

7 ЛАБОРАТОРНА РОБОТА №7 ТЕСТ-ДИЗАЙН ТА ТЕСТ-КЕЙСИ

Мета роботи: ознайомлення з техніками тест-дизайну, отримання базових теоретичних та практичних навичок в групуванні тест-кейсів, здобуття навичок створення тестових випадків (*test cases*).

7.1 Короткі теоретичні відомості

Тест-дизайн (*test design*) – це етап процесу тестування програмного забезпечення (ПЗ), на якому проєктуються і створюються тестові випадки (тест-кейси), відповідно з певними раніше визначеними критеріями якості і цілями тестування [1].

Тестовий випадок або тест-кейс (*test case*) – це сукупність кроків, конкретних умов і параметрів, необхідних для перевірки реалізації функції, що тестується, або її частини.

Тестовий набір (*test suite*):

- набір тестів, що реалізують бізнес-завдання, виконуване системою, що тестується;
- тестовий набір включає тестові сценарії, тестові дані або правила їх генерації [2].

До стандартних атрибутів тест-кейса відносять:

- **ID (*Test Case ID*)** – унікальний ідентифікатор необхідний для зручної організації зберігання і навігації у групі тестів (*Test Suite*), переважно генерується автоматично;
- **назва** – основна тема чи ідея тест-кейса, короткий опис його суті, який дозволяє зрозуміти призначення тест-кейса. Може повторювати тему. Поле обов'язкове для заповнення;
- **тема (*Summary*)** – поле опису основної теми, яка лаконічно і точно описує його суть. Для уніфікованого підходу до викладу теми використовують конструкцію, яка відповідає на питання «Що перевіряємо? Де перевіряємо? З якими даними?». Поле обов'язково для заповнення;

– **попередні умови (Preconditions)** – список всіх необхідних підготовчих дій (налаштування програми, середовища тестування) для виконання даного тест-кейса;

– **кроки для відтворення (Step actions)** – описують послідовність дій для відтворення тестового випадку, які повинні привести до очікуваного результату. Повинні бути короткими і зрозумілими;

– **очікувані результати (Expected results)** – визначають правильну реакцію програми на виконання даних кроків. Повинні бути зрозумілими, однозначними, простими;

– **історія редагування** – лаконічний журнал змін, де відображено ким, як і коли був змінений тест-кейс [1].

Тестові випадки поділяються за очікуваним результатом на **позитивні** та **негативні**.

Позитивний тест-кейс використовує тільки валідні дані і перевіряє, що додаток правильно виконав функцію, що викликається.

Негативний тест-кейс оперує як валідними, так і невалідними даними (мінімум 1 некоректний параметр) і ставить за мету перевірку виняткових ситуацій (спрацьовування валідаторів).

Тестові випадки зручно об'єднувати **за призначенням**:

1. позитивні кейси:

- перевірка функціоналу;
- перевірка дизайну/UI;
- перевірка безпеки.

2. негативні кейси [2].

При складанні тест-кейсів необхідно дотримуватися наступних принципів:

– тему тест-кейса необхідно описувати за принципом «Що перевіряється? Де перевіряється? З яким типом даних (валідні, невалідні)?»;

– наявності опису тест-кейса;

– наявність попередніх умов, де має бути вказано, яка форма, сторінка відкрита, але посилання слід вказувати тільки на головний домен сайту – не на конкретну сторінку;

– на кожен крок повинен бути очікуваний результат;

– в кроках має бути зазначений тип даних, що вводяться (валідні/ невалідні), у разі невалідних – повинні бути наведені приклади таких даних (наприклад, для поля введення електронної пошти невалідними будуть неприпустимі спеціальні символи, без @, без домена, та ін.);

– тест-кейс повинен бути завершеним і цілісним (наприклад, функція відновлення пароля повинна закінчуватися на успішній зміні пароля, а не на відправці листа з посиланням для зміни пароля) [1].

7.2 Завдання до роботи

7.2.1 Створити 6 тест-кейсів для сайту <https://www.automationexercise.com/> для наступного функціоналу:

- форма реєстрації нового користувача;
- форма авторизації користувача;
- форма «Contact Us».

Для кожної форми має бути по 2 тест-кейси – один з позитивним сценарієм (з валідними даними) і один з негативним сценарієм (з невалідними даними).

Шаблон тест-кейса «*Тест_кейси.xlsx*» завантажити з навчальної системи Moodle.

7.2.2. Для зареєстрованого користувача створити 4 тест-кейси для сайту <https://www.automationexercise.com/> для функціоналу роботи з кошиком:

- додавання товару;
- перевірка кількості доданого товару;
- видалення товару;
- оформлення замовлення.

7.2.3 Скриншоти створених тест-кейсів додати до звіту.

7.2.4 Оформити звіт з лабораторної роботи та надати його на перевірку.

7.3 Зміст звіту

7.3.1. Тема та мета роботи.

7.3.2. Завдання до роботи.

7.3.3. Короткі теоретичні відомості.

7.3.4. Хід роботи з виконання завдання до лабораторної роботи.

7.3.5. Висновки, що відображують результати виконання роботи та їх критичний аналіз.

7.3.6 Перелік використаних джерел.

7.4 Контрольні запитання

7.4.1. Дайте визначення поняттю тест-кейс та вкажіть його основні атрибути.

7.4.2. Вказати причини того, чому спочатку проводиться позитивне тестування, а потім негативне.

7.4.3. В чому полягає важливість використання тестових випадків у тестуванні ПЗ?

7.4.4. Вказати переваги та недоліки використання тестових випадків у тестуванні ПЗ на прикладі.

7.4.5. В чому полягає різниця між тест-кейсом та звітом про дефект?

7.5 Перелік джерел посилання

1 Що таке цілісність тест-кейсу або як правильно описати тест-кейс, щоб перевірити функціонал? URL: <https://training.qatestlab.com/blog/course-materials/test-case-description/> (дата звернення 30.08.2025).

2 How to Write Test Cases in Software Testing with Examples. URL: <https://www.guru99.com/test-case.html> (дата звернення 30.08.2025).