

АНАЛІЗ МЕТОДУ ЧЕК-ЛІСТІВ ДЛЯ ТЕСТУВАННЯ ГРАФІЧНОГО ІНТЕРФЕЙСУ

Тестування ПЗ – це процес перевірки відповідності реальної та очікуваної поведінки програмного продукту, що за звичай здійснюється на кінцевому етапі розробки програмного продукту. Також можна сказати, про тестування – є одним із видів контролю якості готового продукту, до якого входить проектування робіт, складання тестів, здійснення тестування та аналіз отриманих результатів [1].

Інтерфейс користувача (ІК) – це інтерфейс між комп'ютером і користувачем, який використовує піктограми, меню та вказівний засіб для вибору функцій для виконання команд.

Користувач зазвичай сприймає лише зовнішній вигляд програмного продукту, тому інтерфейс повинен забезпечувати прийнятний рівень взаємодії користувача з функціоналом інтерфейсу програмного продукту, тобто бути зручним, естетично привабливим та простим для розуміння. Часто саме через зручність графічного інтерфейсу користувач приймає рішення щодо вибору того чи іншого програмного продукту серед існуючих аналогів для постійного користування [2,3]. Тому тестування графічного інтерфейсу користувача є важливою задачею.

Основними об'єктами тестування графічного інтерфейсу є коректне розміщення елементів управління на екранних формах (верстка), граматична та стилістична коректність контенту та текстових повідомлень, які виводяться на екран, формати вводу даних, реакція системи на введення даних користувачем, а також час відгуку системи на запити користувача.

Основним підходом до тестування програмних продуктів є складання тест-кейсів [4] – набору вхідних даних, умов виконання та очікуваних результатів, розроблених з метою перевірки тієї чи іншої властивості або поведінки програмного продукту.

Розрізняють високорівневий та низькорівневий тест-кейси. Високорівневий тест-кейс, на відміну від низькорівневого тест-кейсу не містить конкретних вхідних даних і очікуваних результатів.

У випадку тестування графічного інтерфейсу доцільніше використовувати високорівневі тест-кейси, які можуть бути зведені до конкретних пунктів чек-ліста. Тому часто серед тестувальників побуває думка, що складання тест-кейсів для тестування графічного інтерфейсу є марною витратою часу, оскільки тривалість оформлення навіть високорівневого тест-кейсу є значно вищою, ніж використання звичайного чек-ліста.

Чек-ліст [5,6] – це документ, який містить в собі варіанти ідей для покрокового тестування ПЗ, тому дозволяє вносити зміни в список, безпосередньо в ході виконання тестування. Детальність чек-ліста може залежати від кількості тестувальників, які інформують про всі події, під час роботи та складності готового програмного продукту.

Чек-ліст може бути оформлений у вигляді:

- 1) Списку, в якому послідовність пунктів не має значення (наприклад, список значень якогось поля).
- 2) Списку, в якому послідовність пунктів важлива (наприклад, кроки в короткій інструкції).
- 3) Структурованим (багаторівневим) списком, що дає змогу побачити ієрархію ідей.

Важливо зрозуміти, що не існує обмежень при розробці чек-лістів – головне, щоб вони допомагали в роботі. Іноді чек-лісти можуть виражатися графічно (наприклад з використанням ментальних карт або концепт-карт), хоча зазвичай їх вигляд – це багаторівневий список.

Оскільки в різних проєктах зустрічаються однотипні завдання, доб-ре продумані та акуратно оформлені чек-лісти можуть використовуватися повторно, що значною мірою економить ресурси.

Для того, щоб чек-ліст був дійсно корисним інструментом, він повинен володіти рядом важливих властивостей:

1) Логічність. Чек-ліст складається на основі наявних цілей і для вирішення поставленої задачі. Іноді початківців припускаються помилки, перевантажуючи чек-ліст недоцільними думками, які ніяк не пов'язані одна з одною.

2) Послідовність і структурованість, що досягається за шляхом оформлення чек-ліста у вигляді багаторівневого списку, в якому послідовність існує навіть у випадках, коли пункти чек-ліста не описують ланцюжок дій, оскільки людині зручніше сприймати інформацію у вигляді невеликих груп ідей, перехід між якими є зрозумілим і очевидним.

3) Повнота і ненадмірність. Чек-ліст повинен являти собою акуратну сукупність ідей, в яких ніщо важливе не втрачено.

Правильно створювати та оформляти чек-лісти також допомагає сприйняття їх не тільки як сховища наборів ідей, але і як «вимоги для складання складних тест-кейсів». З цього виходить, що у більшості випадків чек-ліст використовується тестувальником, щоб задокументувати всі ідеї та думки в голові, так як дуже часто продукти достатньо складні, з великою кількістю функцій та перевірок. Однак, сьогодні для типових програмних продуктів розроблено чимало чек-лістів, тому за основу можна брати готовий чек-ліст, але адаптувавши його під особливості свого продукту. Такий підхід значно економить ресурси на тестування та дозволить ознайомитись з досвідом інших тестувальників.

У результаті проведеного аналізу, можна стверджувати, що створення якісного тест-кейса може вимагати

тривалої копіїткої і досить монотонної роботи, яка при цьому не вимагає від досвідченого тестувальника особливих інтелектуальних зусиль, на відміну від складання чек-листів, які повинні відображати у лаконічній формі всі аспекти тестування графічного інтерфейсу користувача.

Список використаних джерел

1. Тестирование. Фундаментальная теория [Електронний ресурс] – Режим доступу до ресурсу: <https://dou.ua/forums/topic/13389/>
2. Інтерфейс користувача [Електронний ресурс] – Режим доступу до ресурсу: https://wiki.cuspu.edu.ua/index.php/Інтерфейс_користувача
3. Кухарчук С.В. Аналіз методів тестування зручності використання веб-сайтів [Електронний ресурс] / С.В. Кухарчук, О.В. Романюк // Матеріали XLVI науково-технічної конференції підрозділів ВНТУ, Вінниця, 22-24 березня 2017 р. – Режим доступу до ресурсу: <https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/17318/3215.pdf?sequence=3>
4. Тест кейсы [Електронний ресурс] – Режим доступу до ресурсу: <https://zhenyabrik.gitbooks.io/matrixcopy/lektsiya-4-ch3-test-keisi.html>
5. ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ БАЗОВЫЙ КУРС [Електронний ресурс] – Режим доступу до ресурсу: https://careers.epam.by/content/dam/epam/by/book_epam_by/Software_Testing_Basics_2_izdanie.pdf
6. Котлярчук Д.В., Романюк О. В. Аналіз методів тестування інтерфейсу користувача [Електронний ресурс] // XLIX Науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2020): тези доповідей. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2020/paper/view/9764>