**Перший рівень**

**Завдання 1**

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

| **Тестування** | **Що перевіряється?** | **Коли застосовується?** | **Обмеження** | **Особливості** |
| --- | --- | --- | --- | --- |
| functional | проводиться тестування функцій компонента або системи. Воно відноситься до діяльності, яка перевіряє конкретну дію або функцію коду. | перевірити, чи є система функціонально досконалою. | можливість пропуску логічних помилок в додатках і високу ймовірність проведення надлишкового тестування;  Логічні помилки в програмному забезпеченні можуть бути пропущені при проведенні функціонального тестування. | Функціональне тестування імітує реальне використання системи.  Виконується в умовах, максимально наближених до умов замовника.  При проведенні функціонального тестування не робиться ніяких припущень щодо структури системи.  Легко виконується вручну. |
| Nonfunctional | Перевіряються якісні характеристики компонента або системи; | для перевірки нефункціональних параметрів, таких як надійність, навантажувальний тест, продуктивність і підзвітність програмного забезпечення. | Кожного разу, коли програмне забезпечення оновлюється, нефункціональні тести виконуються знову.  Через оновлення програмного забезпечення людям доводиться платити за повторну перевірку програмного забезпечення; таким чином, програмне забезпечення стає дуже дорогим. | Забезпечує більш високий рівень безпеки. Безпека є фундаментальною характеристикою, завдяки якій система захищена від кібератак.  Забезпечує можливість завантаження системи таким чином, що будь-яка кількість користувачів може використовувати її одночасно.  Це підвищує продуктивність системи.  Тестові кейси ніколи не змінюються, тому не потрібно писати їх більше одного разу.  Загальні витрати часу менші порівняно з іншими процесами тестування. |
| Тестування, пов’язане зі змінами | Призначене для перевірки виправлення дефектів і перевірки працездатності системи після внесення змін. | після установки програмного забезпечення, для підтвердження працездатності програми | **Smoke testing** тестування поверхні, навмисне упущення частини функціоналу під час покриття.  **Regression testing**  збільшення кількості часу та активності на проекті;  Втрата концентрації під час постійного повторного тестування | **Smoke testing**  Відносно низькі показники часу, витраченого на тестування.  Пріоритетом є забезпечення роботи основного функціоналу.  **Regression testing**  гарантія що зміни в новій версії програми не пошкодять існуючу функціональність;  Прискорити повторні тести;  Гарантія виявлення дефектів на ранніх стадіях розвитку |

**Завдання 2**

2. Поясни, в чому різниця між регресією та ретестингом.

**Re-Test** : проводиться в разі, якщо фіча/функціональність уже мала дефекти, і ці дефекти було нещодавно виправлено.

**Регресійні тести**: власне те, що займає левову частку часу і для чого існує автоматизація тестування. Проводять регресійне тестування AUT тоді, коли потрібно переконатися, що нові (додані) функції застосунку / виправлені дефекти не вплинули на поточну, вже наявну функціональність, що працювала (і протестовано) раніше.

**Другий рівень**

**Завдання 2**

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

На мою думку, проведення функціонального тестування, без перевірки нефункціональних вимог - так, можливе. Функціональні тести пишуться, ґрунтуючись на функціональних вимогах, які можна знайти в специфікаціях, бізнес-вимогах, user story, use case тощо.

Часто буває, що нефункціональні вимоги можуть не описуватися, але це не означає, що їх не існує! Оцінка швидкості роботи системи, зручності, кросплатформенності, безпеки - все це потрібно тестувати, тому що ці характеристики дуже сильно впливають на якість.

Мало хто захоче користуватися сайтом, що "лагає", вічно "лежить" або не є безпечним, навіть якщо він супер-пупер функціональний.

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

**Димові тести**: виконуються кожного разу, коли ми отримуємо новий білд (версію), проекту (системи) на тестування, вважаючи її відносно нестабільною. Нам потрібно переконатися, що критично важливі функції AUT (Application Under Test) працюють відповідно до очікувань. Ідея цього виду тестування полягає в тому, щоб виявити серйозні проблеми якомога раніше, і відхилити цей білд (повернути на доопрацювання) на ранньому етапі тестування, щоб не заглиблюватися в довгі та складні тести, не витрачаючи тим самим час на браковане ПЗ.

**Третій рівень**

**Завдання 3**

Напиши, які нефункціональні вимоги ти хотів/ла б застосувати для продукту твого стартапу.

Опиши перевірки, які б їх перевіряли (3-5 прикладів).

1. Вимоги до апаратури та ПЗ: IOS версія 11.0 і вище; IPhone 7 і старше;

Перевірка буде заключались в тому, що буду перевіряти на різних версіях IOS і різних моделях телефона. Переконатись що додаток не може бути встановлений на непідтримувані пристрої.

1. Слайд шоу світлин котиків в фотогалереї безперервно працює, допоки не буде зупинено мануально.

Перевірка заключається в тому, що вімкну слайд-шоу на тривалий період часу, щоб впевнитись, що зупинити можливо лише вручну.

1. Додаток повинен по закону відповідати правилам GDPR - загальний регламент захисту даних.

Вивести консоль api запросів по пошуку користувачів і впевнитись, що в відповіді сервера немає приватних даних, таких як номер телефона, email.