Урок номер 6

| Види тестування | Що перевіряється | Коли застосовується | Обмеження | Особливості |
| --- | --- | --- | --- | --- |
| Функціональне | Чи працює функціонал системи відповідно до вимог | На ранніх та середніх етапах розробки | Може не враховувати продуктивність , або зручність | Орієнтована на вимоги бізнес -вимоги, або дії користувача |
| Нефункціональне | Якість роботи системи: швидкість , безпека, зручність | Після перевірки основного функціоналу, або паралельно | Може бути складніше виміряти, потребує спеціальних інструментів | Оцінює наскільки система відповідає вимогам продуктивності , та сумісність з різними пристроями. |
| тестування пов'язане зі змінами | Чи після змін у коді збережена стабільність | Після виправлення багів, або додавання функцій | Потребує повного розуміння змін у системі | Використовуються regression , retesting, smoke |
|  |  |  |  |  |
|  |  |  |  |  |

2)

Ретестінг - це повторне тестування бага , регресійне тестування - це повторне тестування всієї системи після внесених змін. Ретестинг застосовується для конкретних дій, Регресія для охоплює ширший обсяг, навіть якщо баг був усунений.

3)

Чи можливе для продукту проведення лише функціонального тестування без перевірки нефункціональних вимог? Обґрунтування

У деяких рідкісних випадках допустимо проводити лише функціональне тестування, без перевірки нефункціональних вимог. Наприклад, це можливо на ранніх етапах створення демо версії , яка має лише основні функції. У такій ситуації важливо просто переконатися, що базові дії працюють (наприклад, кнопки, реєстрація, додавання товару).

Проте у більшості випадків обмеження лише функціональним тестуванням є недоцільним, оскільки нефункціональні характеристики (швидкодія, безпека, зручність інтерфейсу) безпосередньо впливають на користувацький досвід. Якщо не перевірити їх вчасно, користувачі можуть зіткнутися з проблемами після релізу — що призведе до втрати довіри або додаткових витрат на виправлення.

Отже, повноцінне тестування має включати як функціональні, так і нефункціональні вимоги, щоб забезпечити стабільну, зручну та якісну роботу продукту.

4)

Smoke-тестування — це як швидка перевірка, що все в додатку хоча б запускається і працює в загальному.

Наприклад, чи відкривається сайт, чи працює кнопка «увійти», чи не вилітає одразу система. Це наче «перша перевірка на дим» — якщо щось не так, далі тестувати немає сенсу.

Таке тестування дуже корисне після кожного оновлення чи зміни в коді. Воно допомагає не витрачати час дарма. Але бувають випадки, коли smoke можна не робити — наприклад, якщо зміна зовсім дрібна (наприклад, поміняли текст на сторінці або колір кнопки).

Напиши 5 функціональних тест-кейсів, які перевіряли б роботу застосунку.

| **Test Case ID** | | TC1 Завантаження фото кота  TC2 Заборона завантаження фото собаки  TC3 Додавання друга  TC4 Ставлення вподобайки  TC5  Додавання коментаря | **Test Case Description** | | TC1  Перевірка, що користувач може завантажити фотографію кота, і вона відображається у стрічці.  TC2  Перевірка обмеження — система не дозволяє завантажити фото, на якому немає кота.  TC3  Перевірка функції додавання іншого користувача в друзі через його профіль.  TC4  Перевірка, що користувач може поставити вподобайку на фото іншого користувача.  TC5  Перевірка можливості залишити коментар під фото і його відображення у стрічці. | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Created By** | | Elizaveta Kovalenko | **Reviewed By** | | Elizaveta Kovalenko | | **Version** | | 1.0 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Єлизавета Романівна | **Date Tested** | | 3.05.25 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| TC1 | – Встановлений застосунок на пристрої  – Користувач зареєстрований та увійшов у свій обліковий запис  – Дозвіл на доступ до галереї або камери | | |  | TC1 | – Назва файлу: cat\_photo.jpg  – Формат: .jpg  – Розмір: 1.2 MB  – Вміст: фото лише кота (без інших тварин чи людей) | | | | |
| TC2 | – Встановлений застосунок  – Фото з котом підготовлене для завантаження  – Увійдений обліковий запис | | |  | TC2 | – Назва файлу: cat\_photo.jpg  – Формат: .jpg  – Розмір: 1.2 MB  – Вміст: фото лише кота (без інших тварин чи людей) | | | | |
| TC3 | Є інший користувач в застосунку  – Користувач перейшов у профіль іншого користувача | | |  | TC3 | – Ім’я користувача: catlover123  – ID користувача: 987654 | | | | |
| TC4 | – Є фотографія іншого користувача у стрічці  – Користувач має доступ до неї | | |  | TC4 | – Ім’я користувача: catlover123  – ID користувача: 987654 | | | | |
| TC5 | .  – Публікація фото доступна  – Авторизація виконана  – Увімкнений інтернет | | |  | TC5 | – ID фото: catphoto\_002  – Текст коментаря: "Який милий кіт!” | | | | |
| **Test Scenario** | TC1  Перевірити, що користувач може успішно завантажити фото котика відповідного формату та розміру. | TC2  Перевірити, що система блокує завантаження фото, яке не відповідає вимогам (інша тварина, людина, об’єкти). | TC3  Перевірити, що користувач може знайти іншого користувача та додати його в д | TC4  Перевірити, що користувач може поставити “вподобайку” під фото котика | TC5  Перевірити, що користувач може залишити коментар під фото котика. |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Pass / Fail / Not executed / Suspended** | | |  |  |  |
|  |  |  |
| TC1 – Завантаження фото | 1. Відкрити застосунок  2. Перейти до розділу «Додати фото»  3. Вибрати фото кота з галереї  4. Натиснути кнопку «Завантаж | | Фото котика успішно з’являється у стрічці публікацій | | Pass | | |  |  |  |
| TC2 – | 1. Відкрити застосунок  2. Перейти до розділу «Додати фото»  3. Вибрати фото людини/іншої тварини  4. Натиснути кнопку «Завантажити | | З’являється повідомлення про помилку — фото не відповідає вимогам | | Pass | | |  |  |  |
| TC3 – Додавання | 1. Увійти в акаунт  2. Перейти до пошуку користувачів  3. Ввести ім’я або логін  4. Натиснути кнопку «Додати в друзі» | | Запит на дружбу успішно надіслано, кнопка змінюється на «Очікує підтвердження» | | Pass | | |  |  |  |
| TC4 – Ставлення вподобайки | 1. Увійти в акаунт  2. Перейти до стрічки публікацій  3. Знайти публікацію з фото котика  4. Натиснути на іконку “серце” | | Кількість лайків збільшується на 1, сердечко підсвічується | |  | | |  | | |
| TC5 – Додавання коментаря | 1. Увійти в акаунт  2. Відкрити будь-яке фото котика  3. Ввести текст у полі коментаря  4. Натиснути кнопку «Надіслати» | | Коментар з’являється під фото, видно ім’я користувача та текст | |  | | |  | | |
|  |  | |  | |  | | |  | | |

приклади нефункціональних вимог та перевірки для кожної з них:

1. Продуктивність

Вимога: Сторінка курсу має завантажуватись не довше ніж за 3 секунди при середньому навантаженні.

Перевірка: Виконати тест навантаження та зафіксувати час завантаження при одночасному доступі 100 користувачів.

2. Надійність

Вимога: Платформа повинна мати доступність не менше 99,9% часу на місяць.

Перевірка: Налаштувати моніторинг і перевірити статистику за місяць.

3. Масштабованість

Вимога: Система має підтримувати збільшення кількості користувачів без помітного зниження продуктивності.

Перевірка: Провести тестування з поступовим збільшенням кількості одночасних користувачів і перевірити стабільність часу відповіді.

4. Безпека:

Вимога: Всі дані користувачів повинні передаватися лише через зашифровані з'єднання

Перевірка: Виконати перевіркуі провести тестування безпеки

5. Зручність використання

Вимога: Новий користувач повинен знайти і пройти перший урок менш ніж за 3 хвилини без інструкції.

Перевірка: Провести юзабіліті-тестування з новими користувачами і виміряти час виконання завдання.