

[2] С Sharp [Електронний ресурс] : [Веб-сайт]. – Режим доступу: https://uk.wikipedia.org/wiki/C_Sharp (дата звернення 20.09.2023). – Назва з екрана.

Дем'ян Дзямба, Ігор Крошний*, Олександр Овсяк*

магістрант кафедри КН НЛТУ України, Львів, Україна.
mezko.a@nltu.lviv.ua.

* в.о. директора ННІ КНІТ НЛТУ України, Львів, Україна.
<https://orcid.org/0000-0003-0018-7998>. kroshny.igor@nltu.edu.ua.

*професор кафедри КН НЛТУ України, Львів, Україна.
<https://orcid.org/0009-0008-8327-5101>. ovsjak@nltu.edu.ua.

Розроблення фреймворку для автоматизованого тестування сайту “Розетка” з використанням TypeScript та Cypress

Анотація - Розроблено фреймворк для автоматичного тестування сайту “Розетка” з використанням TypeScript та Cypress. У сучасному світі, де веб-сайти та веб-застосунки стають все більш складними та функціональними, надійність та якість їх роботи стають критичними. Саме тестування веб-сайтів відіграє важливу роль у забезпеченні коректності та стабільності роботи програмного забезпечення. Проте, тестування веб-сайтів може бути складною та ресурс-затратною задачею. Ручне тестування, яке виконується людиною, може бути повільним, незахищеним від помилок та вимагати значних зусиль. Тому, актуальність проблемної області автоматизованого тестування веб-сайтів є надзвичайно високою. Використання автоматизованих тестів дозволяє прискорити процес тестування, зменшити ймовірність помилок та забезпечити більш швидке виявлення дефектів.

Ключові слова - Cypress, TypeScript, JavaScript, CI/CD, CSS Selectors, Mocha Reporting, GitLab.

У сучасному світі, де веб-сайти та веб-додатки стають все більш складними та функціональними, надійність та якість їх роботи стають критичними. І саме тестування веб-сайтів відіграє важливу роль у забезпеченні коректності та стабільності роботи програмного забезпечення.

Тестування веб-сайтів може бути складною та ресурс затратною задачею. Ручне тестування, яке виконується людиною, може бути повільним, піддатли-

вим до помилок та вимагати значних зусиль. Крім того, зі зростанням розміру та складності веб-сайтів, кількість тестових сценаріїв та тестових кейсів також збільшується, що ускладнює процес тестування.

Актуальність проблемної області автоматизованого тестування веб-сайтів є надзвичайно високою. Використання автоматизованих тестів дозволяє прискорити процес тестування, зменшити ймовірність помилок та забезпечити більш швидке виявлення дефектів.

На рисунку 1 показано фрагмент коду тестування у Cypress з використанням патерну Page Object. На рисунку 2 представлено логін у Cypress.

```
import HomePage from "../pages/HomePage";

describe("HomePage", () => {
  beforeEach(() => {
    HomePage.navigateTo();
  });

  it("should display correct title", () => {
    HomePage.getTitle().should("contain", "Головна сторінка");
  });

  it("should navigate to login page when login button is clicked", () => {
    HomePage.clickLoginButton();
    cy.url().should("include", "/login");
  });
});
```

Рис. 1. Фрагмент коду тестування у Cypress з використанням патерну Page Object

```
import HomePage from "../pages/HomePage";

describe("HomePage", () => {
  beforeEach(() => {
    HomePage.navigateTo();
  });

  it("should display correct title", () => {
    HomePage.getTitle().should("contain", "Головна сторінка");
  });

  it("should navigate to login page when login button is clicked", () => {
    HomePage.clickLoginButton();
    cy.url().should("include", "/login");
  });
});
```

Рис. 2. Авторизація у Cypress

На рисунку 3 показано програмний код функції для виконання скріншотів у Cypress.

```
export const screenshot = (name: string) => {
  const timestamp = new Date().getTime();
  const fileName = `${name}_${timestamp}.png`;
  cy.screenshot(fileName);
};
```

Рис. 3. Програмний код функції для виконання скріншотів у Cypress

Висновок. В роботі розглянуто всі етапи розробки фреймворку для автоматичного тестування веб-сайту з використанням TypeScript та Cypress. Були розглянуті актуальність предметної області автоматизованого тестування, аналіз предметної області тестування веб-застосунків, а також огляд існуючих систем оцінювання поширення.

Для цього було використано мову програмування TypeScript, яка надає переваги типізації, підказок під час розробки та зменшує кількість помилок. Фреймворк Cypress був обраний як основний інструмент для автоматизованого

тестування, оскільки він надає зручний і простий інтерфейс, має багатий функціонал і підтримує високий рівень надійності.

У роботі були розглянуті різні види тестування, включаючи функціональне тестування, регресійне тестування, тестування взаємодії з користувачем та інші. Для організації тестових сценаріїв був використаний патерн PageObject, що дозволяє структурувати тестовий код та полегшує його підтримку та розширення.

Список використаних літературних джерел

- [1] TypeScript [Електронний ресурс] : [Веб-сайт]. – Режим доступу: <https://www.typescriptlang.org/docs/> (дата звернення 25.09.2023). – Назва з екрана.
- [2] Mocha TypeScript [Електронний ресурс] : [Веб-сайт]. – Режим доступу: <https://mochajs.org/> (дата звернення 25.09.2023). – Назва з екрана.
- [3] Cypress Dashboard TypeScript [Електронний ресурс] : [Веб-сайт]. – Режим доступу: <https://docs.cypress.io/guides/dashboard> (дата звернення 25.09.2023). – Назва з екрана.
- [4] Cypress TypeScript [Електронний ресурс] : [Веб-сайт]. – Режим доступу: <https://www.cypress.io/blog/> (дата звернення 25.09.2023). – Назва з екрана.
- [5] TypeScript TypeScript [Електронний ресурс] : [Веб-сайт]. – Режим доступу: <https://devblogs.microsoft.com/typescript/> (дата звернення 25.09.2023). – Назва з екрана.
- [6] GitLab TypeScript [Електронний ресурс] : [Веб-сайт]. – Режим доступу: <https://about.gitlab.com> (дата звернення 25.09.2023). – Назва з екрана.

Ростислав Завацький, Олександр Сторожук*, Наталія Гіссовська*

магістрант каф. КН НЛТУ України, Львів, Україна,
Zavatskyu.R@nltu.lviv.ua

* доцент каф. ІСКМ, НЛТУ України, Львів, Україна, <https://orcid.org/0000-0001-6566-5271>. storozhuk@nltu.lviv.ua

* асистент каф. ІСКМ, НЛТУ України, Львів, Україна.,
gissovaska.n@nltu.edu.ua.