

Desarrollo de software: una experiencia de Aprendizaje basado en proyectos

Homero Jiménez Pérez

Universidad de la Sierra Sur

jphomero@yahoo.com.mx

Sandra Karina Ramírez Vásquez

Universidad de la Sierra Sur

skrv03@gmail.com

Resumen

En el presente documento se expone la experiencia obtenida al desarrollar un producto de software para el Departamento de Adquisiciones y Recursos Materiales (DARM) de la Universidad de la Sierra Sur, en el Estado de Oaxaca. Este producto de software es el resultado práctico de aplicar una estrategia de Aprendizaje Basado en Proyectos, en materias de la Licenciatura en Informática de la misma universidad. Esta metodología de aprendizaje permite conectar el conocimiento teórico de asignaturas de ingeniería de software, con proyectos de desarrollo que tienen aplicación en el mundo real. En el documento se describe el proceso de desarrollo, a partir de la fase de análisis, hasta la implementación y funcionamiento. Se destaca la responsabilidad de profesores en la preparación del proyecto y en la guía y supervisión del trabajo de los estudiantes. Además, se resalta la experiencia práctica que los estudiantes adquieren durante el proceso y su participación directa en la incorporación consciente de los conceptos y metodologías revisados en la asignatura.

Palabras clave: Experiencia, aprendizaje basado en proyectos, Ingeniería de software, proceso de desarrollo.

Introducción

De acuerdo con Zavala et. Al (2007:19g) el proceso de enseñanza-aprendizaje tradicional se ha “reducido al aprendizaje memorístico de conocimientos, hecho que conlleva la dificultad para que estos puedan ser aplicados en la vida real”. Se rige por características que se han aplicado en la Educación Básica principalmente. Sin embargo, también se pueden encontrar ejemplos de su aplicación en el nivel medio superior y superior. Dichas características se resumen a continuación:

El profesor es el modelo a seguir, a través de la imitación y la obediencia. Éste se encarga de organizar los contenidos de las asignaturas y dirige de forma centralizada la enseñanza.

El método de enseñanza es el mismo para todos los estudiantes. Los estudiantes son sujetos pasivos, que reciben lo que el profesor elige.

Los problemas a resolver se extraen de libros de texto sin que exista una vinculación con el mundo real.

El profesor prefiere no distraer al estudiante con temas que no cubre su asignatura para evitar confusiones.

Es necesario adoptar nuevas estrategias de trabajo en el aula para fomentar que los estudiantes adquieran conocimientos y experiencia en la toma de decisiones, para elevar su desempeño académico y prepararlos para su futuro profesional.

Con el presente trabajo, se pretende motivar a los profesores para utilizar la estrategia de Aprendizaje Basado en Proyectos. Dicha estrategia permite una experiencia más enriquecedora para que los estudiantes refuercen y pongan en práctica los conocimientos adquiridos en el aula, adquieran mayor experiencia en la toma de decisiones y en la colaboración interdisciplinaria. Por último, la evaluación que los profesores realizan estimula la interacción del estudiante con los usuarios potenciales.

El proyecto que se describe en el presente documento, se desarrolló con estudiantes de séptimo semestre de la Licenciatura en Informática de la Universidad de la Sierra Sur (UN SIS). Los estudiantes al momento de ingresar a séptimo semestre ya cursaron las asignaturas de Programación, Bases de Datos e Ingeniería de Software. Se eligió un problema generado en un área de la universidad. El problema se resuelve con el desarrollo de una aplicación de software. Se eligió un grupo de desarrollo de dos estudiantes que son dirigidos por un profesor del área.

El documento se estructura de la manera siguiente: en la sección 2 se presentan inicialmente los antecedentes, donde se describe brevemente la dinámica que se ha llevado a cabo en el desarrollo de las asignaturas de la Licenciatura en Informática. A continuación se explican las causas que definieron el problema, seguida por una descripción del mismo. Se presenta el objetivo que se pretende alcanzar, así como la lista de los participantes en el proyecto, de acuerdo a los roles que llevó a cabo cada participante. En el desarrollo del proyecto, se describen las etapas desde la planeación inicial hasta la construcción, pruebas e implementación del mismo. Finalmente, en la sección 3, se exponen los alcances y las limitaciones que se tienen al trabajar con este enfoque.

2. Desarrollo de contenidos

2.1. Antecedentes

Desde hace siete años, para las materias relacionadas con el desarrollo de software de la Licenciatura en Informática de la Universidad de la Sierra Sur, se eligen problemas reales como ejercicios de práctica. La carrera y la universidad son de reciente creación¹, por lo que durante los primeros años dichas actividades se aplicaron con algunas limitaciones.

Los primeros proyectos desarrollados para dar solución a necesidades dentro de la misma universidad, eran detectados y propuestos por los propios profesores. A raíz del trabajo que se llevó a cabo inicialmente, los responsables de área detectaban

problemas que requerían la solución de software para su solución y acudían a la jefatura de carrera de Informática para solicitar apoyo. A partir de la planeación inicial de los cursos, se determinaba el proyecto que se trabajaría durante un semestre o un año entre dos o tres profesores. Para esto, se debía iniciar el trabajo de planeación, primero entre los profesores, para tener definido el problema al solicitar la solución a los estudiantes. Un producto generado es el software CONAB (López et al., 2009 y Jiménez-Pérez et al., 2010). Esta aplicación se desarrolló por estudiantes de la universidad y aún se utiliza en la biblioteca de la universidad.

En 2010 se formaliza el proyecto “Desarrollo e implantación de software en la UNSIS”, del Cuerpo Académico: Tecnologías e Ingeniería de Software. Se cuenta actualmente con un registro de necesidades de software y de proyectos desarrollados desde el año 2005. Todos los proyectos de software han sido desarrollados por estudiantes, bajo la dirección de profesores. Lo anterior ha hecho posible que los estudiantes obtengan mayor madurez en el proceso de desarrollo de aplicaciones de software.

Sin embargo, a pesar de la experiencia previa, es a partir de este proyecto que se apoya en la estrategia de aprendizaje basado en proyectos. En el desarrollo de este software se pone mayor relevancia en impulsar a los estudiantes para que apliquen sus conocimientos y habilidades en todas las fases del desarrollo. Se inicia desde la definición del problema, pasando por el análisis e interpretación de las necesidades del usuario y la habilidad para tomar decisiones en todo el desarrollo de la aplicación, hasta concluir con la construcción implementación y prueba.

De acuerdo con Maldonado (2008), algunos beneficios del Aprendizaje Basado en Proyectos son los siguientes:

Preparar a los estudiantes para los puestos de trabajo.

Aumento de la motivación

Conectar el aprendizaje de los estudiantes en la escuela con la realidad.

Ofrecer oportunidades de colaboración para construir conocimiento

Aumentar habilidades sociales de comunicación.

Acrecentar habilidades para la solución de problemas

Ofrece oportunidades para realizar contribuciones en la escuela o en la sociedad.

Aumenta la autoestima cuando los estudiantes logran algo que tiene su valor real fuera del aula de clase.

Permite que los estudiantes hagan uso de sus fortalezas individuales.

El aprendizaje basado en proyectos disminuye la forma de trabajo individual, aumenta la capacidad de análisis crítico del estudiante y lo incita a tomar decisiones de acuerdo al contexto del problema. Los beneficios observados motivaron el uso de esta estrategia.

2.2. Elección del problema

En este documento se toma como punto de partida el inicio del curso de séptimo semestre de la asignatura: Calidad de Software. En este semestre, los estudiantes ya tienen los conocimientos necesarios para desarrollar proyectos (productos) de software. Han cursado asignaturas de Diseño Estructurado de Algoritmos, Programación Estructurada, Bases de Datos I, Bases de Datos II, Programación Orientada a Objetos I, Programación Orientada a Objetos II, Ingeniería de Software I e Ingeniería de Software II.

La selección del tema se hizo a partir de una revisión de las necesidades que diferentes áreas de la universidad habían reportado a la jefatura de carrera de Informática. Se eligió atender la necesidad del Departamento de Adquisiciones y Recursos Materiales (DARM) de la universidad. Una de las razones más importantes para elegir este problema fue que la complejidad no era tan alta, y se tenía el interés y la disponibilidad de tiempo de la responsable del área, para definir y especificar los requisitos del software.

2.3. Descripción del problema

El DARM cuenta con un sistema de software que les proporciona el Gobierno del Estado de Oaxaca para el manejo de inventarios y resguardos. Sin embargo llevan a cabo otras tareas que de manera regular realizan manualmente. Las actividades que desarrollan y para las que requerían de una aplicación se listan a continuación:

Realizar la solicitud de compra de materiales y contratación de servicios.

Elaborar la solicitud de pago del material o el servicio contratado.

Controlar la información de proveedores, solicitudes de compra, proyectos autorizados.

La información que se genera en el departamento se almacena en hojas de cálculo, lo que dificulta su manejo e incrementa el tiempo de búsqueda de datos concretos. Además, se requerían consultas y reportes periódicos para conocer lo siguiente:

El padrón de proveedores que trabajan para la institución.

El número de solicitudes de compra que se generan.

Conocer el área que realiza la solicitud de compra.

A partir de la problemática expuesta, se analizó con más detalle el alcance del proyecto y sobre todo la forma en la que los estudiantes desarrollarían la aplicación y se involucrarían en el proceso de desarrollo de software. En esta etapa el profesor tenía claro el alcance del proyecto, por lo que se encargó de relacionar los temas del curso para aplicarlos durante el proceso de desarrollo. El profesor debe acompañar a los estudiantes, permitiendo que éstos tomen las decisiones necesarias, pero debe estar atento para guiar y orientar a los estudiantes cuando surja algún problema de comunicación o de interpretación de las necesidades del usuario.

Con esta estrategia de trabajo se va más allá del aula de clase, se motiva al estudiante para que indague por cuenta propia, para que realice la búsqueda de conocimiento,

que analice e interprete situaciones desconocidas, que proponga soluciones y que sea capaz de implementarlas.

2.3.1 Objetivo

El objetivo que se planteó para el curso es el siguiente:

Desarrollar un producto de software cuya aplicación resuelva una problemática real, mediante el enfoque de Aprendizaje Basado en Proyectos, que permita a los estudiantes involucrarse en todas las etapas del proceso de software para adquirir mayor experiencia en el análisis y la interpretación de necesidades, que motiven la investigación, la búsqueda de conocimiento e incrementen su madurez en la toma de decisiones.

Con el objetivo planteado se busca que los estudiantes adquieran la experiencia de trabajo colaborativo, en un proyecto real de desarrollo. Proyecto que finalmente tendrá una aplicación en el contexto que se desarrolló y no quedará archivado como tantos proyectos finales de clase.

2.4. Participantes del proyecto y roles

Los roles y las personas que jugaron dichos roles se listan a continuación:

Usuario o cliente del sistema. Se encarga de plantear directamente la necesidad que tiene, respecto al producto de software. Para el presente proyecto, la Lic. Ruth Lucía Reyes López, titular del Departamento de Adquisiciones y Recursos Materiales de la Universidad de la Sierra Sur, desempeñó este rol.

Director del Proyecto. Este rol lo llevan los profesores de la Licenciatura en Informática que coordinan y guían el trabajo de los estudiantes. La orientación que los profesores proporcionan a los estudiantes, les sirve de base para el análisis, diseño y sobre todo para la toma de decisiones en las diferentes fases del proyecto. En este proyecto quien

tiene el rol de dirección y asesoría es el M.C. Homero Jiménez Pérez, quien coordinó el trabajo de los estudiantes durante el transcurso del proyecto. El profesor debe apoyar y guiar el trabajo de los estudiantes, pero permitiendo que sean éstos los que aporten las soluciones y que decidan la forma de resolver el problema.

Equipo de desarrollo de software: Estudiantes del séptimo semestre de la Licenciatura en Informática, Waldemar Rojas Hernández y José Alberto Pacheco Antonio. Durante todo el proceso forman el equipo de desarrollo. El rol que desempeñan va cambiando de acuerdo con la etapa del proyecto. Llevan a cabo tareas de planeación, análisis, diseño, codificación, pruebas e implementación. Todas estas tareas forman parte de actividades específicas de su área de estudio.

2.5. Desarrollo del proyecto

2.5.1. Planeación inicial

Este proyecto inicia formalmente con la primera reunión, donde se describe la problemática y las necesidades de información que el DARM tiene. A la primera reunión le siguieron otras dos en donde se estableció el alcance del proyecto. En todas estas reuniones se tuvo la participación del usuario, quien respondió a las preguntas y dudas del equipo de desarrollo, para lograr una comprensión general del problema.

En la siguiente etapa, el equipo de desarrollo presentó un primer análisis de la situación y de las posibles alternativas de solución. Este análisis fue revisado por el director del proyecto, quién los apoyó para entender completamente el problema. Posteriormente, el equipo de desarrollo determinó el modelo de desarrollo de software a seguir. Tomando en cuenta las técnicas de la Ingeniería de Software, analizando las condiciones del problema y considerando el tiempo de entrega (cuatro meses), el equipo de desarrollo decidió utilizar el modelo iterativo. Este modelo permite a los desarrolladores, dividir el proyecto en módulos o partes, y cada una de ellas integrarlas a las anteriores, hasta conseguir el producto final. El proyecto se dividió en tres módulos, siendo el primero la creación de la base de datos y las interfaces principales, el segundo módulo correspondiente al proceso de ingreso de la

información. Finalmente el tercer módulo correspondió al manejo de la información almacenada en la base de datos: modificación, eliminación y generación de órdenes de pago. Para cada módulo se realizó cada una de las actividades que a continuación se detallan.

2.5.2. Descripción o definición de requisitos

De acuerdo con Sommerville (2005: 108) la definición de los requisitos –o requerimientos- es “la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. El proceso de descubrir, analizar y verificar estos servicios y restricciones se denomina ingeniería de requerimientos”.

Las tareas de descubrimiento, análisis y verificación a las que se refiere Sommerville, dan como resultando un entendimiento claro del problema que en cada módulo se debe resolver. La importancia de esta fase, en el proceso de desarrollo de software, la convierte en la base para el inicio del trabajo. Una comprensión errónea del problema provoca la entrega de un resultado que no coincide con lo que el usuario solicita. Este problema es muy común, por lo que el director del proyecto debe poner mucha atención para evitar que el equipo avance a la siguiente etapa, sin tener claro lo que el usuario está requiriendo.

El equipo de desarrollo realizó reuniones para definir los requerimientos de cada módulo. La información obtenida en estas reuniones sirvió para establecer la lista definitiva de requerimientos a cumplir. En esta etapa, se puso especial atención para evitar las dificultades que frecuentemente se presentan:

1. Para el usuario es complicado explicar y especificar lo que necesita exactamente.
2. Para el equipo de desarrollo, no siempre quedan claras las necesidades que el usuario tiene. Es una tarea ardua que requiere de análisis e interpretación. Esta parte es muy importante, debido a que se integran los conocimientos que se adquieren en el aula y se aplican para resolver una problemática real.

3. El equipo de desarrollo debe ayudar al usuario para que en conjunto se concreten y definan los alcances del proyecto.

En todas estas situaciones difíciles, la guía y la orientación del director del proyecto son determinantes para cumplir con los plazos previstos.

2.5.3. Diseño.

La etapa siguiente consiste en realizar un “proceso iterativo mediante el cual los requisitos se traducen en un ‘plano’ para construir el software” (Pressman, 2005: 249). Complementando la definición anterior, el autor cita a McGlaughlin, quién sugiere tres características que se utilizan para guiar la evaluación de un buen diseño (Pressman, 2005:249):

1. “El diseño debe implementar todos los requisitos explícitos contenidos en el modelo de análisis, y debe ajustarse a todos los requisitos implícitos que desea el usuario.

2. El diseño debe ser una guía legible y comprensible para quienes generan código y quienes realizan pruebas, y en consecuencia, dan soporte al software.

3. El diseño debe proporcionar una imagen completa del software desde una perspectiva de implementación”.

El “plano” para construir el software al que hace referencia Pressman (2005) se plasma en un documento de diseño (parecido a un plano arquitectónico). Para realizar esto se necesitan dos elementos básicos:

1. Conocimientos de Lenguaje de Modelado Unificado (UML). Es un lenguaje que “proporciona un conjunto estandarizado de herramientas para documentar el análisis y diseño de un sistema de software. Se basa esencialmente en una técnica orientada a objetos” (Kendall, 2005: 699).

2. Una herramienta de software que comúnmente llamamos “modelador” o “herramienta de modelado”. Existen varias en el mercado, las más completas se pueden utilizar mediante el pago de una licencia. También existen versiones gratuitas. En este proyecto se utilizó una herramienta gratuita llamada StarUML.

Mediante la creación de modelos:

“se captura un conjunto de ideas conocidas como abstracciones de un sujeto. Si un modelo, es muy complicado capturar información relevante necesaria para el entendimiento del problema, manipularlo e implementar la solución, mientras se excluye cualquier información irrelevante que pudiera dificultar nuestro progreso”² (Alhir, 2003: 6).

A partir de los conocimientos obtenidos en materias previas y la experiencia del equipo de desarrollo, éste realizó un diseño UML del proyecto que representa la solución del problema en un lenguaje formal. En esta etapa, se presenta una solución que atiende las necesidades planteadas en la etapa de determinación de requisitos. El equipo de desarrollo debe analizar con detalle las alternativas posibles para encontrar la mejor de ellas. Este es otro momento importante para tomar decisiones.

En todo momento se tiene el apoyo y la guía del director del proyecto. Sin embargo, para lograr el objetivo que los estudiantes adquieran mayor experiencia y madurez en la toma de decisiones, es importante que se motive y dé la seguridad necesaria para que el equipo de desarrollo sea el que proponga la mejor solución. La misma importancia requiere que el director del proyecto evalúe la argumentación y los conocimientos que los estudiantes expongan para proporcionar mayor seguridad del curso que el proyecto debe seguir.

2.5.4. Construcción, pruebas e implementación La siguiente etapa consiste en la codificación del diseño. De acuerdo con Alonso et al. (2005: 86), la etapa de diseño:

“tiene como objetivo traducir en una forma legible para la computadora el diseño desarrollado en la etapa anterior. Esta actividad implica la creación de programas informáticos aplicando las estructuras de programación de algún paradigma y

utilizando un lenguaje apropiado de programación. Como producto de este proceso se obtiene un listado fuente de los programas que definen el software que se está desarrollando”.

Para el caso de este proyecto se utiliza el paradigma orientado a objetos, el lenguaje de programación Java SE, así como el manejador de datos MySQL 5.5 para el almacenamiento de la información.

El resultado de esta etapa es la construcción del software de cada módulo del proyecto. Cada vez que el equipo de desarrollo concluía un módulo, dicho módulo pasaba por el proceso de validación y verificación. Después de la verificación de los módulos en forma independiente, estos fueron integrados para completar el producto final.

Las actividades de verificación y validación a las que se hace referencia corresponden a la etapa de pruebas del software. Según Garreta et al. (2003: 45) la verificación:

“es el conjunto de actividades para asegurar que un sistema o producto (informático) desarrolla o lleva a cabo una función específica con la máxima corrección. Por el contrario, se entiende a la validación como un conjunto de actividades para asegurar que un sistema o producto desarrolla o lleva a cabo la función especificada de acuerdo con los requisitos iniciales del cliente”.

Una vez terminadas estas actividades en cada módulo, el equipo realizó pruebas de integración para concluir la etapa de desarrollo. Alonso et al. (2005:87) menciona que la importancia de esta etapa de pruebas radica en que ésta constituye “un 40% del coste total del desarrollo del proyecto”.

En la etapa de implementación, se lleva a cabo la entrega del producto de software al usuario. Consiste en la explotación del producto en su entorno de ejecución. El riesgo de esta última actividad radica en el impacto que pueda tener en el usuario final. Lo anterior debido a que el usuario está acostumbrado al manejo de su información de cierta manera, lo que provoca una natural resistencia al cambio.

2.6. Evaluación de la materia

Para integrar la evaluación de la materia, el director solicita la entrega del proyecto terminado. Se monitoreó que cada actividad fuera realizada en tiempo y forma. Ante las dificultades presentadas, se observó la actitud de los estudiantes, y sobre todo las decisiones que tomaban para resolverlas. Todo esto en conjunto representaba puntos que se añadían a la puntuación asignada en la entrega del producto final. Es importante mencionar que en este caso, la experiencia previa de los estudiantes en semestres anteriores influyó en el resultado exitoso. Cuando los estudiantes no cuentan con experiencia en la resolución de problemas del mundo real ni con la experiencia de trabajar bajo en equipo, el desarrollo de las actividades sufre retrasos importantes.

1La Universidad se creó en octubre de 2000 y la Licenciatura en Informática inició en octubre de 2004. Dirección electrónica: www.unsis.edu.mx.

2Traducción propia tomada de Alhir (2003: 6).

Conclusión

Al aplicar la estrategia de Aprendizaje Basado en Proyectos, se incentiva la participación directa de los estudiantes en el proceso de enseñanza-aprendizaje, incrementando no sólo sus conocimientos; sino también sus habilidades, lo que será de utilidad en su futuro profesional.

A lo largo del desarrollo del proyecto, los estudiantes aplicaron conocimientos adquiridos en el aula. No solo aplicaron temas vistos en la asignatura de calidad de software, sino además temas de asignaturas que previamente habían cursado. Esto resultó benéfico debido que los estudiantes contaban con el conocimiento teórico relacionado a la problemática que se les presentaba en cada fase del proyecto.

Enfrentaron varios retos. Uno de ellos, debido a la inexistencia de un modelo a seguir en cuanto a la toma de decisiones. Es importante contar con un modelo para analizar cada alternativa de solución posible, ya que a partir de dicho análisis, es posible tomar la decisión más adecuada. Los estudiantes debían tomar decisiones en cada etapa de desarrollo. Cuando los estudiantes se enfrentaban a situaciones donde la decisión era más difícil de tomar, el profesor los apoyaba y guiaba, permitiendo que eligieran la mejor decisión entre las alternativas que tenían.

Este tipo de trabajo requiere un esfuerzo y compromiso mayor tanto por parte de los estudiantes, como del profesor, debido al tiempo que disponen para el desarrollo del software. Es importante resaltar el compromiso que se adquiere con este tipo de proyectos, en donde a diferencia del mundo laboral, donde el equipo de desarrollo está dedicado al 100%, en la escuela existen otras actividades que realizar. La carga de materias que cursan, las tareas y proyectos de las otras asignaturas, así como el trabajo extraescolar que deben realizar, complica planear las horas de trabajo.

Podemos concluir que la experiencia obtenida por los estudiantes en el presente proyecto, así como en trabajos realizados con anterioridad, les permite resolver con mayor rapidez y seguridad cada etapa de desarrollo. Se pretende desarrollar más

proyectos mediante esta estrategia, ya que a los estudiantes les permite comprender y valorar la importancia del trabajo en equipo, ver su aplicación en funcionamiento en una situación real, y mejorar el tiempo de desarrollo y la calidad de los productos.

Bibliografía

Alhir, S. S. (2003). Learning UML. Sebastopol, CA: O'Reilly & Associates.

Alonso, F., Martínez, L., & Segovia, F. J. (2005). Introducción a la Ingeniería de Software. Modelos de desarrollo de programas. España: Delta Publicaciones Universitarias.

Garreta, J. S., Rosalen, R. C., Simón, O. C., Manero, P. M., & Sancho, C. C. (2003). Ingeniería de proyectos informáticos: actividades y procedimientos. Castelló De La Plana: Publicacions de la Universitar Jaume I.

Hernández, C. (2010, Abril). Utilización del trabajo por proyectos para incentivar la Innovación Tecnológica en los estudiantes universitarios. HEKADEMUS. Revista Científica de la Fundación Iberoamericana para la Excelencia Educativa, 03(08), 42-54.

Jiménez-Pérez, H., Ramírez-Vásquez, S.K., López-Salinas, S., Juárez-López, Leyda y Ramírez-Ramírez, R. (2010). CONAB: Resultado de una actividad de evaluación en Ingeniería de Software. Congreso Nacional de Software Libre (CONSOL). México, D.F.

Kendall, K. E., & Kendall, J. E. (2005). Análisis y Diseño de Sistemas. (6ª ed.). México: Pearson Educación.

López, S.E., Juárez, L., Ramírez, R. (Certificado del Registro Público del Derecho de Autor: 03-2009-040310520900-01, 2009). CONAB. Control de Acceso a Biblioteca

(Versión 1.0) [Software de Cómputo]. Miahuatlán de Porfirio Díaz, Oax., México: Universidad de la Sierra Sur.

Maldonado, M. (2008, Septiembre). Aprendizaje Basado en Proyectos Colaborativos. Una experiencia en educación superior [Versión electrónica]. *Laurus*, 14(28), 158-180.

Pressman, R. S. (2005). *Ingeniería de Software. Un enfoque práctico* (6ª ed.). México: McGraw-Hill.

Sommerville, I. (2005). *Ingeniería del Software* (7ª ed.). España: Pearson Educación.

StarUML - The Open Source UML/MDA Platform. Recuperado el 10 de agosto de 2011 de <http://staruml.sourceforge.net/en/>.

Zavala, A., & Arnau, L. (2007). *Cómo aprender y enseñar competencias*. España: Editorial GRAÓ.

Reseña curricular

Jiménez, Homero, Ramírez Vásquez, Sandra Karina

Jiménez-Pérez, H.

Maestro en Ciencias de la Computación por el Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET).

Licenciado en Informática por el Instituto Tecnológico de Oaxaca (ITO).

Profesor-Investigadora de la Licenciatura en Informática en la Universidad de la Sierra Sur (UNISIS). Julio de 2006 a la fecha.

Desarrollador de Software. Grupo Ámbar. Cuernavaca Morelos. Enero 2005 – Diciembre 2005.

Ramírez-Vásquez, S.K.

Estudiante del Doctorado de Gobierno Electrónico en la Universidad de la Sierra Sur (UNSI).

Maestra en Electrónica y Computación por la Universidad Tecnológica de la Mixteca (UTM).

Licenciada en Informática, Instituto Tecnológico de Oaxaca (ITO).

Proyectos de investigación: “Desarrollo e Implantación de Software en la UNSI” y “Desarrollo de Metodologías y Aplicaciones de Minería de Datos para Gobierno Electrónico”

Profesora-Investigadora de la Licenciatura en Informática en la Universidad de la Sierra Sur. Julio de 2004 a la fecha.

Profesora-Investigadora en la Universidad del Mar. Agosto de 2001 a Julio de 2004.

Catedrática en el Instituto Tecnológico de Oaxaca. Enero de 2000 a julio de 2000

Jefa de Carrera de la Licenciatura en Informática. Universidad de la Sierra Sur. Diciembre de 2004 a Julio de 2011.