



Fuente: pixabay.

Herramientas para la gestión de configuraciones en el desarrollo de software

Geovanny Albeiro Hernández Pantoja

Docente de Ingeniería de Sistemas
Universidad Mariana

Danny Roberto Chacón Bonilla

Estudiante de Ingeniería de Sistemas
Universidad Mariana

Según Sommerville (2005) las herramientas de gestión de la configuración son usadas para rastrear las propuestas de cambio, almacenar versiones de componentes del sistema, construir sistemas a partir de dichos componentes, y rastrear liberaciones de las versiones del sistema para los clientes. Sin embargo, los desarrolladores no cuentan con una guía específica, o una forma de tomar una decisión y escoger una herramienta que brinde las ventajas requeridas.

Durante el proceso de desarrollar software hay contratiempos, solicitudes por parte del cliente, cambio de requerimientos que pueden implicar desfases en costos y tiempo, y que además aumentan el riesgo de cometer errores al momento de alterar archivos o códigos que ya habían sido creados. Los cambios que surgen son inevitables, y éstos generalmente dan origen a nuevas versiones, a modificaciones en los archivos por parte de los desarrolladores, que no siempre van a ser informados a todo el equipo, por lo tanto es esencial hacer uso de unas herramientas que faciliten las tareas de este tipo, para que la versión final del software sea de calidad.

En consecuencia, el interés por realizar esta investigación es rescatar las características de las herramientas para la gestión de configuraciones existentes en la actualidad, que son un factor importante en el momento de elegir una de ellas, o cuando se trata de desarrollar un proyecto según el contexto en el que se encuentren. De igual manera, obtener ventajas y conocer las desventajas que presenta cada herramienta para una posterior comparación, determinando cuál ofrece mayor documentación, facilidad en la curva de aprendizaje, transparencia en la implementación, rendimiento, entre otros. Finalmente, la extracción de ventajas y desventajas sobre cada herramienta permite identificar diferencias entre ellas, facilitando la selección de una alternativa.

El objetivo general de esta investigación es evaluar el uso de las herramientas para la gestión de configuraciones, para lo cual se desarrollará los siguientes objetivos específicos: - Caracterizar las herramientas para la gestión de configuraciones, - Desarrollar pruebas con las herramientas para la gestión de configuraciones a través de un experimento, y por último, - Describir de manera comparativa las ventajas y desventajas que existen entre las herramientas para la gestión de configuraciones.

Las herramientas de software libre a evaluar para realizar las pruebas de gestión de configuraciones son:

CVS: es una forma de trabajo relacionada con el desarrollo de software, que guarda el código fuente, almacena las versiones de todos los archivos, y lleva un registro del acceso de los participantes; además permite combinar el código de dos o más programadores que trabajen en un mismo archivo.

Subversión: es una herramienta de control de versiones, que utiliza un árbol de archivos en un repositorio que guarda todos los cambios que se realice a los archivos y a los directorios, y además facilita realizar revisiones de las modificaciones de los componentes del sistema.

Git: su objetivo es mantener gran cantidad de archivos para un extenso número de programadores. Lo que hace diferente a esta herramienta de las otras, es que al realizar la gestión de configuraciones toma una “foto” del estado de cada archivo; si posee modificaciones y si el archivo no ha sufrido cambios, no realiza una copia del mismo.

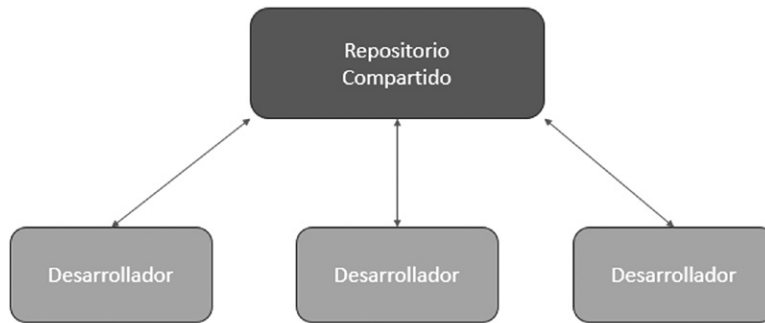


Figura 1. Ejemplo Repositorio Compartido Centralizado
Fuente: <https://git-scm.com>

Mercurial: es un sistema multiplataforma de control versiones, desarrollado en su mayoría en lenguaje Python; facilita la gestión de una gran cantidad de archivos usando un repositorio distribuido, a diferencia de los sistemas de control de versiones tradicionales.

Bazaar: esta herramienta fue diseñada principalmente para facilitar la contribución en grandes proyectos de software libre; soporta flujos de trabajos distribuidos o centralizados, y puede ser usado por un único programador o por varios a través de la red.

Para evaluar las anteriores herramientas se empleará el “Modelo de evaluación abierto para software libre y de código abierto”, compuesto por cuatro fases, que a su vez están conformadas por actividades. Al ser un modelo abierto, las características de las herramientas a evaluar quedan a criterio del evaluador; de la misma manera, si se requiere evaluar características diferentes, puede hacerse adiciones sin inconvenientes.

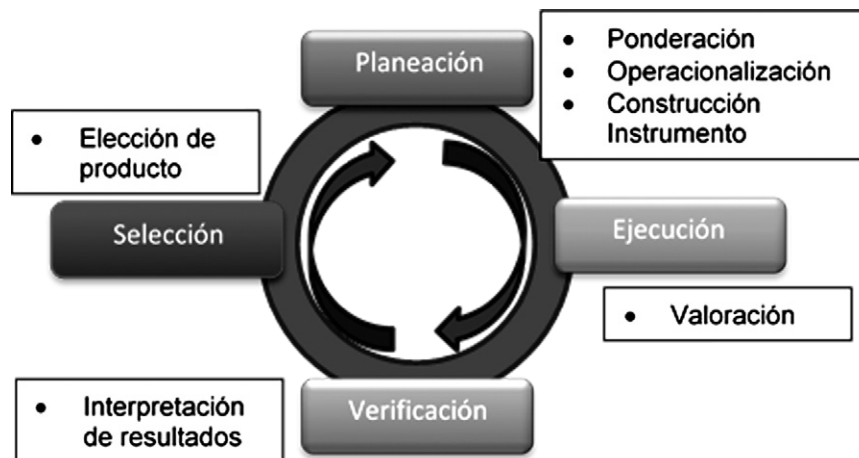


Figura 2. Fases del modelo de evaluación
Fuente: OAM-F/OSS Open Appraisal Model for Free and Open Source Software-2016

Para desarrollar la evaluación de las herramientas, se determinó tomar como población objeto de estudio a los estudiantes de séptimo a décimo semestre del programa de Ingeniería en Sistemas de la Universidad Mariana, para lo cual se realizará una convocatoria con participación voluntaria, en un curso que aborde la temática del desarrollo dirigido por modelos. Esta decisión busca que los datos que puedan aportar los informantes en relación con el uso de las herramientas computacionales, sean obtenidos a partir de una convicción propia de ampliar el conocimiento sobre el funcionamiento de estas tecnologías y sin ningún tipo de restricciones.

En conclusión, al terminar esta investigación se habrá construido una guía con las ventajas y desventajas de las herramientas evaluadas, que servirá de apoyo para la comunidad desarrolladora de software. Ésta será construida con base en los resultados del experimento, organizados de forma comparativa entre todas las herramientas.

Bibliografía

Sommerville, I. (2005). *Ingeniería del Software*. Madrid, España: Pearson.