## Рівні та види тестування

## Beet Seed — відпрацюй навички на базовому рівні

1. Склади порівняльну таблицю функціонального, нефункціонального і пов’язаного зі змінами видів тестування.

|  | функціональне тестування | нефункціональне тестування | пов’язане зі змінами тестування |
| --- | --- | --- | --- |
| що перевіряється | поведінку компоненту або системи | відповідності компонента або системи нефункціональним вимогам | наявність помилок, що були знайдені раніше та нові помилки |
| коли застосовується | коли треба перевірити функціональність системи без чіткого розуміння самої системи (black box) - “ЩО САМЕ робить система?” | коли треба оцінити готовність системи за різними критеріями, які не охоплюються функціональним тестуванням - “НАСКІЛЬКИ ДОБРЕ працює система?” | коли треба переконатися, що раніше викорінені помилки було виправлено, а також виявити помилки, які могли випадково з’явитися в новій версії - Confirmation testing (Re-testing) та Regression testing. |
| обмеження | Існує висока ймовірність зайвого тестування.  Логічні помилки в програмному забезпеченні можуть бути пропущені під час виконання функціонального тестування. |  | Має виконуватись після інших видів тестування, оскільки перевіряє чи з’явилась нова помилка у тест-кейсі, що вже проходили, та чи пофіксили проблеми, які знайшлися при попередніх перевірках |
| особливості | * Підготовка тестових даних має основуватись на специфікацій функцій * Бізнес-вимоги є вхідними даними для функціонального тестування * виконання test cases * перевірка фактичних та очікуваних результатів (AR\ER). |  | Спочатку треба виконати Confirmation testing (re-testing), щоб переконатися, що помилку справді успішно видалено. Test case, у якому була помилка в минулий раз, тепер має пройти без проблем. |

2. Поясни, в чому різниця між регресією та ретестингом (5 речень).

**Ретестінг** перевіряє, чи було виправлено конкретні баги, які було знайдено при попередніх перевірках. Він виконується з тими самими вхідними даними та тим самим середовищем, але з новою збіркою.

**Регресія** не передбачає перевірку того, чи був конкретний баг виправлений. Вона оцінює як вплинули ці виправлення на повторне проходження тест-кейсу. Регресія дає зрозуміти, чи продукт нормально працює з новим функціоналом або з виправленними помилками.

## Beet Sprout — детальніше заглибся в практику.

1. Виконай завдання попереднього рівня.

2. Як ти вважаєш, чи можливе для продукту проведення тільки функціонального тестування, без перевірки нефункціональних вимог?

Якщо так – в яких випадках?

Якщо ні – чому?

Обґрунтуй свою відповідь.

На мою думку, повноцінне тестування продукту вимагає проведення усіх видів тестування - і функціонального, і нефункціонального, і регресіонного з ретестами (аби перевірити виправленні виявленні помилки). Необхідно перевірити систему із різних сторін - і що саме вона робить, і наскільки правильно вона працює.

Відсутність якогось виду тестування можливо у разі, якщо на це наполягає клієнт-замовник продукту.

3. Як ти розумієш необхідність проведення smoke (димового) тестування? Чи завжди воно є доречним?

Метою цього тестування є перевірка «стабільності» системи в цілому, щоб дати зелене світло проведеному більш ретельному тестуванню. Воно перевіряє тільки найважливіший функціонал та не передбачає повторну перевірку дефектів, що були знайдені до цього.

Воно є доречним після випуску нової зборки проекту, аби швидко перевірити, чи система “відгукується”, чи вона взагалі працює.