

UNIVERSIDAD DE COSTA RICA
SISTEMA DE ESTUDIOS DE POSGRADO

EVALUACIÓN DEL USO DE PRUEBAS BASADAS EN MODELOS EN
UN PROYECTO DE LA INDUSTRIA

Trabajo Final de Investigación Aplicada sometido a la consideración de la Comisión del Programa de Estudios de Posgrado en Computación e Informática para optar al grado de Maestría Profesional en Computación e Informática

REBECA OBANDO VÁSQUEZ

Ciudad Universitaria Rodrigo Facio, Costa Rica

2019

Dedicatoria

A mi mamá y mi familia, que siempre me han dado su apoyo incondicional.

Agradecimientos

Quiero agradecer a mis profesores guías Alexandra Martínez, Christian Quesada López y Marcelo Jenkins por el soporte y la guía académica que me dieron. Un especial agradecimiento a todos los participantes del estudio, por su colaboración y el tiempo invertido. También agradezco a los miembros del Grupo de Investigación en Ingeniería de Software Empírica (ESEG) de la Universidad de Costa Rica, por el apoyo y la retroalimentación recibida durante todo el proceso de la investigación.

"Este Trabajo Final de Investigación Aplicada fue aceptado por la comisión del programa de Estudios de Posgrado en Computación e Informática de la Universidad de Costa Rica, como requisito parcial para optar al grado y título de Maestría Profesional en Computación e Informática".

Dra. Gabriela Marín Raventós
Representante del Decano
Sistema de Estudios de Posgrado

Dra. Alexandra Martínez Porras
Profesora Guía

Dr. Cristian Quesada López
Lector

Dr. Marcelo Jenkins Coronas
Lector

Dr. Gustavo López Herrera
Representante Programa de
Posgrado en Computación e Informática

Rebeca Obando Vásquez
Sustentante

Índice general

Portada	i
Dedicatoria	ii
Agradecimientos	iii
Hoja de aprobación	iv
Resumen	ix
Índice de figuras	x
Índice de cuadros	xi
1. Introducción	1
1.1. Problema	2
1.2. Justificación	2
1.3. Objetivos	3
1.3.1. Objetivo General	3
1.3.2. Objetivos específicos	3
2. Marco Teórico	5
2.1. Niveles de pruebas de software	5
2.2. Tipos de pruebas de software	6
2.3. Pruebas de software automatizadas	7
2.4. Pruebas basadas en modelos	7

3. Metodología	10
3.1. Metodología de la revisión de literatura	10
3.1.1. Etapa 1: Diseño de la revisión	10
3.1.2. Etapa 2: Ejecución de la revisión	10
3.1.3. Etapa 3: Análisis de resultados de la revisión	12
3.2. Metodología del caso de estudio	12
3.2.1. Etapa 1: Diseño del caso de estudio	12
3.2.2. Etapa 2: Ejecución del caso de estudio	14
3.2.3. Etapa 3: Análisis de resultados del caso de estudio	14
4. Proceso y resultados de la revisión de literatura	15
4.1. Preguntas de investigación	15
4.2. Proceso de recolección de datos	15
4.3. Resultados de la revisión de literatura	17
4.3.1. Características de los estudios sobre MBT en la industria (RQ1)	17
4.3.2. Aceptación de MBT por parte de profesionales de la industria (RQ2)	20
4.3.3. Métricas usadas en la evaluaciones de MBT en la industria (RQ3)	21
5. Diseño del caso de estudio	24
5.1. Objetivo y preguntas de investigación	24
5.2. Contexto	24
5.2.1. Selección de los sujetos	25
5.2.2. Selección del sistema bajo pruebas: APP1	25
5.3. Proceso y recolección de datos	25
5.4. Instrumentación	27
5.4.1. Encuesta demográfica	28
5.4.2. La herramienta de MBT	28
5.4.3. Material de capacitación	28

5.4.4.	Encuesta de aceptación	29
5.4.5.	Notas de observación	29
5.5.	Métricas de evaluación	30
5.5.1.	Métricas de eficiencia	30
5.5.2.	Métricas de eficiencia	30
5.5.3.	Métricas de aceptación	31
5.6.	Amenazas a la validez	31
5.6.1.	Validez interna	32
5.6.2.	Validez externa	32
5.6.3.	Validez de conclusión	32
6.	Resultados del caso de estudio	33
6.1.	Análisis demográfico de los participantes	33
6.2.	Eficiencia y eficacia de los casos de prueba generados con MBT (RQ1)	33
6.2.1.	Eficiencia	34
6.2.2.	Eficacia	35
6.3.	Nivel de aceptación de MBT (RQ2)	36
6.3.1.	Percepción de utilidad	36
6.3.2.	Percepción de facilidad de uso	38
6.3.3.	Intención de uso	39
6.4.	Lecciones aprendidas	40
7.	Conclusiones	42
7.1.	Trabajo futuro	44
	Bibliografía	48
A.	Artículo de investigación	49

B. Encuesta demográfica	60
C. Encuesta de aceptación	64

Resumen

Las pruebas basadas en modelos (MBT, por sus siglas en inglés) es un enfoque que automatiza el diseño y la generación de casos de prueba a partir de un modelo que representa el sistema bajo prueba. MBT puede reducir el costo de las pruebas de software y mejorar la calidad de los sistemas. Sin embargo, estudios previos señalan la carencia evaluaciones empíricas sobre el uso de MBT y la necesidad transferir los conocimientos sobre este enfoque en el contexto de la industria.

Uno de los pilares de los equipos de desarrollo es la calidad del producto que se entrega, por ello la búsqueda nuevas técnicas que mejoren los procesos de calidad es valiosa para los equipos de desarrollo. De ahí que el objetivo de este estudio fuera evaluar el uso de MBT en un proyecto de una organización de desarrollo de software, con el propósito de analizar su eficiencia, eficacia y aceptación por parte de los ingenieros de software. Para ello, primero se realizó una revisión de literatura con el fin de conocer el estado del arte sobre MBT en la industria. Posteriormente, se realizó un caso de estudio en una empresa ubicada en Costa Rica, que cuenta con varios equipos de desarrollo. En este estudio participaron seis ingenieros de software, quienes modelaron un sistema, y luego generaron y ejecutaron casos de prueba sobre el mismo, usando una herramienta de MBT llamada MISTA.

En la revisión de literatura se identificaron 15 estudios, los cuales reportaban experiencias en el uso de diferentes herramientas de MBT en la industria, pero no se encontraron estudios sobre el proceso de aceptación del enfoque de MBT por parte de los profesionales.

Los resultados del caso de estudio indicaron que todos los participantes pudieron modelar al menos 4 de los 5 requerimientos funcionales, en un lapso de 20 a 60 minutos, logrando una cobertura de decisión de 39 % a 59 % por módulo. Se discutieron los hallazgos más significativos con respecto a la completitud de los modelos y los errores cometidos por los participantes durante las fases de modelado y concretización. Con respecto a la aceptación de MBT por parte de los ingenieros de software, los resultados sugieren que ellos valoran positivamente el enfoque de MBT pero no están satisfechos con la herramienta usada, debido a que no soporta las necesidades del proceso del equipo de ingenieros.

Índice de figuras

3.1. Metodología de la investigación.	11
4.1. Cantidad de estudios por año de publicación.	18
4.2. Cantidad de estudios por bases de datos.	19
5.1. Proceso utilizado en el caso de estudio.	27
6.1. Resultados de la percepción de utilidad.	37
6.2. Resultados de la percepción de facilidad de uso.	38
6.3. Resultados de la percepción de intención de uso.	40

Índice de cuadros

4.1. Soporte de las herramientas reportadas a las etapas del proceso de MBT. . .	19
6.1. Tiempo de modelado y concretización (en minutos) por sujeto.	34
6.2. Cantidad de requerimientos modelados, casos de prueba generados y porcentaje de cobertura obtenida.	36

Capítulo 1

Introducción

En la industria de software, uno de los aspectos más importantes del desarrollo de software es asegurar la calidad del producto a entregar. Existen diferentes tipos, niveles y técnicas de pruebas que buscan asegurar la calidad de un producto, entre las cuales están las pruebas funcionales. El principal objetivo de las pruebas funcionales es identificar no-conformidades entre el comportamiento del software que está siendo probado y sus requerimientos [25].

Las pruebas basadas en modelos (MBT, por sus siglas en inglés: *model-based testing*) son un enfoque para generar automáticamente casos de pruebas con base en modelos que representan el comportamiento del sistema a probar [23], y normalmente se utiliza para realizar pruebas funcionales. La robustez del modelo que representa el sistema bajo prueba y su entorno, es muy importante en la generación de los casos de prueba [23]. El proceso de pruebas basadas en modelos consta de cinco fases [24]:

1. El modelado del sistema con base en los requerimientos
2. La escogencia del criterio de selección de pruebas
3. La generación de los casos de prueba abstractos
4. La concretización de los casos de prueba abstractos
5. La ejecución de los casos de prueba concretizados

Villalobos et al. [26] señalan que los estudios de pruebas basadas en modelos carecen de evaluaciones empíricas o no han sido transferidos a un contexto industrial. Por este motivo, nuestra investigación plantea y conduce un caso de estudio para evaluar empíricamente el enfoque de MBT en un proyecto real de la industria, donde un grupo de ingenieros de software aplican el proceso de MBT usando la herramienta MISTA [10].

1.1. Problema

El estudio se realiza en una compañía *Fortune 100* con presencia en Costa Rica, cuyo negocio principal no es el desarrollo de software, pero tiene múltiples equipos que desarrollan software para proyectos internos de la empresa. El departamento encargado de desarrollar y dar soporte al software seleccionado para este estudio no cuenta con un proceso robusto de verificación de software, ni con un equipo de pruebas de software. Sin embargo, uno de los objetivos de dicho departamento es entregar soluciones con la mayor calidad posible, de manera que esta misión recaerá en los ingenieros de software, quienes deben buscar herramientas o enfoques que les faciliten su trabajo.

El enfoque de MBT se plantea entonces como una alternativa que podrían usar los ingenieros de software de esta organización para cumplir con el objetivo de entregar soluciones de alta calidad y potencialmente reducir el tiempo y costo asociado a las pruebas de software. De ahí que con esta investigación se busca medir la eficacia, la eficiencia y el nivel de aceptación del enfoque de MBT en el contexto de la organización descrita, con el fin de determinar si dicho enfoque puede resultar beneficioso para equipos de desarrollo de la industria en los cuales las tareas de verificación de software recaen sobre los ingenieros de software.

1.2. Justificación

Estudios previos sobre pruebas basadas en modelos [24, 26] señalan la necesidad de generar evaluaciones empíricas y transferir los conocimientos sobre pruebas basadas en modelos a la industria. Adicionalmente, no se encontraron estudios que evalúen específicamente la aceptación del enfoque de MBT por parte de profesionales de la industria.

Por otro lado, si bien la literatura recomienda que sean los expertos en pruebas quienes construyan el modelo del comportamiento del sistema, en algunos casos esto no es posible. Tal es el caso del departamento donde se realizó este estudio, el cual no cuenta con personal

de validación y verificación de software, por lo que resulta interesante evaluar el uso de MBT por profesionales que no son expertos en pruebas (en nuestro caso, ingenieros de software). Por estas razones, creemos que esta investigación puede contribuir a aumentar la evidencia empírica existente sobre el uso del enfoque de MBT en la industria, con respecto a su eficacia, eficiencia y aceptación por parte de un equipo de ingenieros de software.

1.3. Objetivos

A continuación se detallan los objetivos de la presente investigación.

1.3.1. Objetivo General

El objetivo general de esta investigación es *evaluar el uso de pruebas basadas en modelos en un contexto industrial, en términos de su eficacia, eficiencia y aceptación, por parte de los ingenieros de software.*

1.3.2. Objetivos específicos

Los objetivos específicos de la investigación son los siguientes:

1. Identificar los aspectos esenciales de una evaluación empírica orientada a personas, en el contexto de profesionales usando pruebas basadas en modelos.
2. Evaluar la eficacia, eficiencia y aceptación de un enfoque de pruebas basadas en modelos mediante un caso de estudio con ingenieros de software en la industria.

Con el primer objetivo específico se busca identificar cómo se ha evaluado el enfoque de pruebas basadas en modelos en la industria y cómo se ha evaluado la aceptación de tecnología. El segundo objetivo específico busca ejecutar y evaluar el proceso de MBT con un grupo

de ingenieros de software en el contexto de un proyecto de desarrollo en una organización de la industria.

La estructura de este documento se describe a continuación. El capítulo 2 contiene el marco teórico con los conceptos relevantes para este trabajo. El capítulo 3 explica la metodología que se usó para llevar a cabo la investigación. El capítulo 4 presenta la descripción y los resultados de la revisión de literatura realizada. El capítulo 5 contiene la descripción del caso de estudio realizado, incluyendo los instrumentos, métricas y proceso de recolección de datos. El capítulo 6 muestra los resultados obtenidos en el caso de estudio y las lecciones aprendidas. Por último, el capítulo 7 presenta las conclusiones y posibles trabajos futuros.

Capítulo 2

Marco Teórico

Las actividades de verificación y validación de software son de las más importantes durante el ciclo de desarrollo de software, pues estas buscan entregar software de calidad. Las pruebas de software se pueden definir como el proceso de ejecutar el software para determinar si cumple con su especificación de requerimientos y se ejecuta correctamente en los ambientes especificados [29]. Por ello, los requerimientos del sistema son importantes para determinar si el comportamiento del sistema corresponde a un fallo o no.

2.1. Niveles de pruebas de software

En el ciclo de desarrollo del software existen diferentes niveles de pruebas. Un nivel de pruebas se define, según el ISTQB [6], como el conjunto de actividades de pruebas de software manejadas y organizadas en conjunto. Existen cuatro niveles de pruebas [6], que se listan a continuación.

1. *Pruebas de componentes*: son pruebas que buscan probar componentes individuales de software. También se conocen como pruebas unitarias.
2. *Pruebas de integración*: su intención es probar la comunicación entre los componentes de software que fueron probados de manera individual por medio de las pruebas unitarias.
3. *Pruebas de sistema*: prueba la colección de componentes que en conjunto representan un producto entregable.
4. *Pruebas de aceptación*: este nivel de pruebas está enfocado en el comportamiento y capacidades del producto. Los objetivos de estas pruebas son validar el sistema de ma-

nera completa y asegurar que el sistema cumple con las especificaciones funcionales y no funcionales.

2.2. Tipos de pruebas de software

También existen tipos de pruebas, donde un tipo se define como un conjunto de actividades que tienen como objetivo probar características específicas de un sistema. De acuerdo a la ISTQB [6], existen los siguientes tipos de pruebas:

1. *Pruebas de caja blanca*: estas pruebas se basan en la estructura interna (implementación) del sistema, que puede incluir código, arquitectura, flujos de trabajo, etc. El objetivo de las pruebas de caja blanca es verificar el comportamiento interno del sistema.
2. *Pruebas de caja negra*: es una técnica de pruebas que se basa en las especificaciones del sistema, como pueden ser requerimientos, casos de uso, etc. Esta técnica de pruebas se centra en las entradas y salidas del objeto bajo prueba, sin tomar en cuenta su estructura interna.
3. *Pruebas funcionales*: estas pruebas buscan evaluar las funcionalidades que el sistema debe soportar, según sus requerimientos funcionales. Las pruebas funcionales deben ejecutarse en todos los niveles. Normalmente se utilizan técnicas de pruebas de caja negra, debido a que la intención de las pruebas funcionales es verificar el comportamiento del sistema desde un punto de vista externo.
4. *Pruebas no funcionales*: estas pruebas evalúan características del sistema tales como usabilidad, eficiencia o seguridad, las cuales aplican a todo el sistema y por consiguiente no son una funcionalidad específica. Las pruebas no funcionales pueden ser ejecutadas en todos los niveles de pruebas.

5. *Pruebas asociadas a cambios*: Estas pruebas son usadas cuando se agrega nueva funcionalidad o se corrige un defecto en el software. Su objetivo es verificar que el nuevo código no cause ningún problema a las funcionalidades existentes del sistema. Entre este tipo de pruebas se encuentran las pruebas de regresión y las pruebas de confirmación, las cuales pueden ser utilizadas en todos los niveles.

2.3. Pruebas de software automatizadas

De acuerdo con Karhi et al. [19], la automatización de las pruebas de software se puede definir como la automatización de las actividades de pruebas, la ejecución de *scripts*, y la verificación de los requerimientos mediante el uso de herramientas de pruebas automatizadas. Las razones para utilizar pruebas automatizadas incluyen: reducir el tiempo de ejecución de las pruebas e incrementar la eficiencia de las mismas. No obstante, la automatización de las pruebas es un proceso de alto costo y riesgo [19].

2.4. Pruebas basadas en modelos

Las pruebas basadas en modelos es un enfoque automatizado para la creación de casos de prueba (típicamente funcionales) con base en un modelo del sistema bajo prueba (SUT, por sus siglas en inglés: *software under test*), que representa las funcionalidades que se desean validar [27]. El proceso de pruebas basadas en modelos consta de cinco etapas, las cuales se describen a continuación [4, 26]:

1. *Modelado del sistema con base en los requerimientos*: en esta etapa se crea un modelo del SUT a partir de los requerimientos del sistema.
2. *Escogencia del criterio de selección de pruebas*: en esta etapa se selecciona el criterio que se usará para generar y seleccionar los casos de prueba. Los criterios pueden ser de cobertura de estados, de profundidad, y creación aleatoria, entre otros.

3. *Generación de los casos de prueba abstractos:* en esta etapa se generan los casos de prueba abstractos, con base en el modelo del SUT creado y el criterio de selección de pruebas escogido anteriormente. Estos casos de prueba abstractos aún no se pueden ejecutar sobre el SUT, pues son secuencias de operaciones sobre el modelo (no sobre el SUT).
4. *Concretización de los casos de prueba abstractos:* en esta etapa es donde se concretizan los casos de prueba abstractos, para que sean ejecutables y puedan ser aplicados al SUT, mediante alguna herramienta de automatización de pruebas (que permita hacer una correspondencia entre las secuencias de los casos abstractos y los elementos ejecutables del SUT).
5. *Ejecución de los casos de prueba concretizados:* en esta etapa se corren los casos de pruebas generados sobre el SUT y se analizan los resultados obtenidos.

El enfoque de MBT tiene múltiples beneficios con respecto a enfoques más tradicionales de pruebas de software. De acuerdo al estudio terciario realizado por Villalobos et al. [27], MBT tiene mayor probabilidad de encontrar errores que las pruebas tradicionales, y los casos de prueba generados mediante MBT pueden ser de mejor calidad que los producidos manualmente (debido a que las herramientas de MBT pueden generar grandes conjuntos de casos de prueba que puede resultar en una alta cobertura de código). Además, MBT puede ayudar a detectar problemas (errores, inconsistencias, u omisiones) en los requerimientos, y a reducir el tiempo y costo del proceso de pruebas [27]. Este mismo estudio [27] también reporta limitaciones que tiene el enfoque de MBT, las cuales de resumen a continuación. En primer lugar, MBT requiere de un conjunto diferente de habilidades en los ingenieros de calidad, por lo que un ingeniero de calidad tradicional puede encontrar el enfoque MBT difícil de adoptar. En segundo lugar, MBT normalmente se recomienda para organizaciones con cierto nivel de madurez en su proceso de pruebas, debido a su complejidad. En tercer lugar, MBT normalmente se aplica solo a pruebas funcionales, dejando por fuera las prue-

bas no-funcionales. Finalmente, MBT tiene problemas cuando los requerimientos no están actualizados, y puede generar muchos casos de pruebas de poca utilidad si el modelo creado es de baja calidad.

Por otro lado, Utting et al.[24] plantean los siguientes desafíos en relación con MBT: la necesidad de contar con más evidencia empírica sobre la selección del enfoque de MBT y sus herramientas; la necesidad de entender las diferencias y similitudes entre los diferentes enfoques existentes de MBT, para poder usar el más adecuado según el contexto; y la necesidad de considerar pruebas no-funcionales en los estudios de MBT.

Capítulo 3

Metodología

Para cumplir con los objetivos de la investigación, se siguió la metodología ilustrada en la figura 3.1. El primer objetivo específico se abordó por medio de una revisión de literatura, orientada a identificar cómo se ha evaluado el enfoque de MBT en la industria, y cómo se ha evaluado la aceptación de tecnología. El segundo objetivo específico fue abordado mediante el diseño y la ejecución de un caso de estudio, en el que participaron ingenieros de software de la industria. En las siguientes secciones se detallan las etapas y tareas seguidas para completar la revisión de literatura y el caso de estudio.

3.1. Metodología de la revisión de literatura

A continuación se describen las etapas asociadas a la revisión de literatura realizada. La cual se basó parcialmente en las guías de Petersen et al. [20] y Kitchenham et al. [2].

3.1.1. Etapa 1: Diseño de la revisión

El proceso de la revisión de literatura consistió de las siguientes actividades:

1. Definición de las preguntas de investigación.
2. Definición del proceso de búsqueda: esto incluye la selección de los criterios de selección y calidad de los estudios, además de la definición de los datos a extraer.

3.1.2. Etapa 2: Ejecución de la revisión

En esta etapa se procedió a la ejecución de la revisión de literatura, según lo planificado en la etapa anterior, con el fin de identificar y procesar los estudios que ayudarían a responder las

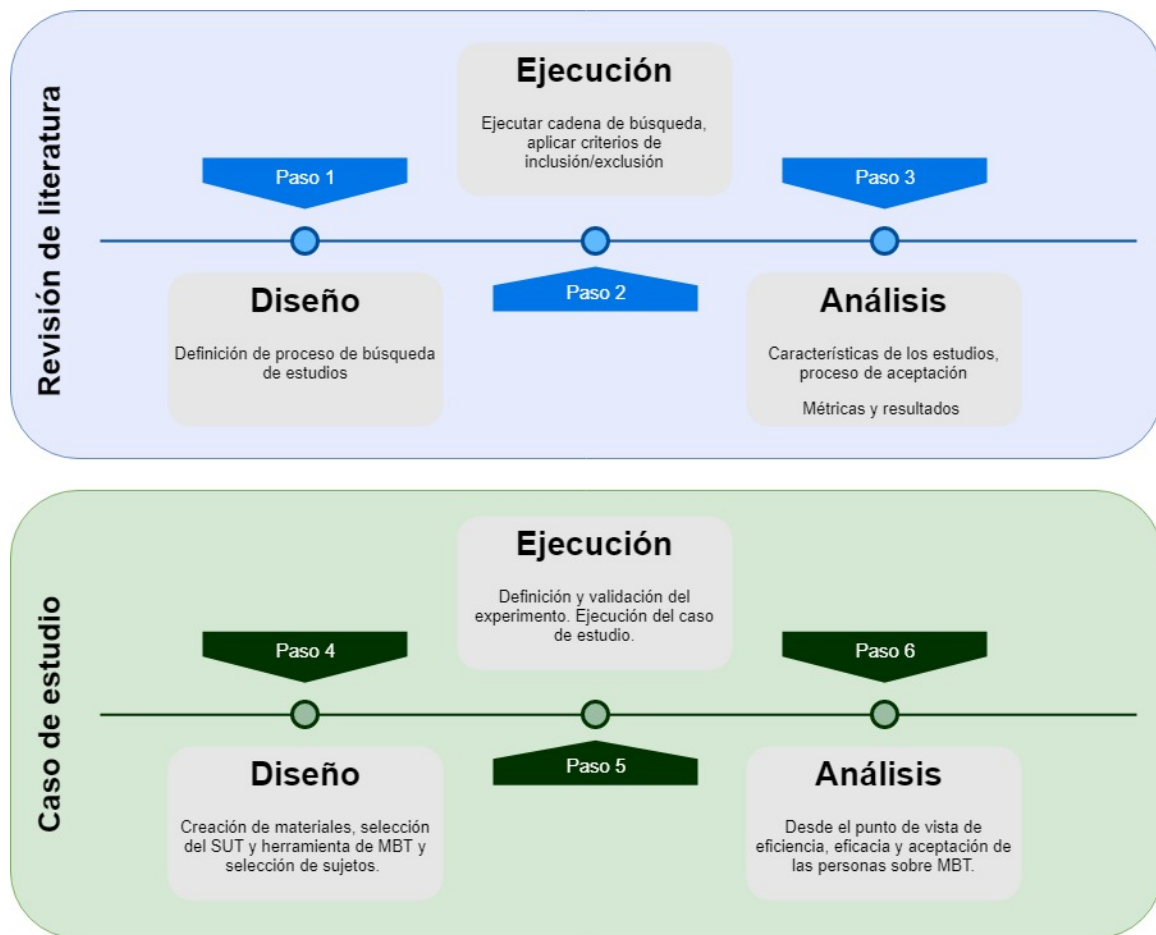


Figura 3.1: Metodología de la investigación.

preguntas de investigación planteadas. La ejecución y recolección de datos se desarrolló en febrero del 2018.

3.1.3. Etapa 3: Análisis de resultados de la revisión

El análisis de los resultados de la revisión se hizo según las preguntas de investigación planteadas. De esta manera, se analizaron (1) las características de los estudios sobre MBT en la industria, (2) los procedimientos o métodos usados para evaluar la aceptación de MBT por profesionales de la industria y (3) las métricas usadas en evaluaciones empíricas sobre la aplicación de MBT en la industria.

3.2. Metodología del caso de estudio

A continuación se describen las etapas asociadas al caso de estudio realizado.

3.2.1. Etapa 1: Diseño del caso de estudio

Los pasos seguidos durante la planificación (preparación) del caso de estudio se detallan a continuación.

1. Definición del objetivo y las preguntas de investigación del caso de estudio.
2. Selección de los sujetos. Se elige a los participantes del caso de estudio, de entre los profesionales de la organización que tienen el perfil deseado.
3. Selección del SUT y obtención de material asociado requerido. En este caso, los ingenieros expertos del sistema nos proveyeron los requerimientos funcionales del SUT.
4. Definición de las métricas y tratamientos: en este paso se definen las métricas a utilizar para la evaluación de eficiencia, eficacia y aceptación.

5. Definición de los datos a recolectar y analizar sobre los sujetos del estudio. Esto incluye la creación de una encuesta demográfica.
6. Definición del instrumento de aceptación de tecnología a utilizar.
7. Definición del proceso de ejecución del caso de estudio: en esta etapa se diseñan los pasos que los sujetos deben seguir para aplicar el enfoque de MBT sobre la aplicación seleccionada.
8. Definición del material de capacitación sobre MBT y MISTA. El material de capacitación incluye una presentación donde se describen los conceptos básicos y el proceso de MBT, así como sus beneficios. Además, la capacitación aborda el uso de la herramienta MISTA.
9. Validación del caso de estudio: al ser un estudio con sujetos, se debe hacer una validación del diseño. Para ello, se hizo una corrida piloto del caso de estudio con 7 sujetos (distintos a los sujetos de la corrida oficial reportada más adelante), usando el mismo SUT y la misma herramienta de MBT. Los resultados del piloto evidenciaron algunos problemas con el diseño inicial, los cuales se presentan a continuación:
 - Algunos participantes no lograron completar la etapa de modelado, lo que resultó en frustración. Se lograron identificar como posibles causas los siguientes aspectos: los requerimientos del SUT no eran lo suficientemente claros, el tiempo para estudiar el material no fue suficiente, y la interacción entre los participantes no ayudó a completar la tarea.
 - Durante la etapa de modelado los participantes vieron y compararon los modelos de sus compañeros, lo que generó incertidumbre y frustración debido a que pensaban que el compañero tenía la respuesta correcta y ellos no. Por ello, esta comparación no fue permitida en la corrida oficial.

Estos problemas resultaron en ruido no controlado en los datos recolectados de la corrida piloto. Debido a esto, se hicieron una serie de cambios la diseño inicial, dando como resultado el diseño final del caso de estudio, que se describe en el Capítulo 5.

3.2.2. Etapa 2: Ejecución del caso de estudio

Una vez validado el diseño del caso de estudio, se hace la corrida oficial (nueva) con nuevos sujetos. Se recolectan los datos especificados durante el diseño, se pasan los instrumentos, y se ejecuta todo el proceso previamente definido y validado, para obtener los resultados del estudio y analizarlos.

3.2.3. Etapa 3: Análisis de resultados del caso de estudio

Con base en los resultados obtenidos para cada sujeto durante la ejecución del caso de estudio, se hacen los análisis correspondientes por variable de respuesta. Usando los datos recolectados con los instrumentos definidos en la etapa 2, se resumen y explican los resultados obtenidos por los sujetos a nivel de eficiencia, eficacia y aceptación .

En los siguientes capítulos se detallan el diseño de la revisión de literatura y sus resultados, así como el diseño del caso de estudio y los resultados obtenidos.

Capítulo 4

Proceso y resultados de la revisión de literatura

En este capítulo se describe el proceso de revisión de literatura realizado, así como sus principales resultados. El proceso de revisión de literatura seguido se basó parcialmente en las guías de Petersen et al. [20] y Kitchenham et al. [2], mas no se puede catalogar como una revisión *sistemática* de literatura (RSL).

4.1. Preguntas de investigación

Para esta revisión de literatura se plantearon las siguientes tres preguntas de investigación:

- RQ1. ¿Cómo se caracterizan los estudios sobre pruebas basadas en modelos en la industria?
- RQ2. ¿Cómo se evalúa la aceptación de MBT por parte de profesionales de la industria?
- RQ3. ¿Cuáles métricas se han usado para evaluar empíricamente la aplicación de MBT en la industria?

4.2. Proceso de recolección de datos

Para responder a las preguntas de investigación, se definió la siguiente cadena de búsqueda, usando tres artículos de control [28, 16, 15] para calibrar los términos:

((“tool*” OR “framework*” OR “automat*”) AND
 (“software” OR “system”) AND
 (“model-based test*” OR “MBT” OR “model based testing”) AND
 (“industr*”) AND
 (“experiment” OR “empirical” OR “study” OR “evaluation” OR “experience report”))

Se usó la base de datos referencial SCOPUS, debido a que esta base indexa artículos de la ACM, IEEE, ScienceDirect y Elsevier. La búsqueda se hizo sobre título, resumen y palabras clave de los artículo. La cadena se filtró usando las reglas de SCOPUS, como se muestra a continuación:

(LIMIT-TO (SUBJAREA, “COMP”) OR LIMIT-TO (SUBJAREA, “ENGI”))

A los estudios obtenidos con la cadena filtrada, se les aplicó el proceso de inclusión y exclusión sobre el título, resumen y palabras clave. Los criterios de inclusión usados fueron los siguientes:

- I1: El estudio posee una evaluación de los datos recolectados
- I2: El estudio menciona y describe la herramienta de MBT utilizada
- I3: El estudio describe el proceso de aplicación de MBT en la industria

Los criterios de exclusión fueron los siguientes:

- E1: El estudio está en un idioma distinto al inglés, o no se puede acceder el texto completo
- E2: El estudio no está relacionado con MBT
- E3: El estudio no corresponde a un artículo científico de conferencia o revista

Para aplicar los criterios de inclusión y exclusión, se usó una hoja de cálculo con la información básica de cada estudio (autores, año de publicación, título, palabras clave) ordenada

por la fecha de publicación (de la más reciente a la más antigua). Dicha hoja contiene una columna para cada criterio de inclusión y exclusión, además de una columna con el resultado final de aplicar los criterios usando la fórmula:

$$(\text{NOT} (\text{E1 OR E2 OR E3}) \text{ AND } (\text{I1 AND I2 AND I3}))$$

Como resultado del proceso de inclusión y exclusión, se obtuvieron quince estudios a analizar. Para cada uno de estos estudios, se extrajeron los siguientes datos asociados a cada pregunta de investigación:

1. Datos extraídos para responder la RQ1: tipo de estudio, foro de publicación, año, herramienta de MBT usada, tipo de modelo, etapas del proceso de MBT en las que puede ser usada la herramienta, SUT usado.
2. Datos extraídos para responder la RQ2: características del proceso de aceptación por parte de las personas que usan MBT.
3. Datos extraídos para responder la RQ3: métricas usadas para evaluar MBT en la industria, resultados y evaluación.

4.3. Resultados de la revisión de literatura

Esta sección presenta los resultados de la revisión de literatura, según las preguntas de investigación planteadas.

4.3.1. Características de los estudios sobre MBT en la industria (RQ1)

La revisión de literatura permitió obtener las principales características de los estudios de MBT en la industria. A continuación se describen los principales hallazgos.

Como lo muestra la figura 4.1, la mayoría de los estudios analizados son del año 2017. Por otro lado, según se puede observar en la figura 4.2, la mayoría de los estudios analizados

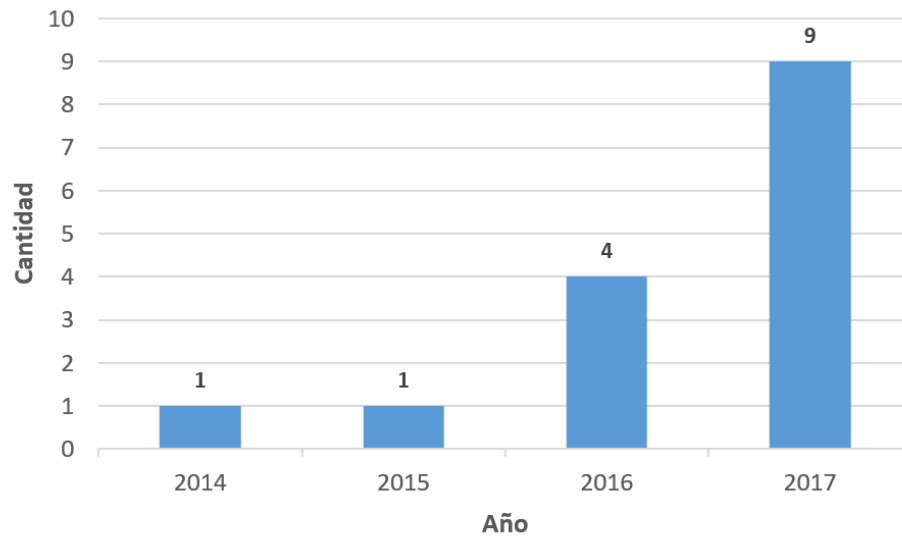


Figura 4.1: Cantidad de estudios por año de publicación.

fueron publicados por la IEEE. En cuanto al tipo de estudio, de los quince estudios analizados, doce fueron reportados como casos de estudio, dos como experimentos controlados en la industria y uno como reporte de experiencia.

Los estudios analizados reportan el uso de diferentes herramientas o *frameworks*, los cuales pueden ser usados en una o más etapas del proceso de MBT. Según se observa en la tabla 4.1, ocho de las herramientas fueron usadas en la etapa de modelado y nueve en la etapa de generación de casos de prueba abstractos. Sin embargo, solo tres herramientas se usaron en la etapa de concretización y dos en la etapa de ejecución de los casos de prueba. Estos resultados muestran una clara tendencia en el soporte que dan las herramientas a las etapas del proceso de MBT: la mayoría dan mayor soporte a las etapas iniciales de modelado y generación de casos de prueba abstracto, que a etapas posteriores de concretización y ejecución de los casos de prueba.

Con respecto al tipo de modelo (de MBT) utilizado en los estudios, ocho de los estudios reportaron el uso de modelos basados en máquinas de estados, un estudio usó un modelo

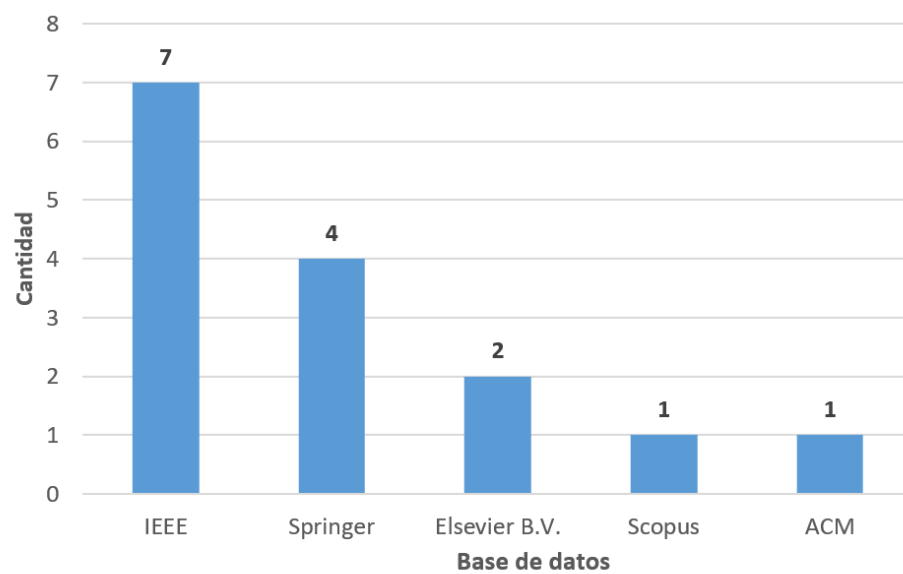


Figura 4.2: Cantidad de estudios por bases de datos.

Cuadro 4.1: Soporte de las herramientas reportadas a las etapas del proceso de MBT.

Estudio	Generación de Modelado	Concretización de casos de prueba	Ejecución de casos de prueba	casos de prueba
ARME & MaTeLo [14]	X			
Asyioco Algorithm [15]		X		
AQAF with AQAT [17]	X	X	X	X
Certifylt & TOCL [7]	X	X		
Conformiq Designer [28]		X		
DTRON [1]				X
ERLY MARSH [5]		X		
FORMAT [11]	X			
Generation tool & MBTCG [21]		X	X	
MoMuT [12]		X		
SART [9]	X			
SBS-Super [16]		X		
SPLOT [13]	X			
The Fraunhofer model [21]				
UncerTum [32]	X	X	X	

por eventos, tres estudios usaron su propio tipo de modelo (específico de la herramienta empleada), y dos estudios no reportaron el tipo de modelo que usaban.

En relación al SUT utilizado en los estudios, se reportaron librerías para software de criptografía, sistemas de alarmas, sistemas de televisión digital, sistemas de condición física, sistemas que recolectan y procesan información biométrica, y sistemas de la NASA. Algunos estudios no tenían información sobre el tipo de SUT usado.

4.3.2. Aceptación de MBT por parte de profesionales de la industria (RQ2)

Ninguno de los estudios analizados reportaba procedimientos o métodos para evaluar el nivel de aceptación por parte de los profesionales de la industria. Sin embargo, algunos estudios [11, 14, 21] mencionan como un aspecto importante el uso de material de entrenamiento sobre cómo modelar el SUT en la herramienta particular de MBT usada. Un estudio [21] menciona el tiempo de modelado como un factor asociado a la aceptación del enfoque de MBT.

Dado que no se obtuvieron resultados sobre métodos o procedimientos para evaluar la aceptación de MBT en la industria, se realizó una segunda búsqueda en mayo del 2018, la cual tenía como objetivo entender el proceso de aceptación de software o herramientas o *frameworks* en la industria, y sus métricas asociadas. Para esto se utilizó la siguiente cadena de búsqueda:

(“adoption process”) AND (“industr*”) AND (“software” OR “tool” OR “framework”)

Los detalles de los resultados obtenidos con esta segunda cadena de búsqueda se explican seguidamente.

Proceso de aceptación tecnológica

Varios trabajos han estudiado el tema de la aceptación de tecnología [8]. De acuerdo a Bettiga y Lamberti [3, 18], el proceso de adopción de una tecnología o herramienta está dividido en dos partes: el cognitivo y el afectivo. La parte cognitiva se puede medir utilizando técnicas como el modelo de aceptación de tecnología (TAM, por sus siglas en inglés: *Technology acceptance model*) [8], el cual mide la aceptación desde tres perspectivas: facilidad de uso, percepción de utilidad, e intención de uso. Sin embargo, para la parte afectiva es necesario medir el nivel de emoción o sentimiento de los usuarios mientras interactúan con el sistema. Para ello se recomienda el uso de sensores u otros mecanismos que permitan medir la respuesta física y sensorial de los usuarios al usar la herramienta o sistema [3].

Otros autores, tales como Jovanović et al. [18], conceptualizan el proceso de aceptación tecnológica en varias etapas y particularmente proponen una metodología para adoptar las metodologías ágiles, que se divide en tres etapas: preparación, transformación del equipo y, por último, una transformación a nivel de la organización. De esta manera se busca la adopción de un nuevo proceso, empezando por los equipos de software y luego motivando su adopción a nivel organizacional.

En síntesis, la respuesta a la pregunta de investigación RQ2 es que los estudios analizados sobre MBT no aportan evidencia sobre el proceso de aceptación de MBT por parte de los profesionales en la industria. No obstante, sí se encontró evidencia de cómo se ha medido el nivel de aceptación de tecnología por parte de las personas. Un ejemplo de ello es el modelo TAM, que se utilizó en el caso de estudio que se describe en el capítulo 5.

4.3.3. Métricas usadas en la evaluaciones de MBT en la industria (RQ3)

Algunos estudios previos han aplicado el enfoque de MBT en áreas específicas de la industria [28, 11, 16, 12]. Por ejemplo, Weißleder y Schlingloff [28] describen la aplicación

de una herramienta de MBT en sistemas embebidos, donde se comparan la eficiencia y el esfuerzo de generar casos de prueba utilizando MBT versus la creación de casos de prueba manuales. Una comparación similar se presenta en el estudio de Dadeau et al. [7].

Otros estudios reportan el tiempo de modelado y el número de casos de prueba generados como métricas que demuestran las ventajas de usar el enfoque de MBT [11, 16]. El trabajo de Gebizli et al. [13] reporta, como medición de esfuerzo, el tiempo requerido por ingenieros de software para crear manualmente los modelos. Además, la **cobertura de código** es usada como una métrica de eficiencia para comparar MBT con las pruebas manuales [16]. La eficacia para encontrar errores es medida contando la cantidad de errores encontrados [16]

También algunos estudios proponen heurísticas para la generación de casos de prueba en diferentes aplicaciones de la industria, tal como el trabajo de Fellner et al. [12], que mide el rendimiento y la calidad de los casos de prueba generados por las heurísticas propuestas.

Finalmente, algunos estudios han evaluado la **correctitud y la completitud** del modelo. Tal es el caso del trabajo de Zhang et al. [32], que presenta un framework llamado UncerTum, el cual permite crear modelos para ser usados en las siguientes fases de MBT, en el contexto de sistemas ciberfísicos (CPS) donde se logra manejar la incertidumbre. Los autores reportan que la correctitud del modelo fue evaluada ejecutando el modelo y la completitud, en términos de conceptos definidos para la incertidumbre para los CPS.

Si bien la mayoría de los resultados de estos estudios muestran buenas experiencias con el uso de MBT para generar casos de prueba, también mencionan que los resultados dependen de un correcto modelado del SUT [11, 14]. Como el tiempo de modelado puede llegar a ser alto, es necesario contar con un buen entrenamiento en modelado [21].

El caso de estudio realizado en esta investigación combina algunas de las métricas de eficiencia, eficacia y aceptación de estudios previos [28, 11, 16, 14, 8]. Nuestra investigación define además un proceso para usar MBT en un proyecto de software real, que podría ser utilizado en contextos similares con el fin de promover la adopción de MBT en la industria.

Con esta revisión de literatura se logró recolectar información sobre los aspectos esenciales de la aplicación de pruebas basadas en modelos en la industria, desde características generales como los años de publicación, tipos de herramientas, métricas utilizadas y recomendaciones hechas en la literatura, que fueron el marco para el diseño del caso de estudio que se define en el capítulo 5. Además, dicha revisión identificó que los estudios analizados no aportaban evidencias ni mediciones sobre la aceptación de MBT por parte de las personas (profesionales). Sin embargo, con los datos obtenidos con la segunda búsqueda realizada, orientada a los procesos de aceptación de nuevas tecnologías, fue posible cubrir los aspectos generales para una evaluación del uso de MBT en personas, como se planteó en el primer objetivo específico.

Capítulo 5

Diseño del caso de estudio

A continuación se detalla el diseño del caso de estudio, incluyendo el objetivo, las preguntas de investigación, el contexto, la selección de sujetos, los instrumentos y el proceso de análisis de datos.

5.1. Objetivo y preguntas de investigación

Este caso de estudio se realizó siguiendo el protocolo de Wohlin et al. [31].

El objetivo del caso de estudio fue evaluar el uso del enfoque de MBT en un contexto industrial, en términos de su eficiencia, eficacia y aceptación por parte de los ingenieros de software. Para alcanzar este objetivo, se plantean las siguientes preguntas de investigación:

- RQ1: ¿Cuál es la eficiencia y eficacia de los casos de prueba generados por ingenieros de software al usar el enfoque de MBT?
- RQ2: ¿Qué nivel de aceptación tiene el enfoque de MBT entre los ingenieros de software?

5.2. Contexto

Este estudio fue ejecutado en una compañía localizada en Costa Rica, que cuenta con varios equipos de desarrollo dedicados a soportar las necesidades internas y externas de la empresa.

La actividad principal de la empresa no es el desarrollo de software, sin embargo, los diferentes equipos de software trabajan en diferentes tecnologías y buscan seguir los estándares de desarrollo de la industria tomando la calidad del producto como uno de los pilares

más importantes de los productos que se desarrollan incluyendo el mantenimiento de las aplicaciones legadas.

5.2.1. Selección de los sujetos

Los participantes fueron seis ingenieros de software de diferentes departamentos, quienes trabajan en el diseño y desarrollo de nuevas soluciones para proyectos internos. Los sujetos fueron seleccionados por conveniencia, de acuerdo con sus prioridades, el tiempo libre y su interés por aprender un nuevo enfoque de pruebas de software. La compañía seleccionada tiene como política que los empleados puedan utilizar algunas horas de trabajo a la semana para aprender nuevas tecnologías que puedan ser útiles en su trabajo diario.

5.2.2. Selección del sistema bajo pruebas: APP1

Para este caso de estudio, se tuvo como objeto una aplicación web que maneja solicitudes internas de la empresa. La aplicación fue liberada a producción en febrero del 2018 y se encontraba en fase de mantenimiento para el momento en que se realizó el estudio. En adelante nos referiremos a esta aplicación como APP1 por razones de confidencialidad. APP1 cuenta con cuatro módulos. Sin embargo se decidió utilizar solamente el módulo principal como el SUT para este caso de estudio, debido a las restricciones de tiempo que tenían los participantes del estudio.

5.3. Proceso y recolección de datos

El proceso seguido para este estudio se muestra en la figura 5.1. El primer paso fue calibrar los instrumentos, que se detallan en la subsección 5.4. Los pasos del 2 al 7 fueron ejecutados por los participantes del estudio. Los detalles se explican a continuación.

- Paso 2. Entrenamiento:

- Una semana antes de la sesión se envió un correo con los enlaces para instalar MISTA y Selenium, los requerimientos de la aplicación a modelar, la documentación sobre MISTA y un video de cómo hacer un modelo en MISTA.
 - Se seleccionó una sala y hora para llevar a cabo la ejecución del caso de estudio.
 - Se les explicó el objetivo del estudio y la dinámica de la sesión de entrenamiento.
 - Se reforzaron los conceptos de MBT y MISTA, repasando los materiales enviados de previo, incluyendo la lista de requerimientos del SUT. Esto se hizo en un lapso de 15 minutos.
 - Se revisó la instalación de las herramientas MISTA, Selenium y la versión específica de navegador.
- Paso 3. Aplicación de la encuesta demográfica: esta encuesta permite obtener los datos de interés sobre los participantes.
 - Pas 4. Modelado de la herramienta: se les da el tiempo para modelar el módulo. Se le solicita a cada participante apuntar el tiempo de inicio y el tiempo final que los llevó finalizar el modelado. Si el participante no termina el modelo en 60 minutos se le dará el modelo previamente creado para que continúe con la siguiente parte.
 - Paso 5. Generación de los casos de prueba: a los participantes se les dio tiempo para agregar los comandos necesarios en el módulo de comandos (MIM) de la aplicación MISTA para poder generar los casos de prueba concretos. Se le solicita a cada participante apuntar el tiempo de inicio y el tiempo final que les llevó finalizar la tarea. Si el participante no genera los casos de prueba en 45 minutos se le dará el modelo previamente creado para que continúe con la siguiente parte.
 - Paso 6. Ejecución de los casos de prueba: se les solicitó a los participantes correr los casos de prueba generados en Selenium con el fin de completar todo el proceso de MBT.

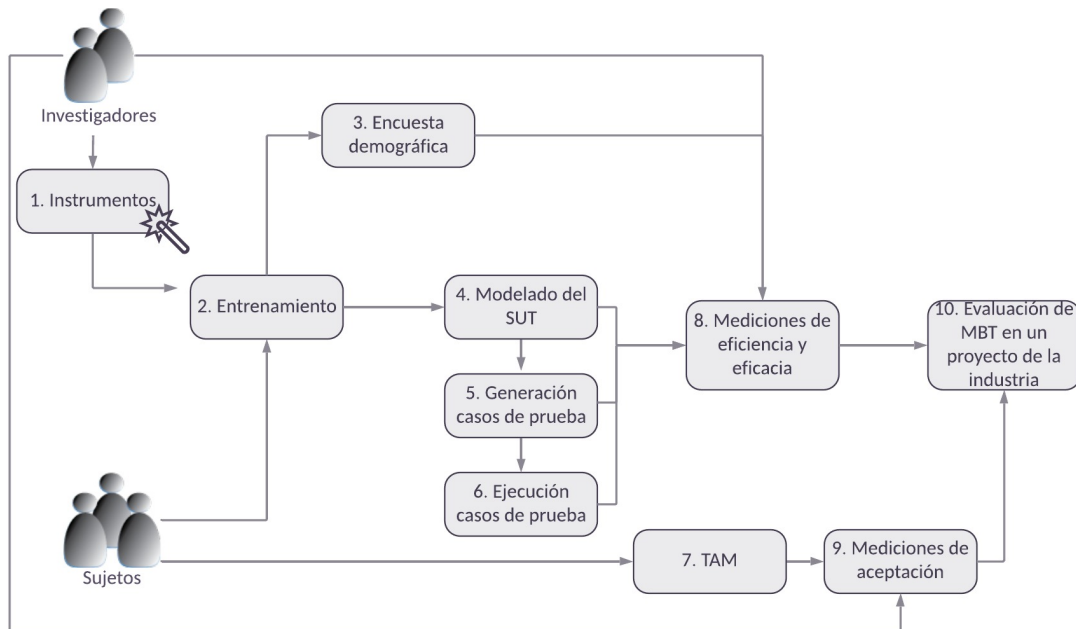


Figura 5.1: Proceso utilizado en el caso de estudio.

- Paso 7. Encuesta de adopción: cuando finalizó el proceso se les solicitó a los participantes llenar la encuesta de adopción.

Los pasos del 8 al 10 fueron llevados a cabo por los investigadores del estudio, quienes recolectaron y computaron todas las métricas necesarias para poder hacer el análisis de los datos y su interpretación.

5.4. Instrumentación

A continuación se describen los instrumentos utilizados en el caso de estudio.

5.4.1. Encuesta demográfica

El objetivo de este instrumento fue recolectar información sobre los sujetos, en particular, su edad, educación, experiencia laboral, cargo actual, tipos de proyectos en los que trabajan, conocimiento previo sobre el SUT y conocimiento previo sobre MBT. Esta encuesta se encuentra en el Apéndice B. La encuesta no solicitó el nombre de los participantes, sin embargo, se le asignó un identificador único a cada participante para poder asociar los resultados de este instrumento con los de instrumentos posteriores.

5.4.2. La herramienta de MBT

MISTA [10] fue la herramienta de MBT seleccionada para este estudio. Esta herramienta fue seleccionada debido a que permite la automatización de los casos de prueba y la ejecución del proceso de MBT. MISTA permite: (i) la generación de casos de prueba abstractos usando diferentes criterios de cobertura, (ii) la concretización de los casos de prueba usando el *model-implementation description* (MIM), y (iii) la ejecución de estos casos de prueba sobre el SUT. Debido a que el SUT seleccionado es una página web, se seleccionó HTML como el lenguaje para concretizar los casos de prueba, *reachability tree* como criterio de cobertura para la generación de los casos de prueba y Selenium [22] como la herramienta para ejecutar los casos de prueba.

5.4.3. Material de capacitación

El material de capacitación tiene las siguientes componentes:

1. Presentación sobre MBT: se crea una presentación para describir conceptos básicos de MBT, el proceso, los beneficios y referencias a otros materiales de la academia. Se envió una semana antes y también se revisó rápidamente (no más de 5 min) durante la sesión de ejecución del caso de estudio.

2. Requerimientos de SUT: se obtienen y revisan los requerimientos con los principales desarrolladores de la aplicación. Basados en la delimitación de APP1, se obtiene un total de 5 requerimientos funcionales, que fueron enviados a los participantes una semana antes, y también se proporcionaron durante la sesión de ejecución del caso de estudio.
3. Material de entrenamiento de MISTA: este material contiene enlaces a la documentación de MISTA, así como un tutorial de cómo usar MISTA y ejemplos prácticos de cómo modelar, generar y ejecutar casos de prueba. La documentación y los tutoriales fueron enviados una semana antes y los ejemplos fueron revisados durante la sesión de ejecución del estudio.

5.4.4. Encuesta de aceptación

Se diseñó una encuesta para evaluar la aceptación del enfoque de MBT por parte de los sujetos. Esta encuesta se dividió en dos secciones: una con preguntas cerradas basadas en el TAM [8] y otra con preguntas abiertas sobre la experiencia que vivieron durante la ejecución del estudio. El fin de estas preguntas fue que ayudaran a explicar los resultados obtenidos. Esta encuesta se encuentra en el Apéndice C.

5.4.5. Notas de observación

Durante las sesiones de ejecución del estudio, se tomaron notas sobre las acciones observadas de los sujetos. El propósito de estas notas fue ayudar a explicar posteriormente algunos de los hallazgos del estudio.

5.5. Métricas de evaluación

Para este caso de estudio se definieron tres variables: la eficiencia, eficacia y el nivel de aceptación. A continuación se definen las métricas utilizadas para medir cada una de estas variables, basadas en Weißleder et al. [28] y Davis [8].

5.5.1. Métricas de eficiencia

- Tiempo de modelado y creación de los casos de prueba: es una medida tomada en minutos desde el momento que cada sujeto empieza a modelar el SUT hasta que termina de generar los casos de prueba abstractos usando MISTA. La recolección de los datos para esta métrica fue realizada por la investigadora para cada uno de los participantes del caso de estudio.
- Tiempo de materialización: es una medida tomada en minutos desde el momento que cada sujeto empieza a agregar los comandos para concretizar los casos de prueba hasta que termina de generar los casos de prueba ejecutables en Selenium. La recolección de los datos para esta métrica fue realizada por la investigadora para cada uno de los participantes del caso de estudio.

5.5.2. Métricas de eficiencia

- Cobertura de código: es una medida porcentual de la cantidad de código fuente probado mediante los casos de prueba generados por la herramienta de MBT. La medida de cobertura se obtuvo por medio de la herramienta OpenCover, que se corrió en el servidor donde estaba la aplicación APP1, y contaba el número de ramas (o decisiones condicionales) ejecutadas por los casos de prueba.
- Cantidad de requerimientos modelados: es una medida de la cantidad de requerimientos que cada participante logró modelar. La cantidad total de requerimientos que fueron

solicitados modelar fueron cinco. Esta métrica fue estimada por la investigadora durante la fase de análisis, usando su conocimiento de la aplicación APP1, y evaluando los modelos creados por cada participante.

5.5.3. Métricas de aceptación

- Percepción de facilidad de uso: es una medida subjetiva del grado de esfuerzo requerido para usar un proceso o método en particular, en una escala Likert de 5 puntos. En nuestro caso, esta métrica mide la percepción de cada participante sobre qué tan fácil es usar la herramienta de MBT empleada: MISTA. Se recolectó mediante la encuesta de aceptación descrita en la sección 5.4.
- Percepción de utilidad: es una medida subjetiva de qué tan efectivo puede ser un proceso o método para lograr un objetivo particular, en una escala Likert de 5 puntos. En nuestro caso, esta métrica mide la percepción de cada participante sobre qué tan útil resulta el enfoque de MBT en su contexto. Se recolectó mediante la encuesta de aceptación.
- Intención de uso: es una medida subjetiva del grado en el que una persona está dispuesta a usar un proceso o método en el futuro, usando una escala Likert de 5 puntos. En nuestro caso, esta métrica mide la percepción de cada participante sobre si usaría el enfoque de MBT en el futuro. Se recolectó mediante la encuesta de aceptación.

5.6. Amenazas a la validez

El análisis de las amenazas a la validez del estudio se describe a continuación.

5.6.1. Validez interna

A pesar de que los participantes no tenían permitido comparar sus modelos, sí podían discutir o hablar entre ellos, lo cual es una amenaza a la validez del estudio, puesto que esta interacción entre los participantes pudo haber afectado el modelo final que cada participante entregó, invalidando la correlación entre la calidad del modelo generado y la experiencia del sujeto o su conocimiento previo del SUT o de MBT.

5.6.2. Validez externa

El estudio no se puede generalizar debido a que es un caso de estudio en pequeña escala. Sin embargo, puede ser de interés para la replicación del proceso y generación de casos similares en la industria.

5.6.3. Validez de conclusión

Debido a que la recolección y el análisis de los datos fue realizado por un solo investigador, esto representa una amenaza a la validez del estudio. Sin embargo, el diseño del caso de estudio fue revisado por dos investigadores más (miembros del Comité Asesor de esta investigación) y validado por un conjunto de profesionales (distinto al conjunto de participantes reportado). Los resultados fueron corroborados en cuanto a consistencia por dos investigadores.

Capítulo 6

Resultados del caso de estudio

Este capítulo presenta los resultados obtenidos del caso de estudio, de acuerdo a las preguntas de investigación planteadas.

6.1. Análisis demográfico de los participantes

En esta sección se describen las características de los seis participantes. Todos los sujetos tienen una edad entre 23 a 33 años de edad, además todos completaron la carrera de ciencias de computación o carreras afines. Cuatro de los sujetos reportaron que se desenvuelven como desarrolladores y otros dos como ingenieros de software utilizando el campo otro de la encuesta B. Todos los participantes trabajan en proyectos de desarrollo y mantenimiento de software además que todos ejecutan tareas de validación y verificación de software.

En cuanto a los datos de experiencia reportados por los sujetos, uno de los participantes reportó tener menos de tres años, cuatro reportaron tener entre 3 y 6 años de experiencia, el último reportó tener más de 10 años de experiencia. Además, 5 sujetos reportaron haber estudiado parcialmente el material de entrenamiento enviado antes de la sesión vía correo electrónico. Uno de los sujetos reportó conocer APP1. Sin embargo, ninguno de los participantes reportó tener conocimiento de MBT o MISTA previo a participar en el estudio.

6.2. Eficiencia y eficacia de los casos de prueba generados con MBT (RQ1)

Aquí presentamos los resultados obtenidos para las métricas de eficacia y eficiencia.

6.2.1. Eficiencia

La tabla 6.1 muestra el tiempo que le tomó a cada participante para modelar el SUT y concretizar los casos de prueba abstractos. Se puede observar que los participantes duraron entre 20 a 60 minutos en la creación del modelo. Además, se logra observar que el participante con conocimiento previo de APP1 (P3) fue capaz de completar el modelo (contemplando todos los requerimientos) más rápido que los otros participantes. Sin embargo P4 y P5 también lograron un tiempo de modelado corto similar a P3, pero ambos participantes no modelaron todos los requerimientos, lo que quiere decir que es un modelo incompleto (como se observa en la tabla 6.2). Por otro lado, el participante P6 fue el que tuvo el mayor tiempo modelando el SUT, esto se puede explicar en que fue el único que incluye validaciones de las entradas de datos en su modelo.

Basado en el resultado de los datos no se encontró ninguna relación entre el tiempo de modelado y estudio previo del material de entrenamiento enviado con anterioridad. Además la experiencia laboral de los participantes no es considerado un factor que afecte el tiempo de modelado debido a que los participantes se enfocaron en modelar los escenarios “positivos”: debido a las restricciones de tiempo los sujetos expresaron que estaban enfocados en entender cómo usar MBT más que en desarrollar un modelo completo.

En la tabla 6.1 también se puede observar el promedio de tiempo de concretización (31

Cuadro 6.1: Tiempo de modelado y concretización (en minutos) por sujeto.

Sujeto	Tiempo de modelado	Tiempo de concretización
P1	33	32
P2	45	29
P3	20	29
P4	23	31
P5	23	37
P6	60	33
Promedio	34	31

minutos) el cual es similar en tiempo promedio de modelado (34 minutos). Sin embargo, de acuerdo con los participantes la concretización de los casos de prueba generados no es difícil, debido a que según ellos esto es un trabajo mecánico que fue comprendido fácilmente gracias al entrenamiento sobre MISTA dado previamente y el ejemplo del modelo utilizado como guía o referencia.

6.2.2. Eficacia

La tabla 6.2 muestra el número de requerimientos modelados, el número de casos de prueba y el porcentaje de cobertura logrado por participante. Usando la herramienta OpenCover [30] se logra analizar que APP1 tiene 732 branches en total. Cada participante logro obtener una cobertura entre 39 % y 59 % comparado con el total de branches que contiene APP1. Estos porcentajes son bajos pero pueden ser explicados con el hecho que la mayoría de los participantes solo se enfocaron en modelar el *happy path*. Además, se logra observar que el participante P4 logra el mayor porcentaje de cobertura (59 %) aunque uno de los requerimientos no fue modelado. Además, en el análisis del modelado creado por P4 se logra observar que el participante agregó varios comandos para la validación de los datos de entrada durante la fase de concretización, logrando así obtener un porcentaje mayor.

Por otro lado el participante P3, el cual tenía conocimiento previo de APP1, modelo todos los requerimientos y fue quien obtuvo el segundo mejor porcentaje de cobertura (54 %). También el participante P6 logra obtener el una cobertura de 54 % quien modeló todos los requerimientos y agregó la validación de datos en su modelo. Luego de analizar los modelos de P3 y P6, se descubre que ambos modelos carecen de los estados necesarios o no agregaron los comandos necesarios en MIM para simular el comportamiento modelado. El análisis de estos datos nos dejó observar interesantes hallazgos: los participantes que modelan todos los requerimientos no van a obtener necesariamente el mayor porcentaje de cobertura comparado con aquellos que modelaron menos requerimientos.

También se puede observar en la tabla 6.2 que los participantes generaron entre 1 y 9

Cuadro 6.2: Cantidad de requerimientos modelados, casos de prueba generados y porcentaje de cobertura obtenida.

Sujeto	Requerimientos	Casos de prueba	Cobertura
P1	5	9	49 %
P2	5	1	39 %
P3	5	4	54 %
P4	4	3	59 %
P5	4	1	44 %
P6	5	8	54 %
Promedio	4.7	4.3	49.8 %

casos de prueba. El participante P2 obtuvo el menor porcentaje de cobertura (39 %) debido a que su modelo solo logró generar un caso de prueba y durante la fase de concretización algunos comandos necesarios fueron agregados incorrectamente (posiblemente debido a errores con el MIM). Luego de analizar la cantidad de casos de prueba en comparación con lo cobertura obtenida, se logra determinar que tener que una gran cantidad de casos de prueba no siempre garantiza un porcentaje de cobertura alto. Por ejemplo es posible que un solo caso de prueba con muchos pasos logre cubrir o ejercitar la misma cantidad de branches que varios casos de prueba cortos.

6.3. Nivel de aceptación de MBT (RQ2)

A continuación se detallan los resultados obtenidos para cada una de las percepciones de aceptación, por parte de los ingenieros de software.

6.3.1. Percepción de utilidad

Los participantes calificaron la utilidad de MBT con un promedio de 4 puntos (de un máximo de 5), según se observa en la figura 6.1. La escala utilizada es de 5 puntos: totalmente

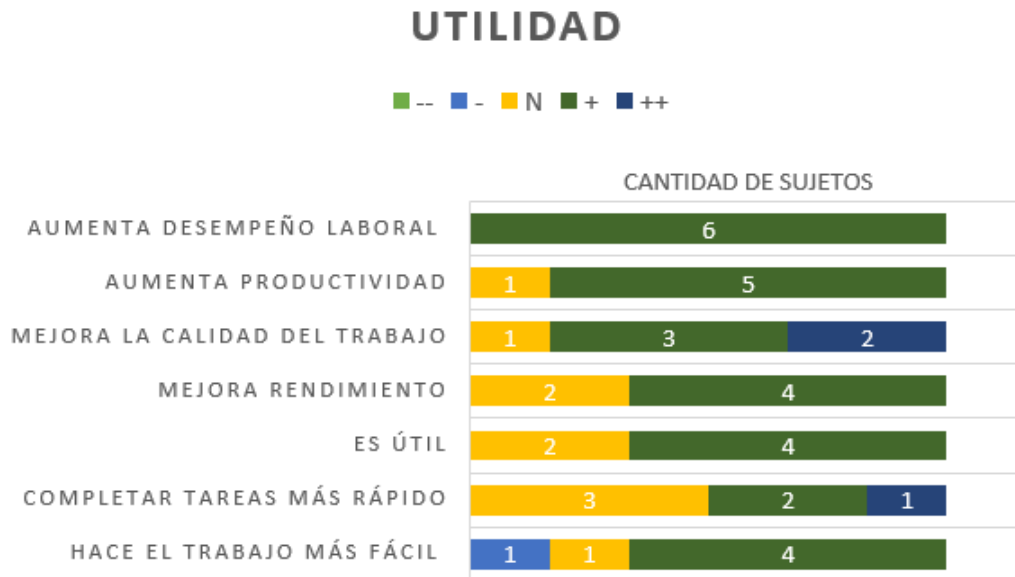


Figura 6.1: Resultados de la percepción de utilidad.

en desacuerdo (--), en desacuerdo (-), neutral (N), de acuerdo (+), totalmente de acuerdo (++). Todos los sujetos reportaron una percepción positiva a afirmación: “Usar MBT con MISTA aumentaría mi desempeño laboral”. Con respecto a las afirmaciones: “Usar MBT con MISTA incrementaría mi productividad” y “Usar MBT con MISTA mejoraría la calidad de trabajo que hago”, cinco sujetos reportaron una percepción positiva mientras que uno fue neutral.

Además, con respecto a las afirmaciones: “Usar MBT con MISTA mejoraría mi rendimiento en el trabajo” y “En general, encuentro MBT y MISTA útiles en mi trabajo”, cuatro sujetos reportaron una percepción positiva mientras que dos fueron neutrales. Por otro lado, tres participantes reportaron una percepción neutral de la afirmación “MBT con MISTA me permitiría completar las tareas más rápidamente”. En este caso las respuestas a las preguntas abiertas revelaron que los participantes creen que MBT es útil pero que MISTA está lejos de ser la herramienta ideal. Además, un participante reportó tener una percepción negativa a la

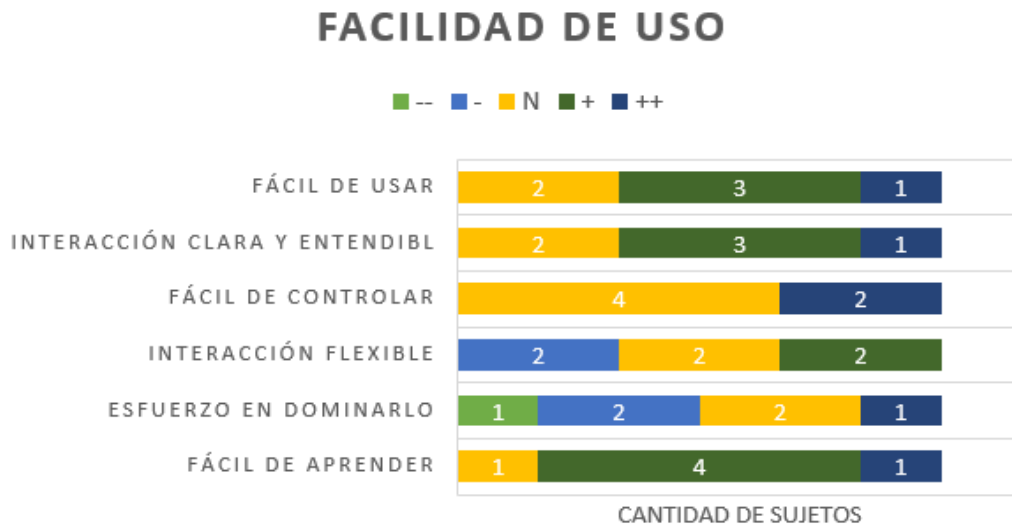


Figura 6.2: Resultados de la percepción de facilidad de uso.

afirmación: “Usar MBT con MISTA haría mi trabajo más fácil”, lo cual creemos que se debe a que el trabajo principal de este participante está relacionado con el diseño y mantenimiento de bases de datos (inferido de las respuestas del participante a las preguntas abiertas), lo que podría explicar por qué el sujeto piensa que MBT no puede hacer su trabajo más fácil.

6.3.2. Percepción de facilidad de uso

Los participantes dieron un puntaje promedio de 3.5 (de un máximo de 5) a las preguntas sobre facilidad de uso de MBT, como se observa en la figura 6.2. La escala utilizada fue de 5 puntos: totalmente en desacuerdo (--), en desacuerdo (-), neutral (N), de acuerdo (+), totalmente de acuerdo (++). En este caso, cuatro sujetos reportaron una percepción positiva y dos neutrales a las siguientes afirmaciones: “En general, encuentro a MBT con MISTA fácil de usar” y “Mi interacción con MBT y MISTA es clara y entendible”. Además, dos participantes reportaron una percepción positiva y cuatro neutrales a la afirmación: “Sería fácil hacer que MBT y MISTA hagan lo que yo quiero”.

Por otro lado, para la afirmación “Puedo interactuar de forma flexible con MBT y MISTA”, se obtuvieron dos percepciones positivas, dos neutrales y dos negativas. Ante la afirmación “Tomaría mucho esfuerzo llegar a dominar el uso de MBT y MISTA”, se obtuvo una percepción positiva, dos neutrales y tres negativas. Fue interesante que todos los participantes excepto uno (con percepción neutral) reportaron una percepción positiva a la afirmación “Aprender a operar MBT con MISTA sería fácil para mí”. Las respuestas a las preguntas abiertas mostraron que los sujetos no recomendaban cambiar nada al proceso de cómo aprendieron a usar MBT, pero sí indicaron que les gustaría tener más tiempo en estudiar la teoría de MBT y ver más ejemplos de modelos.

6.3.3. Intención de uso

Como se observa en la figura 6.3, los participantes dieron un puntaje promedio de 3 (de un máximo de 5) a las preguntas sobre la intención de usar MBT en el futuro. La escala utilizada fue de 5 puntos: totalmente en desacuerdo (--), en desacuerdo (-), neutral (N), de acuerdo (+), totalmente de acuerdo (++). En general, los resultados sugieren que a los sujetos les gustó el enfoque de MBT pero no están satisfechos con la herramienta MISTA. Esto se puede inferir de las respuestas obtenidas a las preguntas de intención de uso, que se presentan a continuación. Cuatro sujetos reportaron una percepción positiva, una neutral y una negativa a la afirmación “Planeo usar el enfoque de MBT en los próximos 6 meses”. Además hubo cuatro percepciones neutrales y dos negativas a la afirmación “Planeo usar MISTA en los próximos 6 meses”.

Las respuestas a las preguntas abiertas revelaron que los participantes desean encontrar y comparar otras herramientas de MBT. También están de acuerdo en que MISTA no es lo suficientemente poderosa para ser utilizada en la industria, pero que fue útil para entender el proceso de MBT.

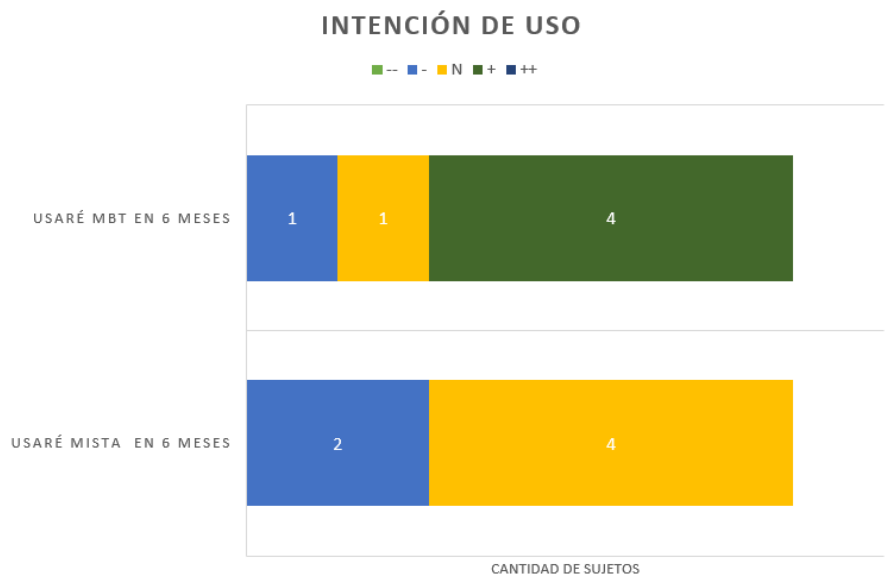


Figura 6.3: Resultados de la percepción de intención de uso.

6.4. Lecciones aprendidas

Un estudio piloto podría eliminar potenciales factores de ruido y desarrollar nuevos instrumentos o refinar los existentes. En nuestro caso, se realizó un estudio piloto para validar el proceso y la recolección de datos definidos en la sección 5.3. Debido a que algunos participantes del estudio piloto expresaron que las tareas de modelado eran complejas y difíciles, se recomienda considerar y controlar la frustración de los participantes en cada etapa del proceso de MBT. Con el fin de controlar eso en nuestro estudio, se tomaron las siguientes acciones:

1. Ofrecer una demostración en vivo durante la sesión de entrenamiento a los sujetos.
2. Especificar claramente los requerimientos del SUT.
3. Tener un “plan de rescate” para cada etapa del proceso de MBT: esencialmente, crear los artefactos que se espera que los sujetos completen, y en caso de que alguno no

logre completar el entregable de una etapa en el tiempo asignado, se le da el artefacto “oráculo” para que el sujeto pueda continuar con las siguientes etapas.

4. No permitir la comparación de los modelos creados por los sujetos durante la sesión.

Otra lección es que la creación de un modelo correcto y completo es crucial dentro del proceso de MBT, puesto que esto afecta directamente la calidad del conjunto de casos de prueba generado. Los participantes del estudio reportaron que el modelado del SUT es la etapa más difícil en todo el proceso de MBT. Además recomendaron hacer más investigación con respecto a las herramientas de MBT existentes para idealmente encontrar mejores herramientas que permitan simplificar o facilitar la creación del modelo. Los sujetos también mencionaron delegar la creación del modelo a otros profesionales que puedan estar más familiarizados con el modelado (en lugar de desarrolladores o ingenieros de software).

En resumen, los beneficios reportados por los sujetos del estudio fueron: que el modelo en sí mismo es una manera fácil de visualizar y entender el SUT, y que el proceso de MBT puede generar casos de prueba más rápido que el proceso manual. Por otro lado, los principales problemas encontrados por los sujetos estuvieron relacionados con la fase de modelado y con la herramienta de MBT usada. En particular, los participantes señalaron las siguientes limitaciones de MISTA: la herramienta no usa la última versión de Selenium, tiene una pobre experiencia de usuario (en especial el MIM). La lección aprendida en este caso es que la elección de la herramienta de MBT es muy importante y puede impactar significativamente los resultados del estudio.

Capítulo 7

Conclusiones

Esta investigación presenta un caso de estudio donde se evaluó el uso de pruebas basadas en modelos en la industria. Mediante una revisión de literatura se obtuvo evidencia del estado actual del uso MBT en la industria, incluyendo características de los estudios realizados así como las métricas utilizadas. Sin embargo, los resultados también evidenciaron la carencia de métodos o procedimientos para evaluar la aceptación de MBT en la industria. Con base en los resultados de esta revisión, se plantea un caso de estudio donde un grupo de ingenieros de software utilicen pruebas basadas en modelos en un proyecto de software, con el fin de poder evaluar la eficiencia, eficacia y el nivel de aceptación.

En este estudio se utilizó MISTA como herramienta de MBT, la cual permite el diseño de los modelos utilizando petri nets de alto nivel, así como la generación de casos de prueba y el proceso de concretización mediante comandos de Selenium. Una aplicación web desarrollada recientemente por la compañía en la que se efectuó este estudio, fue seleccionada como sistema bajo pruebas.

El caso de estudio contó con la participación de seis sujetos, quienes son ingenieros de software que trabajaban para diferentes Departamentos de la compañía seleccionada. Los participantes recibieron primeramente un entrenamiento sobre el proceso de MBT y la herramienta MISTA, y luego llevaron a cabo el proceso completo de MBT sobre el sistema web seleccionado, desde el diseño del modelo hasta la ejecución de los casos de prueba. Durante este proceso se recolectaron diferentes métricas para evaluar la eficiencia y eficacia de MBT, así como su aceptación, por parte de los participantes, como una posible técnica a utilizar en el futuro cercano.

Los resultados sugieren que MBT es una técnica eficiente puesto que se lograron obtener tiempos promedios cortos de modelado y concretización: 34 y 31 minutos, respectivamente. Sin embargo, la mayoría de los sujetos solo modelaron escenarios del flujo principal (*happy*

path) y en muchos casos olvidaron incluir validaciones en la entrada de datos. De hecho, solo un sujeto incluyó este tipo de validación en su modelo, y fue precisamente quien tuvo el tiempo de modelado más alto (60 minutos). Esto quiere decir que los resultados obtenidos en este estudio podrían estar subestimando el tiempo de modelado, puesto que ninguno de los participantes logró crear un modelo completo. Adicionalmente, el conocimiento previo de los sujetos sobre el sistema a modelar puede ser un factor que afecte el tiempo de modelado, debido a que el sujeto que terminó su modelo más rápido fue el único que tenía conocimiento del SUT. En el caso del tiempo de concretización, no se hallaron variaciones importantes entre los sujetos, y esto puede ser atribuido al hecho de que el trabajo de concretizar casos de prueba es un trabajo mecánico simple, donde la experiencia de los sujetos no debería ser un factor que lo afecte.

Los resultados con respecto a la efectividad mostraron que el porcentaje promedio de cobertura fue de un 49.8 %, con solo 4.3 casos de prueba (en promedio) y 4.7 requerimientos modelados por sujeto (de un total de 5 requerimientos). El bajo nivel de cobertura alcanzado posiblemente se deba a que muchos de los requerimientos fueron modelados o concretizados de manera incorrecta. Además, la cantidad de casos de prueba y el porcentaje de cobertura no tienen una relación directa con la calidad del modelo generado.

Con respecto a la aceptación que tiene MBT entre los participantes, los resultados indican que estos tienen una percepción positiva de la utilidad de MBT. Por otro lado, la percepción de facilidad de uso es ligeramente positiva, pues algunos sujetos expresaron sus inquietudes con respecto a las limitaciones que tiene la herramienta MISTA, incluyendo su mala experiencia de usuario y la utilización de versiones obsoletas de diferentes *frameworks*. Por último, la percepción de los sujetos con respecto a la intención de uso sugiere que están interesados en utilizar el enfoque de MBT en el futuro, pero desean buscar y comparar otras herramientas de MBT que puedan satisfacer mejor sus necesidades.

7.1. Trabajo futuro

Dado a que algunos de los participantes del estudio consideraron que el modelado del software bajo prueba era la parte más difícil del proceso de MBT, y muchos de ellos no tenían la suficiente experiencia en modelado, se propone como trabajo futuro replicar el caso de estudio usando sujetos con otro perfil, ya sea ingenieros de calidad o, inclusive, analistas de negocio, quienes podrían tener mejores habilidades de modelado y un mejor conocimiento del software a probar.

Otra línea de posible trabajo futuro sería replicar el estudio con otros tipos de SUT, o bien, usar otras herramientas de MBT que sean más aptas para la industria, con el fin de obtener más evidencias que ayuden a comprobar la validez de los resultados obtenidos en este estudio.

Finalmente, sería interesante hacer un estudio similar con un SUT que posea un conjunto de casos de prueba funcionales (previamente definidos), de manera que se puedan comparar los casos de prueba generados mediante la herramienta de MBT, con los casos de prueba ya existentes, en términos de la eficiencia y eficacia.

Bibliografía

- [1] ANIER, A., VAIN, J., AND TSIPOPOULOS, L. *DTRON: A tool for distributed Model-Based testing of time critical applications*. Estonian Academy Publishers, 2017, p. 75–88.
- [2] B. KITCHENHAM, S. C., AND CHARTERS, S. Guidelines for performing systematic literature reviews. Tech. rep., 2007.
- [3] BETTIGA, D., AND LAMBERTI, L. Exploring the adoption process of personal technologies: A cognitive-affective approach. *The Journal of High Technology Management Research* (2017), 179–187.
- [4] BISWAS, S., KAISER, M. S., AND MAMUN, S. Applying ant colony optimization in software testing to generate prioritized optimal path and test data. *Electrical Engineering and Information Communication Technology (ICEEICT) 2015 International Conference 8* (2015), 1–6.
- [5] BLOM, J., JONSSON, B., AND NYSTRÖM, S. Industrial evaluation of test suite generation strategies for model-based testing. *IEEE International Conference on Software Testing, Verification and Validation Workshops* (2016).
- [6] BOARD, I. S. T. Q. Certified tester foundation level syllabus, 2018.
- [7] DADEAU, F., FOURNERET, E., AND BOUCHELACHEM, A. Temporal property patterns for model-based testing from uml/ocl. *Software Systems Modeling* (2017), 865–888.
- [8] DAVIS, F. D. Perceived usefulness, perceived ease of use, and user acceptance of information technology. 319–340.
- [9] DE OLIVEIRA NETO F., R., T., AND P., M. *Full modification coverage through automatic similarity-based test case selection*. Estonian Academy Publishers, 2016, pp. 124–137.

- [10] DIANXIANG, X. *A Tool for Automated Test Code Generation from High-Level Petri Nets*, 5 ed. Springer-Verlag Berlin Heidelberg, 2011, pp. 308–317.
- [11] ERGUN, B., GEBIZLI, C., AND SOZER, H. Format: A tool for adapting test models based on feature models. *IEEE 41st Annual Computer Software and Applications Conference (2017)*.
- [12] FELLNER, A., KRENN, W., SCHLICK, R., TARRACH, T., AND WEISSENBACHER, G. Model-based, mutation-driven test case generation via heuristic-guided branching search. *15th ACM-IEEE International Conference on Formal Methods and Models for System Design (2017)*, 56–66.
- [13] GEBIZLI, C. S., AND SOZER, H. Model-based software product line testing by coupling feature models with hierarchical markov chain usage models. *IEEE International Conference on Software Quality, Reliability and Security Companion (2016)*.
- [14] GEBIZLI, C. S., AND SÖZER, H. Automated refinement of models for model-based testing using exploratory testing. *Software Quality Journal (2017)*, 979–1005.
- [15] GRAF-BRILL, A., AND HERMANNNS, H. *Model-Based Testing for Asynchronous Systems*. Springer, Cham, 2017, pp. 66–82.
- [16] HERPEL, H. J., LI, J., XIE, J., JOHANSEN, B., KVINNESLAND, K., KRUEGER, S., AND BARRIOS, P. Model based testing of satellite on-board software — an industrial use case. *Aerospace Conference1 (2016)*.
- [17] JOHNSEN, A., LUNDQVIST, K., HANNINEN, K., PETERSSON, P., AND TORELM, M. Experience report: Evaluating fault detection effectiveness and resource efficiency of the architecture quality assurance framework and tool. *IEEE 28th International Symposium on Software Reliability Engineering (2017)*.

- [18] JOVANOVIĆ, M., MESQUIDA, A., MAS, A., AND LALIĆ, B. *Towards the Development of a Sequential Framework for Agile Adoption*. Springer, 2017, pp. 30–42.
- [19] KARHU, K., REPO, T., TAIPALE, O., AND SMOLANDER, K. Empirical observations on software testing automation. *2009 International Conference on Software Testing Verification and Validation (2009)*.
- [20] PETERSEN, K., AND ALI, N. *Identifying strategies for study selection in systematic reviews and maps*. IEEE, 2011, pp. 351–354.
- [21] SCHULZE, C., LINDVALL, M., BJORGVINSSON, S., AND WIEGAND, R. *Model generation to support model-based testing applied on the NASA DAT Web-application - An experience report*. IEEE, 2015.
- [22] SELENIUMHQ. Seleniumhq. <https://www.seleniumhq.org>. Accessed on 2018 – 03- 07.
- [23] SHAFIQUE, M., AND MARCHIONNI, Y. A systematic review of state-based test tools. *International Journal on Software Tools for Technology Transfer* 17 (2013), 59–76.
- [24] UTTING, M., AND LEGEARD, B. *Practical Model-Based Testing: A Tools Approach*. Morgan Kaufmann, 2010.
- [25] UTTING, M., PRETSCHNER, A., AND LEGEARD, B. A taxonomy of model-based testing approaches. *Software Testing, Verification Reliability* (2012), 297–312.
- [26] VILLALOBOS-ARIAS, L., C. QUESADA-LÓPEZ, A. M., AND JENKINS, M. A tertiary study on model-based testing areas, tools and challenges: Preliminary results. *21st Ibero-American Conference on Software Engineering* (2018).
- [27] VILLALOBOS-ARIAS, L., C. QUESADA-LÓPEZ, A. M., AND JENKINS, M. Model-based testing areas, tools and challenges: A tertiary study. *CLEI Electronic Journal* 22, 1 (2019).

- [28] WEISSLEDER, S., AND SCHLINGLOFF, H. An evaluation of model-based testing in embedded applications. *IEEE International Conference on Software Testing, Verification, and Validation* (2014).
- [29] WHITTAKER, J. A. *What is software testing? And why is it so hard?* IEEE, 2000, pp. 70 – 79.
- [30] W.MASRI, AND F.A.ZARAKET. *Chapter Four - Coverage-Based Software Testing: Beyond Basic Test Requirements*. 2016, ch. 4, pp. 79–142.
- [31] WOHLIN, C., RUNESON, P., HÖST, M., OHLSSON, M., REGNELL, B., AND WESSLÉN, A. *Experimentation in software engineering*. Springer Science and Business Media, 2012.
- [32] ZHANG, M., ALI, S., YUE, T., NORGRÉN, R., AND OKARIZ, O. Uncertainty-wise cyber-physical system test modeling. *Software Systems Modeling* 18 (2017), 1379–1418.

Apéndice A

Artículo de investigación

Este trabajo de investigación derivó en un artículo científico, que fue publicado en la conferencia *International Conference on Information Technology & Systems (ICITS' 19)*. Las memorias de esta conferencia las publica la casa editorial Springer. La referencia completa del artículo es la siguiente:

Vásquez, R.O., Quesada-López, C., Martínez, A. (2019) Evaluating Model-Based Testing in an Industrial Project: An Experience Report. In: Rocha Á., Ferrás C., Paredes M. (eds) Information Technology and Systems. ICITS 2019. Advances in Intelligent Systems and Computing, vol 918. Springer, Cham.

Enlace a la publicación en la biblioteca digital Springer Link: https://doi.org/10.1007/978-3-030-11890-7_29

El artículo completo se incluye a continuación.



Evaluating Model-Based Testing in an Industrial Project: An Experience Report

Rebeca Obando Vásquez^(*), Christian Quesada-López,
and Alexandra Martínez

Universidad de Costa Rica, San José, Costa Rica
{yohana.obando,cristian.quesadalopez,alexandra.martinez}@ucr.ac.cr

Abstract. Model-based testing (MBT) is an approach that automates the design and generation of test cases based on a model that represents the system under test. MBT can reduce the cost of software testing and improve the quality of systems in the industry. The goal of this study is to evaluate the use of MBT in an industrial project with the purpose of analyzing its efficiency, efficacy and acceptance by software engineers. A case study was conducted where six software engineers modeled one module of a system, and then generated and executed the test cases using an MBT tool. Our results show that participants were able to model at least four functional requirements each, in a period of 20 to 60 min, reaching a code coverage between 39% and 59% of the system module. We discussed relevant findings about the completeness of the models and common mistakes made during the modeling and concretization phases. Regarding the acceptance of MBT by participants, our results suggest that while they saw value in the MBT approach, they were not satisfied with the tool used (MISTA), because it did not support key industry needs.

Keywords: Model-based testing · Automation · Industry · Experience report

1 Introduction

Assuring the quality of a delivered product is a key part of industrial software development. Different types of testing techniques are used for product quality assurance, including functional tests, whose goal is to detect deviations (non-conformities) of the system under test (SUT) from its functional specifications or requirements [1]. Model-based testing (MBT) is a software testing approach that automates the creation of test cases based on a model of the system under test [2], and is normally used for functional testing. MBT has five phases: modeling the SUT from the requirements, choosing a test selection criteria, generating abstract test cases, concretizing the test cases, and executing these test cases [3,4].

Many MBT approaches lack empirical evaluations or have not been transferred to industrial contexts [4]. This was our motivation for conducting an

industrial case study, where the use of MBT by a group of software engineers on a real system was empirically evaluated. In this study we used MISTA, a model-based testing tool that provides support for all five MBT phases [3]. Our work can shed some light on how MBT can be leveraged by the industry providing some evidence about the efficiency, efficacy and acceptance properties of MBT from the perspective of industry practitioners.

The rest of the paper is organized as follows. Section 2 summarizes relevant related work. Section 3 describes the design of the study. Section 4 presents our results, discussion and lessons learned. Section 5 outlines our conclusions and future work.

2 Related Work

A few studies that apply MBT to specific niches were found in the literature [5–7]. In particular, Weißleder and Schlingloff [5] describe the process to implement MBT in embedded applications using a new tool. Their study compares the efficiency and effort of generating test cases with MBT versus manually. Similar measures were reported by Ergun et al. [6] and Herpel et al. [7], where the time to model and the number of generated test cases were used to show the benefits of applying MBT. Code and branch coverage have also been used as metrics of efficiency when comparing MBT to manual testing [7].

Weißleder and Schlingloff [5] held a workshop with requirements experts and software engineers, in order to get feedback about the advantages of MBT and the need for training on model creation and on the use of MBT. These issues were also discussed by Ergun et al. [6], Gebizli et al. [8] and Schulze et al. [9]. Also, Herpel et al. [7] mentioned that a good understanding of the SUT's requirements was important to obtain a complete and correct model of the SUT. Moreover, Schulze et al. [9] suggest that the design of the model is a factor associated to the adoption process, due the time and training/experience required to complete the model.

According to Bettiga and Lamberti [10], the technology adoption process has two parts: the cognitive and the affective one. The cognitive part can be measured using techniques like the Technology Acceptance Model (TAM) [11], which assesses the adoption of new technology from three perceptions: usefulness, ease of use, and intention to use. The affective part can be measured by sensors or other mechanisms that are able to detect the physical and sensorial reaction (feelings) of users while interacting with the system [10].

Our study combines some of the efficiency, efficacy, and acceptance metrics from previous studies [5–8, 11], and defines a process for using MBT on a software project of the industry, which could be used in similar contexts to help promote industry adoption of MBT.

3 Case of Study Design

This case study was conducted according to Wohlin's protocol [12] and Runeson's guidelines [13] for case studies. The goal of this study was to evaluate the use of

MBT in an industrial context in terms of its efficiency, efficacy, and acceptance by the software engineers. To pursue this goal, we defined the following research questions:

RQ1: What is the efficiency and efficacy of the test cases generated using MBT?

RQ2: What level of acceptance does MBT have among software engineers?

3.1 Context

This study was conducted in a Fortune 100 company located in *country name*, which has several development teams dedicated to support both internal and external needs of the company.

Subjects. The participants were software engineers from different departments who work on designing and developing new solutions for internal projects. The subjects were selected based on convenience, according to their priorities, available time and interest in learning a new testing approach. The company allows their employees to devote some work time to learn about new technologies that can be useful in their daily work.

SUT. The object of the study was a web application that manages requests for internal use, which was released to production one year before the study and was under maintenance at the time of the study. We will refer to this application as APP1 for confidentiality reasons. APP1 has four modules but due to time restrictions of the subjects, only one of its modules was used as SUT in this study.

3.2 Process and Data Collection

The process followed in the study is shown in Fig. 1: the first step was the creation and calibration of the instruments, described in Subsect. 3.3. Steps 2 to 7 were followed by the participants. Participants received training on MBT and MISTA, completed a demographic survey, modeled the SUT, generated the test cases and ran them. The steps 8 to 10 were performed by the researchers of the study, who collected or computed the metrics and did the corresponding data analysis and interpretation.

3.3 Instruments

Next we describe each of the instruments used in this study.

- **Demographic survey:** it gathered information about the subjects: age, background, work experience, role, projects, and prior knowledge of the SUT.
- **MBT tool:** MISTA [14] was the MBT tool selected for this study. It allows the automation of the test case generation and execution processes using high level petri nets. MISTA allows the generation of abstract test cases using different coverage criteria, the concretization of test cases using its

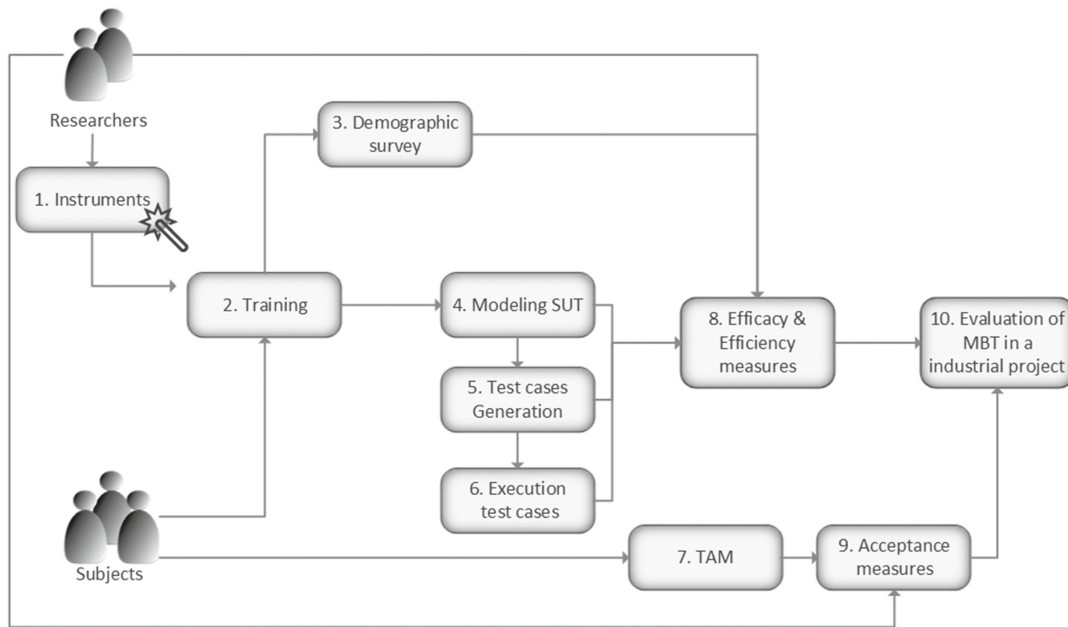


Fig. 1. Process followed in the study.

model-implementation description (MIM), and the execution of these test cases on the SUT. Given that the SUT is a web application, we chose HTML as language, reachability tree as the coverage criteria for test generation, and Selenium [15] as the tool for execution of functional test cases.

- **Training material:** it includes a presentation on MBT, on how to use MISTA, and a practical example of a model created with MISTA, together with links to MISTA tutorials.
- **Acceptance survey:** a survey that aims to assess the acceptance of MBT by the subjects. It has two sections: one with closed questions based on TAM [11] and another with open questions about the experience they went through, and will help explain the results obtained from the first section.
- **Observation notes:** during the sessions where the researchers interacted with the subjects, one researcher took notes about interaction and actions of the subjects.

3.4 Evaluation Criteria

The metrics used to measure the efficacy, efficiency and acceptance of MBT are based on [5, 11] and are described next.

- **Efficiency metrics:** we measured the modeling time (time it took each subject to create the model of the SUT) as well as the concretization time (time each subject took to convert –concretize– the abstract test cases generated from the model into executable test cases that can be run in Selenium). Both times were taken in minutes.

- **Efficacy metrics:** we measured the coverage percentage (percent of decision branches in the control structure of the source code that were executed while running the test cases created by the MBT tool [16]) and the number of requirements modeled (quantity of requirements each subject was able to model, from a total of five). For code coverage, we used a tool called OpenCover, that runs on the APP1 server and counts the number of visited branches. The number of requirements modeled was estimated by the researchers of the study, using their expertise in APP1. Both measures were in an ordinal scale.
- **Acceptance metrics:** we measured the perceived usefulness (degree to which the subjects believe that MBT with MISTA will increase their productivity or efficiency), perceived ease of use (degree to which the subjects believe that MBT with MISTA would be free of effort), and intention to use (degree to which the subjects think they will use MBT with MISTA in the future), according to TAM [11]. All these metrics are ordinal on a 5-point Likert scale.

4 Results and Analysis

4.1 Demographic Analysis of Participants

Here we describe the characteristics of our six subjects. All subjects were between 23 and 33 years old, and had completed a career in computer science or related areas. For subjects reported having a role of developers while two reported being software engineers. All subjects worked on software development and maintenance projects along with software validation and verification tasks. In terms of their general work experience, one subject had less than 3 years, four had between 3 and 6 years, and one had more than 10 years. Additionally, all subjects had 6 or less years working for their organization. Regarding their experience in modeling, four subjects reported to have less than 3 years of experience, one reported to have 6 to 10 years, and one reported to have more than 10 years. Moreover, five of the subjects stated that they had partially studied the material sent by email prior to the training session. One subject reported to know APP1. Finally, none of the subjects had knowledge of MBT or MISTA.

4.2 RQ1: Efficiency and Efficacy of Generated Test Cases

Efficiency. Table 1 shows the time it took each participant to model the SUT and concretize the abstract test cases. We observe that participants spent between 20 and 60 min creating the model. We found that the participant with prior knowledge of APP1 (P3) was able to complete the model (with all its requirements) faster. Participants P4 and P5 also achieved short modeling times, but neither of them finished modeling all the requirements (as seen in Table 2). The participant who took the longest time to model the SUT (P6), was the only one who included validation of entry data as part of his model. We found no

Table 1. Modeling and concretization time (in minutes) per subject.

Subject	Modeling time	Concretization time
P1	33	32
P2	45	29
P3	20	29
P4	23	31
P5	23	37
P6	60	33
Avg	34	31

Table 2. Number of requirements modeled, test cases generated, and coverage percentage achieved, per subject.

Subject	Requirements	Test cases	Coverage
P1	5	9	49%
P2	5	1	39%
P3	5	4	54%
P4	4	3	59%
P5	4	1	44%
P6	5	8	54%
Avg	4.7	4.3	49.8%

pattern relating modeling time to prior study of the material. On the other hand, participants' work experience was not considered a factor that affected the modeling time since they only modeled 'positive' scenarios: the subjects expressed that they focused on understanding how to use MBT rather than on developing a complete model, due to the time restrictions. From Table 1 we can also see that the average concretization time (31 min) is similar to the average modeling time (34 min). According to the participants, concretizing the generated test cases was not difficult (it was mere mechanical work) thanks to the previous training on MISTA and to the example model used as guideline.

Efficacy. Table 2 shows the number of requirements modeled, the number of test cases generated, and the coverage achieved, per participant. Using OpenCover, we found that APP1 had 732 branches in total. Participants achieved a coverage ranging between 39% and 59%. Such low percentages could be explained by the fact that most participants only focused on the 'happy path' when modeling. Participant P4 obtained the highest coverage percentage (59%) despite missing one requirement in his model. An analysis of P4's model shows that commands were added during the concretization stage to validate data entry, hence the higher coverage percentage. Participant P3, who had knowledge about APP1,

modeled all the requirements and attained the second highest percentage coverage (54%). Likewise, participant P6 obtained a coverage percentage of 54% and modeled all the requirements, even adding data entry validations to the model. After analyzing P3's and P6's models, we discovered that they either did not contain a necessary state or did not add the corresponding commands in the MIM to simulate the behavior. This led us to an interesting finding: participants who modeled all the requirements did not necessarily achieved higher coverage than those who modeled less requirements.

We can also observe from Table 2 that participants generated between 1 and 9 test cases. Participant P2 obtained the lowest coverage percentage (39%) because his model only generated one test case and the corresponding concretized test case had some incorrect commands (possibly due to errors with the MIM). After analyzing of the number of test cases versus the coverage percentage achieved, we found that a higher quantity of test cases does not always guarantee higher coverage. For example, it is possible that a single test case with many steps covers (exercises) as many branches as several (shorter) test cases.

4.3 RQ2: Level of Acceptance of MBT Among Participants

Perceived Usefulness. Participants deemed MBT useful, with an average rating of 4 (out of 5) across the 'usefulness' questions. All subjects reported a positive perception of "Using MBT with MISTA would improve my job performance". Regarding the "Using MBT with MISTA would increase my productivity" and "Using MBT with MISTA would improve the quality of the work I do" statements, five subjects reported a positive perception and one was neutral. Besides, for the "Using MBT with MISTA would enhance my effectiveness on the job" and "Overall, I would find MBT with MISTA useful in my job" statements, four subjects reported a positive perception and two were neutral. On the other hand, three participants reported a neutral perception of the "MBT with MISTA would enable me to accomplish tasks more quickly" statement. In this case, responses to open questions revealed that participants believed MBT is a useful approach but MISTA is far from being an ideal tool. Furthermore, one participant reported a negative perception of "Using MBT with MISTA would make it easier to do my job", but we believe this is because his job was mainly related to database design and maintenance (deducted from his responses to open questions), which explains why he thinks that MBT would not make his job easier.

Perceived Ease of Use. Participants gave MBT an average rating of 3.5 (out of 5) across the 'ease of use' questions. Four subjects reported a positive perception and two were neutral on the following two statements: "Overall, I would find it MBT with MISTA easy to use" and "My interaction with MBT with MISTA would be clear and understandable". Also, two participants reported a positive perception of "I would find it easy to get MBT with MISTA to do what I want it to do" and four were neutral. For the claim "I would find MBT with MISTA

flexible to interact with”, we had two positive, two neutral, and two negative responses. On the other hand, the “I would find it takes a lot of effort to become skillful at using MBT with MISTA” statement, had one positive perception, two neutral and three negative perceptions. Interestingly, all but one (neutral) of the subjects were positive towards the statement “Learning to operate MBT with MISTA would be easy for me”. Answers to open questions revealed that the subjects do not recommend any change in the process used to learn MBT, but would like to spend more time studying the theory of MBT and seeing more examples of models.

Intention to Use. Participants gave MBT an average rating of 3 (out of 5) across the ‘intention to use’ questions. Overall, our results show that the subjects liked the MBT approach but were not satisfied with the MBT tool used: MISTA. This is evidenced by the four positive, one neutral, and one negative responses to the statement “I plan to use an MBT approach in the next 6 months”. And also by the four neutral and two negative responses to the “I plan to use MISTA in the next 6 months” statement. Open questions revealed that the subjects wanted to find and compare other MBT tools. Also, they agreed that MISTA was not powerful enough to use as an industry tool, but was useful for understanding the MBT process.

4.4 Lessons Learned

A pilot study can be helpful to remove potential noise factors and develop new instruments or refine existing ones. In our case, we performed a pilot study to validate the process and data collection defined in Sect. 3.2. Given that some participants from the pilot expressed that the modeling tasks were complex and difficult, we recommend to account for and control subjects’ frustration at each stage of the MBT process. To control for that in this study, we (*i*) conducted a demo during the training session, (*ii*) clearly specified the SUT’s requirements, (*iii*) had a *rescue* plan for each stage of the MBT process (essentially, we produced the artifact subjects were expected to complete, and if they did not finish it, we gave them ours so that they could continue the process), and (*iv*) did not allow model comparisons among subjects.

Creating a correct and complete the model is a crucial step in the MBT process, since it directly affects the quality of the generated test suite. The subjects reported modeling the SUT as the hardest stage of the MBT process. They recommended doing more research on MBT tools (to hopefully find better tools that facilitate model creation), and to delegate the creation of the model to other roles (instead of software developers) that are more familiar with modeling.

In summary, the two benefits of MBT reported by the subjects were: that the model itself is an easy way to visualize and understand the SUT, and that the MBT process could generate test cases faster than with the manual process. On the down side, the main problems indicated by the subjects were related to the modeling stage and the MBT tool. Particularly, they pointed out the following

shortcomings of MISTA: it does not use the latest version of Selenium, MIM is not user-friendly, and MISTA had a poor user experience. The lesson learned here is that the choice of MBT tool is of utmost importance and can have a large impact on the results of the study.

5 Conclusions and Future Work

This paper reports on an industrial experience where a group of software engineers used model-based testing in a software project, with the aim of evaluating its efficiency, efficacy and acceptance. In this study, we used MISTA as the MBT tool, which allows the design of a model using high level petri nets, as well as the generation of test cases and their concretization with Selenium. A web application recently developed by the company was selected as the system under test. There were six participants in the study, who were software engineers from different departments within the company. They first received training on MBT and MISTA, and then carried out the entire MBT process over the SUT, from model design to test case execution. Metrics were collected during this process to assess the efficiency and efficacy of MBT as a testing approach, and its acceptance by the participants as a plausible technique to use in the near future.

Our results suggest that MBT is an efficient technique given that both average modeling time and average concretization time were short: 34 and 31 min, respectively. However, we noted that most subjects only modeled the ‘happy path’ scenarios, forgetting to include validation of data entry. In fact, the only subject who included data entry validation in his model took the longest time to model (60 min). This means that our results could be underestimating the modeling time, and that the modeling time depends on the scope of what is being modeled. Also, modeling time seems to be affected by subject’s prior knowledge of the system being modeled, since the subject who finished modeling faster was the one with knowledge of the SUT. With respect to the concretization time, we found that there was not much variance between subjects, and this could be attributed to the fact that concretizing test cases is simple mechanical work that does not require a lot of effort, and therefore should not be significantly affected by the subject’s experience.

The results also suggest that MBT is relatively effective given its average coverage of 49.8% with just 4.3 test cases (on average), for an average number of 4.7 (out of 5) requirements modeled. The relatively low coverage percentage attained (between 39% to 59%) could be explained by the fact that many requirements were incorrectly modeled or concretized. Furthermore, the quantity of test cases and the attained coverage do not directly correspond to the quality of the model generated.

Regarding the acceptance of MBT by the participants of the study, our results show that participants had a positive perception of the usefulness of MBT. Their perception of ‘ease of use’ was barely positive, with some subjects expressing concerns about the limitations of MISTA, including its poor user experience and outdatedness. Lastly, the subjects perception of ‘intention of use’ suggest that

while they were interested in using MBT in the future, they wanted to search and compare other MBT tools, since MISTA did not meet their industry needs.

Given that some subjects considered modeling the SUT as the most difficult part of the MBT process, and some did not have much experience in modeling, we think it would be interesting, as future work, to replicate this study using subjects from different roles such as quality or test engineers, or even business analysts, who might have more developed modeling skills and a broader knowledge of the SUT.

References

1. Utting, M., Pretschner, A., Legeard, B.: A taxonomy of model-based testing approaches. *Softw. Test. Verification Reliab.* **22**, 297–312 (2012)
2. Biswas, S., Kaiser, M.S., Mamun, S.: Applying ant colony optimization in software testing to generate prioritized optimal path and test data. In: 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), pp. 1–6 (2015)
3. Utting, M., Legeard, B.: *Practical Model-Based Testing: A Tools Approach* (2006)
4. Villalobos-Arias, L., Quesada-López, C., Martínez, A.: A tertiary study on model-based testing areas, tools and challenges: preliminary results (2018)
5. Weißleder, S., Schlingloff, H.: An evaluation of model-based testing in embedded applications. In: IEEE International Conference on Software Testing, Verification, and Validation (2014)
6. Ergun, B., Gebizli, C., Sozer, H.: Format: a tool for adapting test models based on feature models. In: IEEE 41st Annual Computer Software and Applications Conference (2017)
7. Herpel, H.J., et al.: Model based testing of satellite on-board software — an industrial use case. In: Aerospace Conference (2016)
8. Gebizli, C.S., Sözer, H.: Automated refinement of models for model-based testing using exploratory testing. *Softw. Qual. J.* **25**(3), 979–1005 (2017)
9. Schulze, C., et al.: Model generation to support model-based testing applied on the NASA DAT web-application - an experience report (2015)
10. Bettiga, D., Lamberti, L.: Exploring the adoption process of personal technologies: a cognitive-affective approach. *J. High Technol. Manag. Res.* **28**(2), 179–187 (2017)
11. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q.* **13**, 319–340 (1989)
12. Wohlin, C., et al.: *Experimentation in Software Engineering* (2012)
13. Runeson, P., et al.: *Case Study Research in Software Engineering: Guidelines and Examples* (2012)
14. Xu, D.: A tool for automated test code generation from high-level petri nets (2011)
15. SeleniumHQ: SeleniumHQ. <https://www.seleniumhq.org>. Accessed 07 Mar 2018
16. Masri, W., Zaraket, F.A.: Coverage-based software testing: beyond basic test requirements. In: *Advances in Computers*, chap. 4, pp. 79–142 (2016)

Apéndice B

Encuesta demográfica

Encuesta inicial

Esta encuesta recolecta información sobre cada participante y su conocimiento previo al experimento.

***Obligatorio**

1. Nombre del participante *

Información general

2. Edad *

Formación académica

3. ¿Cuál es su carrera base? *

Marca solo un óvalo.

Computación

Otro: _____

4. ¿Dónde obtuvo su carrera base? *

Marca solo un óvalo.

Universidad pública

Universidad privada

Experiencia profesional

5. ¿Cuántos años tiene usted de laborar en su organización de desarrollo de software? *

Marca solo un óvalo.

Menos de 3 años

De 3 a 6 años

De 6 a 10 años

Más de 10 años

6. ¿Cuántos años tiene de experiencia en el desarrollo de software? *

Marca solo un óvalo.

- Menos de 3 años
- De 3 a 6 años
- De 6 a 10 años
- Más de 10 años

7. ¿Cuál es su rol principal en la organización de desarrollo de software? *

Marca solo un óvalo.

- Gerente General
- Gerente de Área de Proyectos (Director de proyectos)
- Gerente de Área de Desarrollo
- Gerente de Área de Calidad
- Administrador de Proyectos
- Líder técnico
- Analista de sistemas
- Desarrollador de software
- Ingeniero de Calidad
- Otro: _____

8. ¿Qué tipo de proyectos ejecuta su organización de desarrollo de software?

Marca solo un óvalo.

- Desarrollo y mantenimiento de software
- Validación y verificación (pruebas)
- Ambas

9. ¿Cuántos años de operación tiene su organización de desarrollo de software en Costa Rica? *

Marca solo un óvalo.

- Menos de 3 años
- De 3 a 6 años
- De 6 a 10 años
- Más de 10 años

10. ¿Cuántos años de experiencia tiene en el modelado de software? *

Marca solo un óvalo.

- Menos de 3 años
- De 3 a 6 años
- De 6 a 10 años
- Más de 10 años

11. ¿Tiene conocimiento sobre MBT (pruebas basadas en modelos)? *

Marca solo un óvalo.

Sí

No

12. Describa qué conocimiento tiene sobre MBT

13. ¿Conoce MISTA, herramienta que soporta el enfoque de MBT? *

Marca solo un óvalo.

Sí

No

14. Describa qué conocimiento tiene sobre MISTA

15. ¿Estudió el material brindado sobre MBT y MISTA? *

Material enviado en un correo la semana anterior

Marca solo un óvalo.

Sí

Parcialmente

No

16. ¿Tiene conocimiento sobre la aplicación HVPIP que se va a modelar? *

Marca solo un óvalo.

Sí

No

Apéndice C

Encuesta de aceptación

Level of adoption of MBT using MISTA

Instructions

The below survey contains two sections. The first section has the purpose of evaluate the usefulness and perception of ease of use of MBT with MISTA. On section two contains open questions regarding to usage of MBT with MISTA.

General information

Name:

Start time of modeling:

Start time of concretization:

Finish time:

Section I. Perception of usefulness and ease of use

The table below shows a series of assertions to evaluate the participant's perception of usefulness, ease of use, and intention to use. Please mark with an **X** the cell that is closer to your opinion.

Question	1 Strongly Disagree	2 Disagree	3 Neutral	4 Agree	5 Strongly Agree
1. Using MBT with MISTA would improve the quality of the work I do					
2. MBT with MISTA would enable me to accomplish tasks more quickly					
3. Using MBT with MISTA would increase my productivity					
4. Using MBT with MISTA would improve my job performance					
5. Using MBT with MISTA would enhance my effectiveness on the job					
6. Using MBT with MISTA would make it easier to do my job					
7. Overall, I would find MBT with MISTA useful in my job					
8. Learning to operate MBT with MISTA would be easy for me					
9. I would find it easy to get MBT with MISTA to do what I want it to do					
10. I would find MBT with MISTA flexible to interact with					
11. My interaction with MBT with MISTA would be clear and understandable					
12. I would find it takes a lot of effort to become skillful at using MBT with MISTA					
13. Overall, I would find it MBT with MISTA easy to use					

14. I plan to use MISTA in the next 6 months					
15. I plan to use an MBT approach in the next 6 months					

Section II. Open Questions.

Please read carefully each question, respond according to your opinion and explain your answers.

1. How was your experience using MBT with MISTA on an industry application? Explain your answer

2. How do you believe MBT with MISTA can be adopted in your work? Explain your answer

3. What are the advantages of using MBT with MISTA according to your experience? Explain your answer

4. What are the disadvantages of using MBT with MISTA according to your experience? Explain your answer

5. What were the difficulties that you found during the application of MBT using MISTA? Explain your answer

6. Would you recommend MBT with MISTA in the future? Or would you do some modification in the process? Explain your answer

7. Would you make any changes to the learning process of MBT with MISTA used on this experiment? Explain your answer