

К. А. Чернишов

*Вінницький національний технічний університет, місто Вінниця
Кафедра програмного забезпечення*

І. П. Малініч

*Вінницький національний технічний університет, місто Вінниця
Кафедра комп'ютерних наук*

П. П. Малініч

*Вінницький національний технічний університет, місто Вінниця
Кафедра програмного забезпечення*

МЕТОДИ ЗБОРУ ДАНИХ ДОСВІДУ ВЗАЄМОДІЇ КОРИСТУВАЧА ДЛЯ ВИПРОБУВАЛЬНОГО ЕТАПУ РОЗРОБКИ ЧЕРЕЗ ТЕСТУВАННЯ

Анотація

В даній статті розглядаються основні методи збору даних досвіду взаємодії користувача, їх переваги та недоліки, а також особливості їх застосування на випробувальному періоді розробки через тестування. Приділяється увага використанню запропонованого методу в таких видах низькобюджетних проектів як стартапи.

1. Суть проблеми

Сучасна інфраструктура збору даних досвіду взаємодії користувача дозволяє підвищувати рівень зручності та простота використання он-лайн додатків та Інтернет-ресурсів в ході бета-тестування та цільового використання відповідних програмних засобів. Досвід взаємодії являє собою сприйняття і дії у відповідь користувача, що виникають в результаті чинного чи подальшого використання програмного забезпечення [1].

Розробка через тестування є ключовим способом розробки додатків, комерціалізація яких тимчасово відсутня або винесена на другий план. Причиною тому може бути відсутність гарантій швидкої окупності проекту, або у випадку створення стартапів – застосування праці розробників-волонтерів.

Якість кінцевого продукту прямо пропорційна кількості часу, витраченого на його розробку, а внаслідок вираженої через ступінь закінченості продукту його цінності та підвищення кваліфікації розробника така цінність може бути виражена графіком з параболою за лінійною прогресією, де час, що витрачено на розробку підвищує як цінність, так і якість продукту. Але графік зростання часу на виправлення і підвищення кінцевої ефективності – прямо пропорційний і абсолютно обернений.

2. Існуючі методи і їх недоліки

Серед найбільш поширених розглядаються чотири методи збору даних досвіду взаємодії користувача: використання традиційних систем обліку недоліків програмного забезпечення (написання баг-репортів в баг-трекерах), віддалене журналювання відлагоджувальної інформації, використання інструментарію віддаленого відлагодження та покроковий збір та аналіз даних про дії користувача у інтерфейсах [2].

Використання систем баг-репорту передбачає активності з боку кінцевих користувачів, що не завжди можливо та ефективно. Віддалене відлагодження може виконати функцію збору даних лише у окремих випадках, і тільки під час безпосередньої співпраці з бета-тестувальниками, що робить цей спосіб збору даних не гнучким у випадку нецільового тестування.

Віддалене журналювання є більш ефективним для відслідковування помилок, що виникли з боку серверного додатку [3] і не є гнучким в плануванні користувацького інтерфейсу. Як видно з аналізу представлених методів, найбільш дієвим залишається останній метод, який передбачає алгоритм збору даних без участі користувача або розробника, а також сторонніх засобів відлагодження на етапі тестування.

3. Опис представленого методу

Взятий за основу метод покрокового збору та аналізу даних можна проілюструвати за допомогою математичної багатофакторної кореляційно-регресійної моделі. Для побудови моделі алгоритму аналізу вхідних даних пропонуються наступні етапи:

1. Априорне дослідження проблеми.

Розглянемо гіпотезу про залежність ефективності архітектури користувацького інтерфейсу від величини проміжку часу, витраченого користувачем на виконання однієї конкретної дії та безпосереднього розташування елемента інтерфейсу, що відповідає за виконання такої дії. Для вираження цієї неоднорідної залежності необхідно проаналізувати решту факторів, з яких складається остання.

2. Формування переліку факторів та їх логічний аналіз.

Слід відокремити впливові та залежні групи факторів. До впливових факторів віднесемо:

- Загальна кількість функцій (активностей) сервісу;
- Загальну кількість використань функцій сервісу;
- Загальну кількість використань всіх варіацій доступу;
- Кількість унікальних користувачів сервісу;
- Кількість користувачів сервісу, що користуються ним вперше.

До залежної групи факторів слід віднести:

- Частота використання даної конкретної функції сервісу;
- Частота використання даної конкретної варіації доступу.
- Кількість варіацій доступу до даної конкретної функції сервісу.

4. Застосування та можливості

Розробку з описаним методом було впроваджено у випробувальному режимі у Вінницькому Національному Технічному Університеті для адміністративних панелей систем управління навчальним процесом ВНТУ [4], а також відкрито локальну версію API для студентських розробок, які використовують інтерактивні ігрові та веб-інтерфейси.

Список використаних джерел

1. ISO 9241-210:2010. Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems [Електронний ресурс] // International Organization for Standardization. – Режим доступу : <https://www.iso.org/standard/52075.html> – Назва з екрана. Дата публікації: 01.03.2010, дата звернення 15.12.2018).

2. Інтерфейс "Користувач-комп'ютер" : навч. посіб. для студ. спец. "Програмне забезпечення автоматизованих систем" / В. П. Майданюк, А. М. Петух ; Вінницький держ. технічний ун-т. - Вінниця : ВДТУ, 1999. - 64 с.

3. Яланський Д. А. Підвищення надійності серверних систем шляхом комплексного поєднання різнорівневих методів балансування [Електронний ресурс] / Д. А. Яланський, В. І. Месюра // Матеріали XLVI науково-технічної конференції підрозділів ВНТУ, Вінниця, 22-24 березня 2017 р. - Електрон. текст. дані. - 2017. - Режим доступу : <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2017/paper/view/2157>.

4. Паламарчук Є. А. Розробка динамічного внутрішнього сайту кафедри / Є. А. Паламарчук, Р. Н. Кветний, В. М. Дубовой // Матеріали 9-ї науково-практичної конференції, м. Львів, 21-23 листопада 2017 р. – Львів : Видавництво Наукового товариства ім. Шевченка, 2017. – С. 179-183.