

# **SET (*Software Engineering Tutor*). Una herramienta para la construcción guiada de modelos de dominio**

Arturo Cepeda Pérez, Sergio Bravo Martín, Francisco José García Peñalvo

Universidad de Salamanca, Facultad de Ciencias,  
Plaza de los Caídos s/n, 37008 Salamanca (España)  
[acepeda@usal.es](mailto:acepeda@usal.es), [ser@usal.es](mailto:ser@usal.es), [fgarcia@usal.es](mailto:fgarcia@usal.es)

**Abstract.** This paper presents a CASE tool named *Software Engineering Tutor* (SET), aimed at improving the student and software engineering formation in order to guide the domain and use case models construction. Besides, this tool offers a repository of case studies, trying to make an effort to share experiences around the user community.

**Keywords.** CASE tool, domain model, use case, software engineering formation, educational technologies

## **1. Introducción**

La Ingeniería del Software es una disciplina de las ciencias de la computación que ofrece métodos o técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo. Cada vez resulta más frecuente la consideración de esta disciplina como una nueva área de la Ingeniería hasta el punto de que el Ingeniero del Software se ha hecho imprescindible en el mundo profesional y con todo el derecho.

Son muchas las herramientas CASE<sup>1</sup> que prestan su apoyo al Ingeniero del Software y que contribuyen en gran medida a un aumento de la productividad en el desarrollo de software reduciendo el coste en términos de tiempo y dinero. Estas herramientas se aplican en todos los aspectos del ciclo de vida de desarrollo del software como son la planificación, análisis y diseño del proyecto, documentación (textual y gráfica), cálculo de costes, generación automática de código e incluso detección de errores entre otras.

Este artículo presenta una herramienta CASE más, denominada *SET*, *Software Engineering Tutor*, pero especialmente novedosa en el sentido de que centra su atención en el aspecto formativo y de autoayuda para el Ingeniero del Software, tal y como se puede deducir ligeramente de su nombre. La herramienta ha sido

---

<sup>1</sup> Del inglés *Computer Aided Software Engineering* (Ingeniería del Software Asistida por Ordenador).

desarrollada en el marco de un proyecto fin de carrera de la *Ingeniería Técnica en Informática de Sistemas* en la Universidad de Salamanca. De igual forma ha sido concebida como una herramienta de apoyo a la docencia en general, y en particular dirigida a los talleres de prácticas de la asignatura de tercer curso *Ingeniería del Software*.

## 2. Descripción funcional

Las características funcionales de la herramienta SET se centran fundamentalmente en los conceptos siguientes:

- Un asistente de modelado como mecanismo de apoyo a la construcción de modelos del dominio.
- Un repositorio centralizado con casos de estudio de ejemplo.
- Compatibilidad con otras herramientas CASE.
- Interfaz de usuario basada en diferentes vistas del modelo en construcción.

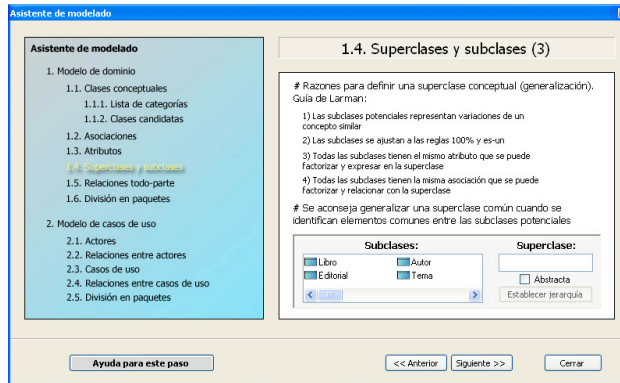
### El asistente de modelado

La idea original de la herramienta busca como objetivo principal iniciar al alumno (y profesional del software) en la fase del análisis orientado a objetos por medio de la construcción guiada de Modelos del Dominio (o modelos conceptuales) y Modelos de Casos de Uso. Ambos modelos se construyen gráficamente por medio de diagramas UML. Sin embargo, su verdadera aportación dentro del mundo de las herramientas CASE reside en su orientación al adiestramiento e instrucción en la construcción de dichos modelos mediante un asistente de modelado integrado. Es precisamente este asistente inteligente el que la convierte en una herramienta única.

El asistente de modelado consiste en un cuadro de diálogo que permite navegar libremente a través de los diferentes pasos que guían el proceso de construcción de los modelos. La mayoría de las estrategias utilizadas para la construcción del asistente, como por ejemplo, la identificación de clases conceptuales basada en listas de categorías, están extraídas de la bibliografía [1][2].

Para cada uno de los pasos se muestra de forma somera en qué consiste, así como los controles necesarios para que el usuario pueda llevar a cabo su cometido; además, existe la opción de ampliar la información relativa al paso en curso, con instrucciones detalladas (y con ejemplos) sobre cómo completarlo.

Fig. 1. Asistente de modelado con la lista completa de pasos a la izquierda y resto de controles.



El asistente está totalmente asociado con el entorno principal, de forma que ambos elementos van a trabajar sobre los mismos datos. Así, los efectos que produzcan las acciones realizadas en el asistente serán visibles de forma instantánea en el entorno.

## El repositorio de plantillas

Toda herramienta que preste un cierto carácter docente precisa de una base de conocimiento útil y de apoyo al aprendizaje. En nuestro caso, *Software Engineering Tutor* dispone de casos de estudio bien conocidos que sirven de ejemplo para el alumno. De acuerdo con este planteamiento, se utiliza el concepto de plantilla como un caso de estudio predefinido (o modelo de dominio estándar), que dota a la herramienta con la facilidad de construir nuevos modelos a partir de otros ya existentes.

Con la finalidad de generalizar el uso de la herramienta, y por supuesto, el de las plantillas, se ha creado un espacio virtual en Internet, en la línea de un **repositorio centralizado**, desde el que se ofrece un conjunto de plantillas estándar con soluciones a casos de estudio bien conocidos. Dichas plantillas se pueden consultar y descargar directamente del repositorio sin necesidad de tener que salir del marco de la aplicación. Además, el repositorio está preparado para que sólo aquellos usuarios que estén registrados puedan interactuar con él.

## Compatibilidad con otras herramientas CASE

La herramienta dispone de formatos propios para mejorar el rendimiento y eficiencia en el trabajo con los documentos. Se diferencian dos tipos:

- Archivos de proyecto (con extensión *.spr*).
- Archivos de plantilla (con extensión *.stp*).

Por otro lado, tanto las **plantillas predefinidas** como los modelos construidos con la herramienta pueden ser exportados de acuerdo a la especificación XMI (*XML*

#### 4 Arturo Cepeda Pérez, Sergio Bravo Martín, Francisco José García Peñalvo

*Metadata Interchange*) para el intercambio de diagramas [3]. De esta forma, la herramienta no se convierte en una aplicación cerrada, sino que se complementa con el resto de herramientas CASE que prestan su apoyo al resto de fases en el ciclo de vida del software.

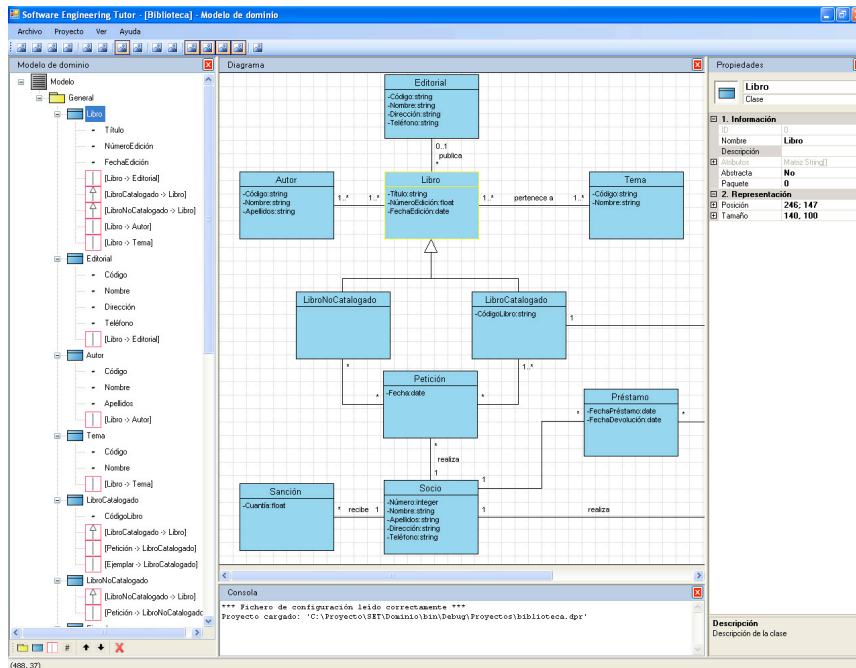
Cabe destacar que, aunque la herramienta ofrece al usuario todo lo necesario para la construcción de los diagramas asociados a los modelos de dominio y casos de uso, la finalidad principal de la aplicación no es la de diagramar, sino tutorizar y asistir en la elaboración de dichos modelos.

#### **La interfaz de usuario**

El marco de la aplicación está dividido en cuatro vistas que ofrecen perspectivas diferentes del modelo en construcción:

- **Vista del Modelo:** Se muestran todos los elementos del modelo en el que se está trabajando en forma de árbol, de manera que están perfectamente clasificados y jerarquizados. Asociada a esta vista hay una barra de herramientas en la cual el usuario tiene disponibles algunas opciones de gestión del modelado, como la creación y eliminación de elementos [4].
- **Vista del Diagrama:** Contiene la superficie de dibujo en la que se muestra una representación del modelo en uso en forma de diagrama. Algunos parámetros pueden ser ajustados directamente mediante la vista de diagrama, como la posición de los elementos [5].
- **Ventana de Propiedades:** En esta vista aparecerán en todo momento las propiedades del elemento seleccionado, ya sean modificables o no. Se listará la totalidad de las propiedades junto a sus respectivos valores actuales en una tabla con dos columnas. Todos los parámetros cuyos valores sean susceptibles de ser modificados podrán ser ajustados manualmente en la vista de propiedades. La selección de un elemento puede realizarse tanto en la vista de modelo como en la vista de diagrama.
- **Consola de operaciones:** Consiste en una consola de texto de sólo lectura en la cual se va dejando constancia de cada acción relevante relacionada con el proyecto en uso.

Fig. 2. Marco principal de SET donde podemos diferenciar la vista del modelo a la izquierda, la vista del diagrama en el centro, la ventana de propiedades a la derecha y la consola de operaciones abajo.



### 3. Descripción técnica

En lo referente a la arquitectura del software, la herramienta se basa en un entorno distribuido en el que se diferencian dos módulos autónomos pero con capacidad para colaborar conjuntamente:

- De un lado el **módulo principal** o núcleo de la aplicación, con una instalación íntegramente en cliente a modo de aplicación local.
- Por otro lado el **repositorio** de plantillas, desplegado en un servidor remoto, totalmente desagregado del módulo principal y accesible a través de una interfaz externa implementada por medio de servicios web.

En términos de implementación, destacar que se han separado las características procedentes de objetos del dominio y lógica del negocio de las estrictamente referidas al aspecto gráfico y representación de los diagramas.

*Software Engineering Tutor* ha sido desarrollado haciendo uso de la *Plataforma .NET de Microsoft (ver. 2.0)*, siendo el lenguaje de programación escogido C# y Visual Studio 2005 el entorno de desarrollo integrado. Algunas de las razones que han motivado la utilización de esta tecnología son:

- Agilidad y rapidez en el desarrollo de prototipos.

## 6 Arturo Cepeda Pérez, Sergio Bravo Martín, Francisco José García Peñalvo

- Potencia para la construcción de interfaces gráficas así como disponibilidad de una completa biblioteca de dibujo, GDI+, utilizada para el sistema de diagramación.
- Integración entre aplicaciones de Windows Forms y Web Services de ASP.NET.
- Gran potencia y versatilidad en la fase de pruebas (especialmente unitarias).

## 4. Conclusiones

*Software Engineering Tutor* pretende ser el germen de un nuevo tipo de herramientas CASE destinadas a la formación de futuros Ingenieros del Software. El claro enfoque docente está marcado por un completo y contrastado asistente que guía la construcción de modelos de dominio y casos de uso.

Una de las ventajas en la incorporación de la herramienta a los talleres de prácticas de la asignatura *Ingeniería del Software* será la unificación del proceso de documentación gracias a la generación automática de informes.

Por otro lado, la iniciativa del repositorio de casos de estudio, amplía no sólo las posibilidades de uso de la herramienta sino también, poder compartir otros casos de estudios construidos por otros miembros de la comunidad de usuarios y profesionales del sector.

También cabe destacar la posibilidad de trabajar con la aplicación cliente en modo *offline* (gracias a su naturaleza distribuida).

Por último, la compatibilidad con los estándares UML y XMI, permite aprovechar los modelos que han sido desarrollados utilizando SET en otras aplicaciones.

## 5. Referencias

1. Larman, C. “*Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process*”. 2nd Ed. Prentice Hall, 2002
2. Shlaer, S., Mellor S. J. “*Object-Oriented Analysis: Modeling the World in Data*”. Yourdon Press, 1988
3. OMG. “*MOF 2.0/XMI Mapping, v2.1.1*”. Object Management Group Inc. Document formal/2007-12-01 [en línea]. Disponible en: <http://www.omg.org/docs/formal/07-12-01.pdf> [Última vez visitado, sep-2008]
4. OMG. “*Unified Modeling Language: Superstructure. Version 2.0*”. Object Management Group Inc. Document formal/05-07-04 [en línea]. <http://www.omg.org/cgi-bin/doc?formal/05-07-04> [Última vez visitado, sep-2008]
5. Rumbaugh, J., Jacobson, I., Booch, G. “*The Unified Modeling Language. Reference Manual*”. 2nd Ed. Addison-Wesley, 2005