1.

| Параметри | Види тестування | | |
| --- | --- | --- | --- |
| функціональне | нефункціональне | пов’язане зі змінами |
| що перевіряється | всі функції системи на відповідність до документації | надійність, продуктивність, зручність та безпека | перевірка ПЗ після внесення будь-яких змін |
| коли застосовується | на всіх рівнях тестування | на системному рівні тестування | після виправлення багів чи внесення змін замовником |
| обмеження | як керівництво для тестування використовується специфікація вимог | неможливість протестувати деякі параметри вручну | проводиться тільки після установки ПЗ |
| особливості | проводиться, щоби усунути помилки, що суперечать бізнес-вимогам, проводиться в першу чергу | проводиться з позиції користувача, відповідає за поведінку системи, а не за пошук багів | використовується щоразу, як отримується стабільний білд |

2. Регресивне тестування проводиться для перевірки працездатності системи після та відсутності помилок після внесення поправок чи доповнень. Ретестінг проводиться підтвердження виправлення помилки та для перевірки функціоналу після виправлення. На відміну від регресії ретестінг проводиться в тому самому оточенні (на тому ж білді), тоді як регресивне тестування може проводитися після суттєвої зміни (або додавання) функціоналу. Регресивне тестування можна проводити паралельно з ретестінгом. При регресивному тестуванні успішні тест-кейси мають бути перевірені повторно, тоді як при ретестінгі перевіряються тільки тест-кейси зі статусом “Faild”.

3. Я вважаю, що функціональне тестування неможливе без проведення нефункціонального. Тому що кожний продукт має свого замовника. А отже і користувача цього продукту (ПЗ). Нефункціональне тестування якраз і служить для того, щоб перевірити, чи задовільняє продукт кінцевого користувача, а саме, чи є він надійним, безпечним та зручним у використанні.

4. Димове тестування я вважаю доречним для перевірки найбільш важливих функцій, щоб вирішити, чи відповідають вони очікуванням та чи варто продовжувати далі заглиблюватися у тестування, наприклад, нової версії (білда). Це потрібно для того, щоб виявити серйозні помилки якомога раніше та повернути цю сборку на доопрацювання програмістам. Воно є доречним в тому випадку, якщо вносяться якісь зміни в функціонал.

5. [Урок 5. Приклад Test Case-у – копія.xlsx](https://docs.google.com/spreadsheets/d/1eiYjCmNOjOiOhgN0ajFWYUujXvJ1-0dB/edit#gid=131938828)