

Base de Conocimiento de los procesos de desarrollo de software a través de un modelo de un sistema de gestión del conocimiento

Sergio Ruiz-Castilla¹, Yulia Ledeneva² y Rocío Morales¹

¹ Universidad Popular Autónoma del Estado de Puebla,
21 Sur 1103, Barrio Santiago, C.P. 72410, Puebla, México
jsorgioruizc@gmail.com, mariadelrocio.morales@upaep.mx

² Universidad Autónoma del Estado de México,
Unidad Académica Profesional Tianguistenco,
Paraje el Tejocote San Pedro Tlantizapan, 52600, Estado de México
yledeneva@yahoo.com

Resumen En la industria del software las organizaciones desarrollan proyectos de software por desarrolladores expertos con conocimiento individual, capitalizando conocimiento colectivo. En este artículo, se propone un modelo de un sistema de gestión del conocimiento para los procesos de desarrollo de software en México, para transferir el conocimiento explícito de los desarrolladores expertos a los no expertos. El sistema de gestión del conocimiento propuesta permite la captura y almacenamiento de activos de conocimiento en una base de conocimiento para su modelado y validación; por otro lado, se plantea la consulta de conocimiento por parte de los desarrolladores no expertos para la transferencia y reutilización del conocimiento. Se acentúan en el proceso de la transición del conocimiento tácito a explícito y culturalizar la voluntad de aportar conocimiento. El objetivo de este artículo consiste en desarrollar una aplicación para un sistema de gestión del conocimiento para la gestión y la consulta del conocimiento.

Palabras claves: Conocimiento tácito, conocimiento explícito, gestión del conocimiento, procesos de desarrollo de software, base de conocimiento.

1. Introducción

Al desarrollar proyectos de software se genera experiencia en los desarrolladores que se capitaliza en las mentes de los desarrolladores como conocimiento individual y en la organización se forja conocimiento colectivo que se debería de compartir para evitar que se tenga que gestionar en cada nuevo proyecto y organización.

El problema que se plantea en este artículo es como transferir el conocimiento de gestión de los Procesos de Desarrollo de Software (PDS) de los desarrolladores expertos y los no expertos. El conocimiento tácito en la mente de los desarrolladores expertos se traduce a conocimiento explícito para poder almacenarse en una base de conocimiento.

Los proyectos de software requieren de una serie de procesos de desarrollo que en poca o gran medida se repiten en otros proyectos, bajo el mismo u otro contexto. La ingeniería de software plantea la reutilización de código, pero es necesario evaluar la reutilización de conocimiento de los PDS.

Se propone un modelo de un Sistema de Gestión de Conocimiento (SGC) que permite la obtención del conocimiento de los PDS de los desarrolladores expertos para ser almacenado en una Base de Conocimiento (BC). Por lo anterior, se desarrolla una herramienta de software que permite el almacenamiento y consulta de activos de conocimiento con ayuda de un motor de búsqueda usando consultas en lenguaje natural por los desarrolladores no expertos.

Cuando el líder de proyectos de software debe desarrollar un proyecto, gestiona los requerimientos del proyecto y los almacena en un banco de requerimientos. Después diseña el software basado en los requerimientos generando experiencia en los desarrolladores de esos requerimientos. Cuando se inicia un nuevo proyecto, un bajo o alto porcentaje de esos requerimientos se repetirán en el mismo o diferente contexto.

La organización cada vez será más experta aprovechando el conocimiento colectivo de su capital humano, sin embargo existirán miembros que se incorporarán a la organización que carecen de los conocimientos que tiene la organización. Tomando en cuenta también los desarrolladores que deciden irse de la organización, que se marchan con el conocimiento tácito consigo mismo.

En este artículo, se plantean el siguiente objetivo: diseñar un modelo de un Sistema de Gestión del Conocimiento que permita la obtención y almacenamiento de activos de conocimiento de los Procesos de Desarrollo de Software en una Base de Conocimiento para transferir y reutilizar dicho conocimiento parcial o completamente en nuevos proyectos de software.

2. Conocimiento

El conocimiento es una mezcla fluida de experiencia estructurada, valores, información contextual e internalización experta que proporciona un marco para la evaluación e incorporación de nuevas experiencias e información. Se origina y se aplica en la mente de los conocedores. En las organizaciones con frecuencia no solo queda arraigado en documentos o bases de datos, sino también en las rutinas, procesos, prácticas y normas institucionales [1].

Según Hessen [2] el conocimiento puede definirse como una transferencia de las propiedades del objeto al sujeto. Considerando que cada sujeto puede tener una diferente perspectiva del objeto. A medida que se conoce algo se logra cierto nivel de abstracción, por lo tanto a mayor nivel de abstracción mayor conocimiento se tiene del objeto.

Por otro lado, Alonso *et. al* [3] determina que el conocimiento es el conjunto completo de datos e información que se usa en la práctica para realizar ciertas acciones y crear nueva información.

2.1. Contexto del conocimiento

Los individuos tienen conocimiento individual que se incrementa día a día gracias a sus experiencias. El conocimiento individual se capitaliza en el grupo de trabajo donde está integrado. El conocimiento puede capitalizarse en las organizaciones de un consorcio o en un corporativo. Finalmente, el conocimiento de todas las organizaciones se capitaliza como el conocimiento de un país entero como se muestra en la Figura 1.

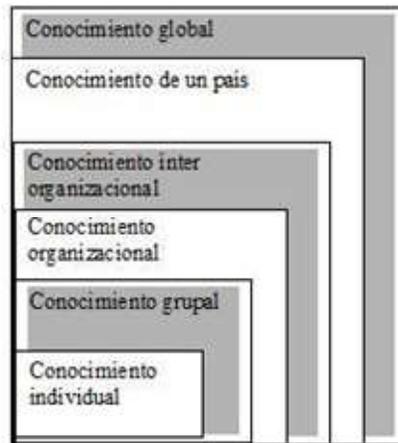


Figura 1. Dimensiones del conocimiento [4].

La propia ISO ocupada en la importancia que tiene la gestión del conocimiento en el apartado 12207 asevera que los recursos humanos de la organización son los encargados de asegurarse que el conocimiento, la información y las habilidades individuales se deben almacenar, reutilizar y mejorar con el tiempo [5]. Es decir, se sigue que las experiencias transformadas en conocimiento se gestionen como activos de la organización que tuvo un costo para generarse y que tiene un alto valor al aplicarse. Si bien el conocimiento no es tangible no implica que no exista, además tiene un valor e incluso se comercializa Davenport y Prusak manejan que la comercialización del conocimiento es una actividad en las organizaciones en forma de cursos y talleres de capacitación y en asesorías de consultores, el precio del conocimiento se establece cuando el proveedor le pone un precio y el cliente está dispuesto a pagar, considerando el beneficio que espera obtener con el nuevo conocimiento [1].

La gestión del conocimiento es una nueva disciplina que estudia las herramientas, técnicas y procedimientos para aprovechar al máximo potencial de los recursos humanos de la organización [6]. Se deberá considerar la experiencia, pericia y el aprendizaje para lograr una ventaja competitiva.

2.2. Conocimiento tácito

El conocimiento se puede clasificar en conocimiento tácito y conocimiento explícito. El conocimiento tácito es aquel que reside en la mente de los individuos y se caracteriza por ser difícil de codificar, formalizar y transmitir [6]. Por tanto, los expertos son los que poseen el conocimiento y le llevan consigo mismo. La propia complejidad de la transferencia del conocimiento tácito puede ser la razón por la cual no se documenta el conocimiento en las organizaciones.

2.3. Conocimiento explícito

El conocimiento explícito como se refiere Gómez y Suárez [6] es el conocimiento que se puede transmitir directamente mediante el lenguaje formal y en forma sistemática. También podemos encontrar este conocimiento en los libros. Es el conocimiento más abundante en forma de textos en las bibliotecas, enciclopedias, páginas Web, y otros medios; y es que tiene mayor transferencia debido a que está documentado y disponible (ver la Figura 2).



Figura 2. El ciclo conocimiento implícito – explícito [8].

3. Sistema de gestión del conocimiento

Un sistema de gestión del conocimiento tiene la misión de coleccionar, almacenar y permitir la consulta del conocimiento en una organización o comunidad con el fin de capitalizar el conocimiento como estrategia competitiva.

3.1. Base de conocimiento

Para almacenar el conocimiento es necesario un repositorio de archivos de los contenidos de los activos de conocimiento. Se usa una base de datos para los datos (metadatos) de los activos de conocimiento. Los anteriores pueden ser textos, aunque pueden contener imágenes. Pueden contener también archivos de sonido o videos. La Figura 3 muestra la arquitectura de la base de conocimiento. Cada activo de conocimiento tiene dos elementos:

- Los datos de los activo de conocimiento.
- Las carpetas que contienen los archivos de los activos de conocimiento clasificados.
- Los datos de los activos de conocimiento tienen la función e índices para el buscador de conocimiento.
- Las carpetas contenedoras de archivos de los activos de conocimiento, tienen la función de almacenar los archivos y su límite es la capacidad del disco duro u otros dispositivos de almacenamiento.

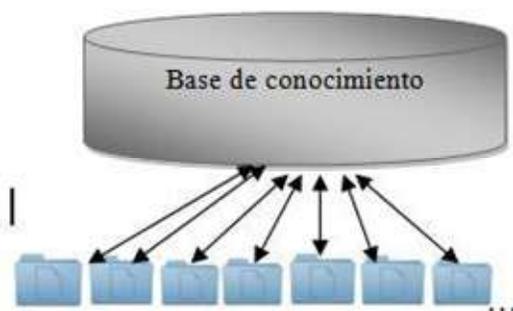


Figura 3. Propuesta de la arquitectura de la base de conocimiento.

El contenido de la base de conocimiento está constituido por el conjunto de archivos de activos de conocimiento:

$$CBC = CAAC1 + CAAC2 + CAAC3 + \dots \quad (1)$$

donde CBC es el contenido de la base de conocimiento y CAACn es el contenido de los archivos de los activos de conocimiento.

4. Administración del conocimiento

Es una actividad en el campo de actividades profesionales y empresariales que se centra en crear, recopilar, organizar y difundir el conocimiento de una organización, en lugar de la información o los datos [7].

La OCDE (Organización para la Cooperación y el Desarrollo Económico) a través del CIIE (Centro para la Investigación y la Innovación Educativa) define un modelo lineal (ver Figura 4) que consiste en 7 etapas [9]:



Figura 4. Modelo lineal de 7 pasos [9].

1. **Producción.** Se refiere a la producción de conocimiento de los individuos y de las organizaciones.
2. **Validación.** Se centra en que el conocimiento se haya aplicado de preferencia varias veces con resultados exitosos.
3. **Cotejo.** Se verifica que el conocimiento que se dispone realmente coincide con el conocimiento validado.
4. **Difusión.** Se determina cuáles son los medios a través de los cuales se hará la publicación e intercambio de conocimiento entre otros miembros de la organización, red o sociedad.
5. **Adopción.** Se motiva a los elementos de la organización para que abandone viejas prácticas y use el nuevo conocimiento que se le ha proporcionado.
6. **Implementación.** La mayoría de elementos de la organización se adoptan el conocimiento y existen las condiciones para dicha adopción.
7. **Institucionalización.** Se logran que la implementación del conocimiento sea algo normal. Se logran que sea una cultura de la organización en bien de todos [9].

5. Modelo propuesto

El modelo propuesto se muestra en la Figura 5. El módulo I, la Gestión de Proyectos (GP), se repite en cada proyecto que se desarrolla, de acuerdo a la metodología elegida y a los estándares internacionales para el desarrollo del software. Se repite el proceso, más no el proyecto, por lo que cada proyecto genera conocimiento nuevo. El equipo de desarrollo aplica conocimiento individual y colectivo para llevar a cabo cada tarea del proyecto. Cuando se finaliza un proyecto,

nacen nuevas experiencias que son conocimiento tácito adquirido. A mayor número de experiencias mayor conocimiento tácito individual y colectivo.

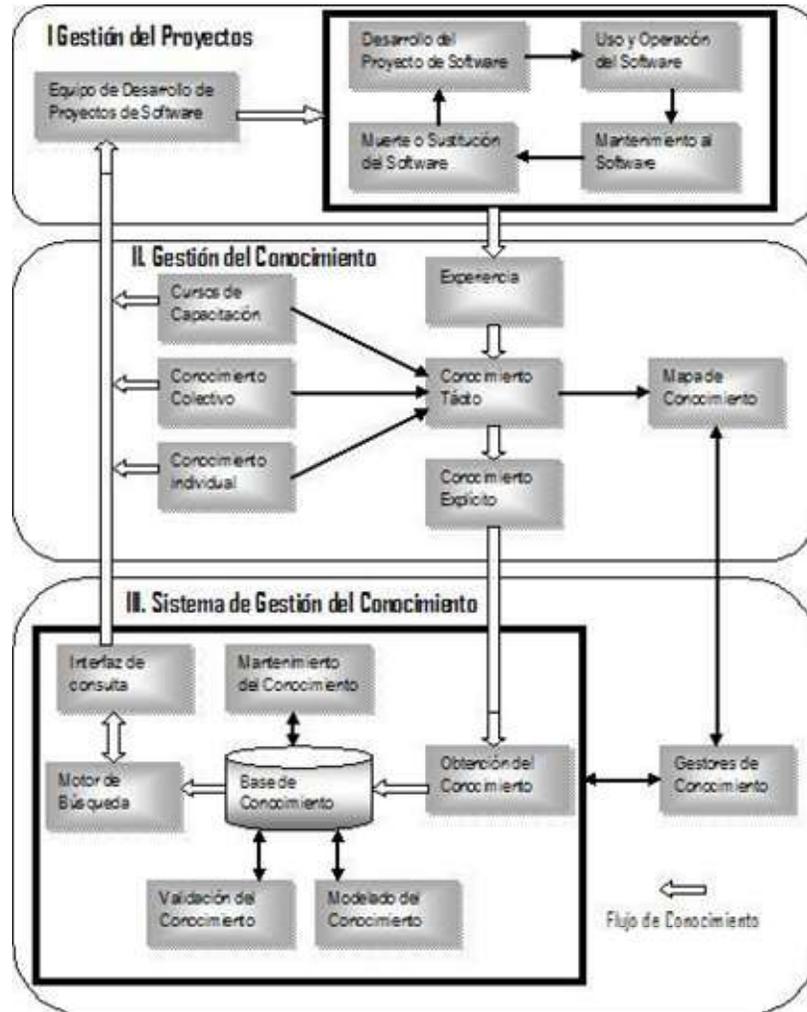


Figura 2. Propuesta del modelo del sistema de gestión del conocimiento para los procesos de desarrollo de software.

En el módulo II de la Figura 5 se muestra el proceso de la Gestión del Conocimiento (GC) referido a la tarea de generación, almacenamiento y uso del conocimiento. El conocimiento tácito se acumula de las experiencias y otras fuentes como, por ejemplo, la capacitación. Para almacenar el conocimiento es necesario

convertirlo del conocimiento tácito a explícito. La gestión del conocimiento es una tarea más que una herramienta tecnológica. Se requiere de cultura de gestión del conocimiento, que determina la importancia y la necesidad de capturar el conocimiento, almacenarse y disponerse para su uso a todos los desarrolladores.

Por último, en el módulo III se muestra el Sistema de Gestión del Conocimiento (SGC) que es la plataforma que se propone para la GC. El SGC facilita la obtención, almacenamiento y uso del conocimiento del conocimiento generado de la ejecución de los procesos al desarrollar el software. Contiene una BC que almacena y permite consultar conocimiento a través de un motor de consulta. La base de conocimiento que se plantea no es una base de conocimiento de reglas, sino más bien una especie de repositorio de activos de conocimiento. Es preciso además que el conocimiento se modele y se valide con el fin de mantener en la BC solo conocimiento de calidad.

En el modelo propuesto (ver la Figura 5) se puede observar una flecha gruesa del flujo del conocimiento que además tiene un ciclo que se repite acumulando más conocimiento organizacional. Una organización con la puesta en práctica del modelo le permite acumular conocimiento a través del tiempo y de explotarse de forma adecuada puede lograr mayor productividad y competitividad. El conocimiento que se administra con el SGC puede ser de carácter privado con acceso restringido o bien puede ser abierto para una comunidad, de acuerdo al tipo de conocimiento que se trate y de las políticas de la organización.

La BC es un conjunto de documentos que forman un compendio de Activos de Conocimiento (AC) que el motor de consultas usará para recuperar el conocimiento consultado por los desarrolladores no expertos. El potencial de la herramienta puede centrarse en el motor de búsqueda, considerando que el conocimiento se presenta como información no estructurada [10-13].

6. Implementación y validación del modelo

Una aplicación de software permite la gestión del conocimiento. El conocimiento tácito se obtiene de los desarrolladores expertos como se muestra en la Figura 6. Los gestores del conocimiento son los que se encargan de obtener y convertir el conocimiento tácito en explícito. Una de las formas para obtener el conocimiento es a través de relatos de las experiencias. Otra forma es cuando el desarrollador experto escribe un texto describiendo el conocimiento adquirido. Otros ejemplos pueden ser experiencias, prácticas, lecciones aprendidas, problemas y soluciones presentados en forma de archivos.

Los textos son activos de conocimiento de la organización que son integrados a la base de conocimiento como archivos de textos (ver la Figura 6). Es preciso catalogar cada activo de conocimiento para su clasificación dentro de la BC. Una vez almacenado el conocimiento se podrá modelar revisando el formato del texto y depurar errores de texto. El software permite que los usuarios del conocimiento califiquen el conocimiento en función de la utilidad obtenida, la validación la hacen entonces los propios usuarios. Cuando un activo de conocimiento obtiene calificaciones muy bajas será eliminado de forma automática. El conocimiento almacenado se puede consultar mediante el motor de consulta, mediante una palabra o

frase del activo de conocimiento que necesita el usuario del conocimiento (ver la Figura 7 y 8).

Es posible introducir conocimiento en formato explícito, en formato de texto que se almacenó en la base de conocimientos. El conocimiento corresponde a temas sobre los procesos de software. Los activos de conocimiento pueden revisarse en su contenido y también para comprobar su valor. Los usuarios del conocimiento pueden acceder y hacer consultas de conocimiento usando una palabra o una frase completa. La herramienta muestra un conjunto de activos de conocimiento disponibles en la base de conocimiento. El usuario del conocimiento elige algún activo de conocimiento y puede leer su contenido. Después de la consulta puede asignar una calificación con el fin de obtener un ranqueo del activo de conocimiento.

Grabar el componente Anular

Núm. de Componente: Tipo de proceso: Calendarización del proyecto

Tipo de Conocimiento: Caso de éxito Desarrollador experto: José José González Hernández

Resumen del componente de conocimiento:

Se consideraron factores del contexto para hacer el calendario del proyecto. Se manejó una holgura media para no castigar el costo del proyecto. Se logró un ajuste entre los factores contextuales y la holgura. La fecha de culminación y la planeada tuvo una diferencia de solo un día.

Contenido del componente de conocimiento:

La administración de software abarca la planeación, calendarización, administración de negocios, manejo del personal, estimación de los costos de software y la administración de calidad.

En este artículo se cubre la planeación y la calendarización de proyectos de software. Muchas son las causas para el fracaso de proyectos de software, se pueden mencionar: entrega tardía, no fiable, costo superior al estimado, características de ejecución pobres. Muchas veces la falla estaba en el enfoque de administración utilizado.

La necesidad de administrar es una distinción importante entre un desarrollo profesional de software y la programación no profesional. La administración de proyectos de software es necesaria debido a que la ingeniería de software profesional siempre está sujeta a restricciones de presupuesto y calendarización, a las que debe ajustarse la organización que desarrolla el software. El trabajo del

Figura 3. Incorporación de un activo de conocimiento.

Consulta de conocimiento

Introducir la(s) palabra(s) del conocimiento a buscar: Buscar...

IdComponenteDeC	IdTipoDeProceso	IdTipoDeConocim	IDDesarrolladorExp	cdcResumen	cdcEstadoDeMod	cdcEstadoDeEva	cdcCalificacion
1	3	4	2	Estimación de costos ..			
*							

Figura 4. Consulta de activos de conocimiento.



Figura 5. Visualización del activo de conocimiento.

7. Conclusiones

El conocimiento es un activo intangible en las organizaciones, sin embargo su valor en la industria del software es el activo más valioso. El conocimiento está en mente de los trabajadores, por lo que es necesario almacenarlo y disponerlo en una plataforma que permita su consulta en nuevos proyectos de desarrollo de software. A lo largo del tiempo los desarrolladores pueden aportar conocimiento que se acumula como conocimiento organizacional, bajo la premisa que se irán los trabajadores, pero el conocimiento se queda en la base de conocimiento. Una organización puede tener más productividad y competitividad en medida que el conocimiento que se gestiona. Es necesaria una cultura de la gestión del conocimiento para lograr su aportación, almacenamiento y utilización, bajo el convencimiento que es una medida que beneficia a toda la industria del software.

Referencias

1. Davenport, T. y Prusak, L.: Cómo las organizaciones manejan lo que saben. Buenos Aires: McGraw Hill. pp. 6 (2001)
2. Hessen, J.: Teoría del conocimiento. México: Época. pp. 20 (2008)
3. Alonso B. Amparo, Guijarro B. Bertha, Lozano T. Adolfo, Palma M. José Tomás, Taboada I. Ma. Jesús: Ingeniería del Conocimiento Aspectos Metodológicos, España: Pearson Prentice Hall. pp. 18 (2004)
4. Nonaka. I. Y Takeuchi, H.: La organización creadora del conocimiento. México: Oxford University Press (1999)
5. IEEE. IEEE/EIA 12207.1-1997. Industry Implementation of ISO/IEC 12207:1995 – Standard for Information Technology – Software Life Cycle Process (1997)

6. Gómez, A. Y Suárez, C.: *Sistemas de Información, Herramientas prácticas para la gestión empresarial*, (2a. ed.). México: Alfaomega Ra-Ma (2007)
7. Barnes S.: *Sistemas de Gestión del Conocimiento Teoría y Práctica*. España: Thompson Editores (2002)
8. Nonaka: *The Knowledge Creating Company*. Harvard Business Review. November – December 1991. Págs. 96-104 (1991)
9. OCDE y la Sociedad del Conocimiento FLACSO-México. *La administración del conocimiento en la sociedad del conocimiento*. Mayol Ediciones: Colombia (2006)
10. García-Hernández, R.A., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: *A Fast Algorithm to Find All the Maximal Frequent Sequences in a Text*, In: Sanfeliu, A., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A. (eds.) CIARP 2004. LNCS vol. 3287, pp. 478-486. Springer-Verlag (2004)
11. García-Hernández, R.A., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: *A New Algorithm for Fast Discovery of Maximal Sequential Patterns in a Document Collection*. In: Gelbukh, A. (ed.) CICLing 2006, LNCS vol. 3878, pp. 514–523, Springer-Verlag (2006)
12. Sidorov, G.: *Lemmatization in automatized system for compilation of personal style dictionaries of literary writers*. In: “Word of Dostoyevsky”, Russian Academy of Sciences, pp. 266-300, (1996)
13. Gelbukh, A. and Sidorov, G.: *Approach to construction of automatic morphological analysis systems for inflective languages with little effort*. Lecture Notes in Computer Science, N 2588, Springer-Verlag, pp. 215–220 (2003)
Gelbukh, A. and Sidorov, G.: *Analizador morfológico disponible: un recurso importante para PLN en español*. En: *Memorias de talleres del congreso internacional Iberamia-2004*, Puebla, México, noviembre 2004, pp. 209-216 (2004)