

MODIFICACIÓN DEL ANTEPROYECTO

Como se indica en el anteproyecto original, titulado “*GCAD: Plug-in de Eclipse para la Gestión de Conocimiento en las fases de Análisis y Diseño de desarrollo distribuido*”, el objetivo principal del PFC consistía en el diseño y construcción de un plug-in de Eclipse que permita dar soporte a la gestión de decisiones en desarrollo distribuido de software.

Sin embargo, se ha planteado un nuevo objetivo que se desea alcanzar con el desarrollo del PFC, siendo necesario modificar el tipo de herramienta, pasando a ser una aplicación Java con arquitectura distribuida cliente/servidor, en lugar de un plug-in para la plataforma Eclipse.

El nuevo objetivo que se desea alcanzar es el desarrollo de un **sistema distribuido con arquitectura cliente/servidor que permita la gestión de conocimiento en el desarrollo distribuido de software**. De este modo, todo el control y conocimiento del sistema pasan a estar en el lado del servidor, mientras que los clientes realizan peticiones a dicho servidor y representan la información de manera visual e intuitiva para los usuarios del sistema.

El motivo fundamental para introducir este nuevo objetivo es mejorar la comunicación y sincronización entre los equipos de desarrollo distribuidos que utilicen este sistema, ya que al estar centralizado el control en el servidor, la información que cada cliente está visualizando en ese momento puede sincronizarse y actualizarse. Por ejemplo, si un usuario está visualizando un grafo de decisiones y otro usuario añade nuevo conocimiento que afecta a ese grafo, el primer usuario será notificado en ese instante y su cliente actualizará el grafo con la nueva información.

El cambio de herramienta a desarrollar, pasando de desarrollar un plug-in a una aplicación Java distribuida, se debe a que existen problemas de funcionamiento del plug-in de Eclipse con RMI (Remote Method Invocation, tecnología utilizada para la comunicación entre el cliente y el servidor). Para comprobarlo, se realizó una prueba de concepto (PoC) con una pequeña aplicación Java actuando como servidor, donde el plug-in tenía que registrarse (mediante RMI) para poder recibir las actualizaciones del servidor ante cambios producidos por otros clientes (plug-ins) y poder así actualizar su interfaz gráfica. Al realizar esta prueba, se detectó que las peticiones de actualización que enviaba el servidor al plug-in, fallaban, porque el plug-in no podía registrarse correctamente en el servidor. Esto es debido a que un plug-in utiliza y carga de manera diferente los recursos que necesita para su ejecución, presentando problemas reconocidos en el uso de comunicaciones y objetos remotos.

Además del motivo anterior, que es el motivo principal que fomenta esta modificación, con el desarrollo de una aplicación Java, no se necesita instalar ni utilizar ningún IDE específico (con el plug-in, era requisito obligatorio utilizarlo bajo el IDE Eclipse) y se presenta una mayor flexibilidad en lo que a la parte de desarrollo de interfaz gráfica se refiere, pues el plug-in estaba limitado a los elementos que provee (vistas, editores, menús, etc).

En resumen, se solicita una modificación con respecto al anteproyecto original en el tipo de herramienta a desarrollar, manteniendo igual los fundamentos teóricos y el campo de

aplicación.

Por tanto, con esta modificación, el nuevo título del anteproyecto sería **GCAD: Sistema Distribuido para la Gestión de Conocimiento en fases de Análisis y Diseño de desarrollo distribuido**, manteniendo los objetivos del anteproyecto original y teniendo en cuenta que será una herramienta cliente/servidor con comunicación remota a través de RMI, como objetivo nuevo añadido.

Para terminar, a los medios software utilizados, habrá que añadir **RMI - Remote Method Invocation** para la comunicación entre el cliente y el servidor, y **Java Swing**, para el desarrollo de la interfaz gráfica de la aplicación.

En Ciudad Real, a 13 de abril de 2011.