

УДК 681.3.07

Шевчук А. – ст. гр. СІМ-52

Тернопільський національний технічний університет імені Івана Пулюя

ПІДВИЩЕННЯ ЯКОСТІ ПРОЦЕСУ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ШЛЯХОМ ВИКОРИСТАННЯ НЕПЕРЕРВНОЇ ІНТЕГРАЦІЇ

Науковий керівник: к.т.н., доц. Луцків А.М.

Shevchyk A.M.

Ternopil Ivan Pul'uj National Technical University

IMPROVING THE QUALITY PROCESS OF SOFTWARE DEVELOPMENT BY USING CONTINUOUS INTEGRATION

Supervisor: Lutskev A. M

Ключові слова: неперервна інтеграція, розроблення програмного забезпечення, якість
Keywords: continuous integration, software development, quality

Процес створення програмного забезпечення має низку етапів життєвого циклу. З метою підвищення якості процесу розроблення програмного забезпечення у 1999 р., Мартіном Фаулером було запропоновано використання концепції неперервної інтеграції (англ. Continuous Integration) — розроблення програмного забезпечення, яке полягає у виконанні частих(періодичних) автоматизованих збирань/компіляцій (build) проекту для якнайшвидшого виявлення та вирішення інтеграційних проблем. Дана концепція передбачає слідування певним правилам та використання спеціалізованих засобів. На сьогодні до найпопулярніших засобів неперервної інтеграції належать:

- Jenkins/Hudson (проект для неперервної інтеграції з відкритим вихідним кодом, написаний на Java);
- TeamCity (серверне програмне забезпечення від компанії JetBrains, написане на мові Java, білд-сервер для забезпечення неперервної інтеграції);
- Travis CI (розподілений web-сервіс для складання та тестування програмного забезпечення, що використовує GitHub в якості хостингу коду);
- IBM Bluemix (реалізація відкритої хмарної архітектури IBM, заснована на Cloud Foundry, яка дозволяє швидко створювати, розгортати і адмініструвати хмарні додатки).

У рамках магістерського дослідження використовується система Jenkins для забезпечення розроблення web-сервісу у рамках технології JavaEE. Яка має можливість шляхом використання різноманітних розширень (plugins) розширювати свій функціонал.

Зупинимось на ключових засобах, які дають змогу підвищити якість створення front-end та back-end компонентів web-сервісу.

Зокрема, пропонується використання розширень для:

- front-end компонентів:

1) тестування веб-інтерфейсу шляхом використання WebDriver та Selenium:

Selenium - це об'єктно-орієнтований Java-додаток, який може аналізувати файли певної структури для того, щоб знаходити в них команди для маніпуляції браузером і команди для виконання певних дій і перевірок з представлення web-сторінки.

WebDriver призначений для забезпечення більш простого і лаконічного інтерфейсу програмування.

- back-end компонентів:

1) модульне та інтеграційне тестування може бути забезпечене шляхом написання автоматизованих unit- та інтеграційних тестів. Система неперервної інтеграції Jenkins дає змогу здійснювати тестування з використанням бібліотек JUnit, TestNG, а також Mockito:

JUnit – це фреймворк, який призначений для написання та запуску тестів для одиниць виконання програмного забезпечення.

TestNG – це фреймворк, який має аналогічний до JUnit функціонал, а також низку додаткових можливостей.

Mockito — бібліотека, яка шляхом використання проху-класів дає змогу перехоплювати виклики до реальних екземплярів об'єктів, що дає змогу проводити тестування коду програми окремо від самої системи.

2) статичні аналізатори коду: PMD і FindBugs:

Основне призначення PMD - пошук неоптимального коду, проблем з продуктивністю, порушень стилю кодування, дублів в кодї і т. д.

Статичний аналізатор FindBugs є власний список, де перераховані і упорядковані всі помилки та способи їх вирішення. Кожній помилці присвоєна відповідна ступінь серйозності: висока, середня чи низька.

У командах розробників, як правило, працюють люди з різним рівнем кваліфікації та стажу, а тому доцільно використовувати засоби, які дадуть змогу контролювати якість форматування коду, для чого доцільно використати розширення CheckStyle.

Система перевірки стилю програмування CheckStyle дозволяє автоматично перевіряти відповідність стилю Java-коду до одного із стилів, який може бути прийнятий в рамках того чи іншого проекту.

Система неперервної інтеграції здійснює отримання коду з репозиторію програмного забезпечення — системи контролю версіями (Git, Mercurial, Subversion).

Веб-інтерфейс системи дає змогу як налаштовувати процес неперервної інтеграції, так і здійснювати його детальний моніторинг усіма членами команди розробників, зокрема: отримання даних з системи контролю версіями, компілювання, збирання, тестування, формування інсталяційних (виконуваних) модулів програми та перегляд результатів й статистичних даних по кожному з етапів, стан того чи іншого модуля.

Використання систем неперервної інтеграції є особливо актуальним при використанні гнучких методологій розроблення, зокрема при TDD-методології. Може бути корисним як при створенні комерційного програмного забезпечення так і при написанні відкритих проектів. Система неперервної інтеграції може бути використана при створенні різних типів програм: мобільних, Web-додатків, Desktop і вбудованих. Може бути використана для різноманітних мов програмування, як інтерпретованих так і тих, які передбачають формування виконуваного бінарного або байт-коду компілятором.