УДК 004.415.532.3

ДОСЛІДЖЕННЯ МЕТОДУ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ «БІЛИЙ ЯЩИК»

Новикова К.В., Люта М.В.
Київський національний університет технологій і дизайну
Розломій І.О.
Черкаський національний університет імені Богдана Хмельницького

Стаття присвячена обґрунтуванню методу тестування програмного забезпечення «Білий ящик». Проаналізовані основні стратегії, стадії та етапи методу тестування «Білий ящик». Були представлені особливості структурного тестування. В результаті була розроблена функціональна модель процесу тестування методом «Білий ящик» відповідно до нотації IDEF0. Складена детальна характеристика кожного етапу тестування методом «Білий ящик».
Ключові слова: тестування програмного забезпечення, «Білий ящик», структурне тестування, функціональна модель, IDEF0.

Постановка проблеми. В даний час створення програмного забезпечення не можливо уявити без процесу тестування, але, на жаль, йому не завжди приділяють належну увагу. Це пов'язано, в основному, з економічними причинами. Виробники прагнучи зменшити витрати на новий продукт часто просто не виділяють належні кошти і час на проведення повноцінного процесу. У більшості випадків проводиться неповноцінне, так зване, «інтуїтивне» тестування у вигляді створення і розробки проектів в тестованій системі.

Мета будь-якого методу тестування безпеки програмного забезпечення полягає в забезпеченні надійності системи в умовах шкідливих атак і програмних дефектів і збоїв. На виробництві вразливість програмного забезпечення (ПЗ) може призвести до фінансових і тимчасових втрат через зупинку або некоректної роботи сис-

теми, тому завчасне її виявлення є актуальним завданням для всіх підприємств.

В даний час існують різноманітні методи виявлення дефектів, помилок і вразливостей програмного забезпечення. Вони мають певні переваги і недоліки, мають різні сфери застосування, що впливає на ефективність і кінцевий результат верифікації.

Аналіз останніх досліджень та публікацій. Останнім часом спостерігається неабиякий інтерес до дослідження різноманітних стратегій тестування програмного забезпечення. Значний внесок в області тестування програмного забезпечення внесли такі науковці: К.В. Рубинов, О.І. Бедердинова, Л.А. Іванова, С.В. Бирюков та інші. Проте, все ще залишається ряд невирішених питань, які дозволять оптимізувати процес тестування програмного забезпечення.

Мета статті. Головною метою статті є аналіз та дослідження методу тестування ПЗ «Білий ящик». Метою дослідження є також розгляд стратегій, стадій та етапів методу тестування «Білий ящик» та розробка функціональної моделі процесу тестування методом «Білий ящик» відповідно до нотації IDEF0.

Виклад основного матеріалу. Метод тестування «Білий ящик» є найбільш поширеним. Його використання призводить до зниження ризиків успішних шкідливих атак, підвищуючи загальну безпеку системи.

«Білий ящик» – це техніка тестування, яка дозволяє перевірити внутрішню структуру програми, її логіку і коректність роботи.

Техніка тестування «Білого ящика» передбачає тестування програмного забезпечення, аналізуючи логіку роботи програми для отримання тестових даних.

У цього методу немає мети виявлення синтаксичних помилок, так як дефекти такого роду зазвичай виявляє компілятор. Методи «Білого ящика» спрямовані на локалізацію помилок, які складніше виявити, знайти і зафіксувати. З їх допомогою можна виявити логічні помилки і перевірити ступінь покриття тестами.

Для застосування методу «Білий ящик» потрібна наявність повної інформації про досліджуване ПЗ (вихідний код, інформація про результати проведених тестувань), що дозволяє провести його повний аналіз на предмет дефектів, помилок і вразливостей. Метод тестування може бути реалізований динамічним і статичним способами.

При статичному аналізі досліджуються вихідні коди компонентів і документовані можливості ПЗ, а при динамічному здійснюється перевірка поведінки ПЗ в реальних умовах із застосуванням спеціалізованого програмного забезпечення (відладчики, профілювальникі) [1].

Стратегія «Білого ящика» включає в себе наступні методи тестування:

1. Покриття операторів. Критерії покриття операторів передбачає виконання кожного оператора програми щонайменше один раз.

2. Покриття рішень. Відповідно до цього критерію необхідно скласти таке число тестів, при яких кожна умова в програмі прийме як справжнє, так і хибне значення.

3. Покриття умов. Записується число тестів достатню для того, щоб всі можливі шляхи виконання програми були пройдені принаймні один раз.

4. Покриття рішень/умов. Відповідно до цього критерію необхідно скласти тести так, щоб результати кожної умови виконувалися хоча б один раз, результати кожного рішення так само виконувалися хоча б один раз, і кожен оператор повинен бути виконаний хоча б один раз.

5. Комбінаторне покриття умов. Цей критерій потребує, щоб всі можливі комбінації результатів умов в кожному рішенні, а також кожен оператор виконався принаймні один раз [2].

Серед особливостей структурного тестування можна виділити наступні:

1. Мінімальна вартість усунення дефекту. Локалізація помилки всередині конкретного програмного модуля не призведе до її міграції в інші частини програми і не потребує витрат на її пошук і усунення.

2. Гарантується, що тести, побудовані на базі вихідного коду, забезпечать його повне покриття.

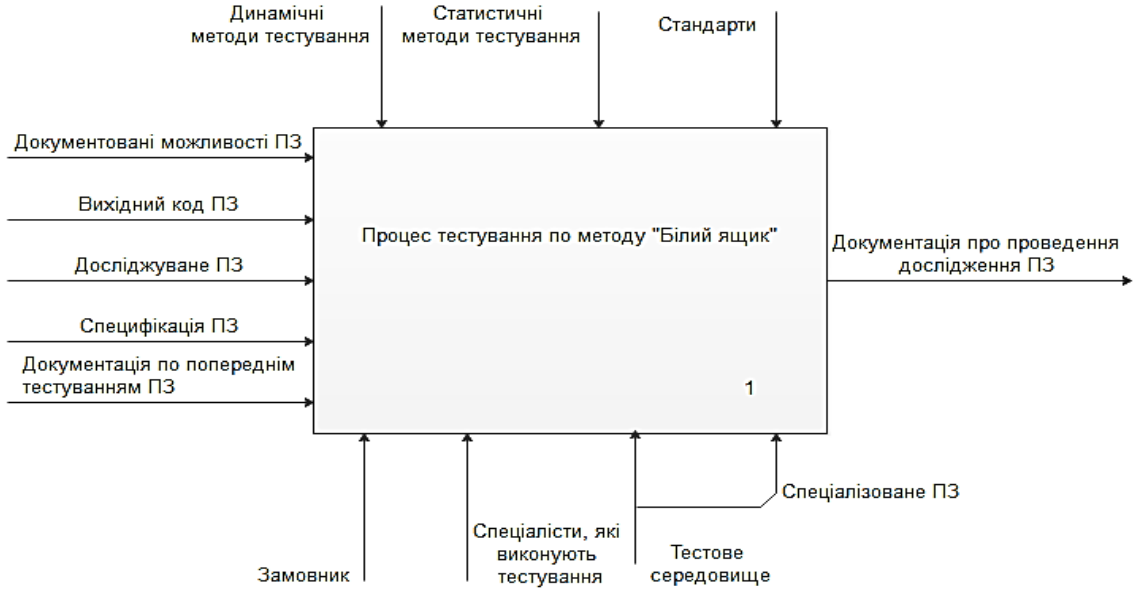

Рис. 1. Контекстна діаграма процесу тестування програмного забезпечення методом «Білий ящик»

3. Можливість відстеження потоку управління і цілісності даних в ході виконання.

4. Тести залежать від вихідного коду, і тестувальник змушений постійно модифікувати їх, слідуючи змін в програмі.

5. Спеціаліст з тестування зобов'язаний чітко розбиратися в перевіряємому коді, що призводить до збільшення витрат ресурсів на тестування.

6. Складність тестування системи в цілому [3].

Функціональна модель процесу тестування методом «Білий ящик» відповідно до нотації IDEF0 представлена на рис. 1.

Вхідними даними для процесу тестування є: вихідний код і бінарні файли ПЗ; документовані можливості програмного і технічного забезпечень; документація по попереднім тестуванням ПЗ.

Документовані (декларовані) можливості програмного і технічного забезпечень включають відомості про структуру, взаємодію компонентів і умов експлуатації програмного забезпечення, про конфігурацію, характеристиках і умовах експлуатації технічного забезпечення, а також специфікації і вимоги на все ПЗ і його компоненти.

Документація по попереднім тестуванням містить інформацію, що описує плани і результати проведених тестів, виявлені дефекти і рекомендації щодо усунення вразливостей.

В результаті проведення тестування ПЗ формується документація про результати проведення дослідження.

В якості виконавців виступають замовник і фахівці, які виконують тестування.

Механізмами виконання є тестова середовище та спеціалізоване ПЗ, що забезпечує проведення дослідження.

Керуючими впливами є стандарти та методи тестування.

Метод тестування «Білий ящик» включає наступні основні стадії і етапи.

1. Підготовча стадія (планування) складається з етапів:

1) проведення аналізу та оцінки ризиків (складання моделі загроз);

2) розробка стратегії тестування;

3) розробка детального плану тестування;

4) розробка сценаріїв тестів і визначення області покриття кожного тесту;

5) підготовка тестового середовища.

2. Стадія проведення тестування включає:

1) проведення валідації та верифікації тестових сценаріїв;

2) проведення статичного аналізу програмного забезпечення і його компонентів;

3) проведення динамічного аналізу програмного забезпечення і його компонентів;

4) проведення аналізу виявлених помилок, дефектів і вразливостей програмного забезпечення.

3. Стадія формування звітної документації є створення звітної документації про результати проведення досліджень.

Етап аналізу та оцінки ризиків включає аналіз загроз і видів вразливостей, присутніх в кожному компоненті ПЗ, ймовірності їх реалізації та оцінка можливих збитків підприємства при виникненні ризиків, вироблення рекомендацій для зниження виявлених ризиків.

Етап розробки стратегії тестів проводиться на основі результатів аналізу ризиків з метою визначення основних заходів і завдань тестування найбільш небезпечних фрагментів коду, тобто області і методів тестування, обсягу перевіряється коду, вимог до архітектури тестового середовища і кваліфікації фахівців, що виконують дослідження.

На етапі розробки детального плану тестування визначаються і узгоджуються з замовником мети, графік тестування і ступінь покриття тестами програми і її компонентів, способи контролю та вимоги до звітної документації.

На етапі створення тестів визначаються сценарії тестів, що описують критерії покриття програмного коду з урахуванням цілей тестування; початкові умови, вхідні дані, очікувані результати та контрольні точки проведення тестів.

Етап підготовки тестового середовища включає налагодження конфігурацій двох складових комп'ютерної техніки і програмного забезпечень, які відповідають необхідним умовам експлуатації.

Стадія проведення тестування ПЗ полягає в виявленні дефектів і помилок в тестованих компонентах ПЗ і підрозділяється на етапи валідації та верифікації, статичного, динамічного аналізу та аналізу результатів проведення тестових сценаріїв.

Етап валідації та верифікації тестових сценаріїв включає інспекцію тестів на відповідність детальному плану і стандартам і аналіз отриманих результатів.

На етапі проведення статичного аналізу проводиться пошук помилок без виконання тестового ПЗ за допомогою аналізу вихідного коду перевіряються компонентів з використанням спеціалізованого програмного забезпечення.

При динамічному аналізі здійснюється перевірка реальної поведінки досліджуваного ПЗ в рамках певних сценаріїв його роботи з використанням спеціалізованого програмного забезпечення.

Етап аналізу результатів тестування включає діагностику та локалізацію виявлених помилок і дефектів в тестованому ПЗ, аналіз досягнутого тестового покриття, прийняття рішення про створення додаткових тестових сценаріїв або про припинення тестування.

На стадії формування звітної документації складається звіт про результати тестування ПЗ, який включає опис сценаріїв проведених тестів, тестові контрольні дані, опис архітектури тестового середовища, ступеня покриття програм сценаріями тестів і статусу отриманих помилок і дефектів, інформацію про документування процесу тестування, список задіяних в ньому фахівців. Для кожної виявленої уразливості вказуються тестовий сценарій, умови її відтворення і методи усунення [4].

Висновки. Таким чином, в статті розглянуто особливості процесу тестування програмного забезпечення, зокрема методу «Білий ящик». Досліджено основні переваги, етапи, стратегії запропонованого методу. На основі проведених досліджень була побудована функціональна модель процесу тестування методом «Білий ящик» за допомогою методології функціонального моделювання і графічного описання процесів IDEF0.

Список літератури:

1. Тамре Л. Введение в тестирование программного обеспечения. – М.: Вильямс, 2003. – 359 с.
2. Василенко Н.В. Модели оценки надежности програмного обеспечения / Н.В. Василенко, В.А. Макаров // Вестник НГУ. – 2004. – № 28. – С. 126-132.
3. Бирюков С.В. Анализ стратегий тестирования програмного обеспечения / С.В. Бирюков // Вестник МГТУ. – 2010. – С. 59-63.
4. Канер С., Фолк Д., Кек Нгуен Е. Тестирование программного обеспечения. – Киев: ДиаСофт, 2000. – 544 с.

Новикова К.В., Лютая М.В.
Киевский национальный университет технологий и дизайна
Розломий И.О.
Черкасский национальный университет имени Богдана Хмельницкого

ИССЛЕДОВАНИЕ МЕТОДА ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ «БЕЛЫЙ ЯЩИК»

Аннотация
Статья посвящена обоснованию метода тестирования программного обеспечения «Белый ящик». Проанализированы основные стратегии, стадии и этапы метода тестирования «Белый ящик». Были представлены особенности структурного тестирования. В результате была разработана функциональная модель процесса тестирования методом «Белый ящик» по нотации IDEF0. Составлена подробная характеристика каждого этапа тестирования методом «Белый ящик».
Ключевые слова: тестирование программного обеспечения, «Белый ящик», структурное тестирование, функциональная модель, IDEF0.

Novykova K.V., Lyutaya M.V.
Kyiv National University of Technologies and Design
Rozlomii I.O.
Cherkassy Bogdan Khmelnitsky National University

SOFTWARE TESTING RESEARCH OF THE «WHITE BOX» METHOD

Summary
The article is devoted to the substantiation of the method of software testing «White Box». The main strategies, stages and stages of the White Box test method are analyzed. Features of structural testing were presented. As a result, a functional model of the testing process using the «White Box» method was developed in accordance with the notation IDEF0. The detailed description of each stage of the testing by the «White Box» method is prepared.
Keywords: software testing, «White Box», structural testing, functional model, IDEF0.