



GENERACIÓN DE PRIMERAS PRUEBAS PARA EL DESARROLLO DE SOFTWARE UTILIZANDO EL MODELO DE DOMINIO Y CASOS DE USO

Olivia Labastida Puertos¹, María José Mirón Chacón² y Enrique Alonso Páez²

1 TecNM: Instituto Tecnológico Superior de Huatusco, 2 TecNM: Instituto Tecnológico Superior de Huatusco.
olabastidap@huatusco.tecnm.mx

Uno de los procesos de desarrollo esenciales en la ingeniería de software, son las pruebas. Se estima que aproximadamente el 30% de la energía de un proyecto, se dedica a esta actividad. La prueba es un proceso dinámico, en el que el software se ejecuta con entradas diseñadas para verificar el comportamiento deseado e identificar defectos. Las pruebas exhaustivas de software suelen ser una tarea imposible debido a su tamaño, por lo que para garantizar la calidad es necesario definir estrategias que puedan identificar conjuntos de pruebas separables y alcanzables que tienen más probabilidades de encontrar defectos. Los casos de uso desarrollados durante la fase de Requisitos representan el comportamiento esperado del sistema desde el punto de vista del usuario y se han utilizado para generar las primeras pruebas, identificando escenarios basados en rutas de flujo básicas y flujos alternos. Sin embargo, al diseñar los casos de prueba, se definen entradas de datos y salidas esperadas específicas, que generalmente se obtienen sin utilizar un procedimiento de trazabilidad, de la información recopilada durante el proceso de elicitación. En este trabajo, se ha aplicado un método para el diseño de los casos de prueba, basándose en los casos de uso y el modelo de dominio, así como en un diccionario del dominio de datos, que contiene información para establecer valores de prueba y facilitar su trazabilidad. El caso de uso se especifica en una carta de descripción que contiene información general sobre el caso y el curso normal de los eventos correspondientes a la interacción entre el actor y el sistema, así como referencias a cursos de acción alternativos. El modelo de dominio captura las entidades, documentos y relaciones, así como sus atributos, que son descritos en un diccionario del dominio, que incluye, entre otras cosas, restricciones de valor, valores típicos y una fuente de información de documentos. Esto permite la derivación de los casos de prueba, que se incorporan en una tabla que describe el procedimiento de prueba, las entradas de datos (que indican el atributo del diccionario de dominio y el valor específico) y la respuesta deseable, así como el resultado observado. Con la aplicación ilustrativa, se generaron 37 casos de prueba sobre 10 módulos, utilizando el método de poda para sólo considerar un caso por cada flujo. La cobertura de la prueba fue del 100% y la identificación de defectos no graves fue del 30%, lo que determinó que el sistema podía funcionar y la corrección de los defectos detectados debía incluirse en su segunda versión. El método postulado demostró practicidad en su uso y contribuyó en la trazabilidad inicial de las primeras pruebas, además, se observa la necesidad de realizar procesos de priorización en los casos de prueba, al identificarse que es importante ampliar la cobertura a un análisis de valores límite, y al mismo tiempo se hace necesario utilizar otras técnicas como las clases de equivalencia robustas en procesos que tengan una complejidad lógica interna mayor, lo que aumentaría el grupo de casos de prueba a aplicar.