

Reporte de proyecto de investigación: desarrollar módulos de software administrativos para el sistema en línea del estímulo al desempeño del personal docente del SNIT utilizando metodología orientada a objetos

Juan Ignacio Cerca Vázquez

Instituto Tecnológico de Celaya
nacho@itcelaya.edu.mx

Luis Alberto López González

Instituto Tecnológico de Celaya
nacho@itcelaya.edu.mx

José Jesús Sánchez Farías

Instituto Tecnológico de Celaya
nacho@itcelaya.edu.mx

José Guillermo Rodríguez Villafaña

Instituto Tecnológico de Celaya
guillermo.rodriguez@itcelaya.edu.mx

Alberto Álvarez Castillo

Instituto Tecnológico de Celaya
nuabli@hotmail.com

Resumen

El sistema en línea del estímulo al desempeño del personal docente del Sistema Nacional de Institutos Tecnológicos (SNIT). Ha sido desarrollado en el año 2012, como un conjunto de estrategias tecnológicas de la dirección de desarrollo

profesional, con el objetivo de apoyar al procedimiento existente en la Dirección General de Educación Superior Tecnológica (DGEST). En el ejercicio 2013 se contó con la participación de 4036 docentes de tiempo completo y 711 con tiempo parcial, pertenecientes a los Institutos Tecnológicos Federales, lo cual implicó la revisión del mismo número de expedientes de solicitud, por una comisión que constó de 418 profesores evaluadores a lo largo del territorio nacional, para este propósito, se utilizó la herramienta de software desarrollada por la dirección de telecomunicaciones, y que fue auxiliar en que la realización del proceso se diera en un tiempo de 2 semanas; Cuando anteriormente se llevaba con un tiempo mínimo de 2 meses, además de las deficiencias en la coordinación y replicación de datos, sin embargo, después de este proceso se detectan por parte de los desarrolladores y la dirección de área responsable, varias áreas de oportunidad que de ser desarrolladas incrementarán las herramientas y funcionalidades que el software ofertó a la comisión de evaluación y subdirectores académicos.

En este mismo ejercicio al abrir la convocatoria se decidió contar con el apoyo del Instituto Tecnológico de Celaya, como proveedor de infraestructura de red, alojamiento, base de datos y administración técnica del servicio. Durante este periodo el área de desarrollo de software del instituto, en coordinación con su contraparte de la dirección de telecomunicaciones, realiza un análisis del proceso automatizado en busca de áreas de oportunidad dentro de los módulos del sistema que no hayan sido proyectadas aún y sumándose a las propuestas que la dirección de desarrollo profesional haya idealizado o solicitado su construcción.

Como resultado de este trabajo conjunto, las áreas de oportunidad mencionadas consisten en realizar la construcción de los módulos (software) de administración que serán utilizados por los usuarios responsables de cada una de las etapas del proceso, y que consisten en:

- a) Módulo que gestionará el subdirector académico con el fin de administrar la comisión de evaluación por plantel del SNIT
- b) Módulos para la generación de las nóminas PROED
- c) Extender las herramientas de administración orientadas a la comisión de evaluación

Concluyendo como objetivo final la integración e implementación de los módulos al software existente, con estricto apego a los lineamientos expedidos por DGEST.

1. Introducción

El Programa de Estímulos al Desempeño del Personal Docente es una convocatoria emitida por la Subsecretaría de Educación a través de la Dirección General de Educación Superior Tecnológica que tiene como finalidad impulsar y reconocer las actividades sustantivas desarrollados por los profesores de los Institutos Tecnológicos Federales y Centros Especializados del SNIT.

En dicha convocatoria pueden participar profesores con categoría asociado o titular, profesor investigador y técnico docente del nivel superior, que estén actualmente desempeñando funciones o tareas que impacten en los factores de calidad en el desempeño, la dedicación y la permanencia en las actividades de docencia, y que cumplan con los dispuesto en el Reglamento Interior de Trabajo del Personal Docente de los Institutos Tecnológicos. Desde el ejercicio 2012 se desarrolló e implementó un sistema en línea para automatizar el procedimiento emitido por la Dirección de Desarrollo Profesional de la DGEST que impacta directamente en la conformación y evaluación de los expedientes virtuales de los profesores participantes. Dicho expediente virtual está conformado por todos los documentos comprobatorios, constancias o evidencias de las actividades realizadas en el año anterior a partir de la emisión de la convocatoria.

2. Metodología

La intención del presente marco teórico es fundamentar utilizando las bibliografías de los contenidos temáticos los conocimientos de ingeniería de software que serán implementados en el desarrollo de la aplicación, se comenzará por definir el tipo de metodología a utilizar que será la orientada a objetos y posteriormente a considerar el esquema de diseño en 3 capas o Modelo-Vista-Controlador con el objetivo de fundamentar y proporcionar una idea clara de los insumos teóricos necesarios para el desarrollo del presente proyecto de investigación.

IBM Rational Unified Process, o RUP, consiste en una plataforma metodológica de desarrollo de software adaptable que tiene como propósito la implementación de prácticas comprobadas y arquitectura configurable. Permite utilizar los componentes de proceso exclusivamente para cada etapa del proyecto. El RUP se basa en 6 principios clave que se definen como:

Adaptar el proceso: Cada proceso deberá adaptarse a las necesidades de cliente considerando como lo más importante la interacción con éste, con enfoque hacia las características propias del proyecto u organización, el tamaño del mismo, así como su tipo o las regulaciones que lo condicionan e influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto en un área subformal para hacer un proceso de satisfacción del software.

Equilibrar prioridades: Los requisitos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un equilibrio que satisfaga los deseos de todos. Gracias a este equilibrio se podrán corregir desacuerdos que surjan en el futuro.

Demostrar valor iterativamente: Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.

Colaboración entre equipos: El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requisitos, desarrollo, evaluaciones, planes, resultados, etc.

Elevar el nivel de abstracción: Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (frameworks) por nombrar algunos. Esto evita que los ingenieros de software vayan directamente de los requisitos a la codificación de software a la medida del cliente, sin saber con certeza qué codificar para satisfacer de la mejor manera los requisitos y sin comenzar desde un principio pensando en la reutilización del código. Un alto nivel de abstracción también permite discusiones sobre diversos niveles y soluciones arquitectónicas. Éstas se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con el lenguaje UML.

Enfocarse en la calidad: El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

Diseño Orientado a Objetos UML

UML (Unified Modeling Language) Consiste en una metodología basada en el lenguaje de modelado que permite, construir y documentar el conjunto de elementos que forman un desarrollo de software orientado a objetos. Fue desarrollado en 1995 por: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas de Desarrollo de Software Asistido por Computadora (CASE). En el proceso de creación de UML han participado, no obstante, otras empresas de la industria como Microsoft, Hewlett-Packard, Oracle o IBM, del mismo modo grupos de analistas y desarrolladores de todo el mundo. Esta notación ha sido ampliamente aceptada debido al prestigio de sus creadores y debido a que incorpora las principales ventajas de cada uno de los métodos particulares en los que se basa: Booch, OMT y OOSE. UML se ha posicionado como una metodología estable y estándar, evitando la creación de trabajos emergentes que basados en los principales métodos implementaban nuevas versiones que incorporaban las técnicas de los demás. Con UML se fusiona la notación de estas técnicas para formar una herramienta compartida entre todos los ingenieros software que trabajan en el desarrollo orientado a objetos. Un modelo representa a un sistema software desde una perspectiva específica. Al igual que la vista de planta y capas de una figura en dibujo técnico nos muestran la misma figura enfocada desde distintos ángulos, cada modelo entrega un panorama con un aspecto distinto del sistema.

En la ingeniería de software se denomina aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales.

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

En la línea de investigación se busca fortalecer la implementación de metodologías de ingeniería de software en el desarrollo de aplicaciones, por tal motivo es una prioridad realizar un adecuado diagnóstico de varias metodologías recientes, con el objetivo de motivar a los estudiantes a implementarlas con nuevos enfoques, una metodología de desarrollo es un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea y abierta cada una de las actividades del ciclo de vida de un proyecto de desarrollo. Es un proceso de software detallado y completo.

Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, incremental, prototipos, espiral). Definen mecanismos, roles y actividades, junto con prácticas y técnicas recomendadas. La metodología para el desarrollo de software en un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Una metodología para el desarrollo de software comprende los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que cumplimos el objetivo por el cual fue creado.³

Se implementa como caso de estudio el uso de la metodología de proceso unificado racional (RUP) que es un marco de trabajo de proceso de desarrollo de software iterativo creado por Rational Software Corporation (IBM, 2003). Se adapta entonces un marco de trabajo de proceso adaptable, RUP proporcionó la combinación de varias metodologías y se vio influenciado por métodos previos como el modelo en espiral. Las consideraciones clave fueron el fallo de proyectos usando métodos

monolíticos del estilo del modelo en cascada y también la llegada del desarrollo orientado a objetos, considerando la naturaleza de la aplicación de software en este paradigma de diseño y el objetivo de elevar el modelado de sistemas a la práctica del desarrollo y de resaltar los principios de calidad que aplicaban a las manufacturas en general al software. El proceso fue diseñado con un modelo orientado a objetos subyacente, usando UML (Unified Modeling Language) utilizando un conjunto de módulos o elementos de contenido, que describen qué es lo que se produce, las habilidades necesarias requeridas y la explicación paso a paso describiendo cómo se consiguen los objetivos de desarrollo. Los módulos principales, o elementos de contenido, son:

Roles (o Actores): Definiendo un conjunto de habilidades, competencias y responsabilidades, son agrupados en una colección de objetos, de acuerdo a la interacción que llevan con el sistema

- Modulo del sistema en turno (Nómina PROED, Subdirector Tecnológico)
- Profesor
- Subdirector
- Evaluador Tecnológico
- Comité evaluador DGEST

Productos de trabajo: Considerando que un producto de trabajo representa algo que resulta de una tarea, incluyendo todos los documentos y modelos producidos mientras que se trabaja en el proceso, en el proceso se desarrollaron los casos de uso pertinentes para cada actor y su interacción con el sistema Figura 1.

De esta forma se muestra la interacción que el usuario final de la entidad especificando la cardinalidad de la solicitud y el almacén de datos de la información, el usuario (Actor) al cual es orientado el módulo. En este caso concreto el Subdirector Académico solicita la lista de evaluadores que existen en su tecnológico con el objetivo de realizar toma de decisiones, el módulo atenderá dicha petición realizando la consulta a la entidad Catálogo de Evaluadores y entregando al Actor

el resultado o respuesta de su petición, de esta manera se elaboran los esquemas de caso de uso para cada una de los actores y módulos del sistema.

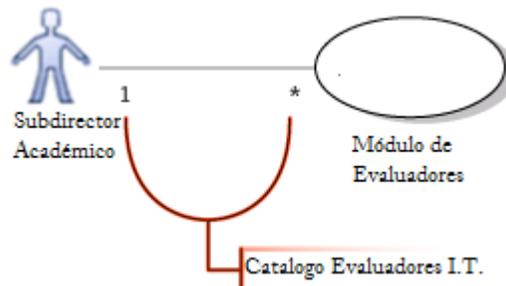


Figura 1 Modelado de caso de uso para el Actor correspondiente.

Tareas o actividades, una tarea describe una unidad de trabajo asignada a una entidad que proporciona un resultado significativo, para este propósito se asigna un conjunto de tareas o actividades por módulo y que representarán cada uno de los esquemas de caso de uso extendido delegando las responsabilidades por entidad del sistema

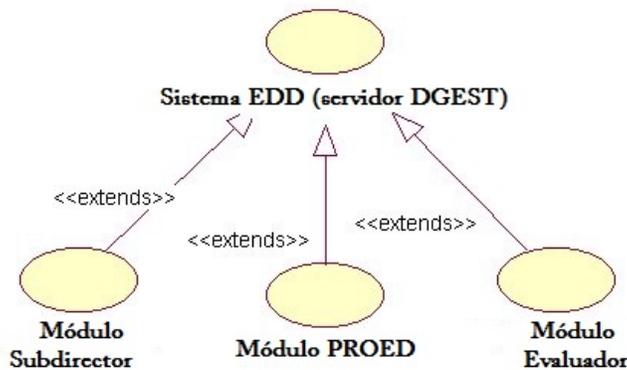


Figura 2 Diagrama de Caso de Uso Extendido.

Con el objetivo preciso de mostrar la delegación de cada módulo en el diagrama extendido se muestra claramente como cada uno de los componentes obtendrá una herencia del módulo padre o central, pero asumiendo su responsabilidad dentro del desarrollo, en este caso y vinculándolo a un diseño en 3 capas resulta relativamente sencillo el que una vez diseñada la aplicación en casos extendidos.

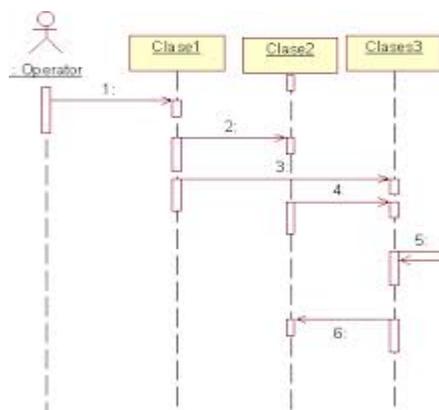


Figura 3 Secuencia permitirá ubicar que clases, entidades y actores se relacionan.

Utilizando el framework de desarrollo web Yii para PHP, en donde el modelo permite la herencia de clases desde un modelo central, en esta etapa del diseño prácticamente el desarrollo se encuentra definido y estructurado, concluyendo en que la abstracción por objetos permitirá a los desarrolladores organizar Clases y Métodos para su codificación final en el modelo de 3 capas Para el ciclo de vida de desarrollo basado en ISO/IEC 12207 Esta norma establece un marco de referencia común para los procesos del ciclo de vida del software, con una terminología bien definida a la que puede hacer referencia la industria del software y proporciona procesos, actividades y tareas de organización, desarrollo, operación y mantenimiento de productos software. Esta norma incluye también un proceso que puede emplearse para definir, controlar y mejorar los procesos del ciclo de vida del software.

La ISO 12207 define un modelo de ciclo de vida como un marco de referencia que contiene los procesos, actividades y tareas involucradas en el desarrollo, operación y mantenimiento de un producto software, y que abarca toda la vida del sistema, desde la definición de sus requisitos hasta el final del uso. Agrupando las actividades de desarrollo dentro de esta norma se generó el esquema:

El nivel de madurez del ciclo de desarrollo se considera gestionado ya que el producto software es desarrollado implementando las etapas de: Planificación, modelado, diseño y codificación, de acuerdo a un esquema definido (UML) y con una metodología (RUP) lo cual permite generar diagramas orientados a objetos que siguiendo este proceso serán fáciles de volver a rehacer.

una aplicación final consistente que satisfaga los requerimientos del usuario, la plataforma seleccionada fue Yii framework para PHP.

En la etapa final de pruebas e implantación son analizados extensamente todos los programas de aplicación recién escritos o modificados de los módulos. Las pruebas son realizadas mientras se desarrolla el sistema y no solo al final, con esto se demuestra la imperfección de programas, manuales o equipos. Las pruebas son aplicadas en el siguiente orden metodológico:

Pruebas De Enlace Con Datos De Prueba: En esta etapa se introducen datos creados especialmente para cada módulo en particular desarrollado, esto se realiza para probar el sistema, primero se introducen datos normales y si el sistema trabaja correctamente se le introducen nuevos datos cada vez más complejos este paso se denomina también pruebas de enlace.

Prueba Completa Del Sistema Con Datos De Prueba: Luego de que las pruebas de enlace se hayan superado el sistema se prueba como una entidad completa, en este paso se presentan los operadores y usuarios finales para el desarrollo de esta etapa.

Prueba Completa Del Sistema Con Datos Reales: Luego de haber superado todas las pruebas anteriores con diversos tipos de datos llega el momento de probar el sistema con datos reales, en esta etapa se hace una comparación entre los resultados que arroja el sistema con los deseados en el diseño y se verifica como es el comportamiento con los datos reales.

Auditoria: Se genera un grupo de usuarios y auditores ajenos al equipo de desarrollo el cual no está involucrado en el diseño y desarrollo del sistema pero si analiza el comportamiento del mismo para determinar si el sistema creado es confiable o no de esta manera notificarlo a los demás usuarios para estar seguros del sistema.

3. Conclusiones

La metodología de desarrollo de sistemas (RUP) es un paradigma de diseño orientado a objetos que se encuentra aún vigente para el modelado de sistemas

enfocados al uso de tecnologías web emergentes, es importante resaltar la facilidad con que el modelado de la solución fue llevado a la etapa de desarrollo de manera sistemática y repetible, es decir si la aplicación volviera a desarrollarse utilizando el mismo modelo volvería a obtenerse el mismo diseño, de esta manera se consigue el objetivo de proporcionar los servicios a los subdirectores académicos de los Institutos Tecnológicos del país, así mismo basados en los datos del ejercicio anterior se espera la revisión aproximada de 4000 expedientes por más de 400 evaluadores a lo largo del territorio nacional, cada uno de los roles definidos en la metodología cuenta con los módulos necesarios para realizar su operación, en la experiencia al equipo de desarrollo queda el uso de un estándar de diseño con el fin de progresar en la madurez del proceso. De esta manera el desarrollo de los módulos de software ha sido concluido de manera integral, la solución fue presentada como complemento a la ya existente y puesta a disposición de la Dirección de Desarrollo Profesional, para incorporarla de acuerdo a sus necesidades, figura 5.



Figura 5 Portal.

Bibliografía y referencias

- [1] PHP y MySQL, Tecnologías para el desarrollo de aplicaciones Web, Cobo, Gómez, Santander, España (2005)
- [2] IBM Rational Unified Process Reference and Certification Guide: Solution Designer, Ahmad K. Shuja and Jochen Krebs, IBM Press, USA 2008

- [3] A Software Project Management for Small to Medium Sized Project. Rakos, John J.Ed. Prentice Hall, USA 1990.
- [4] The Unified Software Development Process, Jacobson, Booch, Rumbaugh, Addison-Wesley, USA 1999
- [5] Lineamientos del Estímulo al Desempeño Docente, Dirección General de Educación Superior Tecnológica, México DF, Abril 2013.
- [6] www.dgest.gob.mx (R: 06/09/13).
- [7] <http://www.snit.mx/informacion/sistema-nacional-de-educacion-superior-tecnologica> (R: 30/08/13).
- [8] <http://www-01.ibm.com/software/awdtools/rup/> (R: 04/10/13).