

Soporte al RUP por medio de una plataforma de cómputo basada en la actividad

Pedro C. Santana, Karla Paulina Calderón Vaca

Facultad de Telemática, Universidad de Colima
Colima, México

{psantana, karla_calderon}@ucol.mx

(Paper received on August 10, 2012, accepted on August 24, 2012)

Resumen. Este trabajo presenta el diseño, desarrollo y evaluación de un sistema de gestión de proyectos de software, el cual brinda apoyo al Proceso Unificado de Desarrollo haciendo uso del cómputo basado en actividades. Resultados iniciales de aceptación de uso dan un 90% de aprobación con respecto a la facilidad de uso.

Palabras clave: Gestores de proyectos, Cómputo Basado en Actividades, Proceso Unificado de Desarrollo.

1 Introducción

La ingeniería en software es una disciplina la cual se encarga de llevar a cabo de manera eficiente la producción y mantenimiento de productos de software los cuales podrán ser modificados con el tiempo. El objetivo de la ingeniería en software es evitar caer en el clásico dominio de la programación tradicional y enfocarse más a cuestiones de gestión [1].

El desarrollo del software consta de ciertas etapas las cuales son muy importantes para lograr el objetivo de desarrollar un software de calidad, estas son las siguientes: especificación de requerimientos, análisis, diseño, codificación y pruebas. A estas etapas se les conoce como el proceso de desarrollo de software, el cual consiste en el conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema de software [2]. El Proceso Unificado de Desarrollo (RUP por sus siglas en inglés *Rational Unified Process*), es un proceso de desarrollo de software, el cual en lugar de tener pasos firmemente establecidos, contiene diferentes metodologías las cuales se pueden adaptar en cada etapa según el contexto y las necesidades del proyecto.

Durante estas etapas, existen diversos problemas con los que el equipo de trabajo puede encontrarse relacionados con la organización del proyecto. Debido a esto, ha surgido diversas herramientas que ayudan a los equipos de trabajo a mantener una adecuada organización, a estos se les conoce como gestores de proyectos de software, los cuales ayudan a coordinar el proceso completo.

En la actualidad ha surgido el modelo llamado cómputo basado en actividades (ABC por sus siglas en inglés *Activity-Based Computing*) el cual permite a los usuarios trabajar con aplicaciones específicas que apoyan la manipulación de determinados tipos de información en la realización de tareas específicas [3].

Este trabajo presenta el desarrollo y evaluación de un gestor de proyectos de software de apoyo al proceso RUP utilizando el modelo ABC.

2 Diseño del sistema

La Fig. 1 muestra la arquitectura general para la construcción de este sistema:

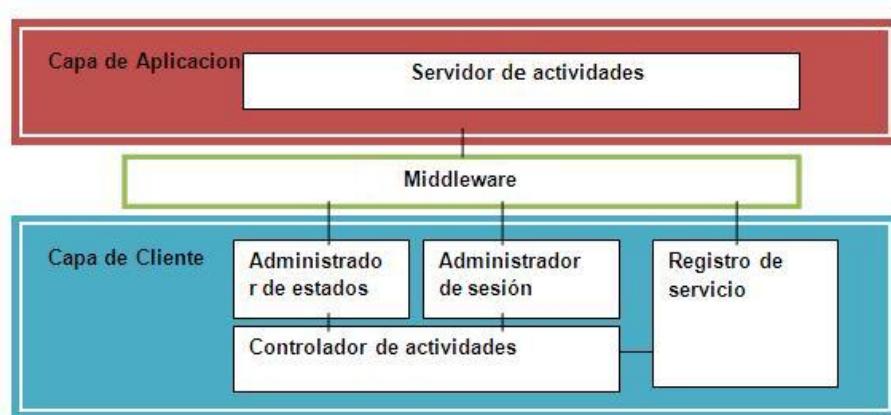


Fig. 1. Arquitectura del sistema.

En la primera parte se muestra la capa de aplicaciones, la cual contiene el “servidor de actividades”, es decir cada una de las actividades que el sistema realiza.

En la segunda parte se muestra la capa de cliente, la cual cuenta con un administrador de estados y de sesión, un controlador de actividades y un registro de servicio. El controlador de Actividades es el vínculo entre el cliente y el servidor.

Un cliente del controlador de actividad registra una o más actividades de administración y mantiene un enlace a la aplicación, la interfaz de usuario a la infraestructura de ABC, que a través del mismo controlador obtiene una lista de actividades para cada usuario.

El Cómputo Basado en Actividades (ABC) consta de los siguientes principios, cada uno aborda distintos problemas.

Centrado en la actividad. Este principio aborda el reto de la aplicación centrada en la informática y apoya las interrupciones en el trabajo al permitir al usuario alternar en las actividades que está involucrado.

Suspensión y reanudación de la actividad: En este principio el usuario puede participar en varias actividades y pueden alternar con otro usuario suspendiendo una actividad y reanudando otra.

Itinerancia de la actividad. La itinerancia de la actividad se rige a través de una serie de eventos de ciclo de vida: registro, acceso, resumen, suspender, salir y sin registro.

Actividades de adaptación. La actividad se adapta a los recursos disponibles.

Actividades compartidas. Se tendrá una lista con todos los usuarios que tendrán acceso a las actividades y podrán reanudar una actividad, es decir dos usuarios podrán trabajar en la misma actividad y también podrán turnarse, cuando un usuario abandone la actividad el otro podrá seguir trabajando en la misma.

Para cubrir estas especificaciones se propuso el uso de una “Barra de actividades” para aplicar el ABC (ver Fig. 2). A continuación se describen las funcionalidades de dicha barra. En el apartado donde dice “Actividades” se listarán todas las actividades que se encuentren registradas en el proyecto. Enseguida se muestran dos actividades, la primera “Análisis” es la que se está realizando en este momento y por lo cual se encuentra habilitada, en cambio la que se llama “ABC” se encuentra deshabilitada debido a que se encuentra en estado de suspensión. Despues se encuentran varios botones los cuales sirven para diversas funciones como: crear, eliminar y guardar actividades. También se encuentra el panel de control para configurar alguna actividad o alguna función de la barra de actividades. Y por último la opción de poder invitar participantes en alguna actividad específica.

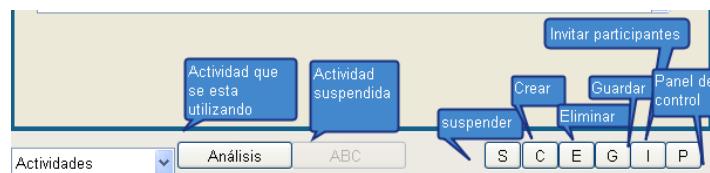


Fig. 2. Barra de actividades.

3 Evaluación

Para medir la aceptación y facilidad de uso de la plataforma, se realizó una evaluación de usabilidad.

Las pruebas de usabilidad se realizaron con un grupo de 30 alumnos de la Facultad de Telemática en la Universidad de Colima. Se aplicaron los instrumentos de evaluación SUS, un cuestionario para recabar los comentarios generales de los usuarios y el Modelo de Aceptación de la Tecnología (TAM).

3.1 SUS

El SUS ha demostrado ser una herramienta de evaluación robusta y fiable [4]. La cual se correlaciona bien con las métricas de usabilidad.

De cada cuestionario SUS se obtuvo una calificación en una escala de 0 (nula usabilidad) a 100 (alta usabilidad). La evaluación por parte de los usuarios dio una calificación de 68.2.

En la Fig. 3 se puede encontrar la distribución de las frecuencias, en las cuales podemos observar que 20 personas dieron una calificación mayor a 60 puntos a la plataforma, lo que nos indica que el sistema cuenta con una usabilidad adecuada, pero que cuenta con áreas de oportunidad. Para esto analizaremos los comentarios generales de los usuarios.

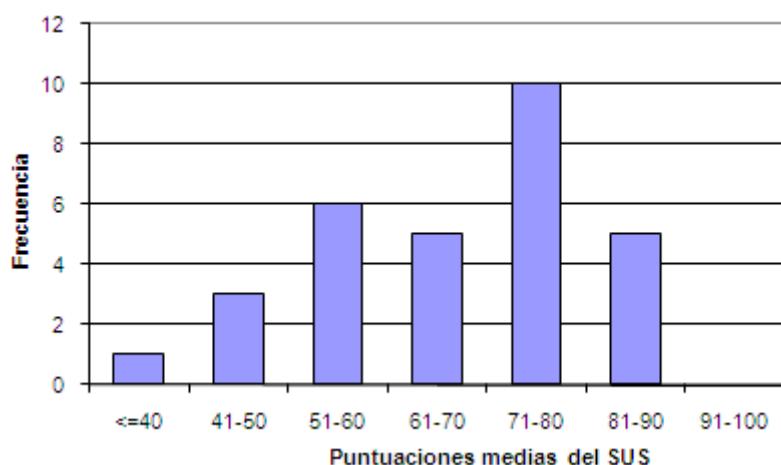


Fig. 3. Distribución de frecuencias de los resultados SUS.

3.2 Evaluación con usuarios

Con los resultados que se obtuvieron de los cuestionarios es posible derivar conclusiones importantes, sobre todo orientadas a fortalezas y debilidades del software para que su operación se adapte más al ambiente del usuario real. De todas las sugerencias sobre las áreas de oportunidad y fortalezas que tiene el sistema, aquellas que contribuirán a lograr los objetivos son las siguientes:

1) Áreas de oportunidad

- Cambiar el tamaño de los botones así como también utilizar imágenes o toda la palabra en lugar de solo utilizar la inicial. El acomodo de los botones puede llegar a causar confusión al usuario.
- Herramientas ilimitadas. Incluir más herramientas para la administración de un proyecto, por ejemplo herramientas para la creación de más diagramas del Lenguaje Unificado de Modelado (UML).

- Compatibilidad. Hacer que el software sea compatible con otros, por ejemplo: MS Word y MS Visio.

2) Fortalezas

- Fácil de utilizar, permite realizar el trabajo de una manera sencilla y eficaz.
- Buena apariencia, sencilla pero entendible y funcional.
- Seguro, debido a que se necesita de un usuario para entrar al sistema.
- Suspensión de actividades. Con esta herramienta puede suspender una actividad y reanudarla después sin perder la información.
- Barra de actividades. Esta barra incluye varias herramientas para realizar proyectos.

En cuanto a las áreas de oportunidad de mejora, uno muy interesante es que los botones causan confusión a los usuarios y proponen que se utilicen imágenes o utilizar toda la palabra en lugar de solo utilizar la primera letra de la palabra ya que así será más entendible. Otras sugerencias importantes son que se deberían de incluir más herramientas UML y que la aplicación sea compatible con programas externos como MS Word y MS Visio, esta sin duda valdría la pena considerarla como una mejora a futuro.

3.3 TAM

Desarrollado por Davis en 1989. Es un modelo muy eficaz probado en la predicción del uso de la tecnología [5].

Tomando en cuenta las respuestas “De acuerdo”, “Muy de acuerdo” y “Totalmente de acuerdo”, la herramienta TAM generó los siguientes resultados.

En la pregunta de la percepción de la facilidad el 90% cree que el sistema es fácil de utilizar, el 83% dijo que es fácil de aprender a usar el software, el 80% dijo que es claro y entendible y finalmente el 87% dijo que es fácil de encontrar información en el software.

En el área de la percepción de la utilidad, los resultados obtenidos fueron, el 77% cree que es muy eficiente, el 80% dijo que mejorará su desempeño, el 87% dijo que mejorará su productividad y finalmente el 83% creen que les sería útil este software.

Con respecto a la actitud hacia el uso del software, se obtuvo una respuesta positiva, el 90% creen que es una buena idea el uso de este software.

Por último, la pregunta de “la intención de uso”, el 73% tienen la intención de usarlo, el 87% de los usuarios dijeron que ellos lo volverían a utilizar, y el 80% tienen la intención de utilizarlo (ver Fig. 4).

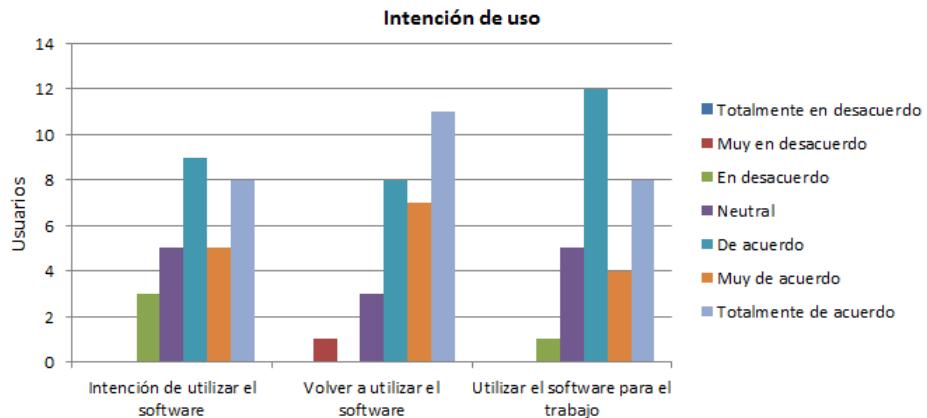


Fig. 4. Intención del uso de la plataforma.

4 Conclusiones

Este trabajo presenta un sistema de gestión de proyectos RUP basado en ABC, la evaluación de este sistema dio como resultado que es útil y permite un mejor desempeño y una mayor eficiencia, además, es considerado como una buena idea. Esto es, ha creado buenas intenciones en los usuarios para seguir usándolo.

Referencias

1. Montesa, J. O. (s.f.). El proceso de Desarrollo de Software. Universidad Politécnica de Valencia, Escuela de Informática 2000.
2. Jacobson Ivar, B. G. El proceso unificado de desarrollo de software. Pearson Addison Wesley.
3. Bardram, J. E. From Desktop Task Management to Ubiquitous Activity-Based Computing. Aabogade 34.
4. Brooke, J. (1996). “SUS: a “quick and dirty” usability scale”, en P. W. Jordan, B. Thomas, B. A. Weerdmeester y A. L. McClelland (Eds.) Usability Evaluation in Industry. London: Taylor and Francis.
5. Davis, F. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. MISQ, 13(3), 319–340.