

УДК 004.891.3: 004.4

**Р. Левицький**

(Тернопільський національний технічний університет імені Івана Пулюя)

## **ВПРОВАДЖЕННЯ BDD МЕТОДОЛОГІЇ В ІТ ПРОЕКТІ**

UDC 004.891.3: 004.4

**R. Levytskyi**

## **ADOPTING BDD METHODOLOGY IN IT PROJECT**

Поведінкова розробка (BDD) – це процес розробки програмного забезпечення, який спрямований на вирішення проблеми реалізації невизначених вимог. Він призваний використовувати доменні знання бізнесу та професіоналів із забезпечення якості, щоб розробники створювали правильне програмне забезпечення.

У якийсь момент організація вирішує, що потрібно пришвидшити релізи і швидше вийти на ринок. Тож вона переходить до Agile методології для досягнення цих цілей. Це може бути ще не повноцінний Agile, але команда працює по спринтах із частими релізами.

Однак, оскільки системи контролю якості намагаються йти в ногу з розробкою, тестування часто стає вузьким місцем при переході до Agile. У відповідь на це, команди вбудовують тестування в процес розробки, що полегшує запобігання лишнім валідаціям, які можуть загрожувати приросту продуктивності, отриманого шляхом переходу на Agile методологію.

На даному етапі не тільки пришвидшується розробка, але й забезпечується якість програмного забезпечення завдяки постійним практикам тестування. Але звідки команда може знати, що процес побудований правильно? Звичайно, це може працювати, але чи це те, що хоче або потребує клієнт?

Призначення BDD якраз і полягає у забезпеченні чіткого визначення та правильного виконання вимог до керівництва роботою з розробки.

Згідно з підходом BDD, під час написання визначення кроку, розробник повинен написати код, щоб визначити операцію, яку слід виконати, коли виконується тестовий крок, який відповідає визначенню.

Деякі засоби автоматизації тестів можуть зменшити технічні бар'єри на шляху впровадження BDD, дозволяючи розробникам автоматизованих тестів писати файли з тестовими кроками з більш простим синтаксисом, ніж Java, .NET або іншу мову програмування. Ці ж інструменти дозволяють в подальшому виконувати визначені кроки як автоматизовані тести, а також надавати звіти про їх виконання в зручному для сприйняття вигляді.

Прикладом такого засобу автоматизації може слугувати реалізація BDD підходу на мові Java, який має назву Cucumber. Даний фреймворк дозволяє записувати тестові кроки англійською, або однією з багатьох інших доступних мов, та асоціювати ці кроки з написаними на мові Java методами, які реалізують ці кроки.

Наприклад, визначення кроку може посилатися на окремий тест, який викликає REST API. Визначення другого кроку може стосуватися іншого тесту, який перевіряє дані відповіді з іншого REST API. Визначення третього кроку може стосуватися тесту, який виконує запит до бази даних і перевіряє дані у наборі результатів.

Як результат, засоби автоматизації тестування забезпечують тестувальникам більше контролю. Їхня впевненість у повноті написаного тестового покриття також зростає. Використовуючи зріле, повністю функціональне рішення для наскрізного тестування, можна легко розпочати автоматизацію тестування, підтримку тестів та природну інтеграцію в існуючий робочий процес CI/CD. На більш високому рівні це дозволяє організації використовувати менше технічних ресурсів для впровадження BDD, тим самим звільняючи ресурси для власне розробки програмного забезпечення.