

## **Test Plan to Verify and Validate Learning Objects**

### **Plan de Pruebas para Verificar y Validar Objetos de Aprendizaje**

**Edgar Serna M.\***

Instituto Tecnológico Metropolitano ITM, Medellín, Colombia. *edgarserna@itm.edu.co*

**Carlos Castro C.**

Universidad de San Buenaventura, Medellín, Colombia. *carlos.castro@usbmed.edu.co*

**Ricardo Botero T.**

Tecnológico de Antioquia, Medellín, Colombia. *rbotero@tdea.edu.co*

#### **ABSTRACT**

The Verification and Validation Techniques —V&V— that are used in software engineering whose functions are to verify that the products are properly develop and validate the specification. This paper proposes the use of these techniques in the development lifecycle of Learning Objects. The proposal is structured in a test plan implementation strategy which corresponds with nature and with the skills and training objectives of the object being designed. In addition, the test plan was based on a collaborative approach that integrally involves the software testing, the training methodology and instructional design. The aim is to develop a series of test activities for sub-products of each of the SEDLO lifecycle phases, structured incrementally until to achieve the guidelines V&V for the specified product.

**Keywords:** SEDLO, Software Engineering, Test execution, Validation, Verification.

#### **RESUMEN**

Las técnicas de Verificación y Validación —V&V— que se utilizan en la Ingeniería de Software tienen como función verificar que los productos se desarrollen correctamente y validar la especificación. En este trabajo se propone el uso de estas técnicas en el ciclo de vida del desarrollo de Objetos de Aprendizaje. La propuesta se estructura en un plan de pruebas cuya estrategia de aplicación corresponde con la naturaleza y con las competencias y objetivos formativos del objeto que se diseña. Además, el plan de pruebas se sustenta en un enfoque colaborativo que involucra integralmente a las pruebas del software, a la metodología de formación y al diseño instruccional. El objetivo es desarrollar una serie de actividades de prueba para los sub-productos de cada una de las fases del ciclo de vida ISDOA, estructuradas de forma incremental hasta lograr los lineamientos de V&V para el producto especificado.

**Palabras clave:** Ejecución de pruebas, Ingeniería de Software, ISDOA, Validación, Verificación.

## **1. INTRODUCCIÓN**

En este trabajo se presenta una propuesta de plan de pruebas para realizar las actividades de V&V en el desarrollo de objetos de aprendizaje —OA—, la cual se basa en la Ingeniería de Software para Desarrollar Objetos de Aprendizaje (ISDOA) —SEDLO, por sus siglas en inglés— (Serna, Castro and Botero, 2011). El objetivo es proponer una serie de actividades de prueba para los sub-productos de cada una de las fases del ciclo de vida, estructuradas de forma incremental, hasta lograr los lineamientos de V&V, asegurar el correcto desarrollo del OA, y que el producto final sea el esperado por el cliente.

El ciclo de vida para OA que propone ISDOA se soporta sobre dos pilares fundamentales: 1) el plan de pruebas y 2) la evaluación de calidad; además, considera que el primero se debe ejecutar de forma incremental y el segundo de forma interactiva, y que ambos se deben realizar de forma paralela a las demás fases. En este artículo se detallan las técnicas de V&V necesarias para llevar a cabo ese plan de pruebas.

El éxito de los modelos formativos, como el modelo aula-profesor, el modelo de estudio independiente, la formación colaborativa y la formación a distancia radica sustancialmente en la construcción cuidadosa del material didáctico en que se soportan. Ese material lo desarrollan grupos especializados, y aplicando procesos de Ingeniería de Software con el objetivo de hacer viable la estrategia del *E-Learning* a través de los OA (McCrea, Gay and Bacon, 2000). El *E-Learning* se puede definir como la adquisición y uso de conocimiento distribuido y facilitado principalmente por medios electrónicos, y de acuerdo con Stokes (1999), se trata de un "aula sin muros"; por otro lado, *Objeto de Aprendizaje* se define como cualquier recurso digital que puede ser reutilizado para apoyar procesos formativos (Serna, Castro and Botero, 2011).

Actualmente, el *E-Learning* y los OA se convierten en una tendencia en la que converge la web y la formación a todos los niveles (Learframe, 2000), lo que implica nuevas técnicas de comunicación a través de redes informáticas, de herramientas multimedia, de motores de búsqueda y de librerías electrónicas, con un sinfín de posibilidades. Pero es común que el desarrollo de OA carezca de un adecuado plan de pruebas, para Verificar y Validar que el producto final cumpla con lo especificado. Como lo señala Ubell (2000), desde mediados de 1900 las escuelas por correspondencia en Europa formaban, a través de correo, en lenguas extranjeras y en taquigrafía, y hoy, las organizaciones empiezan a aplicar el concepto de empresa haciendo énfasis en el *E-Learning* y utilizando, de forma interactiva, proveedores de contenidos, herramientas de creación, materiales de formación, portales, sistemas de distribución y soluciones integradas (Barron, 2000), y en horas y lugares independientes y totalmente en línea. En esta situación es esencial que los procesos de ingeniería, aplicados para desarrollar OA, sean compatibles con la situación específica del logro instruccional, y con la tecnología que requiere el sistema de distribución. Se necesitan objetos que le proporcionen al usuario suficiente reusabilidad e interactividad pero, sustancialmente, que sean de alta calidad.

Tradicionalmente, la evaluación de los OA se realiza al final del proceso, destacándose el papel de los computadores y de los evaluadores profesionales, sin embargo, no suele estar disponible una descripción detallada de métodos de prueba a la medida del *E-Learning*. Una forma de asegurar la formación individualizada consiste en aplicar *evaluaciones* frecuentes en partes intrínsecas del proceso formativo, aunque no se deben confundir con las *pruebas* en el ciclo de vida del producto. Para asegurar el logro instruccional utilizando OA es necesario que se desarrollen adecuadamente y aplicando técnicas de V&V a través de su ciclo de vida. Esto haría que conceptos como pruebas, evaluación y formación no se separen, sino que se combinen estrechamente en los procesos. Dado que el costo de desarrollar unidades de formación altamente interactivas puede ser muy alto, cobra vigencia la necesidad de un plan de pruebas estructurado (Serna, Castro and Botero, 2011) que se puede aplicar paralelamente al ciclo de vida de la Ingeniería de Software, con el objetivo de garantizar que el producto satisface adecuadamente la especificación.

El resto de este documento se estructura de la siguiente forma: en la sección 2, se definen los conceptos clave necesarios para la comprensión temática; en la sección 3, se describe el plan de pruebas; en la sección 4, se introduce el concepto de Verificación y Validación de OA; en la sección 5, se describe la metodología de plan de pruebas propuesta para OA; en la sección 6 se proponen los trabajos futuros, y las conclusiones se presentan en la sección 7.

## 2. ACERCAMIENTO CONCEPTUAL

- *Formación*: es un cambio en el rendimiento de una persona, que se produce de acuerdo con una condición de práctica (Rajadell, 2001).
- *Evaluación*: es un proceso de recopilación e interpretación de datos, con el objetivo de determinar cierto grado de eficacia y eficiencia, con el cual se puede valorar cuantitativa y cualitativamente el avance y los resultados de un proceso (Elola and Toranzos, 2000).

- *Prueba de software*: es una actividad en la que se ejecuta, bajo condiciones específicas y con valores de entrada específicos, un sistema o componente para obtener valores de salida igualmente específicos (Serna and Arango, 2010).
- *Validación*: es un proceso en el que se prueba un producto software para determinar si satisface los requisitos especificados. Responde a la pregunta: ¿estamos desarrollando el producto correcto? (Patton, 2005).
- *Verificación*: es un proceso en el que se prueba un producto software para determinar si el proceso de desarrollo se está ejecutando adecuadamente. Responde a la pregunta: ¿estamos desarrollando correctamente el producto? (Patton, 2005).

Aunque en la literatura de OA se utiliza frecuentemente el término *evaluación* indistintamente como prueba, es preciso aclarar que mientras la primera se refiere a evaluar si el OA cumple con los objetivos, las competencias y las metas formativas definidas previamente, mientras que en la segunda se aplican técnicas de V&V para determinar si el OA se desarrolla adecuadamente, y si el producto final es el que espera el cliente, de acuerdo con los requisitos previamente especificados.

### 3. EL PLAN DE PRUEBAS

Un plan de pruebas detalla un método sistemático para probar un sistema o producto software, y documenta la estrategia que se utilizará para verificar y validar que cumple con las especificaciones de diseño y otros requisitos. Las fases del plan de pruebas son actividades de Verificación y Validación, y de acuerdo con Edward Kit (2011), la segunda se puede dividir en pruebas de bajo nivel y pruebas de alto nivel, dependiendo de qué partes del producto se prueban. Las de bajo nivel prueban los componentes individuales, y para lograrlo es necesario conocer la estructura interna del programa; usualmente se realizan en compañía del equipo de desarrollo. Los tipos de prueba a este nivel son las pruebas de unidad y las pruebas de integración. Las de alto nivel prueban el producto completo, y las aplica el equipo de pruebas. Los tipos de prueba a este nivel son: pruebas de usabilidad, pruebas funcionales, pruebas del sistema y pruebas de aceptación.

El formato del documento del plan de pruebas puede ser tan variado como los productos y las organizaciones en las que se aplica, aunque existen tres elementos importantes que debe contener: la cobertura de la prueba, los métodos de prueba y las responsabilidades de la prueba. La *cobertura* de la prueba indica qué requisitos se verificarán y en qué fases del ciclo de vida del producto; se deriva de las especificaciones de diseño y de requisitos como los estándares de seguridad o los códigos de regulación, donde cada requisito o especificación del diseño deberá tener uno o más formas correspondientes en la Verificación. La cobertura se puede superponer para diferentes etapas del ciclo de vida, pero necesariamente no será exactamente la misma para todas las fases. Por ejemplo, algunos requisitos se pueden verificar durante la prueba del diseño, pero pueden no verificarse durante la prueba de aceptación. Debido a que el producto debe estar diseñado antes de aplicar la prueba, la cobertura también se retroalimenta en el proceso de diseño. Los *métodos de prueba* se determinan a través de estándares, organismos reguladores o acuerdos contractuales, e indican cómo se implementará la cobertura de la prueba. También especifican los equipos de prueba que se utilizarán, y establecen los criterios de pasa/error. Los métodos de prueba utilizados para verificar los requisitos de diseño pueden ir desde medidas muy simples, como la inspección visual, hasta la elaboración de procedimientos de prueba que se documentan por separado. Las *responsabilidades de la prueba* describen qué o quién aplicará los métodos de prueba en cada fase del ciclo de vida del producto. Esto le permite a la organización de la prueba planear, planificar, adquirir o desarrollar equipos de prueba y otros recursos, necesarios para implementar las técnicas de las que cada uno es responsable. Estas responsabilidades también incluyen qué datos se recogerán y cómo se almacenarán y reportarán. Uno de los resultados de un plan de prueba exitoso debe ser un registro o reporte de la Verificación de todas las especificaciones y requisitos de diseño, según lo convenido por las partes interesadas.

### 4. VALIDACIÓN Y VERIFICACIÓN DE OA

El reporte Learning and Skill Council's Distributed and Electronic Learning Group DELG (2002) indica que la inversión en procesos formativos distribuidos con apoyo electrónico debe ser racionalizada, principalmente

debido a que hay que esperar si el dinamismo actual de las TIC en los procesos formativos es una moda, o realmente logrará alcanzar la efectividad y el cubrimiento que promete. Para hacer realidad esas promesas, las futuras implementaciones de OA deben garantizar que el material se entregue a tiempo, con buena calidad y con buen diseño (Learning and Skill Council's, 2002). El papel de los evaluadores será determinar el grado en que se logran los objetivos y competencias formativas, y garantizar la calidad de los contenidos, para lo cual deberán evaluar parámetros como facilidad de uso y sostenibilidad, eficacia pedagógica y conformidad con el área formativa que cubre el objeto. Además, se debe garantizar la conveniencia de la reutilización, la visibilidad y la distribución. Para lograrlo, será necesario implementar técnicas de V&V en el ciclo de vida del desarrollo del OA.

Las técnicas de V&V se utilizan para asegurar la calidad de los productos software, para lo que, comúnmente, se aplican métodos como revisiones, análisis y técnicas de prueba. Algunas actividades de Verificación son: verificar los requisitos, verificar el diseño funcional e individual y verificar el código. En la Ingeniería de Software tradicional para OA los requisitos se expresan como una especificación del sistema, y en la fase de diseño y desarrollo las inconsistencias se resuelven de acuerdo con esa especificación. Los estándares para la especificación de requisitos los establece el grupo de ingenieros de software que desarrolla el objeto, además, a ese grupo se deben vincular asesores pedagógicos y didácticos, quienes participarán en los procesos de diseño, revisión, inspección y logro de competencias, y evaluadores que determinarán el logro de los objetivos formativos del objeto.

La Validación se realiza tradicionalmente aplicando diversos métodos y técnicas de prueba al momento de la implementación, es decir, luego del diseño instruccional y del desarrollo —donde participan estudiantes y profesores—. Aunque los desarrolladores utilizan algunas técnicas de prueba en la Validación dinámica, también es necesario aplicar la Verificación estática al final de cada fase del ciclo de vida. ISDOA propone que el plan de pruebas para desarrollar OA se implemente como una fase paralela al ciclo de vida, convirtiéndola en un pilar sobre el que se soporta todo el proceso de desarrollo, como se observa en la Figura 1.

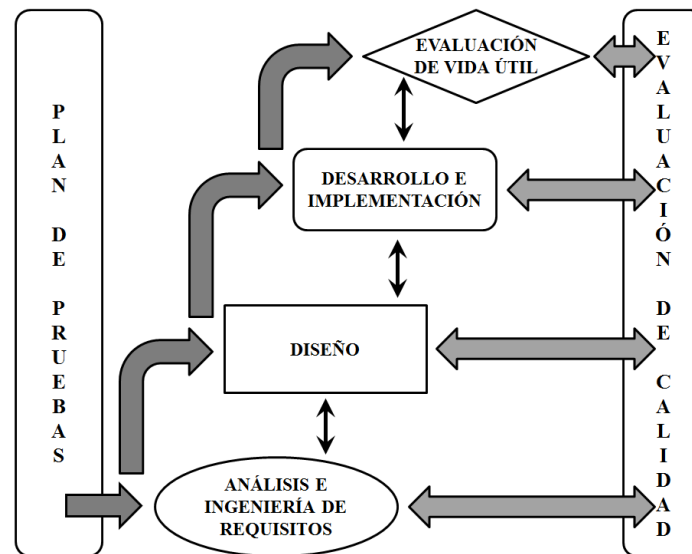


Figura 1. Ciclo de vida ISDOA (Serna, Castro and Botero, 2011)

## 5. PROPUESTA DE PLAN DE PRUEBAS PARA OA

Las aplicaciones de OA presentan un desafío para el proceso de diseño, debido a que deben responder a tiempo, con información precisa y facilidad de uso, para evitar que el público objetivo abandone el objeto o cambie de proveedor de aplicaciones (Nguyen, 2001). Esto es de suma importancia, debido a que los ciclos de tiempo de entrega son cortos, la tecnología está en constante evolución y el número de usuarios es alto, lo que puede llevar a una incapacidad para controlar el entorno de aplicación. Para el plan de pruebas, el desafío lo constituyen

cuestiones críticas como el manejo de consultas, el volumen de usuarios en términos de asignaciones, la seguridad de las cuentas de usuario, la integralidad del objeto, y la Verificación y Validación funcional y estructural.

Las actividades de V&V propuesta en el plan de pruebas para OA se ilustran en la Figura 2. Se propone un plan para desarrollar de forma incremental y paralelamente al ciclo de vida del OA, tal como lo plantea ISDOA. Cada una de estas actividades se ejecuta sobre el sub-producto derivado de cada fase del ciclo, y dado que los OA tienen objetivos formativos, naturaleza y competencias diferentes, el probador puede decidir utilizar la Verificación y Validación completa o la parcial, las pruebas de unidad o las integrales, o las funcionales o las estructurales. Esta decisión también depende del perfil del público objetivo.

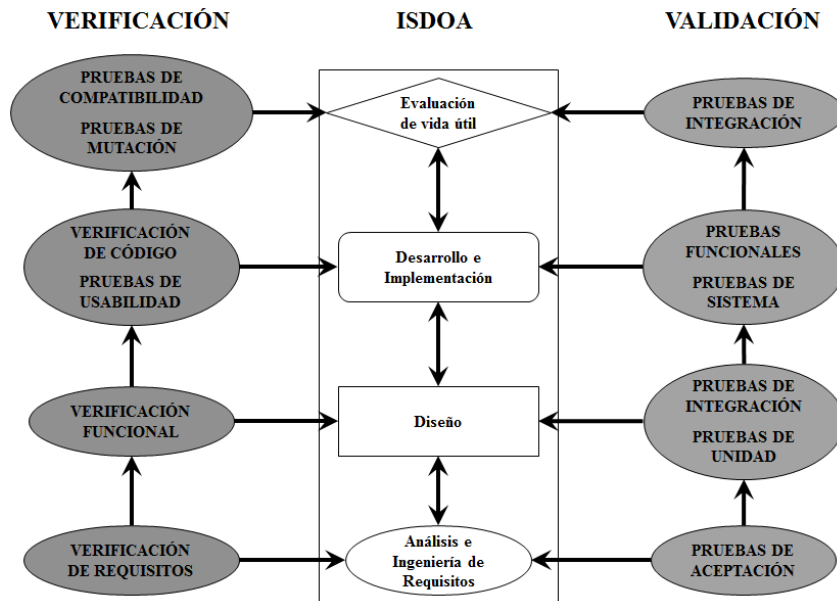


Figura 2. Actividades de V&V del plan de pruebas para OA

### 5.1 ANÁLISIS E INGENIERÍA DE REQUISITOS

- *Verificación de requisitos.* La gestión de requisitos es un enfoque sistémico para elicitar, organizar, documentar, modelar y especificar los requisitos de un sistema (Dustin, Rashka and Paul, 2000), que tradicionalmente se ejecuta como una fase separada del desarrollo de software. Para el desarrollo de OA la Verificación de requisitos consiste en estructurar una matriz de indicadores para verificar la comprensión y formulación de las necesidades del cliente. Es una actividad que realizan conjuntamente el ingeniero de requisitos y el cliente.
- *Pruebas de aceptación.* Luego de verificar los requisitos, el equipo de desarrollo diseña un prototipo no funcional que representa una propuesta de solución a la especificación verificada. Las pruebas de aceptación se aplican para determinar si el prototipo satisface los criterios de aceptación, y para que el cliente decida entre aceptar o no la propuesta de solución. El proceso de la prueba consiste en realizar discusiones para validar la distribución y el pre-diseño, antes de pasar al diseño y al desarrollo. Mediante interacciones hombre-máquina el cliente valida si el prototipo satisface sus necesidades, las características necesarias de función y las limitaciones del sistema especificado (Hsia et al., 1994). Esta actividad la realiza el equipo de trabajo y el cliente con la asesoría de especialistas en didáctica y pedagogía, y con la participación de especialistas en la temática que cubre el OA.

### 5.2 DISEÑO

- *Verificación funcional.* Se realiza después que el cliente verifica los requisitos especificados y que valida el prototipo, y contiene información acerca de la funcionalidad del prototipo. Se relaciona con las pruebas de

usabilidad porque la información acerca del uso del sistema se basa siempre en su funcionalidad. La prueba la realiza el equipo de trabajo sobre un prototipo funcional del OA, y consiste en aplicar una matriz de indicadores, para verificar la validación que el cliente hizo previamente y la evaluación de los especialistas.

- *Pruebas de unidad.* Son un tipo de pruebas de bajo nivel que requieren el conocimiento de la estructura interna del código; su objetivo es asegurar que cada módulo del prototipo funcione de acuerdo con los requisitos especificados, tanto de forma integral como separada. La propuesta consiste en diseñar contratos (Meyer, 1992) que debe cumplir cada módulo, con el objetivo de encontrar los errores que puedan existir en la fase de desarrollo. El equipo de trabajo diseña los contratos de acuerdo con la especificación de requisitos, para que el diseñador valide que los módulos del objeto los cumplen, y para que los desarrolladores validen que la sintaxis y la gramática del código también lo hacen.
- *Pruebas de integración.* Son un tipo de pruebas de bajo nivel que se aplican para validar que los módulos del sistema trabajan bien cuando se integran en componentes. El objetivo principal de estas pruebas es descubrir errores en las interfaces entre los componentes. En esta propuesta, estas pruebas se aplican de forma incremental, es decir, a medida que se integran los módulos. La cuestión es no repetir las pruebas unitarias, sino reunir las características que sean comunes a los módulos, y que no se probaron en las pruebas unitarias. Luego de las pruebas, el producto final de esta fase del ciclo de vida es el prototipo funcional que se implementará en la siguiente fase.

### 5.3 DESARROLLO E IMPLEMENTACIÓN

- *Verificación de código.* Con los resultados de la matriz de indicadores de la Verificación funcional se hace un análisis a la semántica y a la sintaxis del código de programación. Mediante un recorrido a las líneas de código el equipo de desarrollo hace seguimiento a la funcionalidad del prototipo.
- *Pruebas de usabilidad.* La usabilidad se refiere a la facilidad con que un usuario puede operar, preparar entradas e interpretar salidas desde y hacia el sistema y sus componentes. Estas pruebas se centran en probar la usabilidad del OA con la participación de usuarios finales. El proceso consiste en verificar: 1) que los prototipos, validados previamente, se han desarrollado e implementado correctamente, con el objetivo de averiguar si se determinaron las expectativas del cliente y de los especialistas, 2) si el contexto de uso es adecuado para el público objetivo, 3) si el diseño es el esperado y si las interfaces reflejan los requisitos especificados y 4) si contiene información obsoleta, colores no estandarizados, páginas de desplazamiento largo, páginas huérfanas o utilización excesiva de maquetado, entre otras. Para que los usuarios efectúen las pruebas de funcionalidad se propone aplicar listas heurísticas de prueba para evaluar aspectos como navegabilidad, lógica de la aplicación, sensibilidad al contexto, y otras más. El punto central de esta propuesta es que las pruebas de usabilidad identifiquen situaciones complejas, que un usuario puede generar al navegar en la interfaz del objeto o cuando utiliza un patrón de comportamiento estándar o no-estándar. Como respuesta a esas posibles situaciones, los resultados obtenidos se utilizan para modificar las interfaces.
- *Pruebas de sistema.* En estas pruebas se valida la integralidad funcional de los componentes del OA y con otros aplicativos en el sistema organizacional. Con el objetivo de validar integralmente al sistema se ejecutan luego de verificar la usabilidad del objeto. En las pruebas se valida el funcionamiento en línea, la respuesta del hardware, la compatibilidad con otras aplicaciones, las ventanas simultáneas, la respuesta del gestor de bases de datos, entre otras, y en su ejecución se utilizan diferentes tipos de casos de prueba. Por ejemplo, Myers (2004) define algunas categorías útiles: disposición para la prueba, pruebas de tensión, pruebas de usabilidad, pruebas de seguridad, pruebas de rendimiento, pruebas de almacenamiento, pruebas de configuración, pruebas de compatibilidad, pruebas de facilidad de instalación, pruebas de recuperación, pruebas de servicio, prueba de documentación, y prueba de procedimientos.
- *Pruebas funcionales.* Las pruebas funcionales ignoran la estructura interna del sistema y se centran exclusivamente en los resultados que éste arroja luego de procesar, bajo ciertas condiciones, las entradas del

usuario. Se aplican para asegurar que todas las funcionalidades implementadas responden de acuerdo con la especificación funcional definida (Serna and Arango, 2011). En esta propuesta, el objetivo de las pruebas funcionales es demostrar que el producto desarrollado es el correcto. Para lograr este objetivo se proponen técnicas de prueba como entradas aleatorias, particiones de equivalencia, análisis de valores límite, grafos causa-efecto, tablas de condiciones, entre otras. Las características que se deben probar son los formularios de inscripción, los niveles de aceptación, los valores enteros y decimales, el despliegue de datos, las búsquedas y respuestas a las evaluaciones del objeto, y otras que el proceso determine.

#### 5.4 EVALUACIÓN DE VIDA ÚTIL

- *Pruebas de mutación.* Estas pruebas se requieren para determinar la vida útil del OA, porque tienden a ser estáticos con respecto a los progresos y cambios en los esquemas formales de formación. Para ejecutarlas se necesita la participación de especialistas en didáctica y pedagogía, y en cada área que cubre el objeto, además de los estudiantes y profesores que los utilizan. El proceso consiste en verificar si pasa o no una serie de indicadores, con lo que se determina su actualización o retiro. Ejemplos de indicadores son: el público objetivo cambió, las dificultades formativas del público objetivo cambiaron, no se alcanzan las competencias formativas, el área de formación que cubre se fusionó con otras más recientes, entre otros.
- *Pruebas de compatibilidad.* Con el paso del tiempo los productos software tienden a sufrir cambios como respuesta al contexto en el que han funcionado, al mal uso por parte de los usuarios, o por la influencia de virus informáticos. Debido a que los OA tienen una vida útil relativamente corta, las pruebas de compatibilidad, al igual que las de mutación, se aplican para determinar su actualización o retiro. Estas pruebas las realizan los mismos actores de las pruebas de mutación con el objetivo de verificar si el producto, luego de los cambios que genera la evolución del software y el hardware, todavía es compatible con el contexto. El proceso consiste en verificar si aparecen nuevas exigencias en software o en hardware, si la arquitectura de software que lo sustenta es obsoleta, si el software y el hardware necesarios son obsoletos, si un nuevo producto para alcanzar las competencias formativas es más eficiente, y otras que el dinamismo del proceso pueda generar.
- *Pruebas de integración.* Estas pruebas son diferentes a las que se proponen en la fase de Diseño, debido a que el objetivo en esta fase es determinar si el OA conserva la integralidad entre la especificación, el logro de objetivos y las competencias formativas con las que fue diseñado. El proceso lo llevan a cabo los mismos actores de las otras pruebas de esta fase, y consiste en validar que el objeto todavía es el correcto. Se hace una Validación entre los objetivos y competencias que se especificaron para desarrollar el objeto y su cumplimiento actual. Si el OA presenta una alta desintegración debe retirarse, pero si esa desintegración es mediana o baja se opta por actualizarlo.

## 6. CONCLUSIONES

El objetivo de los OA es realizar una interacción multi-medial personalizada de doble vía, lo que se logra con la inclusión de escenarios basados en problemas, estudio de casos interactivos, simulaciones de realidad virtual, e-libros, objetos de aprendizaje cortos, módulos, y proyectos. Debido a que puede existir centros de formación con micro-sistemas de carga, programas de reconocimiento de voz, pantallas de vídeo grandes y bases de datos colectivas, se deben desarrollar con la calidad suficiente como para optimizar todos estos recursos, por lo que el plan de pruebas es una cuestión crítica.

En este artículo se presenta una propuesta de plan de pruebas para objetos de aprendizaje, estructurada de forma incremental en el ciclo de vida propuesto por ISDOA. El objetivo es poder responder a las exigencias formativas de los estudiantes en este siglo, de forma tal que sus expectativas se cumplan, y para que no pierdan el interés rápidamente en sus procesos formativos. Hay que recordar que todos hacen parte de la Generación Digital, y que sus intenciones son diferentes a las de otras generaciones. Por esto es importante que los OA se desarrollen adecuadamente, y con criterios de prueba claros y orientados al logro de los niveles de calidad y de seguridad necesarios.

El asunto más importante acerca del futuro de las pruebas de OA es el enfoque en la efectividad y en la economía. Además, si probar OA es una cuestión crítica para la supervivencia de las instituciones u organizaciones que desarrollan E-Learning, entonces el objetivo final deberá ser la automatización total del plan de pruebas. Este es un trabajo futuro que ya vienen desarrollando algunos integrantes de la comunidad, pero que necesitan mayor participación debido que los desarrollos tecnológicos son abundantes y continuos, a la vez que los estudiantes tienen mayor contacto con los mismos. Esta generación busca satisfacer su intencionalidad formativa de otras formas, muy diferentes a como lo hacían las anteriores generaciones, por eso es importante pensar en que los OA deben pasar por estrictos procesos de prueba antes de ser llevados a alguno de los contextos.

## REFERENCIAS

- Barron, T. (2000). Thinking Thin: The Race for Thin-Client Synchronous E-Learning. <http://www.docstoc.com/docs/27141188/Thinking-Thin-The-Race-for-Thin-Client-Synchronous-E-Learning>, (Oct. 2013).
- Dustin, E., Rashka, J. and Paul, J. (2000). "Automated Software Testing. Introduction, Management, and Performance: Introduction, Management, and Performance." Boston: Addison-Wesley.
- Elola, N. and Toranzos, L.V. (2000). Evaluación Educativa: una aproximación conceptual. <http://www.oei.es/calidad2/luis2.pdf>, (Sept. 2013).
- Hsia, P., Gao, J., Samuel, J., Kung, D., Toyoshima, Y. and Chen, C. (1994). "Behavior-Based Acceptance Testing of Software Systems: A Formal Scenario Approach." *Proceedings of the Third International Conference on System Integration*. Brasil, pp. 293-298.
- Kit, E. (1995). "Software Testing In the Real World: Improving the Process." New York: Addison-Wesley Professional.
- Learframe (2000). Facts, Figures and Forces Behind eLearning. <http://learnframe.com/aboutelearning/elearningfacts.pdf>, (Sept. 2013).
- Learning and Skill Council's. (2002). Report of the Learning and Skills Council's Distributed and Electronic Learning Group. <http://readingroom.lsc.gov.uk/lsc/national/DistElectronicGroup3.pdf>, (Sept. 2013).
- McCrea, F., Gay, R.K. and Bacon, R. (2000). "Riding the big waves: A white paper on B2B E-Learning industry." Thomas Weisel Partners, San Francisco.
- Meyer, B. (1992). "Applying 'design by contract'." *Journal Computer*, vol. 25, No. 10, pp. 40-51.
- Myers, G. (2004). "The Art of Software Testing." USA: John Wiley & Sons.
- Nguyen, H.Q. (2001). "Testing Applications on the Web: Test planning for internet based systems." USA: John Wiley.
- Patton, R. (2005). "Software Testing." USA: Sams.
- Rajadell, P.N. (2001). "Los procesos formativos en el aula: Estrategias de enseñanza-aprendizaje." Editores: Sepúlveda, F. and Rajadell, P.N. *Didáctica general para psicopedagogos*, UNED, España, pp. 465-528.
- Serna, M.E. and Arango, I.F. (2010). "Critical analysis of proposals to generate test cases from use cases for functional testing." *Revista Avances en Sistemas e Informática*, Vol. 7, No. 2, pp.105-113.
- Serna, M.E. and Arango, I.F. (2011). "Software Testing: More than a Stage in the Life Cycle." *Revista de Ingeniería*, No. 35, pp. 34-40.
- Serna, M.E., Castro, C.C. and Botero, T.R. (2011). "SEDLO: Software Engineering for Developing Learning Objects." *Proceedings of the 6th Euro American Conference on Telematics and Information Systems*, Editor: Chagas, do N.R.P., Universidad Politécnica de Valencia, Spain, pp. 347-353.
- Stokes, S. (1999). "Preparing students to take online interactive courses." *The Internet and Higher Education*, Vol. 2, No. 2-3, pp. 161-169.
- Ubell, R. (2000). "Engineers turning to E-Learning." *IEEE Spectrum*, Vol. 37, No. 10, pp. 59-63.

## **Authorization and disclaimer**

*Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.*