

Віктор МАЛЯРСЬКИЙ

*Здобувач третього (освітньо-наукового) рівня вищої освіти,
Тернопільський національний педагогічний університет
імені Володимира Гнатюка,
Тернопіль, Україна
vmalarskij@gmail.com*

РОЗВИТОК СУЧАСНИХ ПІДХОДІВ ДО ОЦІНКИ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Постановка проблеми. Стрімкий розвиток сучасних технологій та постійні зміни в процесі їх розробки зумовлюють відповідні зміни на всіх рівнях життєвого циклу розробки програмного забезпечення, в тому числі і в процесі оцінки та забезпечення його якості. Майбутнім та вже досвідченим фахівцям важливо знати не лише трендові інструменти тестування, а і підходи та процеси, що дозволяють досягати бажаної якості продукту швидше, при цьому витрачаючи менше ресурсів. Розуміння сучасних цінностей в розробці та тестуванні програмного забезпечення надає змогу спеціалістам, та проєктам, над якими вони працюють, отримувати результати роботи в коротші терміни, а також більше відповідати очікуванням клієнтів та кінцевих споживачів.

Сучасні підходи до процесів оцінки та забезпечення якості програмного забезпечення. Тестування програмного забезпечення (ПЗ) є невід'ємною частиною життєвого циклу розробки, що забезпечує якість, надійність і відповідність функціоналу очікуванням користувача. Протягом останнього десятиліття ця галузь зазнала суттєвих трансформацій, пов'язаних із глобальною цифровою трансформацією, зростанням складності ІТ-продуктів, поширенням Agile-методологій та DevOps-підходів. Ці зміни вплинули як на інструменти тестування, так і на ролі та компетентності фахівців.

Однією з найвизначальніших змін стало масове впровадження автоматизованого тестування. Зараз більшість компаній застосовують гібридну модель покриття тестових сценаріїв, де автоматизація охоплює рутинні та повторювані сценарії, особливо для регресійного тестування. Це стало можливим завдяки поширенню фреймворків, що вже мають в собі вбудований функціонал для роботи з елементами застосунків, API та методи перехоплення чи очікування подій, що значно пришвидшує розробку тестів [1]. Також популярним є використання інструментів CI/CD, що інтегрують тестування в процеси впровадження коду.

Особлива увага зараз приділяється навантажувальним тестам та тестам безпеки та продуктивності. З ростом числа кібератак та витоків даних безпека стала критично важливою. Останні роки позначилися розвитком напрямку Security Testing, зокрема Penetration Testing, Static/Dynamic Analysis

(SAST/DAST), а також автоматизованих рішень для виявлення вразливостей. Не менш важливим стало оцінювання продуктивності застосунків. Оскільки використання застосунків стало більш доступним, зросли вимоги до кількості користувачів, що одночасно можуть працювати з ними. Через це розробники почали більш ретельно тестувати вплив великого потоку запитів на швидкість завантаження застосунків.

Розширення варіативності пристроїв і платформ (мобільні, IoT, веб, десктоп) змусило інженерів з тестування адаптувати свої методи до мультиплатформенності. З'явилися рішення для кросплатформеного тестування (наприклад, BrowserStack, Sauce Labs), які дозволяють швидко перевірити ПЗ на різних ОС та пристроях. Також значна частина тестування сьогодні відбувається в хмарному середовищі. Це спрощує масштабування, забезпечує гнучкість та дозволяє зменшити витрати на інфраструктуру.

Інновації у сфері штучного інтелекту (AI) і машинного навчання (ML) також знайшли своє застосування і у сфері тестування. Нові інструменти використовують AI для автоматичного створення тестів, виявлення аномалій, візуального тестування та навіть прогнозування потенційних проблем у коді. ML-алгоритми також допомагають оптимізувати підбір тестових сценаріїв, визначити, які тести є надлишковими або які з них найбільш ефективні для конкретної функціональності застосунку. Також вагомий внесок мають програмні розширення з вбудованим AI у існуючі інструменти автоматизації тестування. Вони дозволяють швидше розробляти фреймворки а також перевіряти їх на помилки.

Як вже було згадано вище, в галузі оцінки та забезпечення якості програмного забезпечення розвинулись не лише технології та інструменти, а і цінності тест-менеджменту та підходи до моделювання тест-планів та процесів життєвого циклу тестування (STLC). Раніше тестувальник часто сприймався як технічний виконавець. В сучасних проєктах його роль трансформувалася у фахівця з забезпечення якості (QA), який не лише виконує тести, а й бере участь у плануванні, ризик-менеджменті, моделюванні поведінки користувача, а також аналізує результати з точки зору цінності для кінцевого споживача. Тому зараз фахівці з забезпечення якості програмного забезпечення повинні також володіти базовими навичками бізнес аналітики, дизайну та проєктного менеджменту.

Щодо самого процесу розробки та тестування програмного забезпечення, то на сучасних проєктах набагато рідше почали використовуватись такі колись популярні методології розробки як каскадна (Waterfall) чи ітеративна де процес тестування виконується окремо від всіх інших. Сучасні команди надають перевагу гнучким методологіям SCRUM чи Kanban, де оцінка якості розпочинається ще на етапі збору вимог до продукту. Цей підхід дозволяє

команді набагато ретельніше та детальніше аналізувати всі потреби замовника або кінцевого споживача, розуміти всі аспекти функціоналу, що розробляється, і в результаті мати більш якісний продукт [2].

Висновки. Підсумовуючи, варто наголосити, що тестування програмного забезпечення в умовах сучасних технологічних тенденцій перестало бути суто технічним процесом. Воно набуло ознак інтегрованої, автоматизованої, інтелектуальної та мультидисциплінарної діяльності, що є запорукою не лише технічної справності, але й відповідності ПЗ ключовим сучасним цінностям: безпеці, якості, доступності та етичності. З огляду на це, майбутнім фахівцям у цій сфері критично важливо постійно оновлювати свої знання, гнучко реагувати на зміни, демонструвати системне мислення та усвідомлювати свою відповідальність не лише за якість коду, але й за суспільний вплив технологій, що ними створюються. У подальших дослідженнях варто детальніше розглянути професійні компетентності сучасних фахівців з оцінки якості програмного забезпечення для формування методик їх розвитку у студентів комп'ютерних спеціальностей вищих навчальних закладів.

ЛІТЕРАТУРА

1. Малярський В. О. Сучасні інструменти та фреймворки автоматизованого функціонального тестування Web-застосунків. Сучасні цифрові технології та інноваційні методики навчання: досвід, тенденції, перспективи : матеріали XIV Міжнар. наук.-практ. інтернет-конф., м. Тернопіль, 7 – 15 лис. 2024 р. / Тернопільський національний педагогічний університет ім. В. Гнатюка, Тернопіль, 2024. С. 121-124.
2. Najihi S., Elhadi S., Abdelouahid R. A., Marzak A.: Software Testing from an Agile and Traditional view. *Procedia Computer Science*. 2022. Vol. 203, P. 775—782. DOI: 10.1016/j.procs.2022.07.116 URL: <https://doi.org/10.1016/j.procs.2022.07.116> .

Олег МЯХКОТА

*Здобувач третього (освітньо-наукового) рівня вищої освіти,
Тернопільський національний педагогічний університет
імені Володимира Гнатюка,
Тернопіль, Україна
aptekar.agronom@gmail.com*

ЦІННІСТЬ ПРИВАТНОСТІ В ЦИФРОВОМУ СВІТІ: ФІЛОСОФСЬКИЙ АНАЛІЗ

У сучасному цифровому середовищі проблема приватності набуває особливої гостроти, виходячи далеко за межі технічної чи правової площини. Вона стає предметом глибокого філософського осмислення, адже торкається базових понять свободи, автономії, контролю та людської гідності. Технології –