**1. Перший рівень.**

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

Порівняння має містити такі блоки:

- що перевіряється;

- коли застосовується;

- обмеження;

- особливості.

| **Тип тестування** | **Що перевіряємо** | **Коли застосовуємо** | **Обмеження** | **Особливості** |
| --- | --- | --- | --- | --- |
| Функціональне | Відповідає на питання, що програма робить? Які дії програма може виконувати, з яких функцій складається та які дії виконують ці функції (логування, додавання товару у кошик тощо). | Від етапу написання коду до моменту фінального релізу. | Перевіряє тільки відповідність вимогам. | Дозволяє внести правки на етапі розробки, якщо брати Юніт та Інтеграційне тестування. Якщо розглядати системне тестування – то це можливість тестування «готового» продукту тестувальниками у ролі кінцевого користувача. Приймальне тестування зазвичай проводить сам замовник. Дає можливість отримати іншу точку зору саме від кінцевого користувача. |
| Нефункціональне | Відповідає на питання, як програма робить? Перевірка швидкості, зручності, можливість працювати на різних системах. | На етапі, коли вже є готовий об’єкт тестування. | Перевіряє тільки наскільки добре працює продукт загалом. | Допомагає перевірити, як продукт працює при певних навантаженнях, стресових сценаріях. Перевіряє юзабіліті. |
| Пов’язане зі змінами | Перевірка програми після певних змін (коли написали новий код, виправили баги). | Скоріш за все паралельно з функ./нефунк. тестуванням. Залежать від моделі тестування. | Обмеженість в часі | Допомагає більш якісно перевірити продукт, так як проводиться детальна та додаткова додаткова перевірка. |

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

В той час, коли розробники не кодять а лише правлять баги (code freeze), тестувальники перевіряють найважливішу функціональність (smoke testing). Якщо тести пройшли успішно, починається регресія (період активного тестування перед релізом). Під час регресії, виявляються та документуються баги, а коли вони знайдені – починається **ретестинг.** Тобто,ретестинг це послідовний крок після регресії. Під час ретестінгу перевіряється виправлений баг.

**2. Другий рівень.**

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

- Якщо так – в яких випадках?

- Якщо ні – чому?

На мою думку, без нефункціонального тестування, не можливо створити хороший продукт. Обидва види тестувань, допомагають зрозуміти/перевірити різні особливості системи. Нефункціональне тестування допомагає отримати інформацію про безпеку, справність та надійність продукту.

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

«Поверхневе» тестування найважливішої функціональності , перевірка ПЗ чи запускається воно та виконує основні функції. Думаю, що smoke тестування завжди доречне, так як потребую менше ресурсів і дозволяє без поглиблення перевірити чи працює продукт.