

Expresando Nuestra Manera de Trabajo con KUALI-BEH: Lecciones Aprendidas por Tic-Tac-S

Emmanuel Arroyo-López¹, Teresa Ríos-Silva¹, Miguel Morales-Trujillo²,
Alejandro Rico-Martínez¹ y Hanna Oktaba²

¹Instituto Tecnológico Superior de San Luis Potosí, Capital, San Luis Potosí, México

²Grupo de Investigación KUALI-KAANS, Universidad Nacional Autónoma de México,
Ciudad de México, México

{emmanuel.arroyo, teresa.rios, alejandro.rico}@tecsuperiorslp.edu.mx
{migmor, hanna.oktaba}@ciencias.unam.mx

Resumen

La incorporación de estudiantes recién egresados al campo laboral puede resultar en un proceso complicado, principalmente por la poca experiencia con la que cuenta el estudiante. En busca de reducir esta brecha, el Instituto Tecnológico Superior de San Luis Potosí, Capital, ha desarrollado un programa de incubación de células de desarrollo de software, conformadas por recién egresados de la carrera de Ingeniería en Sistemas Computacionales. Estas células incursionan en el desarrollo de proyectos de software con clientes reales. En este artículo se describe la experiencia de la célula Tic-Tac-S y la aplicación de KUALI-BEH como marco de trabajo para la definición, mejora y ejecución de un método de desarrollo de software propio, acorde a sus necesidades y contexto.

Palabras clave: Células de desarrollo, incubación, método, KUALI-BEH.

1. Introducción

La ejecución de proyectos de desarrollo de software requiere que los integrantes del equipo de trabajo involucrado apliquen sus conocimientos y habilidades durante un periodo de tiempo establecido, de tal manera que el trabajo realizado permita el cumplimiento de los objetivos establecidos. Para lograr lo anterior, los equipos de trabajo siguen procesos de desarrollo de software apoyados en diversas metodologías, las cuales proporcionan una estructura adecuada o recomendada de la forma de trabajo a los participantes del proyecto.

Sin embargo, la problemática principal que enfrentan los estudiantes recién egresados que se integran a un ámbito laboral, es la falta de experiencia, en particular en el desarrollo de software. Situación que no es particular de una zona o contexto específico.

El Instituto Tecnológico Superior de San Luis Potosí, Capital (ITSSLP-C) no es ajeno a este fenómeno, es por ello que en busca de generar una ventaja competitiva a sus egresados, decidió implementar un proyecto para la conformación de células de desarrollo de software, formadas por alumnos, para que éstos adquieran experiencia en proyectos reales. Esta iniciativa cuenta con el apoyo de los departamentos de Vinculación y de académicos de la carrera de Ingeniería en Sistemas Computacionales (ISC) del Tecnológico.

A partir del surgimiento de la iniciativa, se formó el primer grupo de desarrollo de software: Tic-Tac-S. Tic-Tac-S es una célula de desarrollo de software conformada por egresados de la carrera de ISC del ITSSLP-C. Dicha célula es guiada por profesores de la ISC en la realización de proyectos con clientes reales.

Sin embargo, conforme el primer proyecto asignado a Tic-Tac-S avanzaba, el equipo de trabajo comenzó a enfrentarse al hecho de que la metodología usada se presentaba como una forma de trabajo estricta y pesada, con poca posibilidad de adaptarse al ritmo y hábitos del grupo, dando como resultado la imposibilidad de definir un proceso acorde a las características y necesidades de la célula.

Lo anterior conllevó a incurrir en el uso de malos hábitos como: no documentar, “saltarse” pasos para el análisis e ir directo a la codificación, no establecer roles ni funciones. Con el paso del tiempo se volvió cada vez más complicada la corrección de la forma de

trabajo, provocando que se generaran productos con poca calidad y los tiempos de desarrollo se extendieran fuera de lo establecido.

De tal forma el grupo Tic-Tac-S, comenzó la búsqueda y exploración de alternativas que permitieran la mejora de la célula. En conjunto con profesores del ISC la célula tomó la decisión de definir un proceso propio utilizando al marco de trabajo KUALI-BEH [1]. La elección de este marco de trabajo se basó en las características ofrecidas por KUALI-BEH ya que se adaptaban a las necesidades de Tic-Tac-S, un equipo pequeño y sin un proceso explícito. La flexibilidad para definir un método propio y la viabilidad para la inclusión y aplicación de buenas prácticas obtenidas de estándares y modelos fueron las razones fundamentales para la elección de KUALI-BEH

Este artículo tiene como objetivo describir la experiencia de aplicar KUALI-BEH dentro de la célula de desarrollo de software Tic-Tac-S para la definición y mejora de un método propio. Esta experiencia de colaboración entre Tic-Tac-S y el grupo de investigación KUALI-KAANS, creadores de KUALI-BEH, se gestó entre 2013 y 2014.

El presente documento se organiza de la siguiente manera: en la sección 2 se presentan los antecedentes tanto de Tic-Tac-S como de KUALI-BEH, en la sección 3 se describe la implementación de KUALI-BEH en el grupo de desarrollo de software Tic-Tac-S, en la sección 4 se presentan los resultados de esta experiencia. En la sección 5 se muestran las lecciones aprendidas y para finalizar en la sección 6 se concluye.

2. Antecedentes

En esta sección se describirán tanto el contexto de Tic-Tac-S dentro del ITSSLP-C, como el origen de KUALI-BEH y los elementos que resultaron más importantes para las necesidades de Tic-Tac-S.

2.1 El ITSSLP-C y Tic-Tac-S

El ITSSLP-C fue creado el 05 de julio del 2003 [2]. En la actualidad oferta las carreras de: Ingeniería en Administración; Ingeniería en Mecatrónica; Ingeniería Industrial; e Ingeniería en Sistemas Computacionales. La población estudiantil al año 2015 es de 1400 alumnos, de los cuales 250 son del área de sistemas computacionales con 7 generaciones de egresados [2].

Son profesores de esta carrera los interesados en incubar células de desarrollo de software con estudiantes recién egresados. Con la intención de que los alumnos, que formen parte de estas células, participen de manera activa en proyectos con clientes

reales. Este enfoque intenta proveer a los alumnos de un panorama más cercano a lo que se enfrentarán una vez fuera de la carrera.

Como se mencionó anteriormente, Tic-Tac-S es una de éstas células de desarrollo de software. Tic-Tac-S en su inicio estuvo conformada por 4 alumnos que cursaban 8° o 9° semestres. Los roles de mayor peso, como el líder técnico y administrador de proyectos fueron tomados por 2 profesores del ITSSLP-C.

La primera experiencia de esta célula fue el desarrollo del proyecto *Sistema administrativo* para una institución de educación superior. Este proyecto tenía como objetivo desarrollar los módulos del departamento de control escolar de dicha institución y fue desarrollado del mes de Abril del 2012 al mes de Octubre del 2013. Si bien, al término de este proyecto se logró la implementación de los módulos requeridos, se dejó ver la clara necesidad de contar con un proceso propio que se ajustara al tamaño del equipo, la experiencia y conocimiento de sus integrantes y que fuera lo suficientemente completo para atender de manera adecuada los proyectos del cliente.

Es por estas razones que surge el interés del grupo de trabajo en buscar una alternativa para expresar las maneras de trabajo reales del equipo.

2.2. Trabajos relacionados

En la literatura existen algunas alternativas que permiten la definición y modelado de procesos de software. Por ejemplo, el Object Management Group¹ (OMG) tiene definido desde 2008 la versión 2.0 del Software and Systems Process Engineering Meta-model (SPEM) [3], que es tanto un marco de trabajo como un meta modelo de procesos de ingeniería. SPEM describe un lenguaje y un esquema de representación que expresa formalmente métodos y procesos reutilizando algunos aspectos del Lenguaje de Modelado Unificado (UML) [4].

Sin embargo, con este enfoque sigue existiendo la necesidad de especificar los fundamentos, tanto teóricos como del contexto de uso, de esas prácticas, que en el contexto de la célula de desarrollo aún no estaban identificados.

Por otra parte Humphrey propuso en [5] un proceso para equipos de alto desempeño que desarrollan software, dividido en cuatro grupos de actividades: preparación de las tareas, tareas de desarrollo, tareas de administración y actividades de evaluación.

Sin embargo, dada la generalidad del modelo, es difícil saber qué prácticas serían las adecuadas para

¹ Object Management Group, <http://www.omg.org>

poder construir los métodos asociados al proceso y llevarlo a cabo. Más aún, resulta complejo delimitar que puede ser incluido dentro de cada grupo y que siga considerándose una actividad válida.

Además de estas propuestas se pueden mencionar a modelos de referencia de procesos, tales como CMMI [6], o estándares ISO/IEC como lo son el 24744 [7], 12207 [8] y 29110 [9]. No obstante la naturaleza de estos compendios de buenas prácticas de la industria no se ajustaba a la de la célula de desarrollo, razón por la cual se decidió investigar otras alternativas, búsqueda que desembocó en KUALI-BEH.

2.3. KUALI-BEH

KUALI-BEH es un marco de trabajo para la expresión, comparación y mejora de las formas de trabajo de los practicantes de Ingeniería de Software involucrados en proyectos de software. Este marco de trabajo forma parte del estándar *ESSENCE – Kernel and Language for Software Engineering Methods* [10] del OMG.

KUALI-BEH describe los conceptos comunes y relaciones presentes en cualquier proyecto de software [1]. Los ingenieros de software, o practicantes, que están involucrados activamente en proyectos de software son el principal objetivo de este marco de trabajo. KUALI-BEH consta de 2 componentes principales:

- **Vista Estática:** A través de conceptos comunes presentes en cualquier proyecto de software, se

definen los elementos básicos para construir prácticas que definan la forma de trabajo de un equipo de trabajo. Permitiendo a sus integrantes expresar un método organizado que cumpla con ciertas propiedades.

- **Vista Operacional:** Ésta abarca la ejecución de los métodos definidos mediante los elementos de la vista estática.

KUALI-BEH es una herramienta que permite la estandarización de términos y conceptos para la ejecución de proyectos de software. Además define una estructura de trabajo enfocada a los conocimientos y habilidades que posee cada participante dentro de un proyecto de software.

KUALI-BEH guía a los participantes en proyectos de software a un cambio estratégico sobre sus hábitos de desarrollo, para ser adaptados a una forma de trabajo estructurada, la cual permita sustentar con una base de conocimiento que pueda ser útil para los involucrados en el proyecto.

Al usar KUALI-BEH el grupo Tic-Tac-S buscó cubrir la necesidad de contar con un método propio que se adaptara a su forma de trabajo, la cual está definida directamente por sus integrantes y su característica principal es la de ser recién egresados con poca experiencia en el desarrollo de software. Para fines de este artículo, la percepción de Tic-Tac-S sobre este marco de trabajo es presentada en la Figura 1.

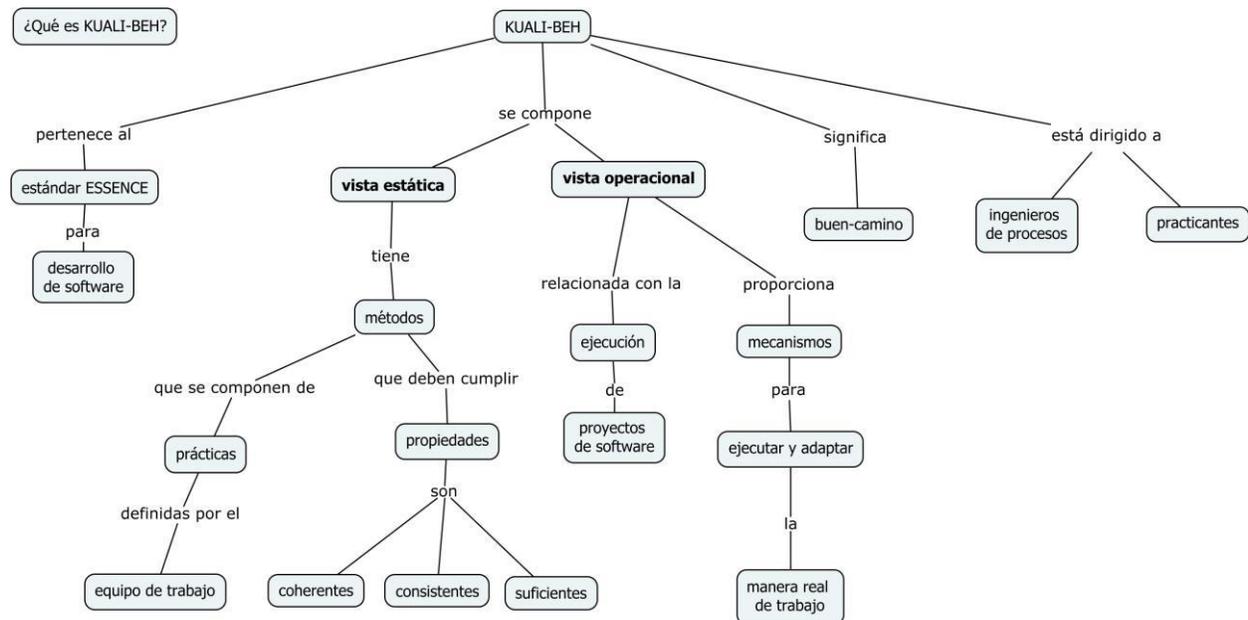


Fig. 1: Mapa conceptual sobre KUALI-BEH.

3. Implementación de KUALI-BEH en una célula de desarrollo del ITSSLP

Un webinar ofrecido por Software Gurú² fue el medio por el cual Tic-Tac-S tuvo el primer contacto con KUALI-BEH, posterior a ello, se dio un primer acercamiento entre el ITSSLP-C y KUALI-KAANS para formalizar una colaboración que permitiera a Tic-Tac-S implementar el marco de trabajo y así atender las necesidades identificadas.

La colaboración se planeó a través de un caso de estudio que constó de 3 fases que permitieran alcanzar los siguientes objetivos:

1. Demostrar la suficiencia de los elementos que conforman una Práctica para describir la manera de trabajo de los practicantes.
2. Medir la factibilidad para el practicante de expresar sus prácticas tácitas utilizando el concepto de Práctica.
3. Identificar el valor obtenido para la organización consecuencia de definir sus propias Prácticas.
4. Medir el esfuerzo requerido para la capacitación del equipo de trabajo.

En las siguientes subsecciones se describen cada una de las fases del caso de estudio.

3.1. Fase 1

La fase 1 del caso de estudio se realizó en el periodo que comprendió del 30 de Abril al 02 de Octubre del 2013 y tenía como objetivos:

- Identificar la manera de trabajo de los practicantes a ser modelada como Práctica(s)
- Expresar la manera de trabajo de los practicantes como Práctica(s)
- Acordar que la(s) Práctica(s) representa(n) la manera de trabajo de los practicantes

Una vez que el grupo de trabajo Tic-Tac-S conoció los objetivos del caso de estudio éstos se dieron a la tarea de estudiar KUALI-BEH para adquirir y comprender a mayor detalle sus elementos. Actividad que involucró una sesión-taller en línea entre Tic-Tac-S y KUALI-KAANS, en donde se clarificaron los conceptos contenidos en la documentación oficial y se dio respuesta a algunas dudas surgidas durante la primera actividad de trabajo.

Además se presentó la plantilla de Práctica de KUALI-BEH como ejemplo para comenzar con la

definición de la manera de trabajo real de Tic-Tac-S. La plantilla de práctica fue, a partir de ese momento, el mecanismo principal para expresar las maneras de trabajo de Tic-Tac-S. Derivado de esta actividad se creó en conjunto con los investigadores la primera práctica de Tic-Tac-S, ver Figura 2.

Práctica			
1.3			
Análisis de Requerimientos			
Objetivo			
Entender la funcionalidad de los requerimientos para poder proponer una solución.			
Entrada		Resultado	
Lista de Requerimientos [Generada]		Esquema del Sistema [Inicial]	
Criterios de Verificación			
El esquema del sistema que se generó cubre las necesidades esperadas por el cliente.			
Guía			
Actividad 1 Analizar la lista de requerimientos			
Input		Output	
Lista de requerimientos		Boceto del sistema	
Tareas (opcional)	Herramientas (opcional)	Conocimientos y Habilidades	Métricas
1. Definir la navegación del sistema 2. Definir las pantallas del sistema		<ul style="list-style-type: none"> • Análisis • Creatividad 	
Actividad 2 Diseñar Esquema del sistema			
Input		Output	
Boceto del sistema		Esquema del sistema [Inicial]	
Tareas (opcional)	Herramientas (opcional)	Conocimientos y Habilidades	Métricas
1. Cotejar el esquema diseñado con la lista de requerimientos	• Edraw	• Manejo de Herramientas de diseño	

Fig. 2: Práctica de Análisis de Requerimientos.

Una vez entendida la forma en que se manejaría el llenado de las plantillas de prácticas, el grupo de trabajo generó un listado de prácticas identificadas que Tic-Tac-S realizaba de manera implícita durante sus proyectos de software.

Basados en ese listado se procedió a la generación de un diagrama que ilustrara la relación entre las prácticas definidas y las etapas en que se agruparían cada una de ellas para poder controlar de mejor manera el avance del método. Una vez que ya se tenía el diagrama de relación de las prácticas se fueron detallando una por una, en esta fase se encontraron diversos obstáculos tales como la confusión de conceptos para el llenado de la plantilla, específicamente sobre la definición de lo que es una actividad y lo que es una tarea, haciendo este proceso más lento de lo planeado, por lo cual se solicitó una comunicación más constante con el equipo de KUALI-BEH para aclarar estos puntos.

² Software Gurú, <http://www.sg.com.mx>

Aclarados los puntos anteriores, se continuó con el llenado de las plantillas de manera secuencial, sin embargo al continuar con este proceso, el grupo se dio cuenta de que era necesario agregar algunas prácticas que no se habían considerado, esta acción derivó en la modificación del diagrama de prácticas.

Basados en el listado de prácticas y el diagrama del método, se continuó con las revisiones sobre los objetivos, entradas y salidas, actividades y tareas de las prácticas para afinar cada práctica.

El proceso de revisión de las prácticas se daba de manera electrónica, siendo el equipo de KUALI-BEH quien hacía las observaciones pertinentes sobre las cuales trabajaría el grupo con el fin de ganar una nueva versión de cada práctica hasta lograr una versión definitiva.

Es importante mencionar, que las observaciones se centraban en el uso correcto de los conceptos y de la plantilla y no en la forma de trabajo que Tic-Tac-S expresaba.

Al concluir esta fase los integrantes del grupo de desarrollo de software definieron su propio método de trabajo, el cual cumplía con las propiedades de *coherencia*, *consistencia* y *suficiencia* establecidas en KUALI-BEH. Es decir, el método era coherente porque todas sus prácticas contribuían al logro del propósito del método, era consistente porque todas las entradas y salidas que requerían sus prácticas eran proveídas y/o consumidas por alguna otra práctica del método, finalmente era suficiente porque el conjunto de prácticas permitía cubrir totalmente el propósito para el que el método fue creado.

Las prácticas ayudaron a plasmar la forma de trabajo de Tic-Tac-S y en consecuencia permitió a sus integrantes tener una mejor visibilidad de su trabajo y así poder controlarlo. Además, un beneficio extra de definir el método, fue la creación de una lista maestra de productos de trabajo que se generaban a lo largo del proyecto. Esto permitió comenzar con la construcción de plantillas de apoyo específicas para cada producto de trabajo.

3.2. Fase 2

La Fase 2 del caso de estudio se dio del 03 de Octubre del 2013 al 01 de Abril del 2014. En esta fase se establecieron como objetivos los siguientes:

- Utilizar las Prácticas en proyectos de software por los practicantes como su forma de trabajo.
- Adaptar y mejorar las Prácticas en base a la experiencia, el conocimiento e influencia externa recabada por los practicantes.

En esta fase, Tic-Tac-S se enfrentó a un nuevo proyecto de desarrollo en el que se implementó y puso a prueba el método que se definió en la Fase 1. El proyecto elegido fue *Ripos* cuyo cliente fue el administrador de la cafetería del Tecnológico. *Ripos* consistió en la generación de una aplicación web que permitía administrar los ingresos, egresos e inventario de la cafetería. La aplicación resultante estuvo compuesta por ocho módulos que ayudaron a mejorar el control del negocio.

Durante esta fase el grupo comprendió que el método aún estaba inmaduro y era necesario realizar algunos ajustes de la versión inicial de las prácticas. En consecuencia la versión inicial del método sufrió cambios en las siguientes prácticas:

- Se consideró que la práctica *Jerarquización de controles* (Etapa 2), dada su baja complejidad podía ser considerada como una actividad de la práctica *Definición de funcionalidad de módulos*.
- Por otra parte, la práctica *Identificación de fuente de datos* (Etapa 2) podía concatenarse con la práctica *Análisis del entorno del sistema*.

Tic-Tac-S realizó ajustes a las prácticas anteriores utilizando las operaciones propias de KUALI-BEH, en particular se aplicaron las operaciones de *fusión* y *concatenación*. Estas operaciones, aplicadas correctamente, permiten mantener las propiedades originales del método.

Siempre que era necesario ejecutar alguna de estas operaciones se registraban los factores causantes de la adaptación en una plantilla específica para este fin. Es de destacar la flexibilidad de las operaciones que permitieron llevar a cabo la adaptación.

Una vez terminada esta fase, el grupo obtuvo un método mejorado y probado durante el desarrollo de un proyecto de software. Este método pasó a formar parte de la base de conocimiento del grupo.

3.3. Fase 3

La última fase del caso de estudio se llevó a cabo del 02 de Abril al 24 de Octubre del 2014. El objetivo establecido fue:

- Adoptar las Prácticas como una manera rutinaria de trabajo.

En esta fase se inició un nuevo proyecto llamado *Stand*, que consistía en la generación de un sistema modular para el diseño y cotización de stands comerciales mediante herramientas web.

Durante el proyecto *Stand* se retomó el método desarrollado por el grupo Tic-Tac-S y se incorporaron nuevos miembros al equipo, a quienes en un primer momento se les capacitó en los conceptos básicos definidos por KUALI-BEH. Posterior a esta capacitación se comenzó con la ejecución del método y sus prácticas en el proyecto *Stand*, es así como se logró incorporar a los nuevos miembros a la forma de trabajo establecida en Tic-Tac-S.

En esta fase se logró que los nuevos integrantes a la célula comprendieran y adoptaran de manera adecuada el método de trabajo de Tic-Tac-S. Además, para fines del proyecto *Stand* se redujo el tiempo de desarrollo en un 10% al estimado, lo anterior derivó de dos razones principales: una ejecución cuidadosa y del método; y el involucramiento de la empresa *Stand* con el equipo de desarrollo conforme se avanzaba en el proyecto. Cabe resaltar que las métricas recogidas durante este proyecto tuvieron que ver con el tiempo que tomó la ejecución de las prácticas.

Al término de la fase, la célula fue consciente de que el método y los hábitos de trabajo de sus integrantes habían madurado, sin embargo aún era necesario mejorar, situación que se planteó lograr mediante el desarrollo de más proyectos.

4. Resultados

Una vez concluidas las 3 fases del caso de estudio, se pudieron identificar los siguientes beneficios:

- El grupo de trabajo Tic-Tac-S cuenta con un método de desarrollo de software propio, ver Figura 3. Dicho método fue definido acorde a las necesidades y capacidades de los miembros que conforman la célula de desarrollo, permitiendo la consecución exitosa de los proyectos que ha enfrentado.
- Una vez que se identificó la necesidad de adaptar al método definido, a raíz de aplicarlo en el desarrollo de un nuevo proyecto, las operaciones de adaptación permitieron modificar de manera ordenada al método. Además permitió generar evidencia de los cambios realizados así como la justificación de los mismos. Dichas modificaciones dieron como resultado una nueva versión del método, ver Figura 4.

- Actualmente se cuenta con 23 prácticas documentadas para las cuales el número total de horas destinadas a su creación fue de 44, es decir se requirió en promedio de 115 minutos para documentar cada práctica.

Método	
Tic-Tac Tic-Tac KUALI-BEH	
Propósito Desarrollar módulos completos para conformar sistemas de software.	
Entrada Solicitud del Cliente Datos Generales del Cliente Formatos del Equipo	Resultado Manuales de Usuario Manual Técnico Trámites Administrativos Sistema [Liberado] Contrato de Satisfacción [Firmado]
Conjunto de Prácticas	
Etapa 1 1.1 Recopilación de Información 1.2 Lista de Requerimientos 1.3 Análisis de Requerimientos 1.4 Modelo del Sistema 2.1 Definición de funcionalidad de módulos 2.2 Establecer Restricciones de los módulos 2.3 Análisis del Entorno del Sistema 2.4 Plan de Desarrollo Etapa 2 3.1 Diseño de Plantillas 3.2 Generación de Pantallas 3.3 Validación de pantallas 4.1 Jerarquización de Controles 4.2 Identificación de Fuentes de Datos 4.3 Identificación de Código Reusable 4.4 Codificación de Pantallas 4.5 Verificación General de Controles Etapa 3 5.1 Diseño de Pruebas 5.2 Ejecución de Pruebas 5.3 Análisis y Corrección de Defectos 6.1 Definir la Estrategia de Migración 6.2 Producir Estrategia de Migración Etapa 4 7.1 Liberación del Sistema 7.2 Documento de liberación del sistema	

Fig. 3: Método documentado.

- El método ha sido seccionado en 7 etapas, las cuales han permitido un mejor control sobre la ejecución del mismo, además de que esta nueva división hace más claro para el resto del equipo y el cliente qué avance se tiene.
- El método de desarrollo de software definido ha sido utilizado en dos nuevos proyectos con la integración de dos nuevos elementos al grupo, sin que esto haya resultado en perjuicio ni para el equipo ni para el producto desarrollado.
- El tiempo destinado para la capacitación de los nuevos elementos fue de 7 horas.
- La participación de los integrantes del equipo en la definición del método logró elevar el compromiso del grupo en la realización de las actividades.

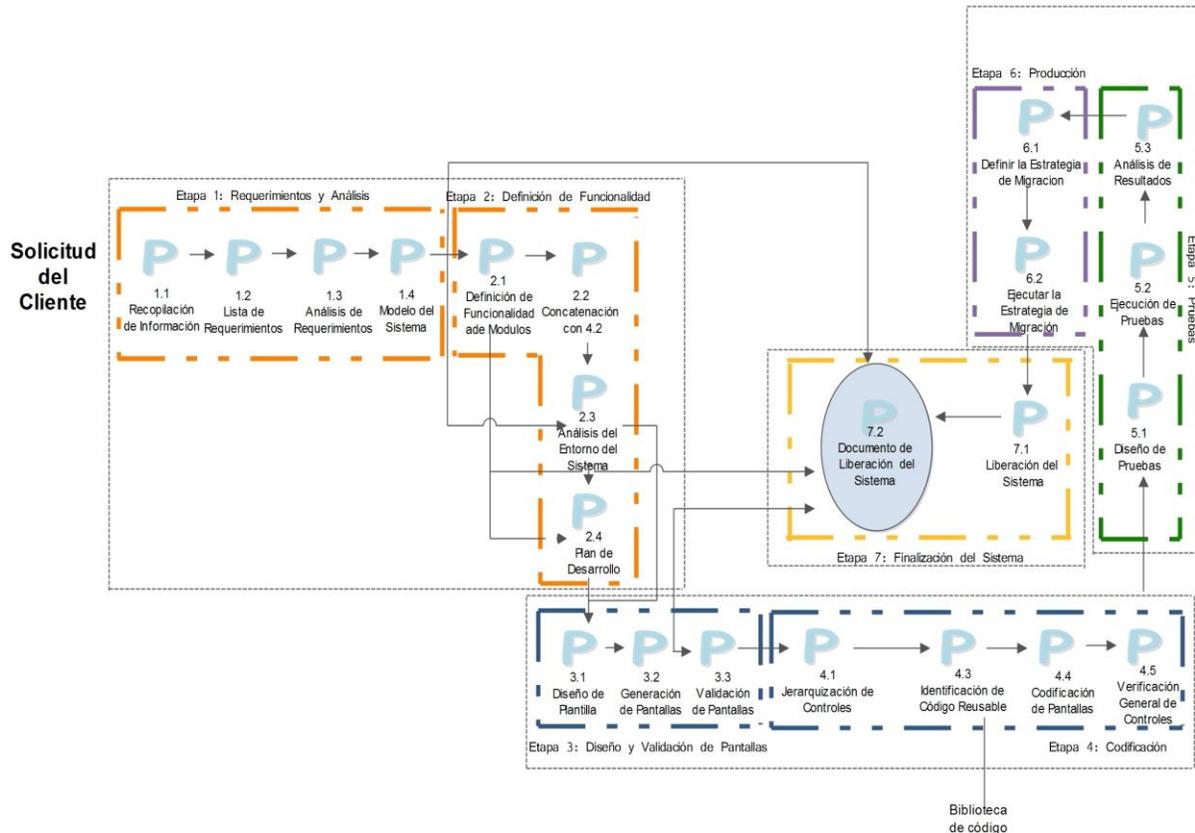


Fig. 4: Segunda versión del método Tic-Tac-S.

Entre las desventajas y oportunidades de mejora se pueden mencionar:

- Para los no iniciados en KUALI-BEH es necesaria una guía que ayude en la delimitación del alcance de la plantilla de prácticas.
- Es necesaria una herramienta electrónica que permita gestionar y compartir la base de conocimiento generada en las organizaciones que implementen KUALI-BEH.
- Sería deseable contar con una herramienta que permitiera llevar el control del método definido, así como apoyar en la adaptación de sus prácticas.

5. Lecciones aprendidas

Las lecciones aprendidas durante esta experiencia fueron las siguientes:

Como Tic-Tac-S, esta experiencia nos permitió darnos cuenta que el proceso utilizado, antes de KUALI-BEH, comenzaba directamente en la línea de programación, únicamente con los datos que proporcionaba el cliente en la primera o segunda

entrevista, esto reflejaba la falta de una forma de trabajo estructurada y dirigida a generar productos de software con calidad, y en consecuencia carecía de un orden bien definido.

Al conocer y comprender KUALI-BEH resultó evidente que para generar una estructura de trabajo óptima es de suma importancia apearse a un orden y a un lenguaje común entre todo el equipo.

KUALI-BEH permitió canalizar el conocimiento de los integrantes en un proyecto. Dentro de las organizaciones debe existir una forma de trabajo significativa que identifique el conocimiento de sus miembros. Lo anterior permitirá definir los métodos adecuados a los diferentes tipos de proyectos que enfrenta la organización y sobre los cuales pueda sustentar la calidad de sus productos de software.

Para el contexto de Tic-Tac-S se observó que un equipo de trabajo debería incluir entre 4 y 6 integrantes para un mejor desempeño del método. Número que a lo largo de los proyectos realizados se ha visto como ideal de acuerdo a las características de los proyectos que atiende el ITSSLP-C. Los integrantes de las células deberán contar con las habilidades y conocimientos necesarios para la aplicación y desarrollo de los productos de software.

Muchas de las ocasiones cuando se sigue una metodología de trabajo que está previamente definida, los equipos de trabajo enfrentan una cuesta arriba al momento de intentar adoptarlo y/o adaptarse. En este caso se logró definir una metodología propia que contempla necesidades particulares para el desarrollo de proyectos y productos de software.

Las prácticas que se diseñaron a través de KUALI-BEH, y que definen la forma de trabajo real de Tic-Tac-S, han permitido comenzar a conformar una base de conocimiento propia que, a la par, está siendo utilizada para los diferentes proyectos de software que enfrenta la célula de desarrollo.

Conforme se fueron completando las prácticas, los participantes pudieron tener una referencia clara del avance de los proyectos. El progreso de las prácticas se fue midiendo de forma cuantitativa y esto permitió ver reflejado el esfuerzo dedicado de cada uno de los participantes en el logro de las metas del proyecto. Las prácticas proveyeron de un panorama confiable a los participantes conforme se avanzaba.

Una limitante de esta experiencia tuvo que ver con la recolección y comparación de métricas. Al ser un proyecto realizado en una entidad de reciente creación y en proceso de maduración, muchos datos históricos no existían lo que imposibilitó realizar comparaciones cuantitativas, por lo que se recurrió a utilizar aspectos cualitativos. Ejemplo de esto, fue el incremento en el compromiso de los alumnos participantes o la facilidad de comprensión del “qué” y el “por qué” tenían que hacer alguna labor.

6. Conclusiones

El grupo de desarrollo pudo experimentar la diferencia de trabajo entre un ambiente académico y uno real. La implementación de KUALI-BEH permitió definir las actividades de los practicantes tomando en cuenta sus habilidades, intereses y metas particulares.

Una vez que se definió el primer método y se aplicó a un segundo proyecto, el esfuerzo necesario para la comprensión de la forma de trabajo por parte de los integrantes se redujo, al retomar lo registrado en la base de conocimiento generada, logrando el objetivo de utilizar un método propio, lo cual resultó ser un aliciente para los integrantes del grupo.

Después de aplicar KUALI-BEH, podemos concluir que provee de flexibilidad al método creado, al permitir incorporar prácticas que no se habían tomado en cuenta, adaptarlas y mejorarlas. La definición de un método propio del grupo fue una labor que requirió del compromiso de todos los integrantes de Tic-Tac-S, profesores y alumnos.

Como trabajo futuro, en el lado académico, profesores de la carrera de ISC tomaron la decisión de incorporar a KUALI-BEH en la asignatura de Ingeniería de Software como un nuevo tópico, dados los resultados hasta ahora obtenidos.

Para la conformación de las siguientes células de desarrollo, el ITSSLP-C estableció KUALI-BEH como herramienta base para la conformación de la manera de trabajo.

En cuanto a Tic-Tac-S, la célula continuará con las mejoras al método definido, ejecutándolo en nuevos proyectos y capacitando a los nuevos integrantes de la célula.

Agradecimientos

Este trabajo fue posible gracias al apoyo del ITSSLP-C y a los departamentos involucrados en la gestación del proyecto del Centro de Desarrollo de Software del Instituto Tecnológico Superior (CEDESITS), así como al Posgrado en Ciencia e Ingeniería de la Computación (PCIC-UNAM) y al Programa de Apoyo a los Estudios de Posgrado (PAEP-UNAM).

Referencias

- [1] KUALI-BEH: Software Project Common Concepts. Normative Annex en ESSENCE – Kernel and Language for Software Engineering Methods. Object Management Group (2014)
- [2] Informe de la 3a Reunión Ordinaria de la H. Junta Directiva. Instituto Tecnológico Superior de San Luis Potosí (2014)
- [3] Software and systems process engineering metamodel (SPEM). Object Management Group (2008)
- [4] Unified modeling language (UML) infrastructure. Object Management (2011)
- [5] Humphrey, W.: Why teams need Operational Processes, Software Engineering Institute, Carnegie Mellon (2010)
- [6] CMMI for development, version 1.3. Software Engineering Institute, Carnegie Mellon University (2010)
- [7] ISO/IEC 24744:2007 Software engineering – metamodel for development methodologies. International Organization for Standardization (2007)
- [8] ISO/IEC 12207:2008 Systems and software engineering - software life cycle processes. International Organization for Standardization (2008)
- [9] ISO/IEC 29110:2011 Software engineering – lifecycle profiles for very small entities (VSEs) – management and engineering guide: Generic profile group: Basic profile. International Organization for Standardization (2011)
- [10] ESSENCE – Kernel and Language for Software Engineering Methods. Object Management Group (2014)