

EL IMPACTO DE LA INGENIERÍA DE SOFTWARE EN EL DESARROLLO DE SISTEMAS INFORMÁTICOS UNIVERSITARIOS

Patricia Zavaleta Carrillo
Dámaris Pérez Cruz
José A. Pérez Rejón*

Resumen

En este trabajo se presentan los resultados obtenidos de la realización de un concurso de proyectos informáticos en los últimos dos años, que tiene como objetivo fomentar en los alumnos el interés por aplicar metodologías de software en el desarrollo de sistemas de información. El desarrollo de software es una tarea esencial en la actividad de cualquier profesional de la computación, el uso de una metodología de ingeniería de software es importante para asegurar la obtención de sistemas de información óptimos y que tengan garantizada su calidad y menores esfuerzos de mantenimiento.

Palabras claves: Ingeniería de software, metodologías, modelos, sistemas informáticos, ANIEI.

Introducción

La tecnología computacional ha avanzado y se ha expandido a tal grado que se ha convertido en una necesidad y su uso es vital para la supervivencia y competitividad de las empresas, ya sean locales, nacionales y, más aún, en grandes consorcios internacionales.

La industria del software es una de las tecnologías que más ha crecido en los últimos años, aun así, la mayoría de las aplicaciones desarrolladas corren de manera local y en menor porcentaje en ambientes *WEB*. De estas aplicaciones, los sistemas informáticos bancarios son los más aceptados por la población y, en menor proporción, aquellos elaborados para otras organizaciones (ayuntamientos, compras en línea, pagos de servicios, etcétera). En este contexto se espera que en los próximos años el desarrollo de aplicaciones *WEB* siga aumentando, así como la confianza en el manejo de aplicaciones con servicios de transacciones electrónicas.

Si nos remontamos a los orígenes del desarrollo de software, nos encontramos con aplicaciones que funcionaban a través switches electrónicos en donde la velocidad, secuencia de pasos y resultados dependían de la habilidad de los operadores para realizar encendido y apagado en el cuarto de máquina. El software ha evolucionado al grado de que hoy en día ejecutar una aplicación es tan sencillo como hacer clic sobre un archivo ejecutable, proporcionar la URL donde se hospeda una aplicación en un navegador o trabajar en grupo a través de aplicaciones distribuidas [1].

Para desarrollar software se requiere el uso de la Ingeniería de Software (IS), la cual se define como: 1) una disciplina de la ingeniería cuya meta es el desarrollo costeable de sistemas informáticos y que comprende todos los aspectos de la producción del software desde las etapas iniciales de la especificación, hasta el mantenimiento después de haber

entregado el sistema [2], y 2) una disciplina o área de la Informática o Ciencias de la Computación que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo [3].

En sus inicios, la IS tuvo fallas, pero éstas se debieron a que en los sistemas no se consideraron situaciones como el año bisiesto, el máximo de edad de una persona, incluso errores en líneas de código con consecuencia de gran impacto. Ya para el siglo XXI se tienen muchos avances, pero aún así el reto está en la migración de sistemas legados, en la incorporación y ejecución de múltiples plataformas, pero sobre todo en el desarrollo de aplicaciones rápidas de gran calidad, dado que el factor tiempo en los grandes corporativos es crítico.

De lo anterior entendemos que el software no se construye mágicamente, sino que se deben seguir ciertos métodos que garantizarán su vida útil y, si fuera necesario, su migración o integración a otros sistemas; sin embargo, esto dependerá de las habilidades del ingeniero en cada etapa. En IS, un método es un enfoque estructurado para el desarrollo de software cuyo propósito es facilitar la producción de software de alta calidad de una forma costeable, basado en la idea de modelos gráficos de desarrollo de un sistema y en el uso de estos modelos como un sistema de especificación o diseño [2]. Estos métodos van desde Métodos para el análisis estructurado (MetE), Métodos orientados a funciones, Métodos orientados a objetos (MetOO), hasta el Lenguaje de modelado unificado (UML).

En cuanto a los Modelos tenemos: Lineal secuencial (ModLS), Construcción de prototipos (ModCP), Desarrollo rápido (ModDR), Evolutivos de proceso del software (incremental, espiral, espiral WINWIN. Desarrollo concurrente), Basado en componentes, de Métodos formales. El modelo que se decida aplicar, será determinado de acuerdo a la naturaleza del proyecto y a las herramientas que se utilizarán para su desarrollo. Un método de desarrollo de software describe cómo modelar y construir un sistema de software de una forma confiable y reproducible, en general los métodos permiten la construcción de modelos desde elementos de modelado que constituyen los conceptos fundamentales para representar los sistemas o fenómenos.

* Docente en la Dependencia de Educación Superior Ciencias de la Información de la Universidad Autónoma del Carmen.

Contexto institucional

En la Dependencia de Educación Superior Área Ciencias de la Información (DES-DACI) de la Universidad Autónoma del Carmen (UNACAR), se ofrecen tres programas educativos (PE), Licenciatura en Informática (LI), Ingeniería en Sistemas Computacionales (ISC) e Ingeniería en Computación (ICO), con los perfiles A, B y D, respectivamente, que establece la Asociación Nacional de Instituciones de Educación en Informática (ANIEI), ver tabla 1. Al ser carreras del área de informática o computación, es necesario que los alumnos desarrollen software a lo largo de su formación profesional y para ello se les incorpora en su malla curricular, cursos o asignaturas del área de Programación e Ingeniería de Software (PIS), cuyos propósitos son: el aprendizaje de la programación, aprendizaje y aplicación de las estructuras de datos, el desarrollo de sistemas e implementación de otras aplicaciones específicas. Con estos cursos se espera que los alumnos desarrollen habilidades de análisis, especificación, diseño, desarrollo, evaluación, mantenimiento y pruebas de los sistemas de software, así como también el uso o generación de herramientas de soporte para el desarrollo de software asistida por computadora (herramientas CASE); sin embargo el grado de habilidad desarrollado dependerá del perfil profesional del alumno.

Objetivo general del PE de LI: “Formar profesionales de excelencia en el área de la informática que enfrenten los retos científicos y tecnológicos; que cuenten con una formación humanística y que estén comprometidos con la sociedad; Capaces de responder a las necesidades actuales y futuras de su entorno relativas al uso de la información; Competentes para analizar, diseñar y evaluar sistemas de información”.

Objetivo general del PE de ISC: “Formar profesionales de excelencia en el área de los sistemas computacionales que enfrenten los retos científicos y tecnológicos; que cuenten con una formación humanística y que estén comprometidos con la sociedad; Profesionales que sean capaces de responder a las necesidades actuales y futuras de su entorno mediante la promoción, coordinación y participación en proyectos de diseño, desarrollo y aplicación en materia de sistemas computacionales con características de productos terminados y competitivos mediante el análisis de situaciones, entornos y procesos productivos, así como la planeación, diseño, construcción, implementación y mantenimiento del software de calidad acorde a las necesidades de la organización”.

Objetivo general del PE de ICO: “Formar profesionales de excelencia en el área de los sistemas computacionales que enfrenten los retos científicos y tecnológicos; que cuenten con una formación humanística y que estén comprometidos con la sociedad; Profesionales capaces de

responder a las necesidades actuales y futuras de su entorno en materia de Computación; Aptos para promover, coordinar y participar en proyectos de diseño, desarrollo y aplicación de tecnologías computacionales acordes con las necesidades de su entorno profesional”.

Tabla 1. Porcentajes de áreas y perfiles de los PE de la universidad con respecto a ANIEI.

Áreas de conocimiento	ANIEI			UNACAR		
	LI (A)	ISC (B)	ICO (D)	LI	ISC	ICO
Entorno Social	25.7	20.0	10.0	27.3	19.5	9.5
Matemáticas	12.5	15.0	17.5	13.6	14.6	16.7
Arquitectura de Computadoras	7.5	7.5	17.5	6.8	7.3	19.0
Redes	7.5	10.0	12.5	6.8	9.8	16.7
Software de Base	7.5	7.5	12.5	6.8	7.3	9.5
Programación e Ingeniería de Software	17.5	17.5	17.5	15.9	19.5	16.7
Tratamiento de Información (TI)	12.5	15.0	5.0	13.6	14.6	4.8
Interacción Hombre-Máquina (IHC)	7.5	7.5	7.5	9.1	7.3	7.1

De los porcentajes totales del área de PIS por cada PE, se tienen cursos que impactan solo en IS, la tabla 2, muestra el número de cursos de esta área y su respectivo porcentaje.

Subdivisión de PIS	Cursos			Porcentaje		
	LI	ISC	ICO	LI	ISC	ICO
Programación	4	6	5	9.1	9.7	11.9
Ingeniería de Software	3	4	2	6.8	9.8	4.8
Total en PIS	7	10	7	15.9	19.5	16.7

Tabla 2. Cursos y porcentajes del área de PIS.

En la tabla 3, se listan los cursos que en su contenido contemplan la aplicación de la IS en el desarrollo de proyectos:

PE	Cursos
LI	Desarrollo de Sistemas de Información para la Oficina (OIS). Laboratorio de Análisis y Diseño Estructurado de Sistemas para la Toma de Decisiones (DSS). Laboratorio de Análisis y Diseño Orientado a Objetos de Sistemas de Información (ERP).
ISC	Ingeniería de Software Métricas y Calidad del Software Reingeniería de Software Administración de Proyectos de Software
ICO	Ingeniería de Software Ingeniería Inversa

Tabla 3. Cursos con contenido de IS.

1 www.aniei.org.mx

2 Según la Real Academia Española de la Lengua, se conoce como asignatura, a cada una de las materias que se enseñan en un centro docente o forman un plan académico de estudios

3 De acuerdo con las áreas y perfiles establecidos por la ANIEI para cada PE.

En la tabla 4, se muestran cursos de otras áreas en los que el alumno debe aplicar la IS:

PE	Cursos
LI	Sistemas de Información (TI) Administración de los Sistemas de Información (TI) Sistemas de Información para Ejecutivos (IHC)
ISC	Laboratorio de Análisis y Diseño Estructurado de Sistemas (TI) Laboratorio de Análisis y Diseño Orientado a Objetos de Sistemas (TI)
ICO	Sistemas de Información (TI)

Tabla 4. Otros cursos en dónde se aplica la IS.

Propiciando la aplicación de la IS, en proyectos universitarios: Caso DES-DACI

En el desarrollo de proyectos de software, es importante seguir una metodología que permita obtener un producto bien documentado, que sea: útil, eficiente, transferible, mantenible, confiable, uniforme y que pueda probarse. La aplicación de metodologías de desarrollo puede disminuir o hacer más eficiente la aplicación de costos, obtener un producto de software con mayor calidad y eficiencia, permitir el reúso y facilitar el mantenimiento del producto de software.

Según Sommerville un SW de calidad debe contar con los atributos de mantenibilidad, confiabilidad, eficiencia y la usabilidad [2], sin embargo, es muy posible que existan sistemas que por su propósito sean más sólidos en alguna característica y débil en otra.

Para Ghezzi, la calidad del producto y el proceso de SW está dada por 1) exactitud, fiabilidad y robustez, 2) desempeño, 3) usabilidad, 4) verificabilidad, 5) capacidad de mantenimiento, 6) reparabilidad, 7) capacidad de evolución y 8) reutilización, 9) portabilidad, 10) comprensibilidad, 11) productividad, 12) oportunidad y 13) visibilidad [4].

La aplicación adecuada de metodologías de desarrollo de software es una habilidad prioritaria en el perfil de los egresados de cualquiera de las carreras relacionadas con el desarrollo de software, en los PE ofrecidos por la DES el 35.5% de los cursos que se imparten requieren que los alumnos entreguen como producto final, un proyecto, el cual en muchos casos, los alumnos no terminan completamente por cuestiones de tiempo de entrega u otros factores.

Para incentivar la culminación de los proyectos, bajo los estándares de la IS, se propuso la creación de un concurso que permita difundir los proyectos que los alumnos desarrollan y de ésta manera motivarlos, para aprender y aplicar metodologías, técnicas y herramientas de la IS en el desarrollo de sistemas informáticos escolares.

El objetivo general del concurso denominado Concurso de Proyectos Informáticos es fomentar en los estudiantes buenos hábitos, al aplicar una metodología formal de IS en el desarrollo de proyectos de SW, además de:

- * Propiciar la comunicación entre alumno-docente.
- * Propiciar el trabajo en equipos de trabajo.
- * Entablar relaciones de asesoría.
- * Incentivar a aplicar adecuadamente metodologías de desarrollo de software.

El concurso se lleva a cabo en un periodo de cinco a seis meses.

Durante éste periodo el equipo de alumnos-asesor, deben planear, desarrollar y documentar el proyecto a entregar. Los proyectos a inscribir pueden ser de algún curso (actual o anterior) que esté contemplado en la malla curricular de su PE, o bien de interés del equipo.

Se consideran tres categorías para los proyectos participantes:

- a) Básica: Proyectos con manipulación de archivos.
- b) Intermedia: Proyectos en los que se administra información en una base de datos (con un manejador de base de datos), y
- c) Avanzada: Proyectos en los que se administra información de bases de datos a través de la web.

Para la evaluación de los proyectos se considera a seis profesores de la DES-DACI que evaluarán a los proyectos en dos dimensiones:

1. Programación del sistema, en la que se verifica que el sistema funcione correctamente conforme a las especificaciones del mismo, asimismo verificar el tiempo y eficiencia de los algoritmos implementados.
2. Diseño y documentación, en la que se verifica la implementación y seguimiento adecuado de algún método y/o técnica de desarrollo de software.

La Academia de Programación e Ingeniería de Software de la DES-DACI, en el año 2006 inició con la planeación del concurso de proyectos informáticos y no fue sino hasta el 2008 cuando se lanzó la primera convocatoria. Hubo poca participación por parte de los alumnos y de los profesores, dado que sólo hubo dos proyectos inscritos, uno en equipo de tres alumnos y una participación individual. Al final, debido a que por parte de los participantes hubo incumplimiento de las bases publicadas, se decidió cancelar el concurso.

En el año 2009 se volvió a lanzar la convocatoria. Esta vez se inscribieron ocho equipos. Llegada la fecha de entrega del proyecto, solo un equipo tenía su proyecto terminado. Dos equipos más se reportaron, pero aún no habían concluido, por lo que se amplió el tiempo de entrega y, por ende, la fecha de evaluación y entrega de resultados.

En esta ocasión sí se llevó a cabo la evaluación. Para ello se dio a cada jurado los tres proyectos entregados. Evaluaron de manera individual y posteriormente se reunieron para deliberar acerca de quién era el equipo ganador. Llegaron a la conclusión de que los lugares quedarían desiertos, ya que los tres proyectos no cubrían los requisitos de documentación ó ejecución.

En el año 2010 se hizo modificación a la convocatoria en el capítulo de la evaluación de los proyectos. En esta ocasión se consideró la participación de los alumnos de la DACI como evaluadores de los proyectos participantes, para lo cual se requería una exposición de los proyectos y la votación sería marcando en una boleta el casillero correspondiente al mejor de los proyectos de cada categoría.

En esta ocasión al lanzar la convocatoria se inscribieron cinco equipos. Se amplió la fecha de inscripción y en esta ampliación se inscribieron seis equipos más; en la fecha de entrega, un equipo de los que

Proyecto	Metodologías		Integrantes	
	Descripción	%	x PE	Total
Laboratorio virtual de estadística	ModCP	70	1 de ISC	5 ISC
Sistema de Información de Almacén	ModLS	80	2 de ISC	
Mini-sistema de altas y bajas de empleados	No hay documentación		2 de ISC	
Ambiente de programación para un lenguaje	No entregó proyecto		2 de ISC 1 de ICO	11 ISC 1 ICO 1 LI
Agenda 2.0	No entregó proyecto		2 de ISC	
Navegación Robótica	No entregó proyecto		2 de ISC	
Simulador de listas ligadas	No entregó proyecto		3 de ISC	
Biblioteca Virtual para programadores	No entregó proyecto		2 de ISC 1 de LI	

Tabla 5. Comparativo del concurso 2009

se inscribieron inicialmente y los seis inscritos al final, entregaron sus proyectos con lo que se tuvieron siete proyectos para evaluar, cinco de los cuales pertenecían a la categoría intermedia y dos a la categoría avanzada.

Se entregó un disco con los códigos de los siete proyectos a cada uno de los jurados que calificarían la parte de programación del sistema; a los jurados de la parte de diseño y documentación se les entregó una carpeta con la documentación entregada de cada proyecto. También se les entregó a todos los jurados un documento con los criterios de evaluación para registrarla y observaciones y/o recomendaciones.

Ese mismo día se montó una exposición de los siete proyectos ante los alumnos de las DES-DACI, para que éstos observaran, evaluaran los proyectos y votaran por el mejor proyecto de cada una de las categorías participantes.

Las tablas 5 y 6 muestran datos comparativos de la participación de los alumnos en los concursos realizados. De los proyectos entregados en ambos concursos se puede observar que hubo una respuesta gradual de los alumnos a la convocatoria en cuanto a participación se refiere y en cuanto a la calidad de proyectos entregados.

En el concurso de 2009 se inscribieron ocho equipos pero de estos sólo tres entregaron su proyecto y de los tres sólo uno funcionaba al 100% aunque le hizo falta documentación, en unos días más de la fecha programada; en el concurso de 2010 se inscribieron once equipos de los cuales siete entregaron su proyecto y de estos tres funcionaban al 100%, tres funcionaban al 80% y uno funcionaba al 50%, en cuanto a documentación, todos entregaron documentación con un porcentaje de completitud que va del 50 al 90%. Así que se puede concluir que a medida que se desarrollen los concursos se puede esperar que:

- Incremente en el número de proyectos participantes,

Proyecto	Metodologías		Integrantes	
	Descripción	%	x PE	Total
Administración de motos	ModLS, MetOO	50	3 ISC	16 ISC 3 ICO
COFIDE: Comisión Financiera de descuentos	ModLS, MetOO	80	1 ISC 1 ICO	
Sistema de inscripción a cursos no curriculares	ModLS, MetOO	85	3 ISC	
Food Server	ModLS, MetOO	80	3 ISC	
Sistema de control de alumnos	ModLS, MetOO	90	3 ISC	
Sistema de control de Zapaterías	ModLS, MetOO	100	3 ISC	
Sistema administrador de Videoclub Sinergia	ModLS, MetOO	95	2 ICO	
Proyecto uno sin nombre	No entregó proyecto		1 ISC	11 ISC 1 ICO 1 LI
Sistema de Encriptamiento de Mensajes (SIEM)	No entregó proyecto		2 ISC	
Sistema de códigos de protección y corrección de errores (SISCOPECORE)	No entregó proyecto		2 ISC	
Proyecto dos sin nombre	No entregó proyecto		3 ISC	

Tabla 6. Comparativo del concurso 2010

- Incremente la calidad de los productos,
- Aumento en la experiencia en el uso de metodologías de software.

En ambos concursos el 86% de los alumnos que se inscriben son los del PE de ISC, el 4% son de LI y el 10% son de ICO, esto es debido a que los alumnos del PE de ISC tienen un mayor porcentaje de cursos en los que desarrollan proyectos.

Conclusiones

Las metodologías de software permiten obtener sistemas informáticos que son confiables, que permiten su evolución y tienen un mayor grado de calidad y satisfacción para el cliente.

Desarrollar la habilidad y actitud para aplicar metodologías de

desarrollo de software es importante para los egresados de carreras relacionadas con la computación y la informática.

Mostrar a los alumnos universitarios proyectos en los que se

usan metodologías de desarrollo de software, les permite ver la diferencia de la aplicación de las mismas y a su vez, los motiva a considerar su uso en el desarrollo de sistemas.



Referencias

1. Gacitúa Bustos, R. A.: *Métodos de desarrollo de software: el desafío pendiente de la estandarización*. Theoria, Vol. 12, pp. 23-42(2003)
2. Sommerville, I.: *Ingeniería de Software*. México D.F. Addison Wesley. (2002).
3. Pressman, R. S.: *Ingeniería del SW Un enfoque Práctico*. Mc Graw Hill (2002).
4. Ghezzi, C.; Jazayeri, M.; Mandrioli, D.: "Fundamentals of Software Engineering". Capítulo 1 *Software Engineering: A Preview*. 2da. Edición. Prentice Hall. (2003).