

Український державний університет імені М.П. Драгоманова

Факультет математики, інформатики та фізики

Кафедра комп'ютерної та програмної інженерії

ЛАБОРАТОРНА РОБОТА №2

з курсу

«Автоматизація тестування програмного забезпечення»

Тема: Стратегія тестування

Студентки: Назаренко Евеліна Юріївна

Група 41ПЗз

Факультет математики, інформатики та фізики

Викладач: Кархут В. Я.

Київ 2026

Завдання 3

Стратегія тестування

1. Вступ

Опис тест-стратегії для тестування ключових модулів веб-платформи Rozetka в умовах обмежених ресурсів:

- 2 QA інженери (3–5 років досвіду),
- 2 місяці на тестування.

Мета стратегії — забезпечити максимальне покриття критичної функціональності магазину при мінімальних ризиках.

2. Пріоритетні типи тестування

2.1. Функціональне тестування (найвищий пріоритет)

Причина: Rozetka — це e-commerce, де основна цінність — можливість знайти товар, переглянути його та оформити покупку. Будь-яка помилка в цих модулях напряму впливає на продажі.

Фокус:

- пошук,
- фільтри,
- картка товару,
- кошик,
- оформлення замовлення.

2.2. Тестування критичного шляху (Critical Path Testing)

Причина: Це найважливіший бізнес-процес: **пошук → товар → кошик → оформлення → підтвердження.**

Цей шлях перевіряється **щодня**, тому що:

- він приносить основний дохід,
- будь-який збій = втрата продажів.

2.3. Регресійне тестування

Причина: Сайт великий, зміни часті, а команда маленька. Повна регресія неможлива — робимо **скорочену щотижневу**.

Фокус:

- критичні модулі,
- функції, які часто змінюються,
- інтеграції (доставка, оплата — поверхнево).

2.4. Димове тестування (Smoke Testing)

Причина: Після кожного оновлення потрібно швидко перевірити, чи сайт взагалі працює.

Покриття:

- головна сторінка,
- пошук,
- відкриття товару,
- додавання в кошик.

2.5. Юзабіліті тестування

Причина: Користувачі часто кидають кошик через незрозумілий інтерфейс. Мінімальні перевірки допоможуть виявити критичні UX-проблеми.

Фокус:

- зрозумілість кнопок,
- повідомлення про помилки,
- логічність навігації.

2.6. Мінімальне безпекове тестування

Причина: Повний security-аудит неможливий у межах 2 місяців, але критичні точки треба перевірити.

Фокус:

- XSS у формах,
- SQL Injection у пошуку,
- валідація полів (телефон, email),
- неможливість змінити ціну через DevTools.

3. Послідовність тестування

3.1. Етап 1 — Димове тестування

Після кожного релізу:

- доступність сайту,
- пошук,
- картка товару,
- кошик.

3.2. Етап 2 — Функціональне тестування ключових модулів

Порядок:

1. Пошук товарів
2. Каталог і фільтри
3. Картка товару
4. Кошик
5. Оформлення замовлення
6. Авторизація/реєстрація

3.3. Етап 3 — Тестування критичного шляху

Виконується щодня:

- пошук → товар → кошик → оформлення.

3.4. Етап 4 — Регресійне тестування

Щотижневе, скорочене:

- ключові модулі,
- функції, які змінювалися останнім часом.

3.5. Етап 5 — Юзабіліті та мінімальна безпека

Виконується паралельно, невеликими блоками.

4. Стратегія звітності

4.1. Щоденні звіти

Формат:

- виконані тести,
- знайдені дефекти,
- статус критичного шляху,
- блокери.

4.2. Щотижневі звіти

Формат:

- прогрес по модулях,
- кількість дефектів за пріоритетами,
- ризики,
- статус регресії.

4.3. Фінальний звіт

Містить:

- % виконаних тестів,
- % успішних тестів,
- список дефектів,
- рекомендації щодо покращення.

5. Обґрунтування вибору стратегії

Ця стратегія оптимальна, тому що:

- **2 QA інженери** не можуть покрити весь сайт → фокус на критичному шляху.
- **2 місяці** — достатньо для ключових модулів, але не для всього сайту → потрібна пріоритизація.
- **Щоденне тестування критичного шляху** мінімізує ризики втрати продажів.
- **Регресія щотижня** дозволяє контролювати якість при частих оновленнях.
- **Мінімальне безпекове тестування** покриває найнебезпечніші точки без перевантаження команди.
- **Юзабіліті** важливе для конверсії, але не потребує багато часу.