

7. Мейерс С. Эффективный и современный C++. М.: Вильямс, 2016. - 304 с.

НИКІТЧЕНКО Я.Ю., РЕЗАНОВА В.Г.

АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ ВЕБ-ЗАСТОСУНКУ З ВИКОРИСТАННЯМ BDD-ТЕХНОЛОГІЙ

NIKITCHENKO Y.Y., REZANOVA V.G.

AUTOMATED TESTING OF A WEB APPLICATION USING BDD TECHNOLOGIES

Purpose and tasks. The aim of the work are to consider various approaches to software testing, to consider the features of BDD, as well as to create a framework for automated testing of web application using BDD-technologies

The task is to automate all processes of checking criteria to make process of testing cheaper and faster.

Object and subject of research. The object of the research is automation and programming of the testing framework.

The subject of the study is is the process of creating and implementing testing steps using BDD-technology to check that all criteria was met.

Вступ

Тестування — це широкий процес, який складається з декількох взаємопов'язаних процесів. Іншими словами сукупності процесів.

Часто говорячи про тестування, люди уявляють собі картинку, в якій спеціаліст перевіряє чи за всіма параметрами програма працює ідеально. Але якість не є абсолютною, це суб'єктивне поняття. Тому тестування не може повністю забезпечити коректність програмного забезпечення. Воно тільки порівнює стан і поведінку продукту зі специфікацією. При цьому треба розрізняти тестування програмного забезпечення і забезпечення якості програмного забезпечення, до якого належать усі складові ділового процесу, а не тільки тестування.

Існує багато підходів до тестування програмного забезпечення, але ефективне тестування складних продуктів - це по суті дослідницький процес, а не тільки створення і виконання рутинної процедури.

Тестування пронизує весь життєвий цикл ПЗ, починаючи від проектування і закінчуючи невизначено довгим етапом експлуатації. Ці роботи безпосередньо пов'язані із завданнями управління вимогами та змінами, адже метою тестування є якраз можливість переконатися у відповідності програм заявленим вимогам.

Тестування - процес також ітераційний. Після виявлення та виправлення кожної помилки обов'язково слід повторити тести, щоб переконатися у працездатності програми. Більше того, для ідентифікації причини виявленої проблеми може знадобитися проведення спеціальної додаткової перевірки. При цьому потрібно завжди пам'ятати про фундаментальний висновок, зроблений професором Едджером Дейкстри у

1972 році: "Тестування програм може служити доказом наявності помилок, але ніколи не доведе їхню відсутність!".

Постановка завдання

Тестування програмного забезпечення є невід'ємною частиною створення програмного продукту. Від того, наскільки досконало проведені тести, залежить те, як скоро проект буде зданий остаточно, і чи буде необхідність згодом усувати помилки.

Люди схильні помилятися, людські помилки можуть призводити до порушення нормальної роботи програмного забезпечення на всіх стадіях розробки, причому наслідки цього можуть бути найрізноманітнішими — від незначних до катастрофічних.

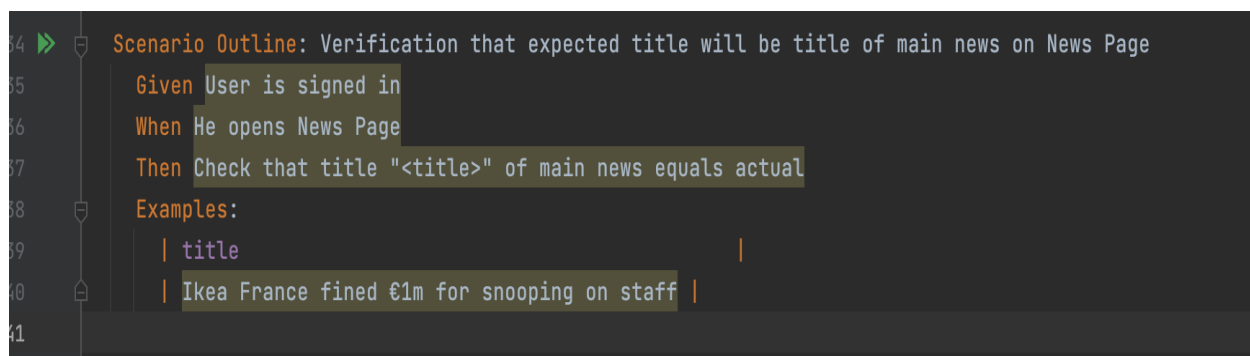
Отже, мета і завдання роботи - це розглянути різноманітні підходи до тестування програмного забезпечення, розглянути особливості BDD, а також створити фреймворк для автоматизованого тестування.

Основна частина

У залежності від переслідуваних цілей види тестування можна умовно розділити на наступні типи:

1. Функціональні.
2. Нефункціональні.
3. Пов'язані зі змінами.

Сучасні проекти все частіше висувають високі вимоги до покриття автоматичними тестами. У наш час писати тести не просто ознака хорошого тону, але одна з вимог, яка пред'являється до коду. Все частіше ми чуємо такі аббревіатури, як TDD (Test Driven Development) і BDD (Behaviour Driven Development) і багато строго слідують цим підходам в розробці.



```
34 > Scenario Outline: Verification that expected title will be title of main news on News Page
35   Given User is signed in
36   When He opens News Page
37   Then Check that title "<title>" of main news equals actual
38   Examples:
39   | title |
40   | Ikea France fined €1m for snooping on staff |
41
```

Рисунок 1 - Приклад сценарію BDD

Основною ідеєю даної методології є поєднання в процесі розробки чисто технічних інтересів і інтересів бізнесу, дозволяючи тим самим керуючому персоналу і програмістам говорити на одній мові.

Для спілкування між цими групами персоналу використовується предметно-орієнтована мова, основу якої становлять конструкції з природної мови, зрозумілі неспеціалісту, яка зазвичай виражає поведінку програмного продукту і очікувані результати.

Проте за кожним словом даного сценарію скривається певна функція, яка на виконує певні кроки, а вкінці порівнює отриманий результат з очікуваним.

```
@Given("User is signed in")
public void signIn() { new SignInPage(getDriver()).signIn(); }

@When("He opens News Page")
public void openNewsPage() { new HomePage(getDriver()).newsButtonClick(); }

@Then("Check that title {string} of main news equals actual")
public void checkMainTitle(String string) {
    String mainTitle = new NewsPage(getDriver()).mainNewsTitle();
    assertThat(string)
        .as("description: Title of given main news differ from actual")
        .isEqualTo(mainTitle);
}
```

Рисунок 2 - Методи, які виконують кроки сценарію

Висновки

Проведення перевірки якості продукту є невід’ємною частиною розробки програмного забезпечення. Тестування не тільки дозволяє перевірити, чи всі вимоги до проекту виконані, а й допомагає у виявленні помилок та дефектів, що пом’якшує наслідки і ризики втрат.

У вільному доступі існує багато проектів, де застосовується BDD, є документація та роботи. Це є доказом того, що BDD має широкий потенціал використовуватись надалі при тестуванні. Крім того, дана технологія дуже полегшує взаємодію між замовником та виконавцем. Використовуючи її, вони будуть розмовляти на “одній” мові, тому що більшість сценаріїв для тестування пишеться на чистій англійській мові.

Використання BDD значно економить час і робота може виконуватись значно швидше, бо замовник може сам проаналізувати, що йому потрібно, а спеціалісту лише доведеться перенести слова замовника на необхідну мову програмування.

Ключові слова: тестування, процес розробки програмного забезпечення, Behavior-Driven Development.

Література

1. Що таке тестування програмного забезпечення та яке його значення? [Електронний ресурс]. Режим доступу:

<https://www.quality-assurance-group.com/shho-take-testuvannya-programnogo-zabezpechennya-ta-yake-jogo-znachennya/>

2. Тестування програмного продукту [Електронний ресурс]. Режим доступу: <http://lib.mdpu.org.ua/e-book/vstup/L11.htm>

3. Basics of BDD in testing, by Alex McPeak [Електронний ресурс]. Режим доступу: <https://crossbrowsertesting.com/blog/development/what-is-bdd>

4. The Pros and Cons of Behavior-Driven Development, by Emilie Maxie [Електронний ресурс]. Режим доступу:

<https://www.verypossible.com/blog/the-pros-and-cons-of-behavior-driven-development>

ВОЛІВАЧ А.П.

СПЕЦИФІКА ВИКЛАДАННЯ ДИСЦИПЛІНИ «ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ»

VOLIVACH A. P.

SPECIFICS OF TEACHING THE DISCIPLINE "INFORMATION SYSTEMS AND TECHNOLOGIES"

Purpose - is to apply the modern approach to teaching the discipline "Information Systems and Technologies" with the use of «Standards and Guidelines for Quality Assurance in the European Higher Education Area (ESG)».

The theoretical bases of researches are the use of information technologies and their influence on the learning outcomes during the study of the subject of information systems and technologies.

It has been that the discipline "Information Systems and Technologies" implies the continuous use of IT, using modern computer programs. Studying this subject forms the students modern principles and approaches to work with information and provides practical skills to solve production issues related to their future profession.

Keywords: information technologies (IT), education, of higher education.

Вступ

Вища освіта України перебуває у стані реорганізації та модернізації, і потребує нового якісного підходу до діяльності ВНЗ. Зокрема це стосується механізмів підготовки висококваліфікованого компетентнісного випускника, який володітиме новітніми знаннями, уміннями, навичками у вирішенні виробничих ситуацій, матиме бажання постійно удосконалювати професійний рівень, буде придатним до працевлаштування за обраною професією.

Закон України “Про вищу освіту” від 1.07.2014 р. № 1556-VII [1] передбачає застосування нового якісного підходу до освітніх послуг в суспільстві, насамперед це стосується змін структури освітніх програм, форм і методів організації навчального процесу.

Входження України в європейські структури з дотриманням вимог Болонської декларації привело до змін організації освіти у вищих навчальних закладах. В основу цих змін закладено: методологію узгодження освітніх структур та навчальних планів, відповідно до проекту Тюнінг «Гармонізації освітніх структур в Європі» («Tuning Educational Structures in Europe») [2] – основи розробки освітніх програм, що