**Level 1.** Порівняльна таблиця видів тестування:

|  | що перевіряється | коли застосовується | обмеження | особливості |
| --- | --- | --- | --- | --- |
| Функціональне тестування | Функціі, які ПЗ повинно виконувати згідно документації | На всіх етапах розробки продукту | - Можливість упущення логічних помилок в ПЗ | - Може базуватися або на специфікації, або на бізнес-процесах |
| Нефункціональне тестування | Властивості, які не є функціями ПЗ | Коли треба протестувати як система веде себе | - Не всі розуміють їхню цінність | - Підвищує якість ПЗ та рівень безпеки система та її даних  - Майже не тестується мануально |
| Тестування пов’язане зі змінами | Чи було виправлено дефект та чи відповідає змінене ПЗ вимогам | Після того як були внесені зміні до ПЗ або виявлено та виправлено дефект | - Виконується тільки за умови внесення змін до ПЗ | - Використовуються вже наявні текст-кейси |

2. *В чому різниця між регресією та ретестингом?* Ретестинг проводиться у випадку, коли фіча/функціонал мав дефекти, і ці дефекти були нещодавно виправлені. Одним словом, ми пере-перевіряємо виправлений баг. В той час як регресійне тестування проводиться тоді, коли потрібно переконатися, що нові (додані) функції ПЗ чи виправлені дефекти не вплинули на поточний, вже існуючий функціонал, що працював (та був протестований) раніше. Простими словами, ми перевіряємо що в раніше працездатній функціональності не з’явилися помилки, викликані змінами в програмі.

**Level 2.***Чи можливе для продукту проведення тільки функціонального тестування, без перевірки нефункціональних вимог?*

Напевно можливо, але проведення виключно функціонального тестування позбавляє можливості ще покращити якість фінального продукту. Тому, на мою думку, проведення нефункціонального тестування є необхідним!!! Бо завдяки йому можна перевірити як продукт буде себе вести майже “в реальному житті”. Виявлення і виправлення дефектів у продуктивності, зручності системи або її роботи на різних пристроях\браузерах безперечно позитивно скажется на якості фінального продукту

Я можу припустити, що нефункціональне тестування можна не проводити у випадку коли незначні зміни були релізнуті та немає ресурсів провести нефункціональне тестування. Але я думаю таке можливо тільки при умові, що нефункціональне тестування проводилось вже тривалий час і рішення поставити нефункціонального тестування буде дуже короткотривалим. Але все одно, краще провести нефункціональне тестування.

*3. Яка необхідність проведення smoke (димового) тестування? Чи завжди воно є доречним?*

Димове тестування проводиться після кожного білда, щоб перевірити чи виконує ПЗ свої основні функції. Тобто, завдяки йому перевіряється стабільність ПЗ та готується підґрунтя для подальшого, більш детального тестування. Я б сказала, что димове тестування доречне в більшості випадків, бо не потребує багато часу, але з його допомогою можна швидко зрозуміти чи доречно зараз починати більш детальне тестування чи треба повертати ПЗ назад для виправлення дефектів.

**Level 3.** 1. Функціональні тест-кейси <https://docs.google.com/spreadsheets/d/12RLJ0R_XST7SgpeqjfkbhrydtQE_oPCFnpH6aDqMcT4/edit?usp=sharing>

2. Нефункціональні вимоги застосунку та перевірки, що їх перевіряють

1. Застосунок має підтримуватися наступними операційними системами на мобільних пристроях: iPhone: iOS 12 та вище (iPhone 6 та вище) та Android: 6.0 та вище - Compatibility testing
2. Застосунок має бути зручним у використанні (user-friendly) - Usability testing
3. Застосунок має працювати без збоїв одночасно на 2000 пристроях - Performance and Load testing
4. Застосунок має відповідати A11Y вимогам - Accessibility testing