**Рівень 1**

**Завдання 1**

| **Пункт** | **Функціональне тестування** | **Нефункціональне тестування** | **Пов’язане зі змінами видів тестування** |
| --- | --- | --- | --- |
| що перевіряється | * інтерфейс, * код * робочі процеси * архітектура системи * системні вимоги | * Навантаження, работа в стресових умовах, * можливість богаточасово використовувати додаток, * можливість встановлення/оновлення/видалення додатку, * зручність використання додатку, * поведінку системи після збоїв * поведінка системи після збільшення бази даних | * поведінка системи після виправлення багу * можливість міграції на іншу операційну систему * поведінка системі після додавання нових функцій |
| коли застосовується | розглядають поведінку системи у цілому | тестування того, “Як” система працює | після виправлення бага, це перевірка що проблема була дійсно вирішена |
| обмеження | можливо упущення логічних помилок у програмному забезпеченні | * для тестування база даних має бути реалістичною * складно придумати test-case для перевірки | не має |
| особливості | імітує фактичне використання системи; | забезпечує ефективну,безпечну, надійну роботу програми. | проводиться після проведення будь-яких змін у системі, може бути вузькоспрямованим як повторне або санітарне тестування, або більш об'ємне як регресійне. |

**Завдання 2**

Регресія та ретестинг

Це звучить дуже схоже, однак, коли ретестінг перевіряє, чи виправлено конкретні **відомі** дефекти, регресійне тестування шукає **непередбачені** дефекти які могли виникнути через деякі зміни, реалізовані в програмі. Регресійне тестування є невід'ємною частиною обов’язкового тестування, у той же час ретестінг виконується тільки за потреби( коли є відома помилка). За рахунок того, що ретестинг має вужчий напрямок, його складніше автоматизувати, тоді як для регресійного тестування розроблено купу автоматизованих програм.

**Рівень 2**

**Завдання 2**

Наразі я не думаю що можливо не використовувати нефункціональне тестування. Нефункціональні тести важливі, оскільки вони враховують всі проблеми з продуктивністю та безпекою додатків. При створенні мобільних додатків, сайтів треба враховувати низку важливих критеріїв, а не тільки функціональність ПО. Яка вірогідність, що користувачі будуть і далі використовувати сайт/додаток якщо він довго відкривається на пристрої або наприклад хтось може отримати твою приватну інформацію? Використання нефункціональних тестів необхідно в першу чергу для перевірки якості та надійності продукту, його привабливості для користувача.

**Завдання 3**

Якщо ми не проводимо smoke тестування на ранніх стадіях, дефекти можуть виникнути на наступних стадіях, де це може коштувати дорого. Краще витратити час на верхньорівневу швидку перевірку, ніж припуститися серйозної помилки, яка тягтиметься через весь проект. Щодо доречності, то можливо немає необхідності проводити smoke тестування якщо ми виконуємо регресійне тестування.