspectivas. Para lograr que el aprovechamiento de las computadoras en el proceso docente tenga un papel relevante, se hace necesario dotarlas de un software educativo de calidad, lo que debe medirse en términos del conocimiento que sean capaces de representar y transmitir.

Sánchez, J define el concepto de Software Educativo, como cualquier programa computacional cuyas características estructurales y funcionales sirvan de apoyo al proceso de enseñar, aprender y administrar.<sup>5</sup>

Según Rodríguez, R es una aplicación informática, que soportada sobre una bien definida estrategia pedagógica, apoya directamente el proceso de enseñanza aprendizaje constituyendo un efectivo instrumento para el desarrollo educacional del hombre actualmente.<sup>6</sup>

Pere Marqués lo define como "programas educativos y didácticos para designar genéricamente los programas para computadoras creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje"<sup>7</sup>.

Finalmente, los Software Educativos se pueden considerar como el conjunto de recursos informáticos diseñados, con la intención de ser utilizados en el contexto del proceso docente – educativo, que potencie un aprendizaje para toda la vida.

Los software educativos a pesar de tener rasgos esenciales básicos y una estructura general común se presentan con características muy diversas. No se puede afirmar que exista un solo tipo de software educativo, sino que este ha venido evolucionando de acuerdo con las innovaciones tecnológicas, generalmente pueden caracterizarse en sus variantes de software de escritorio o software para la web por ser altamente interactivos, a partir del empleo de recursos multimediales, como enlaces, hipertextos, hipermedias, videos, sonidos, fotografías, diccionarios especializados, explicaciones de experimentados profesores, ejercicios y juegos instructivos que apoyan las funciones de evaluación y diagnóstico.

Es necesario entonces considerar como se han enfocado estos software atendiendo a los polos en los cuales se ha movido la educación:

---

<sup>5</sup> Sánchez, J. Construyendo y aprendiendo con el computador. – /s.n. /, 2006. -- p. 12

<sup>6</sup> Introducción a la informática educativa / R. Rodríguez Lamas, R. .. [et al.] . -- Pinar del Río: Universidad Hermanos Saiz, 2000.-- 151 p.

<sup>7</sup> Pere Marqués. El software educativo. – Barcelona: Universidad Autónoma, 2003. – p. 32-36.

- Un software educativo de tipo algorítmico: predomina el aprendizaje vía transmisión del conocimiento. El rol del alumno es asimilar el máximo de lo que se le transmite.
- Un software de tipo heurístico: el estudiante descubre el conocimiento interactuando con el ambiente de aprendizaje que le permita llegar a él.

Considerando la función educativa pudiéramos asumir que "dentro de los productos con un predominante enfoque algorítmico se pueden considerar los denominados sistemas tutoriales, sistemas entrenadores y libros electrónicos; mientras que en los que predomina el enfoque heurístico se pueden encontrar los simuladores, juegos educativos, sistemas expertos y sistemas tutoriales inteligentes de enseñanza".<sup>8</sup>

Veamos a continuación algunas de sus principales características.

- **Sistemas Tutoriales:** sistema basado en el diálogo con el estudiante, adecuado para presentar información objetiva, tiene en cuenta las características del alumno, siguiendo una estrategia pedagógica para la transmisión de conocimientos. Estos pueden ser: programas lineales, programas ramificados y entornos tutoriales.
- **Sistemas Entrenadores:** se parte de que los estudiantes cuentan con los conceptos y destrezas que van a practicar, por lo que su propósito es contribuir al desarrollo de una determinada habilidad, intelectual, manual o motora, profundizando en las dos fases finales del aprendizaje: aplicación y retroalimentación.
- **Libros Electrónicos:** su objetivo es presentar información al estudiante a partir del uso de texto, gráficos, animaciones, videos, etc., pero con un nivel de interactividad y motivación que le facilite las acciones que realiza.
- **Simuladores:** su objetivo es apoyar el proceso de enseñanza – aprendizaje, semejando la realidad de forma entretenida.
- **Juegos Educativos:** su objetivo es llegar a situaciones excitantes y entretenidas, sin dejar en ocasiones de simular la realidad.
- **Sistemas Expertos:** programa de conocimientos intensivo que resuelve problemas que normalmente requieren de la pericia humana. Ejecuta muchas funciones secundarias de manera análoga a un experto, por ejemplo, preguntar aspectos importantes y explicar razonamientos.
- **Sistemas Tutoriales Inteligentes de enseñanza:** despiertan mayor interés y motivación, puesto que pueden detectar errores, clasificarlos, y explicar por qué se producen, favoreciendo así el proceso de retroalimentación del estudiante.

No obstante, existen otros "criterios donde se muestran clasificaciones que pueden proporcionar categorías claras y útiles a los profesores teniendo en cuenta el grado de control del programa sobre la actividad de los alumnos y la estructura de su algoritmo".<sup>9</sup>

Por ejemplo podemos mencionar:

- **Bases de datos:** proporcionan datos organizados, en un entorno estático, según determinados criterios, y facilitan su exploración y consulta selectiva. Las bases de datos pueden tener una estructura jerárquica (si existen unos elementos subordinantes de los que dependen otros subordinados, como los organigramas), relacional (si están organizadas mediante unas fichas o registros con una misma estructura y rango) o documental (si utiliza descriptores y su finalidad es almacenar grandes volúmenes de información documental: revistas, periódicos, etc.). En cualquier caso, según la forma de acceder a la información se pueden distinguir dos tipos: bases de datos convencionales y bases de datos tipo sistema experto.

<sup>8</sup> Rodríguez Lamas, R. [et al.] Introducción a la informática educativa. Pinar del Río: Universidad Hermanos Saiz, 2000. 151 p.

<sup>9</sup> Pere Marqués. El software educativo. Barcelona: Universidad Autónoma, 2003. – p. 56-72.

- Constructores: Son programas que tienen un entorno programable. Facilitan a los usuarios elementos simples con los cuales pueden construir elementos más complejos o entornos. De esta manera potencian el aprendizaje heurístico y, de acuerdo con las teorías cognitivistas, facilitan a los alumnos la construcción de sus propios aprendizajes, que surgirán a través de la reflexión que realizarán al diseñar programas y comprobar inmediatamente, cuando los ejecuten, la relevancia de sus ideas. Se pueden distinguir dos tipos de constructores: los específicos y los lenguajes de programación

A partir del 2004 en Cuba, surge una tendencia, a integrar en un mismo producto, todas o algunas de estas tipologías de software educativos. A este nuevo modelo de software se le ha denominado "HIPERENTORNO EDUCATIVO o HIPERENTORNO DE APRENDIZAJE, lo cual no es más que un sistema informático basado en tecnología hipermedia que contiene una mezcla de elementos representativos de diversas tipologías de software educativo"<sup>10</sup>.

### **Consideraciones metodológicas para desarrollar software educativo.**

Previo al proceso de elaboración de un software educativo resulta imprescindible tener en cuenta dos elementos de suma importancia:

- 1- Determinar la existencia de un problema educativo a resolver.
- 2- Asegurar que la computadora efectivamente posee ventajas cualitativas sobre otros medios educativos para resolver el problema.

Existen diversas metodologías que tratan este tema. Vamos a analizar una, que tiene su base en el denominado modelo de cascada y resume diferentes aspectos relacionados con el diseño de materiales instruccionales. Consta de cinco fases o etapas: análisis y requerimientos, diseño, construcción, prueba y mantenimiento.

Análisis y requerimientos: Resulta de vital importancia, ya que en esta etapa se realiza una descripción detallada del objeto de estudio y se elaboran todas las especificaciones, tanto las que se relacionan con la construcción, como con el uso del software. En esta etapa debe quedarnos claro entre otras cuestiones: la necesidad de elaborar el producto (problema pedagógico a resolver), la existencia de otros materiales didácticos, el público al que va dirigido, los objetivos pedagógicos que se pretenden cumplir, los contenidos a tratar y los medios para presentarlos, las herramientas que se utilizarán para el desarrollo, los equipos de trabajo que se conformarán, el hardware necesario tanto para realizadores como para usuarios, la factibilidad técnica y económica de su producción, las formas de distribución y la primera versión del cronograma de trabajo. El resultado más significativo de esta etapa es la escritura de la primera versión del guión.

Diseño: En esta etapa se obtendrá una información detallada de cómo estará estructurado el programa, como progresa o fluye a través de cualquier opción posible dentro de él, elegida por el usuario o por la computadora. Debe incluir, por tanto, un análisis de modularidad y jerarquía (la utilización de mapas conceptuales favorece el trabajo), y tener en cuenta todos los requerimientos del público al que está dirigido y ante todo el diseño de la interfaz de cada una de las pantallas. Aquí se define la organización interna del producto (directorios, archivos, etc.). También debe quedar definido el protocolo de pruebas que se empleará.

Construcción: Aquí se cumplen dos tareas de singular importancia: la obtención y edición de todos los medios que serán empleados y la programación, es decir, la codificación de los módulos definidos con anterioridad. Al final de esta fase se debe obtener un código claro y documentado, así como trabajar por la utilización de herramientas y bibliotecas comunes. Si además de los sistemas de ayuda que se programen para asistir al usuario durante la ejecución del software se decide incluir otra documentación, manual o recomendaciones, deben ser escritas en esta etapa.

Prueba: Es necesaria una comprobación sistemática para buscar los posibles errores; se debe velar por el cumplimiento satisfactorio de los objetivos seleccionados con la confiabilidad del software desde los puntos de vista conceptual, de la utilización y de la representación o codificación. En la evaluación sistemática del prototipo y del producto final, deben participar

---

<sup>10</sup> Los softwares educativos. Documento digital. MINED.

no solo el colectivo de realización sino también otros expertos en informática educativa y de la materia en cuestión, además de una representación del público al que está dirigido el software.

Mantenimiento: La correcta utilización de una metodología en el desarrollo de un software posibilita el mantenimiento efectivo de éste. Se hace necesario actualizar los comentarios del código y la documentación correspondiente para hacer cualquier modificación que garantice la competitividad del producto. Un registro de usuarios permite obtener, de forma real, un análisis riguroso de dificultades y errores en el software, así como de sus aciertos.

En las etapas mencionadas con anterioridad, resulta muy importante, que el jefe de proyecto vele porque se cree y mantenga la documentación mínima, así como la comunicación entre los distintos miembros de los equipos de trabajo.

Hacer un software educativo no es solo una tarea de ingenieros, sino la extrapolación en el ámbito digital de lo que un docente hace diariamente: crear materiales educativos, sólo que en este caso, estos materiales serán utilizados en un contexto específico. Es sumamente importante para ello tener en cuenta en el proceso de producción los aspectos pedagógico, informático y comunicativo, para garantizar la realización de un producto de calidad.

## **Bibliografía**

- Bauza, G. El guión multimedia. Anaya Multimedia. Madrid, 1997.
- Booch G. Software Architecture and the UML. Disponible en: <http://www.rational.com/uml>
- Fernández, Mariano. Modelos de desarrollo de software educativo. Disponible en: <http://www.sadpro.ucv.ve/agenda/online/vol6n2/a16.htm>
- González, J: Multimedia en educación. Centro de estudios de software para la enseñanza., ISPEJV, La Habana, 1995.
- Introducción a la informática educativa / R. Rodríguez Lamas, R.... [et al.] . -- Pinar del Río: Universidad Hermanos Saíz, 2000.
- Jacobson, I. Applying UML in The Unified Process. Rational Software. Disponible en: <http://www.rational.com/uml>
- Labañino C y Toro, M. Multimedia para la Educación. Editorial Pueblo y Educación. La Habana. 2001.
- El Software educativo. Su influencia en la escuela cubana. Disponible en: <http://www.monografias.com>.
- Pere Marqués. El software educativo. – Barcelona: Universidad Autónoma, 2003.
- Sánchez J. Construyendo y Aprendiendo con el Computador. – Barcelona: /s.n. /, 2006.
- Usandizaga, I y Otros. Nuevos Sistemas Hipermedia Educativa. Revista de la asociación para el desarrollo de la Informática en la educación, 1995.
- Valencia, Maria Eugenia. Un método de desarrollo de aplicaciones educativas hipermedia. Disponible en: <http://www.tise.cl/archivos/tise97/trabajos/trabajo2/index.htm>
- Zavala R. Diseño de un Sistema de Información Geográfica sobre internet. Tesis de Maestría en Ciencias de la Computación. Universidad Autónoma Metropolitana-Azcapotzalco. México, D.F. 2000 Disponible en: <http://www.angelfire.com/scifi/jzavalar/apuntes/IngSoftware.html>