

Сучасні методологічні підходи до тестування програмного забезпечення

Вінницький національний технічний університет

В статті розглянуто особливості та методологічні засоби налагодження тестувального циклу в процесі підготовки фахівців з інженерії програмного забезпечення.

Ключові слова: тестування, BDD, TDD, unit-тест, програмне забезпечення, сценарій.

Modern methodological approaches to software testing

In the article the methodological characteristics and debugging testing cycle in the software engineers training process.

Keywords: testing, BDD, TDD, unit-test, software, scenario.

Науково-технічний прогрес стимулює прогресивні зміни в усіх сферах діяльності сучасного суспільства. Сучасні технології дозволяють розробляти нові рішення, спрямовані на забезпечення високоякісного рівня життя та підвищення комфорту в користуванні. На сьогоднішній день найактивнішою нішею програмного забезпечення є серверні та веб-додатки. Тому, особливо актуальним питанням в процесі підготовки фахівців з розробки програмного забезпечення є вибір засобів розгортання та неперервної інтеграції розроблених проектів.

В циклі розробки програмного продукту важливу досить важливу роль відіграють засоби тестування. Сучасні методології дозволяють удосконалювати продукт, знаходити помилки, виявляти потреби користувача на етапі проектування та розробки. Так новітні підходи до тестування передбачають створення тестових ситуацій до написання робочих модулів, які мають бути задіяними у тестовій ситуації.

Метою дослідження є виявлення сучасних методів для тестування програмного коду.

Об'єктом дослідження постають методи практичного застосування, такі як, TDD, BDD. Предметом дослідження вбачаються засоби проведення тестування програмних продуктів.

Тестування програмного забезпечення (Software Testing) – перевірка відповідності між реальною і очікуваною поведінкою програми, що здійснюється на кінцевому наборі тестів, обраному певним чином [1].

Розглянемо основні два підходи до тестування програмного забезпечення, які найактивніше використовуються в сучасному розробницькому процесі.

TDD (test-driven development) – підхід до розробки та тестування, при якому Керована тестами розробка вимагає від розробника створення автоматизованих модульних тестів, які визначають вимоги до коду безпосередньо перед написанням самого коду. Тест містить перевірки умов, які можуть або виконуватися, або ні. Коли вони виконуються, кажуть, що тест пройдено. TDD – процес інтерактивний. Додаючи в клас щось нове, ви спочатку пишете тест на новий функціонал і тільки потім створюєте мінімальну кількість коду, що реалізує потрібну поведінку. Домігшись успішного проходження тесту, можна задуматися про якість коду та зробити його рефакторинг [2].

Використання TDD надає наступні переваги при тестуванні програмного забезпечення. Тести

дозволяють робити рефакторинг коду без ризику його зламати. При внесенні змін в добре протестований код, ризик появи нових помилок значно нижче. Якщо нова функціональність призводить до помилок, тести, якщо вони звичайно є, одразу ж це покажуть. При роботі з кодом, на якому немає тестів, помилку можна виявити через значний час, коли з кодом працювати буде набагато складніше. Якщо існуючий код добре покритий тестами, розробники будуть відчувати себе набагато вільніше при внесенні архітектурних рішень, які покликані поліпшити дизайн коду.

Керована тестами розробка сприяє більш модульному та гнучкому коду, що призводить до менших, більш спеціалізованих класів, зменшенню пов'язаності і чистіших інтерфейсів. Оскільки пишеться лише той код, що необхідний для проходження тесту, автоматизовані тести покривають всі шляхи виконання.

Тести можуть використовуватися набагато ефективніше, ніж документація. Документація та коментарі в коді, на відміну від тестів, можуть застарівати, що може збивати з пантелику розробників, які вивчають код.

BDD (behavior-driven development) – розширення підходу TDD до розробки та тестування, при якому особлива увага приділяється поведінці системи, модуля. Як правило, такі тести ілюструють і тестують різні сценарії, які цікаві безпосередньо клієнту системи. У зв'язку з цим при складанні таких тестів часто використовується фреймворки, що мають синтаксис, що забезпечує читаність тестів не тільки програмістом, а й представниками замовника [2].

Якщо TDD використовується для написання тестів програмістами для програмістів, BDD-тести можуть бути написані, наприклад, технічними менеджерами або тестувальниками, що робить можливим їх використання не тільки при формальній TDD-розробці, але й при складанні компонентних тестів, а також при формалізації вимог до системі.

Компонентні тести, які використовують синтаксичні конструкції BDD, як правило, тестують зовнішній інтерфейс модуля та не звертають уваги на його внутрішню реалізацію. Такі тести можуть бути написані безпосередньо після узгодження зовнішнього інтерфейсу системи, тим самим формалізувати вимоги до системи та процес фінального тестування. Наявність компонентних тестів, що повністю покривають функціонал системи дозволяє безболісно змінювати її внутрішню реалізацію. У зв'язку з цим час життя таких компонентних тестів може значно перевищувати час життя юніт-тестів, що покривають код окремих складових модуля: внутрішня архітектура і реалізація модулів може змінитися, що призведе до заміни й юніт-тестів [3].

До компонентних і формалізуючих BDD тестів вже не може бути застосований ітеративний підхід TDD. Самі тести можуть не бути такими атомарними і реалізація, необхідна для їх виконання, не завжди тривіальна.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Software testing [Електронний ресурс] // Режим доступу: https://en.wikipedia.org/wiki/Software_testing

2. Тестирование при TDD и BDD [Електронний ресурс] // Режим доступу: <http://blog.bausov.pro/tdd-bdd-tests/>

3. Testing via BDD [Електронний ресурс] // Режим доступу: <https://habrahabr.ru/post/139674/>

Орехов Олег Русланович, студент групи ІІІ-15 МС, Вінницький національний технічний університет, Вінниця.
Кобылянська Ірина Миколаївна, кандидат педагогічних наук, доцент кафедри БЖДПБ, Вінницький національний технічний університет, Вінниця.

Orekhov Oleg R., student of group ІІІ-15 МС, Vinnytsia National Technical University, Vinnitsa.

Kobyljanska Iryna M., PhD, assistant professor of Department of Health and Safety Studies, Vinnytsia National Technical University, Vinnitsa