

Generación de pruebas del sistema en el desarrollo del proyecto ADAGIO mediante la aplicación de NDT

S. Moreno-Leonardo, J.G. Enríquez, L. Morales y F.J. Domínguez-Mayo

Universidad de Sevilla, Departamento de Lenguajes y Sistemas Informáticos
Escuela Técnica Superior de Ingeniería Informática. Avda. Reina Mercedes s/n. 41012 Sevilla
{sara.moreno,jose.gonzalez,leticia.morales}@iwt2.org, fjdominguez@us.es

Resumen. La ingeniería guiada por modelos (MDE) se ha utilizado en los últimos años para promover mejores resultados en el desarrollo de aplicaciones web, en el campo que se ha denominado ingeniería web guiada por modelos (MDWE). Una de las ventajas de aplicar MDWE es que ofrece una solución para reducir el coste de las pruebas sin afectar a su ejecución ni la calidad de las mismas. *Navigational Development Techniques* (NDT), es una metodología que proporciona soporte para todas las fases del ciclo de vida del desarrollo de un proyecto de software, proponiendo transformaciones automáticas entre dichas fases, sin embargo, en este trabajo, aunque se describe brevemente cómo se ha hecho uso de NDT para la definición de las fases de requisitos y análisis, se hace hincapié en el uso de la metodología para la definición de la fase de pruebas de un proyecto real denominado ADAGIO. La aplicación de esta metodología proporciona un mayor índice de cobertura de pruebas del sistema y, consecuentemente, un incremento en la calidad del producto.

Palabras clave: Metodología, Ingeniería web guiada por modelos, NDT, Testing

1 Problema a resolver y quién lo resuelve

En la confluencia de los ámbitos de consultoría y de innovación, existe una línea de negocio orientada a *Business Intelligence*, la cual engloba el desarrollo de los servicios de oficinas técnicas para la explotación de la información, benchmarking de herramientas, consultoría tecnológica especializada, y diseño de soluciones para la cartera de clientes.

No obstante, hasta el momento no se ha culminado la extensión de esta línea de negocio hacia otras áreas ya consolidadas hoy en día en el mercado empresarial como pueden ser: *Big Data*, *Open Data*, *Linked Data*, geolocalización o máquinas de aprendizaje (*Machine Learning* o *ML*) entre otras. Esto puede ser debido a que los datos de interés para los diversos ámbitos de la investigación, en general, se encuentran localizados en sistemas distribuidos y heterogéneos, que, a su vez, implementan estructuras y semánticas de datos heterogéneas.

Ante esta problemática, nace el proyecto ADAGIO, cuyo objetivo principal es ofrecer a los investigadores la posibilidad de integrar y consolidar fuentes de datos heterogéneas, mostrando una visión unificada de ellas, permitiendo simplificar la tarea de búsqueda en diferentes repositorios, así como la relación entre las fuentes encontradas. Teniendo en cuenta el objetivo del proyecto, las pruebas para garantizar

que todo el proceso de integración funciona correctamente, y consecuentemente, la calidad del producto, cobran una importancia vital.

ADAGIO, es un proyecto financiado por el Centro para el Desarrollo Tecnológico Industrial (CDTI), entidad pública del Ministerio de Economía, Industria y Competitividad de España. Esta iniciativa está lanzada por el consorcio del grupo de investigación de Ingeniería Web y Testing Temprano (IWT2) de la Universidad de Sevilla y la empresa Servinform S.A.

2 Tecnologías usadas

El modelo funcional de ADAGIO propone 4 módulos principales: (i) recuperación periódica de datos, (ii) proceso de reconciliación de entidades, (iii) administración y gestión de las fuentes de datos, y (iv) explotación de información a través de una aplicación web.

Este artículo se centra en el módulo iv, siendo el objetivo principal de este la creación de consultas por parte de los usuarios del sistema, en un lenguaje lo más natural y de alto nivel posible, aplicando técnicas supervisadas de ML, con el fin de extraer conocimiento subyacente en las fuentes de datos previamente procesadas en los otros módulos.

La definición de este producto software ha sido llevada a cabo siguiendo las recomendaciones de la metodología NDT [1]. NDT, es una propuesta englobada dentro del paradigma de la ingeniería web guiada por modelos, aunque, actualmente, puede ser aplicada para la especificación de cualquier tipo de sistema software. Actualmente, NDT es un marco de trabajo completo, llamado NDTQ-Framework, que brinda soporte para todas las fases del ciclo de vida del desarrollo de un proyecto software, proponiendo transformaciones automáticas entre las diferentes fases del ciclo de vida, que ayuda al ingeniero a seguir un proceso sistemático y automatizado y a reducir considerablemente el coste del desarrollo del mismo.

Para aplicar esta metodología, se ha utilizado como herramienta de soporte NDT-Suite [2] que trabaja bajo el paraguas de la herramienta Enterprise Architect (EA) [3]. Las herramientas más importantes de NDT-Suite son: (i) NDT-Profile, que implementa un perfil definido sobre la herramienta EA, donde se definen todos los modelos y artefactos asociados para trabajar con NDT, (ii) NDT-Quality, que toma como entrada un proyecto realizado en base a NDT-Profile y comprueba que la trazabilidad y las normas de NDT se cumplen en el proyecto indicado, (iii), NDT-Driver, que implementa un conjunto de procedimientos automáticos que permite llevar a cabo transformaciones entre los distintos modelos NDT definidos en un proyecto desarrollado con NDT-Profile, y (iv) NDT-Report, con la que seremos capaces de generar toda la documentación definida en el proyecto de forma automática.

3 Resultados obtenidos

Como se mencionó anteriormente, el conjunto de herramientas que proporciona NDT-Driver se ha utilizado para: (i) definir tanto la fase de requisitos como la fase de

análisis del proyecto ADAGIO, y (ii) para la generación de pruebas tempranas del sistema definido. Cabe destacar, que las transformaciones automáticas que proporcionan estas herramientas fueron aplicadas, por una parte, entre las fases de requisitos y análisis y, por otra, entre las fases de requisitos y pruebas.

Durante la definición de la fase de requisitos y análisis del proyecto ADAGIO, se definieron un total de 25 requisitos funcionales (RF). Los requisitos funcionales se basan en diagramas de actividad, por lo que NDT-Driver genera para el plan de pruebas del sistema una prueba para cada uno de los caminos posibles que existan en el diagrama de actividad que define cada RF.

En la parte izquierda de la Fig. 1, se muestra cómo ha sido definido el requisito funcional "RF-01. Parametrización de la búsqueda" de la plataforma ADAGIO, siendo posible apreciar que lo señalado con una línea de mayor grosor es una de las posibles rutas del diagrama de actividad, considerándolo como una de las pruebas generadas para la plataforma ADAGIO. Así pues, en la parte derecha de la misma figura, se observa una de las pruebas específicas generadas para este requisito funcional.

Una vez obtenidas todas las pruebas del sistema cubriendo cada una de las rutas posibles, se puede generar un plan de prueba mediante el uso de la herramienta NDT-Report. Esta herramienta define una plantilla para la descripción de cada una de las pruebas generadas automáticamente. Esta plantilla incluye la definición de las pruebas del sistema de dos maneras diferentes: una tabla en la que se encuentran descritos: el nombre, la versión, la fecha de creación, los autores, las relaciones, la descripción y las condiciones previas y posteriores de la prueba; y el diagrama de actividad de la prueba, que muestra todas las actividades que el actor asociado a dicha prueba debe llevar a cabo para realizar la prueba satisfactoriamente.

Para asegurar la calidad del producto software definido, la definición del plan de pruebas de la plataforma web se realizó por parte de la empresa (S) y, por parte del grupo IWT2 usando NDT.

Por parte de IWT2, se hizo uso de la herramienta NDT-Driver, tomando los diagramas de actividad definidos para cada RF y, de forma automática, se transformaron para generar las diferentes pruebas del sistema. En este sentido, el nivel de cobertura del plan de prueba será del 100%.

La Tabla 1 muestra que la empresa definió un total de 76 pruebas de sistema e IWT2 un total de 136, aumentando en un 44% las generadas por la empresa. Por lo tanto, el uso de NDT hace palpable que la calidad del producto software que se espera obtener se aumenta considerablemente.

Finalmente, y con el objetivo de elevar el nivel de automatización de la ejecución del plan de pruebas, se utilizó la herramienta Selenium [4]. Gracias al uso de esta herramienta, los autores lograron reducir notablemente el costo de ejecutar el plan de prueba para cada iteración del proyecto.

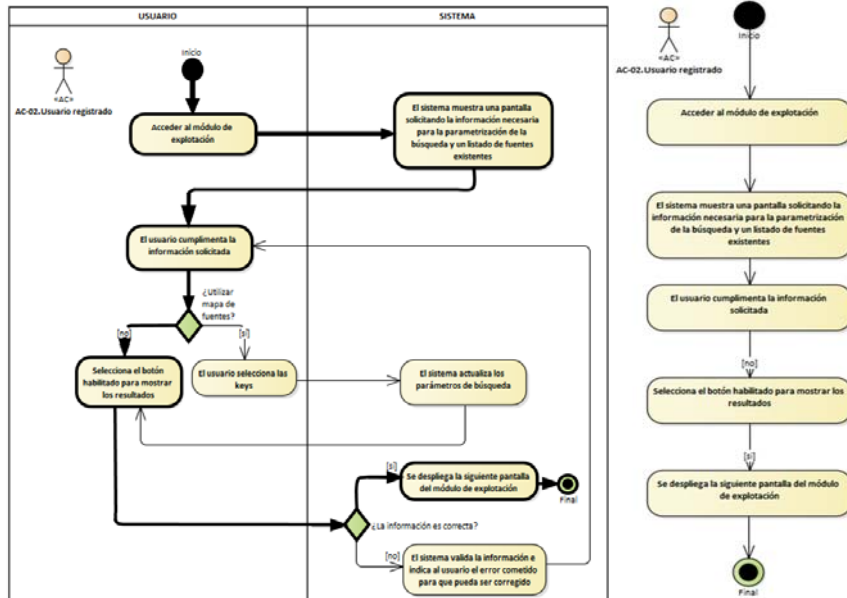


Fig. 1. Definición de un caso de uso con EA y prueba del sistema generada por NDT.

Tabla 1. Definición del plan de pruebas por empresa (S) e IWT2 (I)

FR	System Tests		FR	System Tests		FR	System Tests		FR	System Tests	
	S	I		S	I		S	I		S	I
MADMON											
FR-01	4	7	FR-02	6	9	FR-03	1	3	FR-04	4	7
FR-05	6	9	FR-06	1	3	FR-07	4	7	FR-08	6	9
FR-09	1	3	FR-10	4	7	FR-11	6	9	FR-12	1	3
FR-13	4	7	FR-14	6	9	FR-15	1	3	FR-16	1	2
MEXPL											
FR-01	2	5	FR-02	3	7						
MGEN											
FR-01	1	2	FR-02	1	2						
MGEST											
FR-01	1	2	FR-02	1	2	FR-03	4	7	FR-04	6	9
FR-05	1	3									
TOTAL	20	38		22	39		16	31		18	28

References

- Escalona, M.J., Aragón, G.: NDT. A model-driven approach for web requirements. IEEE Trans. Softw. Eng. 34, 377–394 (2008).
- García-García, J.A., Escalona, M.J., Domínguez-Mayo, F.J., Salido, A.: NDT-Suite: A Methodological Tool Solution in the Model-Driven Engineering Paradigm. J. Softw. Eng. Appl. 7, 206–217 (2014).
- Enterprise Architect <http://www.sparxsystems.com/products/ea/>: Enterprise Architect. In: Último acceso: Marzo 2018.
- Selenium: Selenium website documentation. RA-MA Ed. Last accessed Febr. 2018.