

Лабораторна робота #7

Автоматизація тестування вебсайту на прикладі сайту kpi.ua

Мета роботи: Навчитися використовувати засоби автоматизації тестування на прикладі Playwright для автоматизованого тестування вебсайтів, перевірити доступність сторінок, правильність перенаправлення та функціональність ключових елементів на сайті.

Крок 1: Перевірка завантаження головної сторінки

```
//Крок 1
const { test, expect } = require('@playwright/test');

test('Перевірка завантаження головної сторінки', async ({ page }) => {
  const response = await page.goto('https://kpi.ua');
  expect(response.status()).toBe(200); // Перевірка статусу 200 OK
});
```

```
PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 2 tests using 1 worker

ok 1 test.spec.js:4:1 > Перевірка завантаження головної сторінки (4.1s)
ok 2 test.spec.js:10:1 > Перевірка HTTP → HTTPS перенаправлення (2.9s)

2 passed (8.8s)
```

Крок 2: Перевірка коректності HTTP → HTTPS перенаправлення

```
//Крок 2
test('Перевірка HTTP → HTTPS перенаправлення', async ({ page }) => {
  const response = await page.goto('http://kpi.ua');
  expect(response.status()).toBe(200); // Перевірка статусу 200 OK
  expect(page.url()).toBe('https://kpi.ua/'); // Перевірка, що URL змінився на HTTPS
});
```

```
PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 2 tests using 1 worker

ok 1 test.spec.js:4:1 > Перевірка завантаження головної сторінки (4.1s)
ok 2 test.spec.js:10:1 > Перевірка HTTP → HTTPS перенаправлення (2.9s)

2 passed (8.8s)
```

Крок 3: Тестування роботи меню навігації

```
// Крок 3
const { test, expect } = require('@playwright/test');
test('Тестування роботи меню навігації', async ({ page }) => {
  await page.goto('https://kpi.ua');
  const aboutUsLink = await page.locator('a.sf-depth-1.menuparent:has-text("університет")');
  await aboutUsLink.waitFor({ state: 'visible' });
  await aboutUsLink.click();
  expect(await page.title()).toContain('Про університет');
});
```

```
PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test
```

```
Running 1 test using 1 worker
```

```
ok 1 test.spec.js:18:1 > Тестування роботи меню навігації (4.8s)
```

```
1 passed (6.4s)
```

Крок 4: Перевірка внутрішніх посилань

```
// Крок 4
const { test, expect } = require('@playwright/test');
test('Перевірка внутрішніх посилань', async ({ page, context }) => {
  test.setTimeout(40000);
  await page.goto('https://kpi.ua');
  const links = await page.locator('a[href^="/"]');
  const totalLinks = await links.count();
  console.log(`Є ${totalLinks} внутрішніх посилань`);
  for (let i = 0; i < totalLinks; i++) {
    const link = links.nth(i);
    const href = await link.getAttribute('href');
    if (!href || href === '#' || href.startsWith('javascript:')) {
      console.warn(`Пропущено некоректне посилання [${i + 1}]: ${href || 'немає атрибута href'}`);
      continue;
    }
    const fullUrl = `https://kpi.ua${href}`;
    console.log(`Перевірка (${i + 1}): ${fullUrl}`);
    try {
      const response = await context.request.get(fullUrl, { timeout: 15000 });
      const status = response.status();
      expect(status).toBe(200);
      console.log(`Перевірено (${i + 1}): ${fullUrl}`);
    } catch (error) {
      console.error(`Помилка (${i + 1}): ${fullUrl}\n${error.message}`);
    }
    await new Promise(resolve => setTimeout(resolve, 500));
  }
  console.log('Перевірка завершена');
});
```

```

PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 1 test using 1 worker

    1 test.сpec.js:29:1 > Перевірка внутрішніх посилань
Є 199 внутрішніх посилань
Перевірка (1): https://kpi.ua/
Перевірено (1): https://kpi.ua/
Перевірка (2): https://kpi.ua/
Перевірено (2): https://kpi.ua/
Перевірка (3): https://kpi.ua/searcher
Перевірено (3): https://kpi.ua/searcher

1 passed (5.3m)

```

Крок 5: Тестування адаптивності сторінки

```

// Крок 5
const { test, expect } = require('@playwright/test');
test('Тестування адаптивності сторінки', async ({ page }) => {
  await page.goto('https://kpi.ua');
  await page.setViewportSize({ width: 1200, height: 800 });
  expect(await page.isVisible('header')).toBe(true); // Перевірка, чи видимий header
  await page.setViewportSize({ width: 375, height: 667 });
  expect(await page.isVisible('header')).toBe(true);
});

```

```

PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 1 test using 1 worker

    ok 1 test.сpec.js:58:1 > Тестування адаптивності сторінки (4.2s)

1 passed (6.1s)

```

Крок 6: Тестування часу завантаження сайту

```

// Крок 6
const { test, expect } = require('@playwright/test');
test('Тестування часу завантаження сайту', async ({ page }) => {
  const startTime = Date.now(); // Час до завантаження сторінки
  await page.goto('https://kpi.ua');
  const endTime = Date.now(); // Час після завантаження сторінки
  const loadTime = endTime - startTime;
  console.log(`Час завантаження сайту: ${loadTime} мс`);
  expect(loadTime).toBeLessThan(5000);
});

```

```
PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 1 test using 1 worker

  ok 1 test.сpec.js:68:1 > Тестування часу завантаження сайту (3.9s)
Час завантаження сайту: 2605 мс

1 passed (5.6s)
```

Крок 7: Перевірка наявності всіх основних елементів на головній сторінці

```
// Крок 7
const { test, expect } = require('@playwright/test');

test('Перевірка наявності основних елементів на головній сторінці', async ({ page }) => {
  await page.goto('https://kpi.ua');

  const logoSelector = 'img[alt="Толовна"]';
  await page.waitForSelector(logoSelector);
  expect(await page.isVisible(logoSelector)).toBe(true);

  const menuSelector = '#superfish-main';
  await page.waitForSelector(menuSelector);
  expect(await page.isVisible(menuSelector)).toBe(true);

  const newsBlockSelector = '.field--name-body';
  await page.waitForSelector(newsBlockSelector);
  expect(await page.isVisible(newsBlockSelector)).toBe(true);

  const searchBarSelector = 'a[title="search"]';
  await page.waitForSelector(searchBarSelector);
  expect(await page.isVisible(searchBarSelector)).toBe(true);

  const devices = [
    { width: 1920, height: 1080 },
    { width: 768, height: 1024 }
  ];

  for (const viewport of devices) {
    await page.setViewportSize(viewport);
    expect(await page.isVisible(logoSelector)).toBe(true);
    expect(await page.isVisible(menuSelector)).toBe(true);
    expect(await page.isVisible(newsBlockSelector)).toBe(true);
    expect(await page.isVisible(searchBarSelector)).toBe(true);
  }
});
```

```
PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 1 test using 1 worker

  ok 1 test.сpec.js:68:1 > Перевірка наявності основних елементів на головній сторінці (4.1s)

1 passed (5.7s)
```

Крок 8: Тестування доступності сайту

```
// Крок 8
const { test, expect } = require('@playwright/test');
test('Тестування доступності сайту', async ({ page }) => {
  await page.goto('https://kpi.ua');
  console.log('Сторінка завантажена');
  const elements = await page.locator('[alt], [role], [aria-label], [aria-labelledby], [aria-hidden], [tabindex], [aria-live], [aria-expanded], [aria-describedby]');
  const elementCount = await elements.count();
  console.log(`Перевірка доступності для ${elementCount} елементів із спеціальними атрибутами...`);
  let issues = 0;
  for (let i = 0; i < elementCount; i++) {
    const element = elements.nth(i);
    const alt = await element.getAttribute('alt');
    const role = await element.getAttribute('role');
    const ariaLabel = await element.getAttribute('aria-label');
    const ariaHidden = await element.getAttribute('aria-hidden');
    const tabIndex = await element.getAttribute('tabindex');
    const ariaLive = await element.getAttribute('aria-live');
    console.log(`Елемент №${i + 1}:`);
    console.log(`  alt: ${alt || 'Немає'}`);
    console.log(`  role: ${role || 'Немає'}`);
    console.log(`  aria-label: ${ariaLabel || 'Немає'}`);
    console.log(`  aria-hidden: ${ariaHidden || 'Немає'}`);
    console.log(`  tabindex: ${tabIndex || 'Немає'}`);
    console.log(`  aria-live: ${ariaLive || 'Немає'}`);
    if (alt && role && ariaLabel) {
      issues++;
    }
  }
});
```

```
PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test
```

```
Running 1 test using 1 worker
```

```
1 test.spec.js:115:1 > Тестування доступності сайту
Перевірка доступності для 47 елементів із спеціальними атрибутами...
```

```
Елемент №1:
```

```
alt: Головна
role: Немає
aria-label: Немає
aria-hidden: Немає
tabindex: Немає
aria-live: Немає
```

```
ok 1 test.spec.js:115:1 > Тестування доступності сайту (6.2s)
```

```
1 passed (7.7s)
```

Крок 9: Перевірка завантаження сторінок іншими мовами

```
// Крок 9
const { test, expect } = require('@playwright/test');
test('Перевірка тексту на англійській сторінці', async ({ page }) => {
  await page.goto('https://kpi.ua');
  console.log('▶ Початок перевірки доступності перемикача мови');
  const languageSwitch = page.locator('a[href="/en/"]');
  const isSwitchVisible = await languageSwitch.isVisible();
  console.log(`🔍 Результат: перемикач ${isSwitchVisible ? 'доступний для взаємодії' : 'не знайдено на сторінці'}`);
  expect(isSwitchVisible).toBe(true);

  console.log('👉 Перемикаємося на англійську версію сайту...');
  await languageSwitch.click();
  await page.waitForURL('**/en/', { timeout: 5000 });

  const currentURL = page.url();
  console.log(`✅ Перемикання виконано. Поточний URL: ${currentURL}`);
  expect(currentURL).toContain('/en/');
});
```

```

console.log('🔍 Аналіз тексту слогану...');
const siteSlogan = page.locator('div.site_slogan a[rel="home"]');
const sloganText = await siteSlogan.textContent();
const sloganCleaned = sloganText?.trim();
console.log('📄 Текст слогану: "${sloganCleaned}"');
expect(sloganCleaned).toContain('National Technical University of Ukraine');
expect(sloganCleaned).toContain('Igor Sikorsky Kyiv Polytechnic Institute');

console.log('🔍 Перевірка пунктів меню...');
const menu = page.locator('#superfish-main');
const menuText = await menu.textContent();
const menuItems = ['University', 'Education', 'Science', 'Admission', 'Student Life'];

menuItems.forEach(item => {
  const exists = menuText.includes(item);
  console.log(`- Пункт "${item}" ${exists ? 'присутній ✅' : 'відсутній ❌'}`);
  expect(menuText).toContain(item);
});

console.log('🔍 Збір заголовків сторінки...');
const headers = await page.locator('h1').allTextContents();
headers.forEach((header, index) => {
  console.log(`- Заголовок [${index + 1}]: "${header.trim()}"`);
});

console.log('🎉 Усі тести пройдено успішно.');
```

PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 1 test using 1 worker

```

ok 1 test.сpec.js:145:1 > Перевірка тексту на англійській сторінці (7.0s)
  ✨ Початок перевірки доступності перемикача мови
  🔍 Результат: перемикач доступний для взаємодії
  ✨ Перемикаємося на англійську версію сайту...
  ✨ Перемикання виконано. Поточний URL: https://kpi.ua/en/
  🔍 Аналіз тексту слогану...
  ✨ Текст слогану: "National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute""
  🔍 Перевірка пунктів меню...
    - Пункт "University" присутній ✅
    - Пункт "Education" присутній ✅
    - Пункт "Science" присутній ✅
    - Пункт "Admission" присутній ✅
    - Пункт "Student Life" присутній ✅
  🔍 Збір заголовків сторінки...
    - Заголовок [1]: "Mission:"
  🎉 Усі тести пройдено успішно.
```

1 passed (8.5s)

Крок 10*: Перевірка завантаження зовнішніх ресурсів

```

// Крок 10
const { test, expect } = require('@playwright/test');
test('Перевірка завантаження зовнішніх ресурсів', async ({ page }) => {
  const resources = new Map();
  page.on('request', req => {
    resources.set(req.url(), {
      url: req.url(),
      startTime: Date.now(),
      status: null,
      endTime: null
    });
  });
});
```

```

page.on('response', res => {
  const resource = resources.get(res.url());
  if (resource) {
    resource.endTime = Date.now();
    resource.status = res.status();
  }
});
await page.goto('https://kpi.ua');
await page.waitForTimeout(5000);

const results = Array.from(resources.values()).map(r => {
  const loadTime = r.endTime ? r.endTime - r.startTime : null;
  return { ...r, loadTime };
});
results.forEach(r => {
  const statusText = r.status ? (r.status < 400 ? 'без помилок' : `Помилка (${r.status})`) : 'Невідомо';
  const loadText = r.loadTime ? (r.loadTime > 3000 ? 'Повільне' : 'Оптимальне') : 'Невідомо';
  console.log(`
    URL: ${r.url},
    Час завантаження: ${r.loadTime ? r.loadTime + 'мс' : 'Невідомо'},
    Статус: ${r.status ? r.status : 'Невідомо'},
    Оцінка: ${loadText}, ${statusText}
  `);
});
const errors404 = results.filter(r => r.status === 404);
const errors500 = results.filter(r => r.status === 500);
const slowResources = results.filter(r => r.loadTime !== null && r.loadTime > 3000);

expect(errors404.length).toBe(0);
expect(errors500.length).toBe(0);
expect(slowResources.length).toBe(0);
});

```

PS C:\3 курс\ТЗЯПЗ\lab_7> npx playwright test

Running 1 test using 1 worker

1 test.spec.js:191:1 > Перевірка завантаження зовнішніх ресурсів

URL: https://kpi.ua/,
Час завантаження: 93мс,
Статус: 200,
Оцінка: Оптимальне, без помилок

URL: https://kpi.ua/libraries/drupal-superfish/css/superfish.css?soduq8,
Час завантаження: 19мс,
Статус: 200,
Оцінка: Оптимальне, без помилок

URL: https://www.youtube.com/youtubei/v1/log_event?alt=json,
Час завантаження: 66мс,
Статус: 200,
Оцінка: Оптимальне, без помилок

1 passed (11.5s)