

**Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»**



ЗВІТ

про виконання лабораторної роботи № 3

з дисципліни

«Аналіз програмного забезпечення»

тема роботи:

Написання тест-кейсів (Test Case).

Виконав(ла): ст. гр. 121 – 22- 2

Скороход Роман Іванович

Прийняла: асистентка,
Шевченко Ю.О.

**Дніпро
2025**

Практична робота №3

Тема: Написання тест-кейсів (Test Case)

Мета: Набування навичок у написанні тест-кейсів різних пристроїв

Очікувані результати навчання: уміння підписувати особисту документацію з використанням єдиного цифрового підпису за допомогою різних сервісів і додатків.

Короткі теоретичні відомості

Тест-кейс – це професійна документація тестувальника, це послідовність дій, спрямована на перевірку будь-якого функціоналу, що описує як прийти до очікуваного результату.

Тема кейса – описова назва тесту, яка спрощує його пошук та розуміння його змісту.

У темі тест-кейсі не повинно бути залежностей від інших тест-кейсів і також нечітких формулювань та зайвої деталізації.

Навички, необхідні для написання тест-кейсів:

- *Уміння збирати та аналізувати вимоги до продукту.* Якщо немає формально описаних вимог (специфікацій) – потрібно вміти їх збирати у розробників, аналітиків, користувачів, тощо.

- *Уміння розділяти систему на складові, тобто робити декомпозицію.* Тобто, потрібно не тільки вміти бачити систему як ціле, а й вміти розкласти її на складові частини.

Це дуже корисна навичка для проведення функціонального тестування, де перевіряється кожна складова продукту.

- *Уміння розставляти пріоритети.* Тест-дизайнер повинен вміти

відрізняти важливіше від менш важливого, а також розставляти пріоритети тестування.

- *Уміння формулювати свої думки (письмово та усно).* Це вміння важливе для тестувальника в принципі. Тест-дизайнеру воно дуже допомагає при створенні тест-кейсів.

Знання технік тест-дизайну, Результатом виконання тест кейсу можуть бути:

- Позитивний результат (**pass**) – фактичний результат дорівнює очікуваному результату.
- Негативний результат (**fail**) – фактичний результат не дорівнює очікуваному результату. У цьому випадку знайдено помилку.
- Виконання тесту заблоковано (**blocked**) – після одного з кроків продовження тесту неможливе. У цьому випадку також знайдено помилку.

Важливі моменти при складанні тест-кейса:

Розуміння вимоги, за якою складається кейс

Скласти й описати ефективний тест-кейс не вийде, якщо немає розуміння, яким саме чином повинен бути реалізований той чи інший функціонал програми. Ефективніше буде витратити сили та час на уточнення неточностей і в підсумку написати «робочий» тест-кейс, ніж спершу скласти абсолютно неправильний, зрозуміти в процесі тестування, що він не спрацював як треба, знову витратити час на уточнення, і тільки після цього повністю переробити та отримати робочий варіант.

Простота та легкість для розуміння

Важливо, щоб тест-кейс був описаний зрозумілими словами, без використання спеціальної термінології та складних мовних конструкцій, так як його можуть проходити фахівці різного рівня кваліфікації та спеціалізації

(досвідчені тестувальники та початківці, керівники проекту, розробники, аналітики та навіть співробітники замовника, не завжди володіють достатнім рівнем знань у галузі комп'ютерних технологій).

Окрема увага приділяється наборам тестових даних

Якщо тест передбачає використання будь-яких даних, слід звернути увагу на правильність їх складання та подальші вказівки. В іншому випадку такий тест є марним

Завдання.

Придумайте об'єкт тестування. Це повинен буде будь-який об'єкт, який складається мінімум з 5 частин. (годинник, скейт, велосипед, монітор і т.п.) Напишіть не менше 20 тест кейсів до цього об'єкту, що зможуть оцінити його якість з точки зору продукту. Створіть *.pdf документ, де кратко опишіть ваш об'єкт та потім опишіть ваші тест-кейси.

Приклад Test Case: «Обчислення $1 + 1$ в калькуляторі».

Назва: Перевірка обчислення $1 + 1$.

Pre-condition: Відкрито стандартний калькулятор, система готова до введення виразів.

Кроки:

- Натиснути кнопку "1".
- Натиснути кнопку "+".
- Натиснути кнопку "1".
- Натиснути кнопку "=".

Expected Result: Відображається число 2

Post-condition: Екран калькулятора показує результат, система готова

до наступного обчислення.

Виконання роботи

1.Спочатку треба обрати об'єкт для тест кейсів, у нас це буде велосипед .

2.Наводимо опис об'єкта

Велосипед



Велосипед — це транспортний засіб, який приводиться в рух мускульною силою людини через педалі. Призначений для пересування по дорогах, прогулянок, спорту та подорожей. Складається з механічних і конструктивних елементів, які забезпечують стабільність, керованість і безпечну експлуатацію.

Основні частини велосипеда:

1. Рама — основна несуча конструкція, до якої кріпляться всі інші елементи.
2. Кермо — елемент керування напрямком руху.
3. Колеса — забезпечують рух велосипеда; складаються з обода, спиць, втулки та покришки.
4. Педалі і система приводу — передають зусилля ніг на заднє колесо через ланцюг і зірочки.
5. Гальмівна система — забезпечує зупинку або зменшення швидкості.
6. Сидіння — забезпечує комфортне положення під час їзди.
7. Передача — дозволяє змінювати зусилля для підйомів або швидкої їзди.
8. Ланцюг — з'єднує систему передачі з заднім колесом.
9. Світловідбивачі або фари — підвищують безпеку під час руху в темний час.
10. Підніжка — дозволяє ставити велосипед у вертикальне положення без опори.

3. Тепер створюємо тест кейси

Test cases

Test Case 1

Назва: Перевірка обертання педалей

Pre-condition: Велосипед стоїть на підставці, ланцюг натягнутий.

Кроки:

- Обернути педалі руками.

Expected Result: Колеса починають обертатися.

Post-condition: Привід працює коректно.

Test Case 2

Назва: Перевірка гальмівної системи переднього колеса

Pre-condition: Велосипед рухається повільно.

Кроки:

- Натиснути передній гальмівний важіль.

Expected Result: Переднє колесо зупиняється, велосипед сповільнюється.

Post-condition: Гальма працюють ефективно.

Test Case 3

Назва: Перевірка гальмівної системи заднього колеса

Pre-condition: Велосипед рухається.

Кроки:

- Натиснути задній гальмівний важіль.

Expected Result: Заднє колесо зупиняється плавно, велосипед сповільнюється.

Post-condition: Гальма працюють справно.

Test Case 4

Назва: Перемикання передач

Pre-condition: Велосипед стоїть, ланцюг змазаний.

Кроки:

- Натиснути перемикач швидкості на кермі.

Expected Result: Ланцюг переміщується на іншу зірочку.

Post-condition: Передача змінена, перемикання без шуму.

Test Case 5

Назва: Перевірка стабільності рами

Pre-condition: Велосипед стоїть на рівній поверхні.

Кроки:

- Легко натиснути на раму зверху.

Expected Result: Відсутні люфти або скрипи.

Post-condition: Рама стабільна, без пошкоджень.

Test Case 6

Назва: Регулювання висоти сидіння

Pre-condition: Велосипед стоїть нерухомо.

Кроки:

- Послабити фіксатор сидла.
- Підняти або опустити сидіння.
- Зафіксувати положення.

Expected Result: Сідло надійно зафіксоване, не рухається.

Post-condition: Позиція сидла збережена.

Test Case 7

Назва: Перевірка повороту керма

Pre-condition: Велосипед стоїть на місці.

Кроки:

- Повернути кермо вліво і вправо.

Expected Result: Колесо плавно повертається, без блокування.

Post-condition: Кермо повертається у вихідне положення.

Test Case 8

Назва: Перевірка накачування шин

Pre-condition: Шини спущені.

Кроки:

- Приєднати насос до вентиля.
- Накачати до тиску 3 бар.

Expected Result: Шини пружні, не спускають повітря.

Post-condition: Велосипед готовий до поїздки.

Test Case 9

Назва: Перевірка світловідбивачів

Pre-condition: Велосипед у темному приміщенні.

Кроки:

- Посвітити ліхтариком на світловідбивачі.

Expected Result: Вони яскраво відбивають світло.

Post-condition: Видимість на дорозі підвищена.

Test Case 10

Назва: Перевірка дзвінка

Pre-condition: Дзвінок встановлений на кермі.

Кроки:

- Натиснути кнопку дзвінка.

Expected Result: Чіткий гучний звук.

Post-condition: Дзвінок справний.

Test Case 11

Назва: Перевірка руху без сторонніх звуків

Pre-condition: Велосипед змащений, всі деталі затягнуті.

Кроки:

- Проїхати 10 метрів.

Expected Result: Відсутні скрипи, клацання, брязкіт.

Post-condition: Робота механізмів плавна.

Test Case 12

Назва: Перевірка переднього колеса

Pre-condition: Велосипед стоїть на місці.

Кроки:

- Обернути переднє колесо вручну.

Expected Result: Колесо обертається вільно, без тертя.

Post-condition: Вісь колеса справна.

Test Case 13

Назва: Перевірка заднього колеса

Pre-condition: Велосипед стоїть на підставці.

Кроки:

- Обернути педалі вперед.

Expected Result: Заднє колесо обертається рівномірно.

Post-condition: Привід працює правильно.

Test Case 14

Назва: Перевірка ланцюга

Pre-condition: Велосипед стоїть.

Кроки:

- Візуально оглянути ланцюг.
- Перевірити натяг.

Expected Result: Ланцюг змащений, не провисає.

Post-condition: Передача крутного моменту стабільна.

Test Case 15

Назва: Перевірка роботи підніжки

Pre-condition: Велосипед стоїть без опори.

Кроки:

- Відкинути підніжку.
- Залишити велосипед стояти.

Expected Result: Велосипед стоїть вертикально, не падає.

Post-condition: Підніжка тримає вагу велосипеда.

Test Case 16

Назва: Перевірка освітлення

Pre-condition: У велосипед встановлено фару.

Кроки:

- Увімкнути фару.

Expected Result: Світло яскраве, спрямоване вперед.

Post-condition: Освітлення працює справно.

Test Case 17

Назва: Перевірка передач під навантаженням

Pre-condition: Велосипед у русі.

Кроки:

- Перемкнути швидкість під час педалювання.

Expected Result: Передача змінюється плавно, без зривів ланцюга.

Post-condition: Передача коректно перемикається.

Test Case 18

Назва: Перевірка стійкості велосипеда

Pre-condition: Велосипед стоїть на рівній поверхні.

Кроки:

- Легко штовхнути з боку.

Expected Result: Велосипед не падає одразу.

Post-condition: Центр ваги збалансований.

Test Case 19

Назва: Перевірка руху по нерівній поверхні

Pre-condition: Велосипед на гравійній дорозі.

Кроки:

- Проїхати 20 метрів.

Expected Result: Велосипед зберігає стійкість, кермо не вібрує сильно.

Post-condition: Ходова частина працює стабільно.

Test Case 20

Назва: Перевірка цілісності гальмівних тросів

Pre-condition: Велосипед у спокійному стані.

Кроки:

- Візуально оглянути троси.
- Натиснути на гальмівні ручки.

Expected Result: Троси не розтягнуті, натяг збережений.

Post-condition: Гальма функціонують без затримок.

Висновок: У ході виконання лабораторної роботи було обрано об'єкт тестування — велосипед. Були створені 20 тест-кейсів, що охоплюють основні функціональні елементи конструкції: систему приводу, гальмівну систему, освітлення, керування, стійкість та комфорт користування. Кожен тест-кейс має опис умов, кроків виконання, очікуваного результату та постумов. Результатом виконання роботи стало набуття практичних навичок у створенні тест-кейсів, що дозволяють перевіряти якість продукту з точки зору користувача та інженера з тестування. Отримані знання можуть бути застосовані під час тестування як фізичних об'єктів, так і програмних систем.

Відповіді на контрольні запитання

1. Навіщо потрібні тест-кейси?

Тест-кейси потрібні для перевірки правильності роботи системи або продукту відповідно до вимог. Вони дозволяють систематизувати процес тестування, уникнути пропуску важливих перевірок, відтворювати знайдені помилки та підтверджувати їх виправлення. Тест-кейси забезпечують прозорість тестування і є основою для звітності про якість продукту.

2. Основні атрибути Test Case:

- Назва (Title) — короткий опис мети тесту.
- Pre-condition — умови, які повинні бути виконані перед тестом.
- Кроки (Steps) — послідовність дій, що виконуються під час тестування.

- Expected Result — очікуваний результат виконання тесту.
- Post-condition — стан системи після завершення тесту.
- (Додатково можуть бути: ID, пріоритет, тип, статус, автор, дата створення тощо.)

3. Типи тест-кейсів:

- Позитивні — перевіряють, що система працює правильно при коректних даних.
- Негативні — перевіряють, як система реагує на некоректні або несподівані дії.
- Функціональні — перевіряють виконання вимог і функцій.
- Нефункціональні — перевіряють продуктивність, зручність, безпеку тощо.
- Регресійні — перевіряють, що після змін у коді старий функціонал не зламався.
- Smoke-тести — базові тести для перевірки, що система взагалі запускається та працює.

4. Що таке негативний тест-кейс?

Негативний тест-кейс — це сценарій, який перевіряє, як система поводить себе при некоректних діях користувача або неправильних вхідних даних. Його мета — виявити помилки, запобігти збоєм і перевірити надійність системи.

Приклад: у полі «вік» ввести текст замість числа — система має видати повідомлення про помилку.

5. Що повинен знати тестувальник?

Тестувальник повинен знати:

- основи життєвого циклу розробки ПЗ (SDLC);
- методи тестування та техніки тест-дизайну;
- принципи роботи з баг-трекерами, системами контролю версій, тест-менеджмент інструментами;

- основи аналітики та складання документації (тест-кейси, чек-листи, звіти);
- базові знання програмування або SQL для перевірки даних;
- уважність, логічне мислення та комунікаційні навички.

6. Скільки основних принципів тестування?

Існує 7 основних принципів тестування програмного забезпечення:

1. Тестування демонструє наявність дефектів, але не їх відсутність.
2. Вичерпне тестування неможливе.
3. Раннє тестування економить час і ресурси.
4. Дефекти концентруються в певних модулях.
5. Парадокс пестициду — тести треба регулярно оновлювати.
6. Тестування залежить від контексту.
7. Відсутність дефектів не означає якість продукту.