



Polibits

ISSN: 1870-9044

polibits@nlp.cic.ipn.mx

Instituto Politécnico Nacional

México

Fuenlabrada Velázquez, Sergio; Miranda Chávez, Edna Martha; Sesma Martínez, Mario Alberto

Ingeniería de Software: Flexibilidad en el Diseño

Polibits, núm. 33, 2006, pp. 8-10

Instituto Politécnico Nacional

Distrito Federal, México

Disponible en: <http://www.redalyc.org/articulo.oa?id=402640446006>

- ▶ Cómo citar el artículo
- ▶ Número completo
- ▶ Más información del artículo
- ▶ Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Ingeniería de Software: Flexibilidad en el Diseño

MCC. Sergio Fuenlabrada Velázquez
MSI Edna Martha Miranda Chávez
Ing. Mario Alberto Sesma Martínez
Profesores de la UPIICSA-IPN

El presente trabajo surge del proyecto de investigación *Desarrollo de Software con Interfaces Parametrizables de Fácil Utilización*, con clave CGPI2003/1673, registrado en la Coordinación General de Posgrado e Investigación del Instituto Politécnico Nacional.

INTRODUCCIÓN

El desarrollo de un país se mide de acuerdo con el nivel educativo de sus habitantes, ya que éste permite incrementar el bienestar social, las expectativas de vida y fomentar las actividades creativas e innovadoras. El sistema educativo de cada país es diferente, pues responde a su propio contexto, situación social, política y económica; sin embargo, hay elementos comunes que permiten compararlos entre sí, como son los niveles de calidad, equidad, pertenencia, eficiencia, relevancia, vigencia, congruencia, suficiencia, viabilidad y participación de los diferentes sectores.

El modelo educativo está integrado por un conjunto de actividades dirigidas tanto a la formación de profesionales y personas con alto grado de instrucción, como al desa-

rrollo del conocimiento humanístico, científico y tecnológico, agrupadas en una unidad programática con la denominación de carrera profesional, programa de posgrado o programa de investigación. Los modelos han sido objeto de muchas investigaciones y propuestas, con las más diversas tendencias, mismas que no se tocarán a detalle en el presente trabajo por corresponder propiamente a especialistas en educación. El enfoque con que se tratará el tema es el de cómo las tecnologías de información pueden apoyar a un modelo educativo.

HIPÓTESIS Y DESARROLLO

La integración del software como herramienta para el desarrollo de las actividades humanas ha llegado a un punto tal, que la necesidad de una adaptación más rápida a los requerimientos cambiantes del entorno de cualquier organización es fundamental.

La tecnología actual influye y condiciona la forma de educar, pero al mismo tiempo, nosotros podemos condicionar a la tecnología. Lo importante es dejar de hablar de hardware y software, e ir más allá; analizar las diferentes alternativas que la tecnología ofrece y relacionarlas con el tipo de educación que se pretende dar a los alumnos.

Las organizaciones requieren Sistemas de Información dinámicos, que modifiquen su comportamiento de acuerdo al contexto y fomenten el aprendizaje, por lo que se necesita establecer una estrategia que produzca software altamente flexible.

Para lograr este dinamismo se debe considerar la estructura y organización del software a construir, establecer los requerimientos a satisfacer y tener una estrategia metodológica tal como la Ingeniería de Software para garantizar la calidad esperada.

La presente investigación tiene como base la hipótesis de que un modelo educativo apoyado en tecnologías de información puede incidir en la mejora de la capacidad de lectura, lógica matemática y la capacidad motriz fina y gruesa. Esto es importante en casos como los niños con síndrome de Down, quienes tienen como característica una disminución en su capacidad tanto intelectual como física; además, su percepción del mundo es diferente y en la mayoría de los casos particular.

La comprobación de esta hipótesis requirió establecer una estrategia para generar software altamente parametrizable, con flexibilidad tal que permita *calibrarlo* de acuerdo a las necesidades particulares y personalidad del usuario final; asumimos como parametrizable la capacidad de

modificar el comportamiento de los principales elementos del software: información y procesos.

Para aplicar las tecnologías de la información en una institución educativa, se requiere algo más que comprar computadoras nuevas y crear un sitio de red. El éxito del uso de la tecnología en el proceso enseñanza-aprendizaje depende en mucho de la capacidad de introducir cambios en la forma de visualizar su aplicación por el personal docente, a fin de desarrollar e implantar planes de estudio concretos, innovadores y con visión a futuro, tomando en cuenta no sólo los cambios producidos en la tecnología, sino también los cambios en la sociedad, para proporcionar a los egresados conocimientos que les permitan colocarse y desarrollarse profesionalmente.

El proceso para la generación, actualización y difusión de programas de estudio se consideró altamente factible para verificar la flexibilidad del software. Un punto que se detectó en el desarrollo del proyecto de investigación fue la necesidad de sistematizar la aplicación de cuestionarios, por lo cual se determinó construir un prototipo que permitiera facilitar este proceso.

El desarrollo de prototipos de software requiere utilizar una estrategia que permita determinar su nivel de adecuación y cumplimiento de los requerimientos establecidos, lo que hace necesario el manejo de un catálogo de versiones que permiten conocer este nivel.

MÉTODOS EXPERIMENTALES

Para articular el desarrollo de los prototipos es necesario establecer algunas consideraciones conceptuales, mismas que se incluyen a continua-

ción. Un sistema es un conjunto de elementos que están interrelacionados con un objetivo común, los cuales tienen una organización y una estructura; dicha organización permite:

- a. Establecer un conjunto de relaciones entre los componentes del sistema.
- b. Determinar la participación de los componentes en la contribución total de la unidad sistema.
- c. Delinear su forma en cualquier instante dado, la cual deberá conservarse a pesar de los cambios.
- d. Establecer una identidad en propiedad como unidad y un marco de referencia.

Por su parte la estructura de los sistemas permite:

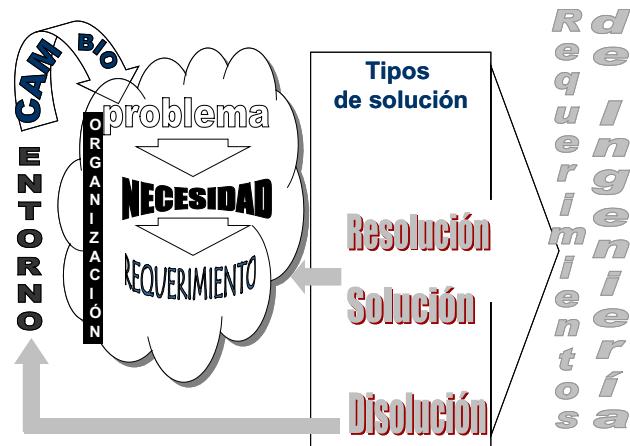
- a. La presencia de interacciones en los componentes, todo esto en un espacio proporcionado.
- b. Determinar el espacio de todas las posibles clases de iteración y que puedan ser afectadas por el entorno.
- c. Que el sistema pueda cambiar de estructura sin perder su identidad, siempre y cuando su organización se conserve.

El manejo de la información se puede realizar considerando tres enfoques diferentes: el sintáctico, el semántico y el pragmático. Por su parte, el procesamiento puede ser: selectivo, de agregación y de cálculo.

ESTRUCTURA DE DISEÑO DE SOFTWARE FLEXIBLE

En la **Figura 1** se muestra la estructura del proceso de creación de software:

1. El entorno modifica al sistema (organización) a través del cambio.
2. El cambio provoca en la organización un problema, entendido éste como el obstáculo físico o lógico que evita lograr un objetivo.
3. El obstáculo provoca la necesidad de eliminarlo.
4. Esta necesidad y el contexto de la organización establecen los requerimientos.
5. Lo anterior lleva al Ingeniero de Software a buscar un tipo de solución al problema:
 - a. Disolución: Busca eliminar las causas (cambio) del problema.
 - b. Resolución: Resuelve los síntomas del problema.
 - c. Solución: Mantiene bajo control al problema.



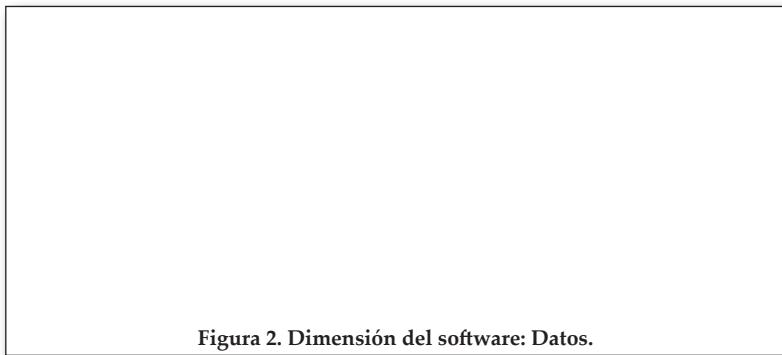


Figura 2. Dimensión del software: Datos.

6. Una vez establecida esta situación se determinan los requerimientos funcionales y no funcionales.
7. Se identifican los elementos que forman el software:
 - a. **Los datos.** Establecen la estructura, considerando cuatro bases de datos, como lo muestra la **Figura 2.**: los datos, los parámetros, las reglas de negocio y el conocimiento.
 - b. **Los procesos.** Establecen el comportamiento del software.
 - c. **El tiempo.** Indica cuando sucede cada uno de los elementos del software.
2. Cuestionario en Red, que sistematiza la gestión de cuestionarios desde su creación, aplicación y evaluación de resultados; permite implementar cualquier tipo de cuestionarios en red. Prototipo de software versión 1.1.
3. **Mi amigo Down**, que integra un modelo educativo que induce la capacidad de lectura y lógica matemática. Se señala como versión 0, debido a que no se ha podido concretar alguna prueba de campo durante este periodo, que permita evaluar la validez del prototipo. Prototipo de software educativo versión 0.

CONCLUSIONES

En conclusión, para poder comprobar el objetivo de demostrar que a través del uso del software con interfaces parametrizables y de fácil calibración se ayuda al desarrollo de las personas, se crearon tres prototipos:

1. **Sistema de Programas de Estudio**, que automatiza el proceso de generación, actualización y difusión de los programas de estudio de cualquier escuela de nivel superior. Prototipo de software versión 1.

REFERENCIAS

- [1] PFleeger, Shari Lawarence; *Ingeniería de software, teoría y práctica*; Prentice Hall, 2002.
- [2] Sommerville, Ian; *Ingeniería de Software*, Pearson; 2001.
- [3] Acholes, Checkland; *La metodología de los sistemas suaves en acción*; Noriega Editores, 1994.
- [4] Reporte del proyecto de investigación, *Desarrollo de software con interfaces parametrizables*

de fácil utilización con clave CGPI2003/1673, registrado en la Coordinación General de Posgrado e Investigación del Instituto Politécnico Nacional, 2004.