

АНАЛІЗ МЕТОДІВ ТЕСТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА

¹Вінницький національний технічний університет

Анотація

Проаналізовано різні підходи та методи тестування інтерфейсу користувача. Розглянуто особливості та методи тестування зручності використання інтерфейсу користувача.

Ключові слова: тестування, інтерфейс користувача, дефекти інтерфейсу, юзабіліті, доступність.

Abstract

Different approaches and methods of user interface testing are analyzed. Features and methods of user interface usability testing are considered

Keywords: testing, user interface, interface defects, usability, accessibility.

Вступ

Тестування інтерфейсу користувача є одним з етапів життєвого циклу розробки програмного забезпечення, яким зазвичай нехтують, щоб пришвидшити процес створення програмного продукту і випустити його на всесвітній ринок. Така ситуація склалася у зв'язку з розповсюдженням різноманітних CASE-засобів, які дозволяють прискорити процеси створення коду з прийнятним інтерфейсом.

Інтерфейс користувача (ІК) повинен не лише забезпечувати коректну взаємодію користувача з функціоналом програмного продукту, а й бути зручним, естетично привабливим та простим для розуміння. Саме ІК впливає на прийняття рішення користувачем щодо вибору того чи іншого програмного продукту серед наявних аналогів.

Особливу увагу сьогодні приділяють доступності (accessibility) ІК для усіх категорій користувачів, зокрема й людей з обмеженими можливостями. Так, використання мілкового шрифту, маленьких кнопок і піктограм може зробити ІК недоступним для людей старших 40 років, а використання лише кольору для розрізнення певних елементів інтерфейсу робить його недоступним для людей з вадами кольорового сприйняття, частка яких становить близько 10% потенційних користувачів [1]. У США відповідно до розділу 508 «Стандарти доступності електронних та інформаційних технологій» від 1998 року закону «American Workforce Rehabilitation Act» за недотримання вимоги доступності ІК може настати адміністративна відповідальність. Так, компанія Apple у свій час сплатила 250 тис. доларів компенсації за недоступність iTunes [2].

Таким чином, тестування інтерфейсу користувача є актуальною задачею.

Мета дослідження – аналіз методів тестування інтерфейсу користувача.

Об'єктом дослідження є процес тестування інтерфейсу користувача.

Предмет дослідження – методи тестування інтерфейсу користувача.

Результати дослідження

В ході дослідження було виконано аналіз функціонального та Usability тестування, з яких обрано найпопулярніших 4 методи тестування інтерфейсу користувача, а саме ручне тестування, автоматизоване тестування, тестування зручності використання та чек-лист. Вимоги до інтерфейсу користувача будуть розглянуті в ході аналізу дослідження.

Функціональне тестування інтерфейсу користувача, як правило складається з 5 фаз [3], що аналогічно

тестуванню будь-якого іншого компонента ПЗ:

- 1) Повна вибірка вимог ІК.
- 2) Перевірка повноти покриття ІК.
- 3) Складання повноцінних тест-вимог та тест-планів, які перевіряють ІК.
- 4) Виконання покрокових етапів тестування та перебір інформації результату виконаних тестів.
- 5) Ознайомлення з витягом звіту проблем, які демонструють розбіжну поведінку системи або вимог, що відсутні або не співпадають з деякими елементами інтерфейсу.

Відмінність може з'являтися у ході перевірки вимог, щодо ІК на кожному кроці тестування.

ІК перевіряється за сценаріями тест-плану, який описує дії користувачів за роботою у готовому продукті [4]. Запис сценарію може відбуватися природною або формальною мовами. Оператор виконує заплановане тестування у ручному або системному режимі.

У результаті виконання тестових прикладів збирається інформація, яка в подальшому застосовується в технології аналізу елементів форми або тексту на екрані.

Повнота покриття ІК передбачає, що в результаті виконання всіх тестових кроків, кожний елемент інтерфейсу був використаний хоча б один раз.

У звіті проблем ІК можуть міститися опис невідповідності до вимог та реальних дій систем або опис проблем у вимогах до ІК. Основною проблемою у вимогах є їх тест-непридатність, викликана неконкретністю і розпливчатиими формулюваннями. Наприклад, вимога "ПЗ повинно достатньо швидко надавати користувачу необхідну інформацію про його запит" є тест-непридатною та неконкретною, оскільки не вказаний конкретний час виконання на запит користувача.

Існують вимоги ІК, що значною мірою ускладнюють тестування і не є тест-непридатним. Такі вимоги зазвичай описують суб'єктивну властивість інтерфейсів, які не точно визначені або виміряні в ході виконання тестів. Вимоги до аналізів вимагають чіткого представлення, яка перевірка буде у якого елемента інтерфейсу, і яка його властивість буде вимірюватись під час виконання тестування. Основними визначеннями «покриття ІК» є функціональне і структурне покриття.

Виконання функціонального тестування ІК можливо різними методами, як оператором, так і за допомогою інструментарію, який автоматизує виконання покрокових тестів.

Ручним тестуванням ІК – називається процес, коли тестувальник-оператор виконує покроковий перебір всіх можливих сценаріїв. Сценарій містить послідовність кроків оператора в ході тесту, результати тестування, які відображаються на інтерфейсі порівнюються з вимогами. Таблиця – це типова форма запису послідовності сценарію під час проведення ручного тестування. У першій колонці записується опис кроків сценарію (дії оператора), у другій – час очікування реакції системи, наступна колонка призначена для запису інформації, чи збіглась запланована відповідь системи з дійсною та перерахування розбіжностей. Зручність даного методу у тому, що коректність у зміні елементу інтерфейсу проводиться безпосередньо користувачем.

Автоматизоване тестування – це використання автоматизованих інструментів, які імітують поведінку «споживача», за ручним сценарієм тестування ІК. Такий метод використовується у випадках, коли вхідна інформація записана деякою мовою, а оператори відтворюють дії користувачів – переміщення курсора, введення команд та активізацію елементів інтерфейсу. Очікуваний стан ІК визначають різними методами. Наприклад, передача інформації в ІК і одержання інформації аналізу можуть бути виконані двома способами, якщо є доступ до елементів інтерфейсу [5]:

1) Позиційний – це процес взаємодії з елементом інтерфейсу, який відбувається за допомогою задавання розмірів і точних координат.

2) Ідентифікатор – це взаємодія з елементами за допомогою одержання відповіді від інтерфейсу на основі його унікального ідентифікатора у межах вікна.

Головна мета розроблення ПЗ та ІК – це тестування надійності та комфорту для користувачів. Коли продукт задовольняє вимогам критерію, тільки тоді передається клієнтам, якщо продукт не відповідає критеріям, він допрацьовується.

Аналіз функціонального тестування дозволяє сформувати базове уявлення про можливі методи тестування інтерфейсу користувача. Кожен із методів по своєму є продуктивним і необхідним в конкретних ситуаціях. Але не потрібно і забувати про недоліки кожного із методу. Наприклад, ручне

тестування потребує об'ємні людські та часові ресурси. Цей недолік зазвичай проявляється при проведенні регресійного тестування. Іншим недоліком ручного тестування є його залежність від людського сприйняття. У свою чергу, якщо виникає необхідність в зміні покрокового тестування, то це легко виконати додавши чи видаливши деякі пункти в сценарії. Автоматизоване тестування у свою чергу використовує позиційний метод тестування, що передбачає зміну значної частини сценаріїв в системі тестів при кожній зміні інтерфейсу системи, що безперечно є недоліком такого методу і свідчить про можливість його використання лише для систем зі сталим інтерфейсом. Ідентифікаційний метод автоматизації тестування більш стійкий до змін розташування елементів інтерфейсу, але зміни тестових прикладів можуть знадобитись у випадку зміни логіки роботи елементів інтерфейсу. Тому ці два методи підходять для перевірки усіх елементів та самого інтерфейсу.

Однак інтерфейс, який працює коректно, не обов'язково буде зручним для використання. Тому важливою складовою тестування інтерфейсу користувача є тестування зручності використання (usability testing), адже від цього у користувача формується загальне бажання й надалі працювати з додатком, або шукати більш зручний аналог.

Тестування зручності використання – метод, який встановлює ступінь зручності використання, що буде зрозумілим та привабливим для будь-якого користувача [6]. Оскільки тестується зручність, тому метод має давати оцінку по таким пунктам:

- 1) Efficiency (продуктивність) – період, необхідний для виконання поставленого завдання.
- 2) Accuracy (правильність) – кількість помилок при роботі з ПЗ.
- 3) Recall (згадування в пам'яті) – час, який знадобиться для згадування всіх функцій програми.
- 4) Emotional response (емоційний відгук) – емоційна реакція користувача під час і після припинення роботи з додатком.

З офіційних досліджень, тест зручності рекомендують проводити з 5-8 учасників. Статистика показує, що 5 учасників розкривають 88% вимог тесту, як зображено на рисунку 1 [7].

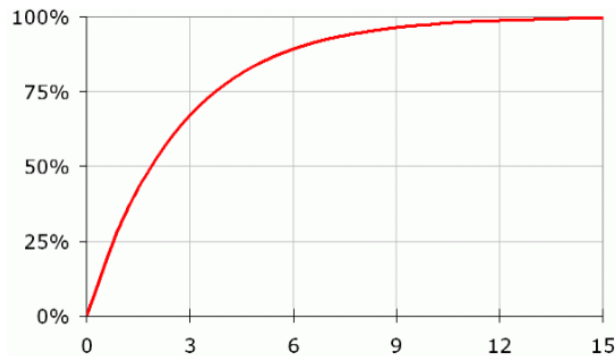


Рисунок 1 – статична значущість учасників

До основних методів тестування юзабіліті відносять такі [8]:

- 1) Карткове сортування – метод, при якому користувачі сортують елементи інтерфейсу одночасно до декількох категорій.
- 2) Зворотне карткове сортування – метод, який перевіряє зрозумілість спроектованої структури навігації інтерфейсу.
- 3) Тестування сподівань – метод, при якому користувач, дивлячись на роздрукований інтерфейс, може розповісти про функціонал та властивості елементів.
- 4) Думки вголос – метод, в якому учасники постійно говорять про свої думки та емоції вголос, під час використання ІК.

Також потрібно сказати, що існують ще такі методи: eye-tracking, оцінка сприйняття дизайну, конструктивна взаємодія, фокус-груп, експертна оцінка, евристична оцінка, паперове прототипування. Але на жаль вони мають свої недоліки, а саме: можлива розбіжність очікуваної від запланованої поведінки системи; неспроможність отримати достовірні статистичні дані; потрібен великий колектив

операторів для перевірки одного методу; якщо є недосвідчений посередник, то виникає ймовірність появи недостовірних результатів досліджень або малої реальної користі.

Отже, виходячи з раніше сказаного, дані методи рідко застосовуються, а у деяких випадках їх використання недоцільне, що не можна сказати про метод чек-лістів.

Чек-ліст [6, 8] – це документ, який описує, що має бути протестовано, і дозволяє фіксувати проведену роботу та відстежувати інформацію про ПЗ. Детальність чек-ліста буде залежати від рівня звітності та складності програмного забезпечення.

Основною властивістю чек-лістів є: необмежена кількість варіантів проходження тесту, яка може збільшуватися, по ходу проходження тестування ПЗ. Головними факторами, що зумовлюють популярність методу чек-лістів, є:

- 1) Не упускати необхідних тестів.
- 2) Рівномірний поділ задач між тестувальниками за рівнем їх кваліфікації.
- 3) Достовірність збереженого звіту за результатами тестів.

Тому повинні бути такі колонки:

- 1) Ідентифікатор перевірки.
- 2) Статус перевірок:
 - версія ПЗ;
 - версія ОС;
 - ПІБ тестувальник.
- 3) Статистика результату перевірки.

Перевагами даного методу можна виділити:

- 1) Всі тестувальники розуміють по своєму пункт перевірки, тому в ході тестування можливе додаткове доповнення важливої розбіжності або помилки.
 - 2) Результат повторного проходження тестового сценарію, буде завжди викликати ті же самі помилки в роботі ПЗ, а не показувати нестикування тесту, де не зрозуміло чому помилка іноді виникає, а іноді ні.
 - 3) Менших затрат часового ресурсу на підтримку тесту.
 - 4) Статистика: хто, що проходив з детальним розбором всіх дій.
 - 5) Можливість комбінувати тест в залежності від уподобань співробітників.
- При цьому виникає можливість оцінити стан готовності продукту, до випуску.

Висновок

Таким чином, були показані переваги існуючих методів для тестування інтерфейсів користувача. Дослідження довело, що дані методи добре себе зарекомендували для розробників, яким необхідно прикласти велику кількість ресурсів до розробки ПЗ. Отже, проаналізувавши все вище сказане, можна зробити висновок, що використання ручного методу з чек-лістами є доцільним для більшості поставлених задач, де необхідно виконати тестування інтерфейсу користувача.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Кулямин В.В. Лекция 9. Принципы создания удобного пользовательского интерфейса [Електронний ресурс] – Режим доступу до ресурсу: <file:///C:/Users/ks/Downloads/programming.technologies.component.approach.09.pdf>
2. Accessibility. Как мы делаем Яндекс доступным людям с ограниченными возможностями и почему считаем это важным [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/company/yandex/blog/258477/>
3. Тестування інтерфейсу користувача. Електронна підтримка та сучасні інформаційні технології у інтерфейсах користувача [Електронний ресурс] – Режим доступу до ресурсу: https://msn.khnu.km.ua/pluginfile.php/278098/mod_resource/content/3/%D0%9B%D0%B5%D0%BA%D1%86%D1%96%D1%8F7.pdf
4. «Проектування інтерфейсів користувача» [Електронний ресурс] – Режим доступу до ресурсу: https://msn.khnu.km.ua/pluginfile.php/315542/mod_resource/content/3/Lekz7.pdf
5. РОЗДІЛ 4. ВИКОРИСТАННЯ СУЧАСНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРИ ПРОЕКТУВАННІ ТА ТЕСТУВАННІ ІНТЕРФЕЙСІВ КОРИСТУВАЧА [Електронний ресурс] – Режим доступу до ресурсу: <http://elar.khnu.km.ua/jspui/bitstream/123456789/1415/5/Rozdil4.pdf>

6. 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>
7. Тестування на зручність [Електронний ресурс] – Режим доступу до ресурсу: [Електронний ресурс] – Режим доступу до ресурсу:
8. Навчальний посібник «Методи тестування та оцінки якості програмного забезпечення із застосуванням Pairwise тестування» для студентів денної та заочної форми навчання – Полтава: ПолтНТУ, 2016. – 391 с.

Котлярчук Дмитро Володимирович – студент групи ІПІ-18мс, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: ratatyu9898@gmail.com

Романюк Оксана Володимирівна – к.т.н., доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: romaniukoksanav@gmail.com

Dmitry Kotlyarchuk – student of group IPI-18ms, Faculty for Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: ratatyu9898@gmail.com

Oksana Romaniuk – Candidate of Technical Sciences, Associate Professor of the Software Chair, Vinnytsia National Technical University, Vinnytsia, e-mail: romaniukoksanav@gmail.com