

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Програма автоматизованого тестування вебресурсів»

Студента 2 курсу, 2-2мз
групи,
спеціальності 121 «Інженерія
програмного забезпечення»
освітньої програми «Інженерія
програмного забезпечення»

Лаврінєнка Владислава
Сергійовича

підпис студента

Науковий керівник
кандидат педагогічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Котенко Наталія
Олексіївна

підпис керівника

Гарант освітньої програми
кандидат педагогічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Котенко Наталія
Олексіївна

підпис гаранта

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь магістр

Освітня програма 121 «Інженерія програмного забезпечення»

Затверджую

Зав. кафедри інженерії програмного
забезпечення та кібербезпеки

Криворучко О. В.

«13» грудня 2022 р.

Завдання

на випускн кваліфікаційну роботу студентів

Лавріненку Владиславу Сергайовичу

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи «Програма автоматизованого
тестування вебресурсів»»

Затверджена наказом ректора від «09» грудня 2022 р. № 3339

2. Строк здачі студентом закінченої роботи 1 грудня 2023

3. Цільова установка та вихідні дані до роботи

Мета роботи розробка програми для автоматизованого тестування
вебресурсів з метою перевірки актуальних курсів валют

Об'єкт дослідження система автоматизованого тестування валютних курсів,
яка включає в себе програмний код для взаємодії з вебсайтами та API, а також
інфраструктуру для виконання тестів та зберігання результатів.

Предмет дослідження розробка програми, що використовує технології, такі
як Java, Selenium WebDriver та RestAssured, для автоматизованого тестування
та аналізу валютних курсів.

4. Консультанти роботи із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)
ВСТУП

РОЗДІЛ 1 ОГЛЯД СУЧАСНОГО СТАНУ АВТОМАТИЗОВАНОГО
ТЕСТУВАННЯ ВЕБРЕСУРСІВ

1.1. Загальний огляд вебресурсів

1.2. Значення автоматизованого тестування

1.3. Виклики

1.4. Мета

1.5. Дослідження технологій та інструментів для автоматизованого
тестування вебресурсів

1.6. Висновок до розділу 1

РОЗДІЛ 2 ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТИ ДЛЯ
АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ВЕБРЕСУРСІВ

2.1. Вступ

2.2. Selenium WebDriver

2.3. Мова програмування Java

2.4. Бібліотека RestAssured

2.5. Звітінговий інструмент ExtentReports

2.6. Патерн "Тестовий об'єкт"

2.7. Інші інструменти та фреймворки

2.8. Висновок до розділу 2

РОЗДІЛ 3 РОЗРОБКА ТА РЕАЛІЗАЦІЯ ПРОГРАМИ
АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ

3.1. Розробка та реалізація програми автоматизованого тестування

3.2. Вимоги до програми та функціональні можливості

3.3. Розробка та реалізація програми автоматизованого тестування

3.4. Висновок до розділу

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ТЕХНІЧНЕ ЗАВДАННЯ

ТЕСТУВАННЯ ДОДАТКА

ДОДАТКИ



6. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>	07.11.2022	07.11.2022
2.	<i>Розробка та затвердження завдання на роботу магістра (стац/заоч)</i>	13.12.2022	13.12.2022
3.	<i>Вступ та перелік літературних джерел</i>	24.02.2023	24.02.2023
4.	<i>Розробка технічного завдання</i>	15.03.2023	15.03.2023
5.	<i>Розділ 1. Огляд сучасного стану автоматизованого тестування вебресурсів</i>	10.04.2023	10.04.2023
6.	<i>Розділ 2. Технології та інструменти для автоматизованого тестування вебресурсів</i>	24.05.2023	24.05.2023
7.	<i>Розділ 3. Розробка та реалізація програми автоматизованого тестування</i>	06.09.2023	06.09.2023
8.	<i>Розробка програми та методики тестування</i>	18.10.2023	18.10.2023
9.	<i>Написання наукової статті</i>	17.05.2023	17.05.2023
10.	<i>Керівництво користувача</i>	25.10.2023	25.10.2023
11.	<i>Висновки та пропозиції</i>	01.11.2023	01.11.2023
12.	<i>Здача випускної кваліфікаційної роботи на кафедрі (перша перевірка)</i>	06.11.2023	06.11.2023
13.	<i>Підготовка автореферату та презентації доповіді</i>	06.11.2023	06.11.2023
14.	<i>Попередній захист випускної кваліфікаційної роботи</i>	20.11.2023 – 24.11.2023	20.11.2023 – 24.11.2023
15.	<i>Здача зброшурованої випускної кваліфікаційної роботи</i>	1.12.2023	27.11.2023
16.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>	2.12.2023	29.11.2023
17.	<i>Підготовка до публічного захисту випускної кваліфікаційної роботи</i>	05.12.2023- 06.12.2023	05.12.2023- 06.12.2023

7. Дата видачі завдання «13» грудня 2022 р.

8. Науковий керівник випускної кваліфікаційної роботи _____

Котенко Н.О.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми _____

Котенко Н.О.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент _____

Лаврінченко В.С.

(прізвище, ініціали, підпис)

АНОТАЦІЯ

Відповідно до мети дослідження випускна кваліфікаційна робота присвячена розробці та реалізації програми автоматизованого тестування вебресурсів. У сучасному світі, де вебдодатки та вебсайти є невідмінною частиною бізнесу та повсякденного життя, надійність та якість цих ресурсів є критичними факторами успіху. Автоматизоване тестування дозволяє забезпечити швидке та ефективне виявлення помилок, забезпечуючи високу якість та надійність вебдодатків.

Метою цієї кваліфікаційної роботи є розробка програмного забезпечення, що дозволить автоматизувати процес тестування вебресурсів. Для досягнення цієї мети, розглядаються основні проблеми та виклики, пов'язані з автоматизованим тестуванням вебдодатків, такі як взаємодія зі складними вебелементами.

У рамках дослідження, проводиться аналіз існуючих методик та інструментів автоматизованого тестування вебресурсів. На основі цього аналізу, розробляється програма, яка використовує сучасні технології та підходи до автоматизації тестування. Згадаан програма забезпечує можливість створення та виконання тестових сценаріїв, аналізу результатів тестування та генерації звітів. Підсумком цієї роботи є розроблена програма автоматизованого тестування вебресурсів, яка демонструє ефективність та надійність у виявленні помилок.

Випускна кваліфікаційна робота на тему Програма автоматизованого тестування вебресурсів містить 61 сторінку, 5 малюнків. Перелік використаних джерел налічує 13 найменувань.

Ключові слова: АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ, АВТОМАТИЗАЦІЯ, JAVA, SELENIUM, ВЕБДОДАТОК, API, TESTNG, RESTASSURED, EXTENTREPORTS, EXTENTSPARKREPORTER.

ABSTRACT

The Master's thesis is dedicated to the development and implementation of a program for automated testing of web resources. In today's world, where web applications and websites are an integral part of both business and everyday life, the reliability and quality of these resources are critical factors for success. Automated testing allows for rapid and efficient error detection, ensuring high quality and reliability of web applications.

The aim of this work is to develop software that automates the process of testing web resources. To achieve this goal, we examine the key issues and challenges associated with automated testing of web applications, such as interacting with complex web elements.

Within the scope of the research, an analysis of existing methodologies and tools for automated testing of web resources is conducted. Based on this analysis, a program is developed that utilizes modern technologies and approaches to automation testing. It enables the creation and execution of test scenarios, analysis of testing results, and generation of reports. The culmination of this work is the development of an automated testing program for web resources, which demonstrates efficiency and reliability in error detection.

The graduation thesis on the topic "Automated Testing Program for Web Resources" consists of 61 pages and includes 5 figures. The list of used sources comprises 13 references

Keywords: AUTOMATED TESTING, AUTOMATION, JAVA, SELENIUM, WEB APPLICATION, API, TESTNG, RESTASSURED, EXTENTREPORTS, EXTENTSPARKREPORTER.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

API – Інтерфейс Програмування Додатків;

HTML – Мова Розмітки Гіпертексту;

CSS – Каскадні Таблиці Стилів;

XML – Розширювана Мова Розмітки;

JSON – Формат Об'єктів JavaScript;

HTTP – Протокол передачі гіпертексту;

REST – Представлення Стану Ресурсу (архітектурний стиль);

URL – Уніфікований локатор ресурсів;

SQL – Мова структурованих запитів;

Selenium – Відомий набір інструментів для автоматизації тестування вебдодатків;

Java – Високорівнева об'єктно-орієнтована мова програмування;

IDEA – Універсальне інтегроване середовище розробки, також відоме як IntelliJ IDEA, яке підтримує розробку на Java та інших мовах програмування;

DOM - Об'єктна модель документу.

					<i>ДТЕУ 121 023-9.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	<i>Програма автоматизованого тестування вебресурсів</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			19.09.23		ПС	2	42
Керівник	Котенко Н.О.			19.09.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Гарант	Котенко Н.О.			19.09.23				
Розробив	Лавріненко В.С.			19.09.23				
					<i>Перелік умовних скорочень</i>			

ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1 ОГЛЯД СУЧАСНОГО СТАНУ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ВЕБРЕСУРСІВ.....	7
1.1. Загальний огляд вебресурсів.....	7
1.2. Значення автоматизованого тестування.....	8
1.3. Виклики автоматизованого тестування.....	9
1.4. Аналіз існуючих методик та підходів до автоматизованого тестування вебресурсів.....	10
1.5. Дослідження технологій та інструментів для автоматизованого тестування вебресурсів.....	11
1.6. Висновки до розділу 1.....	12
РОЗДІЛ 2 ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТИ ДЛЯ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ВЕБРЕСУРСІВ.....	14
2.1. Вступ до другого розділу.....	14
2.2. Selenium WebDriver.....	16
2.3. Мова програмування Java.....	16
2.4. Бібліотека RestAssured.....	17
2.5. Звітінговий інструмент ExtentReports.....	19
2.6. Патерн "Тестовий об'єкт".....	20
2.7. Інші інструменти та фреймворки.....	21
2.8. Висновки до розділу 2.....	22
РОЗДІЛ 3 РОЗРОБКА ТА РЕАЛІЗАЦІЯ ПРОГРАМИ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ.....	24
3.1. Розробка та реалізація програми автоматизованого тестування.....	24
3.2. Вимоги до програми та функціональні можливості.....	26
3.3. Розробка та реалізація програми автоматизованого тестування.....	27
3.4. Висновок до розділу 3.....	35
ВИСНОВКИ ТА ПРОПОЗИЦІЇ.....	38
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	41
ТЕХНІЧНЕ ЗАВДАННЯ.....	43
ПРОГРАМА ТА МЕТОДИКА ТЕСТУВАННЯ.....	46
ДОДАТКИ.....	55
АКТ ВПРОВАДЖЕННЯ РОБІТ.....	60

					<i>ДТЕУ 121 02з-9.МР</i>			
Зм.	Аркуш	№ докум.	Підпис	Дата	<i>Програма автоматизованого тестування вебресурсів</i>	Стаді	Аркуш	Аркушів
Зав. каф.	Криворучко О.В.			01.11.23		Зміст	3	42
Керівник	Котенко Н.О.			01.11.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Гарант	Котенко Н.О.			01.11.23				
Розробив	Лавріненко В.С			01.11.23				
					<i>Зміст</i>			

ВСТУП

У сучасному цифровому світі вебдодатки та вебресурси займають ключову роль у житті людей. Надійність, функціональність та безпека цих вебресурсів є надзвичайно важливими для задоволення потреб користувачів та успішного функціонування бізнесу.

Традиційні методи ручного тестування вимагають багато часу та ресурсів. Це призводить до затримок у розробці, невпевненості щодо якості продукту та збільшення ризиків. Як описано Gundecha (2015).

Автоматизоване тестування вебресурсів стає незамінним інструментом для забезпечення якості та ефективності розробки. Воно дозволяє автоматизувати процес тестування, зменшуючи час, зусилля та ризики, пов'язані з ручним тестуванням.

На основі проведеного аналізу та дослідження розроблена програма автоматизованого тестування вебресурсів. Ця програма включає можливості створення та виконання різноманітних тестових сценаріїв.

Дослідження також передбачає аналіз існуючих методик та інструментів тестування вебресурсів, зокрема, використання популярного і потужного інструменту Selenium та мови програмування Java для створення тестових сценаріїв та автоматизації тестового процесу.

У дипломній роботі використовувались середовища розробки та інструменти, такі як IntelliJ IDEA, Selenium WebDriver та мова програмування Java, що дозволило забезпечити зручну та потужну платформу для розробки та виконання автоматизованих тестів вебресурсів.

					<i>ДТЕУ 121 0з2-9.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	<i>Програма автоматизованого тестування вебресурсів</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			24.02.23		<i>В</i>	<i>4</i>	<i>42</i>
Керівник	Котенко Н.О.			24.02.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Гарант	Котенко Н.О.			24.02.23				
Розробив	Лавріненко В.С.			24.02.23	<i>Вступ</i>			

Мета дослідження: полягає в розробці та реалізації програми автоматизованого тестування вебресурсів.

Об'єктом дослідження є автоматизоване тестування вебресурсів.

Предмет дослідження – розробка та реалізація програми автоматизованого тестування вебресурсів.

Предметна область даного дослідження є автоматизоване тестування вебресурсів, зокрема вебсайтів та вебдодатків.

Пошуки шляхів досягнення мети обумовили необхідність визначення наступних завдань:

- дослідити сучасний стан автоматизованого тестування вебресурсів, включаючи технології та інструменти, що використовуються для цього;
- проаналізувати існуючі методики та підходи до автоматизованого тестування вебресурсів, включаючи використання різних інструментів та фреймворків;
- розробити програмне забезпечення для автоматизованого тестування вебресурсів, яке включатиме у себе можливості створення та виконання тестових сценаріїв, аналізу результатів тестування та генерації звітів;
- провести експерименти та тестування розробленої програми на реальних вебресурсах для оцінки її ефективності та надійності;
- створити висновки щодо результатів дослідження та розробленого програмного забезпечення, а також запропонувати можливі шляхи подальшого вдосконалення та розвитку в даній галузі.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			6

РОЗДІЛ 1

ОГЛЯД СУЧАСНОГО СТАНУ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ВЕБРЕСУРСІВ

1.1. Загальний огляд вебресурсів

Сьогодні вебресурси стали невід'ємною частиною суспільного та комерційного життя. Вони використовуються як засіб комунікації, поширення інформації, проведення електронних операцій та багато іншого. Вебресурси можуть бути у вигляді вебсайтів, вебдодатків, онлайн-магазинів, соціальних мереж, електронних сервісів та інших інтерактивних платформ.

Вебресурси працюють на основі клієнт-серверної архітектури, де клієнтська сторона взаємодіє з вебсервером для отримання необхідних даних та виконання різних операцій. Вебсайти можуть мати складну структуру, включаючи HTML-сторінки, CSS-стили, JavaScript-скрипти та інші компоненти, які забезпечують їх функціональність та взаємодію з користувачем.

Для забезпечення якості та надійності вебресурсів, важливо проводити тестування, яке дозволяє виявити помилки, недоліки та проблеми функціональності. Автоматизоване тестування вебресурсів є ефективним підходом, який дозволяє автоматизувати виконання тестових сценаріїв та отримувати швидкі та точні результати.

У цьому розділі роботи розглянемо загальний огляд вебресурсів, їх структуру, функціональність та особливості. Варто зазначити важливість автоматизації тестування та використання Selenium WebDriver з мовою програмування Java для досягнення цілей тестування вебресурсів.

					<i>ДТЕУ 121 023-9.MP</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	<i>Програма автоматизованого тестування вебресурсів Огляд сучасного стану автоматизованого тестування вебресурсів</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.		Криворучко О.В.		10.04.23		P1	7	42
Керівник		Котенко Н.О.		10.04.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Гарант		Котенко Н.О.		10.04.23				
Розробив		Лавріненко В.С		10.04.23				

1.2. Значення автоматизованого тестування

Автоматизоване тестування вебресурсів відіграє важливу роль у забезпеченні якості та надійності вебдодатків. Значення цього підходу полягає в наступних аспектах:

1. Підвищення ефективності: Автоматизоване тестування дозволяє виконувати широкий спектр тестів швидко та ефективно. Воно автоматизує повторювані завдання, звільняючи тестерів від рутинної роботи і дозволяючи їм сконцентруватись на більш складних аспектах тестування.

2. Підвищення точності: Автоматизовані тестові сценарії дозволяють уникнути людських помилок, які можуть стати причиною неправильного аналізу результатів тестування. Вони виконуються з високою точністю та узгодженістю, забезпечуючи надійний результат.

3. Виявлення помилок: Автоматизоване тестування дозволяє виявляти помилки та проблеми в ранній стадії розробки. Це дозволяє знизити витрати на виправлення помилок пізніше в процесі розробки, коли вони стають більш складними та коштовними для виправлення.

4. Забезпечення якості: Автоматизовані тестові сценарії дозволяють перевіряти функціональність, сумісність, продуктивність та інші аспекти вебресурсів з високою ретельністю. Це забезпечує високу якість продукту та задоволення користувачів.

5. Підтримка неперервної інтеграції: Автоматизоване тестування є важливою складовою процесу неперервної інтеграції, де вебдодатки регулярно тестуються на відповідність очікуванням. Це допомагає забезпечити стабільність і надійність продукту під час швидкого впровадження змін та оновлень. Як описано Mr Navneesh Garg (2014).

						Аркуш
					ДТЕУ 121 023-9.МР	8
Зм.	Аркуш	№ докум	Підпис	Дата		

Використання Selenium WebDriver та мови програмування Java надає потужний інструментарій для реалізації автоматизованого тестування вебресурсів. Цей підхід дозволяє забезпечити високу якість, ефективність.

1.3. Виклики автоматизованого тестування

Автоматизоване тестування вебресурсів, хоча і забезпечує багато переваг, також виникають деякі виклики та проблеми, які варто враховувати. Нижче описані деякі з них:

1. Динамічний контент: Вебресурси часто містять динамічний контент, який змінюється залежно від дії користувача або зміни стану системи. Це може ускладнити написання стабільних тестових сценаріїв, оскільки потрібно враховувати різні стани та умови, в яких може знаходитися вебресурс.

2. Синхронізація елементів: При автоматизованому тестуванні вебресурсів часто виникає потреба визначати момент, коли певний елемент стає доступним для взаємодії або зникати. Це вимагає врахування різних механізмів синхронізації, щоб забезпечити правильне взаємодію тестового скрипта з вебелементами.

3. Сумісність з різними браузерами: Вебдодатки повинні працювати на різних браузерах, таких як Chrome, Firefox, Safari тощо. Кожен браузер має свої особливості та різні реалізації вебстандартів, що може призводити до виникнення проблем при автоматизованому тестуванні.

4. Управління тестовими даними: Для виконання тестів потрібно мати достатньо тестових даних, щоб перевірити різні сценарії взаємодії з вебресурсом. Керування тестовими даними може стати складною задачею, особливо при великому обсязі тестів та необхідності їх підтримки та оновлення.

						Аркуш
					ДТЕУ 121 023-9.МР	9
Зм.	Аркуш	№ докум	Підпис	Дата		

5. Обробка помилок та винятків: Під час автоматизованого тестування можуть виникати помилки та винятки, такі як недоступність сервера, невідповідність очікуваних результатів тощо. Важливо мати механізми для виявлення та обробки таких ситуацій для забезпечення стабільності тестових скриптів. Як зазначено в книзі Mr Navneesh Garg (2014).

1.4. Аналіз існуючих методик та підходів до автоматизованого тестування вебресурсів

Основною метою даної роботи є розробка та реалізація програми автоматизованого тестування вебресурсів з використанням Selenium WebDriver та мови програмування Java. Головною метою є забезпечення ефективності, швидкості та якості процесу тестування вебдодатків.

Конкретні цілі дослідження включають:

1. Розглянути основні проблеми та виклики, пов'язані з автоматизованим тестуванням вебресурсів.
2. Проаналізувати існуючі методики та інструменти автоматизованого тестування вебресурсів.
3. Розробити програму автоматизованого тестування, яка використовує Selenium WebDriver та мову програмування Java.
4. Провести експериментальне тестування розробленої програми на різних вебресурсах для оцінки її ефективності та надійності.
5. Зробити висновки щодо результатів дослідження та запропонувати рекомендації для подальшого вдосконалення програми автоматизованого тестування.

Це дослідження спрямоване на покращення процесу тестування вебресурсів, забезпечуючи більшу ефективність, швидкість та надійність завдяки використанню автоматизації та потужних інструментів, таких як Selenium WebDriver та мова програмування Java.

						Аркуш
						10
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-9.МР	

Додатково, метою є ідентифікація можливих викликів та проблем, які можуть виникнути під час розробки та використання програми автоматизованого тестування.

Результати дослідження допоможуть розробити оптимальні підходи та стратегії для використання Selenium WebDriver та мови програмування Java в автоматизованому тестуванні вебресурсів. Це дослідження є важливим кроком у напрямку поліпшення процесу розробки та впровадження вебдодатків, забезпечуючи їх високу якість та надійність.

Аналіз існуючих методик та підходів до автоматизованого тестування вебресурсів є важливою частиною дослідження. Цей підрозділ присвячений огляду різних підходів та методик, що використовуються в індустрії для автоматизації тестування вебресурсів. Він включає аналіз різних фреймворків та інструментів, які найчастіше використовуються для реалізації автоматизованого тестування.

Дослідження також враховує переваги та недоліки кожного підходу, їх придатність для різних типів вебресурсів та особливості їх використання. Це дозволяє виявити найефективніші та найпідходящіші методи та підходи для розробки програми автоматизованого тестування вебресурсів з використанням Selenium WebDriver та мови програмування Java. Як описано Gundecha (2015).

1.5. Дослідження технологій та інструментів для автоматизованого тестування вебресурсів

У цьому підрозділі проводиться детальне вивчення технологій та інструментів, які використовуються для автоматизованого тестування вебресурсів. Передбачається дослідження Selenium WebDriver, який є одним з найпоширеніших інструментів для автоматизації вебтестування. Вивчення

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			11

включатиме огляд основних функціональних можливостей WebDriver та його інтеграцію з мовою програмування Java Mr Navneesh Garg (2014).

Додатково, будуть розглянуті інші технології та інструменти, які можуть допомогти у покращенні процесу автоматизованого тестування. Це можуть бути фреймворки для тестування, такі як TestNG або JUnit, які надають зручні інструменти для створення, виконання та управління тестовими сценаріями. Також будуть розглянуті інші корисні інструменти, які спрощують створення та підтримку автоматизованих тестів, наприклад, системи контролю версій, засоби реєстрації даних, засоби генерації звітів тощо.

В результаті вивчення технологій та інструментів, буде з'ясовано, які з них найбільш підходять для розробки програми автоматизованого тестування вебресурсів з використанням Selenium WebDriver та мови програмування Java.

Результатом цього розділу буде готове програмне забезпечення, яке здатне автоматизувати тестування вебресурсів та забезпечити високу якість та надійність вебдодатків.

1.6. Висновки до розділу 1

У цьому розділі надано загальний огляд теми магістерської дипломної роботи "Програма автоматизованого тестування вебресурсів". З'ясовано значення автоматизованого тестування вебресурсів і виявлено виклики, пов'язані з його реалізацією. Описано мету дослідження та наведено завдання, які необхідно виконати для досягнення цієї мети. У наступних розділах будуть детально розглянуті аналіз існуючих методик, вивчення технологій, розробка програмного забезпечення, експериментальне використання та аналіз результатів тестування.

У цьому розділі був проведений огляд теми магістерської дипломної роботи "Програма автоматизованого тестування вебресурсів". Встановлено, що автоматизоване тестування вебресурсів має велике значення для

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			12

підвищення якості та ефективності розробки, оскільки дозволяє швидко та ефективно виявляти помилки та недоліки вебдодатків.

З'ясовано, що існують різні методики та підходи до автоматизованого тестування вебресурсів, а також розглянуті різні технології та інструменти, що використовуються для цієї цілі. Відзначено, що використання мови програмування Java та інструменту Selenium є поширеними при розробці програм автоматизованого тестування.

В рамках магістерської дипломної роботи планується розробка програми автоматизованого тестування вебресурсів, яка базуватиметься на сучасних технологіях та підходах. Програма буде забезпечувати можливість створення та виконання тестових сценаріїв, аналізу результатів тестування та генерації звітів.

Очікується, що результати цієї роботи допоможуть покращити якість вебресурсів та зробити процес тестування більш ефективним. Розроблена програма автоматизованого тестування вебресурсів буде використовуватись для тестування реальних вебдодатків, а результати експериментів дозволять оцінити ефективність та надійність програми.

У наступних розділах роботи будуть розглянуті деталі аналізу існуючих методик, вивчення технологій та інструментів, проектування та реалізації програмного забезпечення, а також проведення експериментів та аналіз результатів тестування.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			13

РОЗДІЛ 2

ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТИ ДЛЯ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ВЕБРЕСУРСІВ

2.1. Вступ до другого розділу

У цьому розділі розглядаються різноманітні технології та інструменти, що застосовуються для автоматизованого тестування вебресурсів. З урахуванням швидкого розвитку вебтехнологій та збільшення складності вебдодатків, автоматизоване тестування стає необхідністю для забезпечення якості та ефективності розробки вебресурсів.

Метою цього розділу є огляд різних технологій та інструментів, які допомагають в реалізації програми автоматизованого тестування вебресурсів. Вивчення цих технологій та інструментів дозволить підібрати оптимальний набір засобів для реалізації поставлених цілей дослідження.

Розділ починається з огляду Selenium WebDriver, який є одним з провідних інструментів для автоматизованого тестування вебресурсів. Він надає потужні можливості для взаємодії з вебелементами та виконання різноманітних дій на вебсторінках.

Далі будуть розглянуті інші важливі компоненти, такі як мова програмування Java, яка використовується для реалізації тестових сценаріїв, а також інші інструменти та фреймворки, що сприяють ефективному тестуванню вебресурсів.

В цілому, цей розділ допоможе розширити розуміння про технології та інструменти, які використовуються для автоматизованого тестування

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 023-9.МР</i>			
Зав. каф.		Криворучко О.В.		24.05.23	<i>Програма автоматизованого тестування вебресурсів</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Котенко Н.О.		24.05.23		Р2	10	42
Гарант		Котенко Н.О.		24.05.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Розробив		Лавріненко В.С		24.05.23				

вебресурсів, що буде корисним при подальшій розробці програми автоматизованого тестування.

У цілому, цей розділ допоможе розширити розуміння про технології та інструменти, які використовуються для автоматизованого тестування вебресурсів. Це знання стане надзвичайно корисним при подальшій розробці програми автоматизованого тестування та в підготовці до тестування вебресурсів, сприяючи вдосконаленню процесу розробки та забезпечуючи високу якість вебдодатків.

Крім огляду інструментів та технологій, розділ також розглядає основні підходи до створення автоматизованих тестових сценаріїв для вебресурсів. Він вказує на важливість ретельного проектування тестових кейсів, обрання оптимальних стратегій тестування та створення надійних і підтримуваних тестових наборів.

Однією з ключових тем розділу є використання мови програмування Java для створення тестових сценаріїв. Java є популярним вибором для автоматизації тестування завдяки своїй платформонезалежності, багатому екосистемному середовищу та великій кількості бібліотек і фреймворків для тестування. Розділ надає огляд базових концепцій Java, необхідних для створення автоматизованих тестів, а також приклади коду для демонстрації цих концепцій у практиці.

Ці знання стануть надзвичайно корисними для розробників, тестувальників та інженерів якості, які прагнуть забезпечити високу якість та надійність своїх вебдодатків через ефективне автоматизоване тестування.

						ДТЕУ 121 02з-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			15

2.2. Selenium WebDriver

Selenium WebDriver є одним з найпоширеніших інструментів для автоматизованого тестування вебресурсів. Він забезпечує розширені можливості для взаємодії з вебелементами, навігації по вебсторінках, введення даних та перевірки результатів.

Цей інструмент підтримує різноманітні браузерери, такі як Chrome, Firefox, Safari, та дозволяє виконувати тестування на різних платформах. Використання Selenium WebDriver дозволяє автоматизувати тестові сценарії та прискорити процес перевірки функціональності та стабільності вебресурсів.

Selenium WebDriver базується на програмному інтерфейсі для керування браузером і надає розширені можливості для елементарного та складного взаємодії з елементами сторінки, знаходження елементів за різними атрибутами, виконання кліків, заповнення полів вводу, навігації по сторінці та багато іншого.

До переваг Selenium WebDriver входить його зручність в поєднанні з мовою програмування Java, що дозволяє розробникам створювати потужні тестові сценарії та використовувати багато функцій для забезпечення надійності та стійкості вебресурсів.

Загальна інтеграція Selenium WebDriver з мовою програмування Java і деякі інші функціональні можливості цього інструменту роблять його потужним інструментом для автоматизованого тестування вебресурсів. В наступних розділах будуть розглянуті інші технології та інструменти, які можна поєднувати з Selenium WebDriver для створення програми автоматизованого тестування. Як описано Gundecha (2015).

2.3. Мова програмування Java

Мова програмування Java є однією з найпопулярніших мов у сфері розробки програмного забезпечення, включаючи автоматизоване тестування

						Аркуш
						16
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

вебресурсів. Вона відома своєю простотою, надійністю та переносимістю, що робить її ідеальним вибором для розробки програмного забезпечення тестування.

Java пропонує широкий спектр функцій та бібліотек, які сприяють створенню потужних тестових сценаріїв та автоматизації вебресурсів. Вона має багато вбудованих інструментів, що спрощують розробку, таких як керування пам'яттю, обробка виключень, робота зі збірниками та інші.

Java є об'єктно-орієнтованою мовою програмування, що дозволяє розробникам структурувати свій код у вигляді класів та об'єктів, сприяючи повторному використанню коду та покращенню його читабельності. Крім того, Java підтримує механізм успадкування, поліморфізму та інтерфейсів, що сприяє гнучкості та розширюваності програм.

Java також має розгалужену та екосистему розширень, які дозволяють розробникам використовувати додаткові бібліотеки та інструменти для покращення процесу автоматизованого тестування. У контексті роботи з Selenium WebDriver, Java надає зручний API, який дозволяє взаємодіяти зі збірником Selenium та забезпечує багато корисних функцій для автоматизації тестування вебресурсів.

Висока продуктивність, широкі можливості та велика спільнота розробників роблять мову програмування Java привабливим вибором для реалізації програми автоматизованого тестування вебресурсів. Її синтаксис, інструменти та екосистема допомагають розробникам ефективно створювати та підтримувати надійні та розширювані автоматизовані тестові сценарії.

2.4. Бібліотека RestAssureda

Бібліотека RestAssured є потужним інструментом для автоматизованого тестування вебсервісів та перевірки їхнього функціоналу. Ця бібліотека, розроблена на мові програмування Java, спеціалізується на тестуванні RESTful

						Аркуш
					ДТЕУ 121 023-9.МР	17
Зм.	Аркуш	№ докум	Підпис	Дата		

API, і вона входить до складу інструментів, які допомагають забезпечити надійність та якість вебдодатків.

Основні переваги використання бібліотеки RestAssured включають в себе:

- Простоту та зручність використання: RestAssured надає простий та інтуїтивно зрозумілий інтерфейс для взаємодії з API. Завдяки цьому, розробникам легко створювати запити та перевіряти відповіді без великих зусиль.
- Підтримку RESTful стандартів: RestAssured розроблена з урахуванням основних принципів REST, що дозволяє ефективно тестувати ресурси, методи, коди стану і інші аспекти RESTful API.
- Можливість виконання різноманітних запитів: RestAssured дозволяє виконувати HTTP-запити, такі як GET, POST, PUT, DELETE та інші, а також передавати параметри, тіла запиту і заголовки.
- Перевірку відповідей: За допомогою RestAssured можна легко перевіряти відповіді від сервера, включаючи коди статусу, вміст, формати даних (наприклад, JSON або XML) та інші параметри.
- Підтримку автентифікації та авторизації: RestAssured підтримує різні методи автентифікації, включаючи базову автентифікацію, авторизацію з використанням жетонів та інші.
- Можливість валідації JSON та XML: RestAssured має зручні засоби для валідації JSON та XML відповідей, дозволяючи перевіряти структуру та значення даних.
- Легку інтеграцію з іншими інструментами: RestAssured можна легко інтегрувати з іншими інструментами для тестування, такими як TestNG або JUnit.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			18

В загальному виразі, бібліотека RestAssured є надійним інструментом для автоматизованого тестування вебсервісів на мові програмування Java, який спрощує процес тестування RESTful API і допомагає забезпечити високу якість програмного забезпечення як зазначено в книзі Alan Richardson (2017).

2.5. Звітинговий інструмент ExtentReports

ExtentReports – це потужний інструмент для створення звітів та візуалізації результатів автоматизованих тестів у фреймворках на мові програмування Java. Цей інструмент забезпечує можливість створення докладних та професійних звітів, що допомагають в розумінні результатів тестування та виявленні проблем в програмному забезпеченні.

Основні переваги використання ExtentReports включають в себе:

- Зрозумілі та ефективні звіти: ExtentReports дозволяє створювати красиві та інтерактивні звіти з деталями про виконані тести, результатами, статистикою та графіками. Це робить звіти більш зрозумілими та корисними для всієї команди.
- Підтримка різних форматів виведення: ExtentReports підтримує виведення результатів тестування в різні формати, такі як HTML, PDF, або друк на папері. Це робить звіти доступними для всіх учасників проекту.
- Інтеграція з різними фреймворками та інструментами: ExtentReports легко інтегрується з популярними фреймворками для автоматизованого тестування, такими як TestNG або JUnit, а також з іншими інструментами, такими як Selenium WebDriver.
- Візуалізація даних: Інструмент надає можливість візуалізувати дані в графіках та діаграмах, що допомагає виділити ключові показники та проблеми.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			19

- Можливість додавати прикріплені файли: ExtentReports дозволяє додавати прикріплені файли до звітів, що корисно для відстеження стану програмного забезпечення, наприклад, знімків екрана або лог-файлів.
- Множинна підтримка локалізації: Інструмент підтримує можливість створення звітів на різних мовах для більш широкого кола користувачів.

Узагальнюючи, ExtentReports є потужним інструментом для створення докладних, інформативних та ефективних звітів для автоматизованих тестів на мові програмування Java. Він сприяє покращенню процесу тестування та полегшує аналіз результатів для всієї команди розробників.

2.6. Патерн "Тестовий об'єкт"

Патерн "Тестовий об'єкт" (Test Object Pattern) є популярним інструментом у світі автоматизованого тестування, особливо в контексті інтерфейсного тестування вебдодатків. Цей патерн допомагає впоратися з динамічно генерованим вебконтентом та взаємодіяти з елементами вебсторінки, що завантажуються асинхронно або за допомогою JavaScript.

Основні ідеї та концепції, що лежать в основі патерну Тестового об'єкта:

- Представлення сторінки як об'єкта: Кожна сторінка вебдодатку та її елементи (кнопки, поля вводу, таблиці тощо) моделюються як об'єкти в програмному коді тестувальних сценаріїв.
- Методи для взаємодії: Для кожного елемента на сторінці створюються методи, які дозволяють здійснювати дії над цим елементом (наприклад, натискання на кнопку, введення тексту тощо).

						Аркуш
						20
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

- Загальний доступ до сторінки: Тестовий об'єкт надає загальний доступ до всіх елементів сторінки та її функціональності.
- Автоматизація очікувань: В паттерні Тестового об'єкта часто використовуються методи для перевірки стану сторінки та її елементів, що дозволяє автоматизувати перевірку очікуваних результатів.
- Сховання деталей реалізації: Деталі роботи з DOM (Document Object Model) та іншими технічними аспектами реалізації сторінки приховуються внутрішньо в тестовому об'єкті, що робить тестовий код більш стабільним та зрозумілим.
- Підтримка багатьох браузерів і платформ: Тестові об'єкти можуть бути розроблені таким чином, щоб підтримувати різні браузери та платформи без змін у тестовому коді.
- Зручність у використанні та підтримці: Підходяща реалізація патерну Тестового об'єкта допомагає зробити тестовий код більш зрозумілим та підтримуваним, зменшуючи витрати на обслуговування.

Загалом, патерн Тестового об'єкта є корисним інструментом для реалізації структурованих та стабільних тестів вебдодатків, які взаємодіють зі складним та динамічним вебконтентом. Він допомагає розробникам автоматизованих тестів підтримувати високу якість програмного забезпечення та полегшує роботу з елементами вебсторінок.

2.7. Інші інструменти та фреймворки

Поміж Selenium WebDriver та мовою програмування Java існує широкий спектр інших інструментів та фреймворків, які допомагають покращити процес автоматизованого тестування вебресурсів. Наведемо кілька популярних з них:

						Аркуш
						21
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

TestNG: Це розширений фреймворк для тестування, який надає розширені можливості управління тестовими сценаріями, підтримку анотацій, групування тестів, залежності між тестами та багато іншого. Використання TestNG разом з Selenium дозволяє легко організувати та виконувати тестові набори.

Cucumber: Це інструмент, який використовує природну мову для опису тестових сценаріїв. Він надає можливість писати тестові кейси у зрозумілому для всіх стейкхолдерів форматі. Cucumber співпрацює з Selenium, дозволяючи створювати легко зрозумілі та легко підтримувані тестові сценарії.

JUnit: Це популярний фреймворк для модульного тестування у мові Java. Він дозволяє створювати та виконувати набори тестів, перевіряти очікувані результати та аналізувати їх стан. Використання JUnit разом з Selenium спрощує створення та виконання автоматизованих тестів.

Appium: Це відкрите програмне забезпечення для автоматизації тестування мобільних додатків, яке також підтримує автоматизацію вебресурсів на мобільних пристроях. Appium дозволяє тестувати вебдодатки на різних платформах, таких як Android та iOS, використовуючи Selenium WebDriver API.

Ці інструменти та фреймворки розширюють можливості Selenium та Java, допомагаючи забезпечити ефективніше, швидше та надійніше автоматизоване тестування вебресурсів.

2.8. Висновки до розділу 2

У цьому розділі розглянуто основні інструменти та технології, які використовуються при розробці програми автоматизованого тестування вебресурсів. Selenium WebDriver, мова програмування Java та інші фреймворки дозволяють створити потужну та гнучку систему для автоматизації тестування.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			22

Selenium WebDriver є одним з найпопулярніших інструментів для автоматизованого тестування вебресурсів. Він надає широкі можливості для маніпуляції з елементами вебсторінки, виконання дій користувача та перевірки очікуваних результатів. З використанням Selenium WebDriver можна створювати стабільні та надійні тестові скрипти.

Мова програмування Java є однією з найпоширеніших мов для розробки програмного забезпечення. Вона має потужний набір бібліотек, фреймворків та інструментів, які сприяють розробці та підтримці автоматизованих тестів. Використання Java дозволяє забезпечити надійність, переносимість та розширюваність програми автоматизованого тестування.

Додатково, інші інструменти та фреймворки, такі як TestNG, Cucumber та JUnit, допомагають зручніше організувати тестові сценарії, забезпечити повноту тестового покриття та полегшити співпрацю між командою розробників та тестувальників.

Загалом, використання Selenium WebDriver, мови програмування Java та додаткових інструментів дозволяє створити потужну програму автоматизованого тестування. Як описано Gundecha (2015).

						Аркуш
						23
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

РОЗДІЛ 3

РОЗРОБКА ТА РЕАЛІЗАЦІЯ ПРОГРАМИ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ

3.1. Розробка та реалізація програми автоматизованого тестування

У цьому розділі детально описані етапи розробки та реалізації програми автоматизованого тестування вебресурсів для перевірки актуальних курсів валют. Проект був розбитий на логічні компоненти, що дозволило забезпечити структурованість та зручну організацію всього процесу.

Початковий етап включав вибір необхідних технологій та інструментів. Для автоматизованої взаємодії з вебдодатками використовувався Selenium WebDriver, що дозволяє автоматизувати взаємодію з елементами сторінок. Для взаємодії з API Національного банку України був обраний інструмент RestAssured. Ця комбінація технологій надала програмі можливість взаємодіяти з обома джерелами даних.

Наступний етап передбачав розробку модулів та компонентів програми. З метою забезпечення кращої підтримки та обслуговування використано паттерн Page Object, що дозволило відокремити логіку взаємодії з інтерфейсом від основної логіки тестів. Це спростило додавання нових функцій та змін до програми без необхідності змінювати всю структуру.

На наступному етапі була проведена реалізація тестових сценаріїв. Кожен тестовий сценарій включав в себе отримання курсів валют з вебсайту

					<i>ДТЕУ 121 023-9.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		06.09.23	<i>Програма автоматизованого тестування вебресурсів</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Котенко Н.О.		06.09.23		<i>РЗ</i>	<i>24</i>	<i>42</i>
Гарант		Котенко Н.О.		06.09.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Розробив		Лавріненко В.С		06.09.23				
					<i>Розробка та реалізація програми автоматизованого тестування</i>			

та API, порівняння отриманих даних та виведення результату тесту відповідно до стандартів звіту ExtentReports. Це забезпечило автоматичну перевірку актуальності даних та коректність їх відображення.

Останнім етапом було тестування та налагодження програми. Були проведені автоматизовані тести, результати яких порівнювалися з даними з джерел. При виявленні розбіжностей аналізувалася логіка та вносились корекції. Це дозволило забезпечити надійність та точність програми.

Завершальним етапом було документування програмного продукту, описання його структури, функціональності та використаних технологій. Це створило базу для подальшого розвитку та підтримки програми. Всі ці етапи спільно забезпечили розробку ефективного та функціонального фреймворку для автоматизованого тестування вебресурсів з використанням валютних курсів як тестової інформації.

Тестова інфраструктура: Була врахована можливість покриття коду автоматизованими тестами. Велика увага була приділена розробці тестів, що дозволяє ретельно перевіряти роботу програми та оперативно виправляти помилки. Це забезпечує більш високу якість програми та підвищує рівень довіри до неї.

Архітектура була розроблена з використанням сучасних стандартів розробки та кращих практик, що спрямовані на досягнення максимальної ефективності, гнучкості та зручності в подальшій розробці та підтримці програми. Впровадження цих принципів допомагає забезпечити стабільну та ефективну роботу програми в довгостроковій перспективі. Чітка взаємодія між компонентами допомагає зручно впроваджувати нові модулі чи можливості без необхідності внесення значних змін.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			25

3.2. Вимоги до програми та функціональні можливості

Під час розробки програми автоматизованого тестування вебресурсів були визначені важливі вимоги та функціональні можливості для забезпечення найвищої якості та ефективності процесу тестування.

Першою ключовою вимогою є здатність програми отримувати актуальні курси валют. Це досягається через взаємодію з вебсайтом та API Національного банку України. Отримання точних та актуальних даних є необхідною передумовою для об'єктивної оцінки коректності ресурсів.

Другою важливою вимогою є реалізація автоматизованого тестування. Програма повинна самостійно проводити тестування, порівнюючи дані з вебсайту та API. Це забезпечує консистентність та надійність результатів, уникнення помилок та спрощення процесу тестування для користувачів.

Додатково, вимоги передбачають візуалізацію результатів тестів. Ця функція допомагає користувачам зручно аналізувати результати та вчасно реагувати на можливі відхилення. Використання звіту ExtentReports дозволяє створити зрозумілий та інформативний звіт, який допомагає легко визначити результати тестів.

У підсумку, розроблена програма відповідає важливим вимогам та функціональним можливостям. Вона забезпечує надійне отримання даних, автоматизоване порівняння та аналіз результатів, спрощуючи процес тестування та допомагаючи забезпечити якість вебресурсів.

Були докладно визначені та обрані компоненти, що найкращим чином відповідають вимогам розробки програми автоматизованого тестування вебресурсів. Основні технології та інструменти включають:

Selenium WebDriver: Цей інструмент використовується для автоматизованої взаємодії з вебдодатками. Його можливості дозволяють відтворити дії користувача на вебсайті, виконати перевірки та отримати необхідну інформацію.

						Аркуш
						26
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

RestAssured: Ця бібліотека дозволяє здійснювати взаємодію з API, зокрема з API Національного банку України. Вона надає зручний спосіб виконання HTTP-запитів, обробки відповідей та отримання необхідних даних для порівняння.

Обрані технології та інструменти є популярними та добре підтримуються спільнотою розробників. Вони відповідають завданням автоматизованого тестування вебресурсів та забезпечать надійність та ефективність роботи програми.

Під час вибору технологій та інструментів, була врахована їхня сумісність, гнучкість та можливість інтеграції. Selenium WebDriver був обраний через його широкий спектр можливостей у взаємодії з різноманітними вебдодатками. Використання RestAssured дозволяє зручно та ефективно взаємодіяти з API Національного банку України.

Вибір цих інструментів забезпечує стабільну роботу програми та підвищує рівень автоматизації в процесі тестування. Ці технології є відкритими та добре підтримуються, що дозволяє вдосконалювати програму в майбутньому та адаптувати її до змінних вимог і вебресурсів.

3.3. Розробка та реалізація програми автоматизованого тестування

У цьому розділі наведено код програми автоматизованого тестування, яка розроблена для перевірки курсів валют на вебресурсі та API Національного банку України. Код був написаний мовою програмування Java з використанням бібліотек Selenium WebDriver для взаємодії з вебсайтом та бібліотеки RestAssured для роботи з API.

Нижче наведений приклад роботи програми, який демонструє взаємодію з вебсайтом та API, отримання курсів валют та проведення автоматизованого тестування:

						Аркуш
						27
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

Приклад роботи програми наведено на рисунку 3.1.

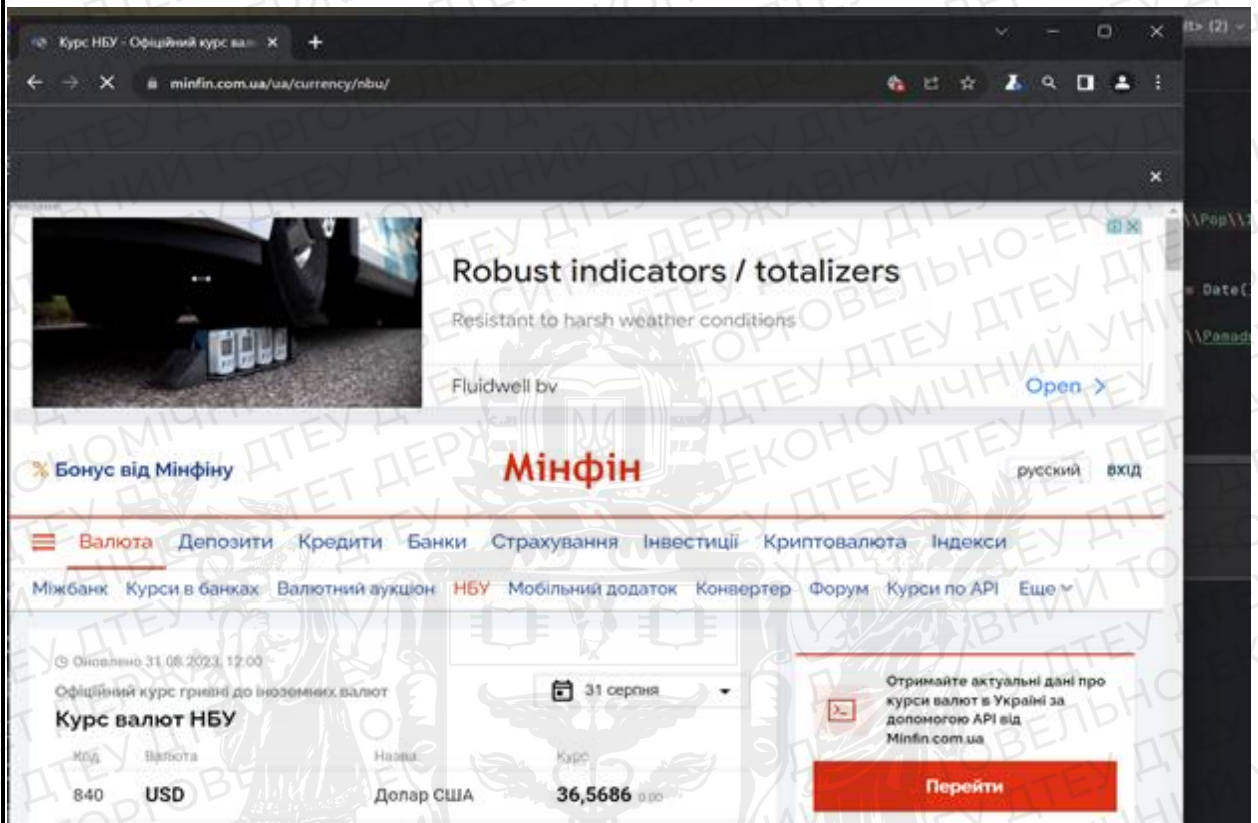


Рис 3.1 Сайт мінфіну

Запуск та відкриття браузера. Програма отримує через API офіційний курс валют з сайту НБУ далі, програма звертається до сайту знаходить курс доллару США на сайті minfin.com.ua (рис. 3.2.)

					ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		28

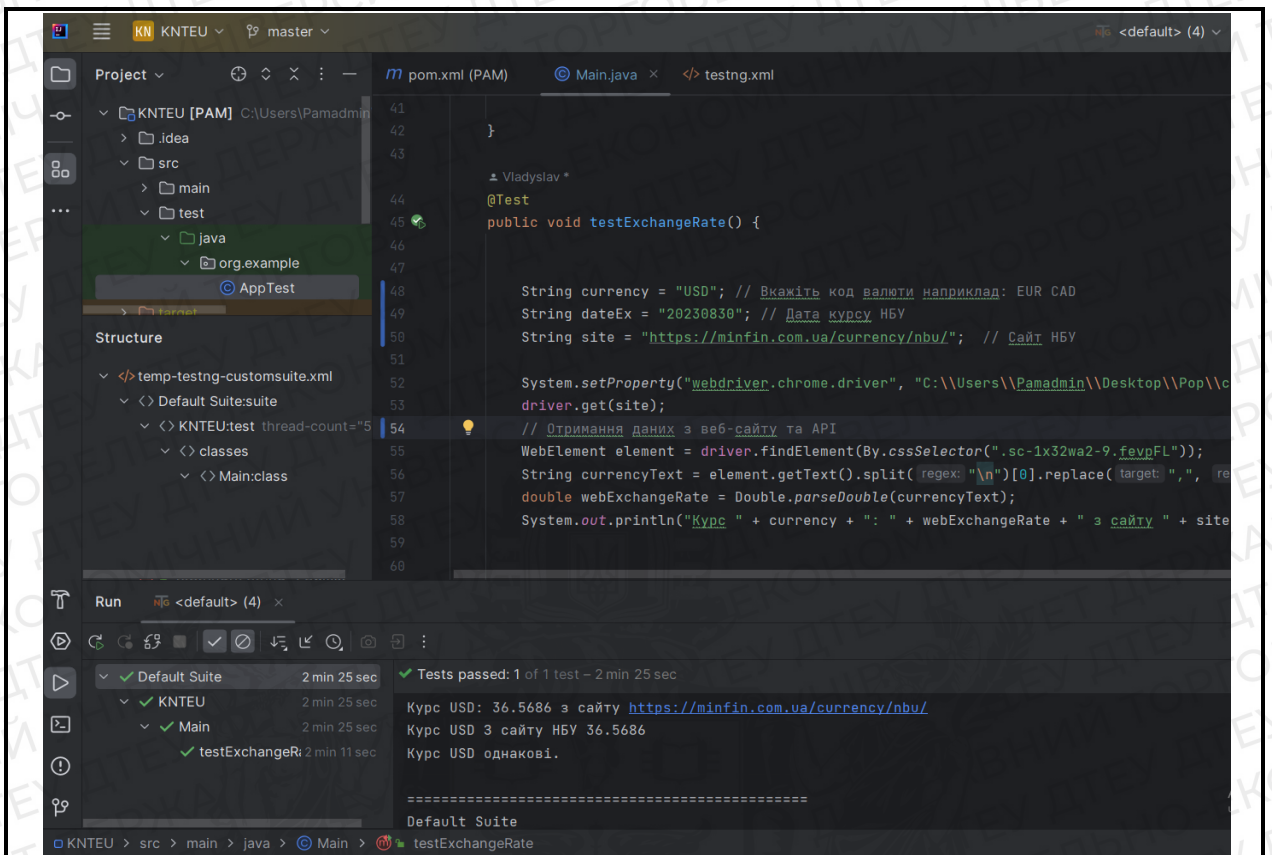


Рис 3.2 Запущена програма в IntelliJ IDEA

Отримуємо в консолі результати на кожну операцію, для того щоб відслідкувати помилку на кожному етапі виконання коду (рис. 3.3).

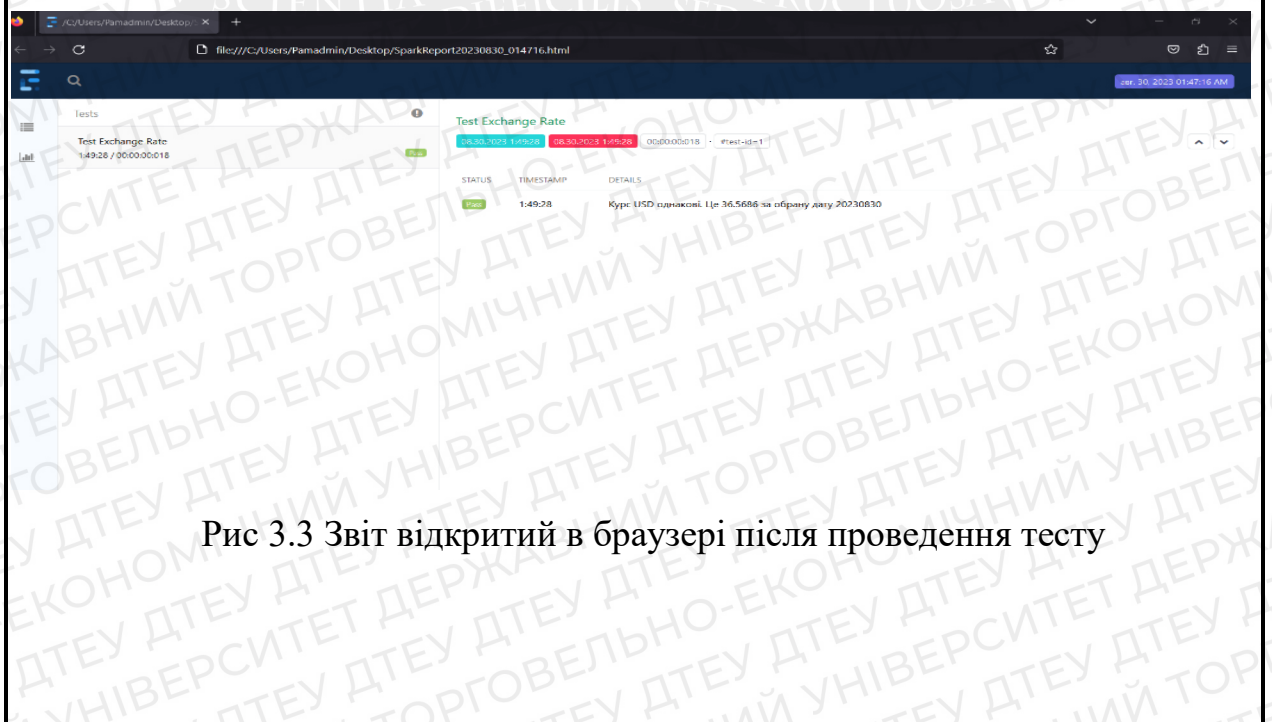


Рис 3.3 Звіт відкритий в браузері після проведення тесту

					Аркуш
					29
Зм.	Аркуш	№ докум	Підпис	Дата	

ДТЕУ 121 023-9.МР

1) @Test анотація:

Ця анотація позначає метод як тестовий.

Метод testExchangeRate буде викликаний для виконання тестування курсу валюти.

2) Змінні:

currency: Рядок, що містить код валюти, для якого перевіряється курс (у цьому випадку, "USD").

dateEx: Рядок, що містить дату, на яку потрібно отримати курс (у цьому випадку, "20230830").

site: Рядок, що містить URL-адресу вебресурсу Національного банку України, де пошук виконується курсу валюти.

3) System.setProperty:

Встановлює шлях до вебдрайвера браузера Chrome.

4) driver.get(site):

Відкриває вебсайт за URL-адресою, вказаною в змінній site.

5) Отримання даних з вебсайту:

Знаходить елемент на сторінці, який містить курс валюти та витягує текст з цього елемента.

Конвертує текст у числовий формат та зберігає його у змінну webExchangeRate.

6) Отримання даних з API:

Формує URL-адресу для запиту до API Національного банку України, використовуючи код валюти та дату.

Відправляє GET-запит до цього API та отримує відповідь.

Витягує курс валюти з отриманої відповіді та зберігає його у змінну apiExchangeRate.

7) Закриття вебдрайвера:

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			30

Менеджер часу очікування сторінки на виконання скриптів налаштовується на 30 секунд.

Закрива вебдрайвер.

8) Створення звіту:

Створює об'єкт ExtentReports, який відповідає за створення звітів.

Додає ExtentSparkReporter для генерації HTML-звіту.

9) Порівняння курсів та логування:

Порівнює курс валюти, отриманий з вебсайту і API.

Виводить результат порівняння в консоль та додає в звіт за допомогою бібліотеки ExtentReports.

Якщо курси валют співпадають (різниця менше 0.01), тест вважається успішним і позначається як PASS. В іншому випадку, тест вважається неуспішним і позначається як FAIL.

10) @AfterClass анотація:

Ця анотація позначає метод, який виконується після виконання всіх тестів.

Метод tearDown виконує завершальні дії, такі як очищення звіту та закриття вебдрайвера.

Під час розробки програми автоматизованого тестування вебресурсів були визначені важливі вимоги та функціональні можливості для забезпечення найвищої якості та ефективності процесу тестування.

Першою ключовою вимогою є здатність програми отримувати актуальні курси валют. Це досягається через взаємодію з вебсайтом та API Національного банку України. Отримання точних та актуальних даних є необхідною передумовою для об'єктивної оцінки коректності ресурсів.

Другою важливою вимогою є реалізація автоматизованого тестування. Програма повинна самостійно проводити тестування, порівнюючи дані з

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			31

вебсайту та API. Це забезпечує консистентність та надійність результатів, уникнення помилок та спрощення процесу тестування для користувачів.

Додатково, вимоги передбачають візуалізацію результатів тестів. Ця функція допомагає користувачам зручно аналізувати результати та вчасно реагувати на можливі відхилення. Використання звіту ExtentReports дозволяє створити зрозумілий та інформативний звіт, який допомагає легко визначити результати тестів.

Перспективи та майбутні розширення: зокрема, планується інтеграція з базою даних для збереження історії курсів валют. Це дозволить відстежувати зміни курсів протягом різних періодів та проводити аналіз динаміки. Додатково, можна розглянути можливість автоматичної побудови графіків для візуалізації цих змін.

Ще однією перспективною можливістю є додавання автоматичної розсилки на електронну пошту. Це дозволить користувачам отримувати сповіщення про зміни курсів валют за певний період, що може бути корисним для трейдерів та аналітиків.

Крім того, інтеграція з процесом CI/CD може покращити управління та автоматизацію розгортання програми. Це дозволить швидше впроваджувати нові функції та зміни.

Загалом, подальший розвиток програми може значно розширити її можливості, зробити її більш корисною та зручною для користувачів, а також сприяти вдосконаленню фінансового аналізу та прийняттю рішень.

Інтеграція з Базою Даних: Один із можливих напрямів розширення полягає у впровадженні інтеграції з базою даних. Збереження історії валютних курсів дозволить аналізувати їх динаміку за різні періоди, що може бути надзвичайно корисним для трейдерів та аналітиків. Створення структурованої бази даних спростить відстеження змін та сприятиме більш детальному фінансовому аналізу.

						Аркуш
					ДТЕУ 121 023-9.МР	32
Зм.	Аркуш	№ докум	Підпис	Дата		

Автоматична Розсилка на Пошту: Ще однією значущою можливістю є впровадження автоматичної розсилки повідомлень на електронну пошту. Це може відігравати важливу роль для трейдерів та інвесторів, які прагнуть отримувати оперативну інформацію про зміни валютних курсів. Автоматизовані сповіщення допоможуть вчасно реагувати на ринкові зміни та впливати на прийняття рішень.

Інтеграція з CI/CD: Західний розвиток індустрії розробки програм передбачає інтеграцію з системами Continuous Integration та Continuous Deployment (CI/CD). Це відкриває можливість автоматизувати процес розгортання програми, впроваджувати нові функції та виправлення безпосередньо використовуючи засоби автоматичної збірки та розгортання.

Інтеграція з Базою Даних:

впровадження інтеграції програми з базою даних є однією з ключових перспектив для подальшого розвитку. Це надасть програмі можливість зберігати, організувати та аналізувати історію курсів валют з різних джерел. Ця функціональність відкриє широкі можливості для трейдерів, аналітиків та інших користувачів, які цікавляться фінансовою сферою.

Переваги Інтеграції з Базою Даних: Зберігання Історії: Інтеграція з базою даних дозволить зберігати історію курсів валют за різні періоди часу. Це забезпечить можливість аналізу змін валютних курсів та виявлення тенденцій.

Аналітичні Звіти: Завдяки базі даних можна буде побудувати аналітичні звіти, графіки та діаграми, які допоможуть в більш детальному розумінні динаміки валютних ринків.

Зручність Клієнтів: Для клієнтів це може стати інструментом для визначення оптимальних точок входу та виходу з угод на основі історичних даних.

Приклад Коду для Інтеграції з Базою Даних:

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			33

```

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.List;

public class DatabaseIntegration {

    private static final String DB_URL = "jdbc:mysql://localhost:3306/currency_history";
    private static final String DB_USER = "your_username";
    private static final String DB_PASSWORD = "your_password";

    public void saveExchangeRatesToDatabase(List<ExchangeRate> rates) {
        try (Connection connection = DriverManager.getConnection(DB_URL, DB_USER, DB_PASSWORD)) {
            String query = "INSERT INTO exchange_rates (currency, rate, date) VALUES (?, ?, ?)";

            for (ExchangeRate rate : rates) {
                try (PreparedStatement preparedStatement = connection.prepareStatement(query)) {
                    preparedStatement.setString(1, rate.getCurrency());
                    preparedStatement.setDouble(2, rate.getRate());
                    preparedStatement.setDate(3, new java.sql.Date(rate.getDate().getTime()));
                    preparedStatement.executeUpdate();
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}

```

Приклад можливого використання.

Для реалізації інтеграції з базою даних у програмі автоматизованого тестування валютних курсів може бути використана технологія Java Database Connectivity (JDBC). Цей підхід дозволяє взаємодіяти з реляційними базами даних, такими як MySQL, PostgreSQL тощо, і забезпечує збереження та організацію даних для подальшого аналізу.

У вищенаведеному коді створено клас DatabaseIntegration, який містить метод saveExchangeRatesToDatabase, відповідний за збереження обмінних курсів до бази даних. Для встановлення з'єднання з базою даних використовуються параметри DB_URL, DB_USER та DB_PASSWORD.

Метод saveExchangeRatesToDatabase отримує список об'єктів ExchangeRate, які містять інформацію про валютний код, курс та дату. Параметри для запиту SQL заповнюються відповідно до полів об'єкта

						Аркуш
						34
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

ExchangeRate. Метод executeUpdate() виконує запит для додавання даних до бази.

Ця інтеграція дозволить зберігати історію курсів валют у базі даних, що в свою чергу дозволить аналізувати їх динаміку за різні періоди часу. Збереження даних у базі спростить роботу з історичними даними та забезпечить більш широкі можливості аналізу для прийняття рішень на основі фінансової інформації.

3.4. Висновок до розділу 3

У результаті даного дослідження та розробки програми автоматизованого тестування валютних курсів можна підсумувати наступне. Розроблена програма успішно виконує свою мету, забезпечуючи отримання актуальних даних з вебресурсів та API. Вона спрощує процес тестування та перевірки, покращуючи точність та ефективність.

Цей проект показав значимість автоматизації валютного аналізу та можливості використання сучасних інструментів для автоматизованого тестування. Отримані результати підтверджують важливість надійних та точних методів перевірки фінансової інформації.

Рекомендації для подальших досліджень включають розширення функціональності та вдосконалення інтерфейсу програми, а також розгляд можливості інтеграції з процесом CI/CD. Такі доповнення можуть покращити якість та використання цієї програми у практичних умовах.

Розробка програми: Початковий етап розробки включав аналіз вимог до функціональності програми. Завдяки чіткому визначенню цілей та завдань, була створена концептуальна модель програми, включаючи її архітектурну структуру. Після аналізу вимог було вибрано необхідні технології для реалізації програми. Використання Selenium WebDriver та RestAssured дозволило ефективно поєднати веб та API-тестування.

						Аркуш
					ДТЕУ 121 023-9.MP	35
Зм.	Аркуш	№ докум	Підпис	Дата		

Використання паттерну Page Object допомогло уникнути дублювання коду та спростити обслуговування.

Досягнуті результати: Розроблена програма успішно виконує свою мету – забезпечує отримання актуальних даних з вебресурсів та API для подальшого аналізу валютних курсів. Автоматизація процесу тестування та перевірки курсів дозволяє виконувати ці завдання швидко та ефективно, зменшуючи можливість помилок. Програма виявилась надійним інструментом для виявлення розбіжностей між курсами валют з різних джерел. Це підвищує достовірність та точність фінансового аналізу. Крім того, вона може слугувати інструментом для сповіщення про невідповідності та забезпечення своєчасного реагування.

Використання сучасних інструментів для автоматизованого тестування виявилось дуже результативним. Селектори Selenium та RestAssured дозволили ефективно інтегрувати різні види тестування в одну програму. Модульна структура програми зробила її гнучкою та зручною для розширення.

Можливі подальші розширення та вдосконалення: Для максимізації користі від програми та покращення її функціональності та зручності користування, можна розглянути наступні напрямки розвитку:

1. Інтеграція з базою даних: Введення бази даних дозволить зберігати та відстежувати історію курсів валют з різних джерел. Це допоможе зробити аналіз більш глибоким та точним, а також дозволить користувачам переглядати зміни курсів валют за різні періоди часу.

2. Автоматична розсилка на пошту: Додавання функціоналу для автоматичної розсилки сповіщень на електронну пошту може бути корисним для користувачів, які бажають отримувати своєчасну інформацію про зміни курсів валют. Наприклад, програма може надсилати щоденні звіти або сповіщення при певних рівнях змін.

						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			36

3. Розширення списку валют: Варто розглянути можливість розширення переліку валют для аналізу. Це може бути корисно для користувачів, які цікавляться не лише основними валютами, а й іншими екзотичними валютами.

4. Інтеграція з іншими фінансовими ресурсами: Для більш повного аналізу фінансового стану можна розглянути можливість інтеграції програми з іншими фінансовими ресурсами, такими як біржові дані, новини про фінансові ринки тощо.

5. Покращення інтерфейсу: Додавання інтерактивної та зручної для користування графічної оболонки може зробити програму більш дружелюбною для користувачів. Це може включати інтерактивні графіки, зручні кнопки та інші елементи. 6. Інтеграція з CI/CD: Для забезпечення автоматичної та безперервної доставки можна розглянути можливість інтеграції програми з процесом Continuous Integration (CI) та Continuous Deployment (CD). Це дозволить автоматично тестувати та розгортати оновлення програми.

Загальний висновок: Результати даного дослідження та розробки підтверджують важливість автоматизованого тестування валютних курсів та використання сучасних інструментів для покращення точності та ефективності фінансового аналізу. Розроблена програма демонструє свою ефективність в реальних умовах та має потенціал для подальшого розвитку та вдосконалення.

						Аркуш
					ДТЕУ 121 023-9.MP	37
Зм.	Аркуш	№ докум	Підпис	Дата		

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

Висновки стосовно розробленої програми для автоматизованого тестування валютних курсів:

- Ефективність тестування: Програма надає ефективний та надійний засіб для автоматизованого тестування валютних курсів. Вона дозволяє швидко та точно перевіряти збіг курсів з різних джерел даних.
- Гнучкість та розширюваність: Архітектура програми дозволяє легко розширювати її функціональність. Модульна структура робить її гнучкою та легко змінюваною для введення нових валют або джерел даних.
- Надійність джерел даних: Програма успішно взаємодіє з вебсайтами та API для отримання даних. Однак важливо постійно моніторити та підтримувати ці джерела даних для надійності результатів тестування.
- Можливості розширення та покращення: Пропозиції для подальшого розвитку програми включають розширення функціональності, додавання нових джерел даних, автоматизацію визначення дати тестування, розширену звітність та використання хмарних ресурсів для підвищення ефективності.
- Аналіз та вимірювання результатів: Докладний аналіз ефективності та точності програми важливий для забезпечення її

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 023-9.МР			
Зав. каф.		Криворучко О.В.		01.11.23	Програма автоматизованого тестування вебресурсів Висновки та пропозиції	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		01.11.23		ВП	38	42
Гарант		Котенко Н.О.		01.11.23		Факультет інформаційних технологій		
Розробив		Лавріненко В.С.		01.11.23		2 курс, 2мз група		

- найвищої якості та надійності. Систематичне порівняння результатів з іншими джерелами та інструментами є ключовим.

У цілому, програма для автоматизованого тестування валютних курсів виявилася корисною та потужною для валютного аналізу та перевірки точності даних. Проте важливо продовжувати її розвиток та підтримку, а також розглядати можливості для розширення функціональності та інтеграції з іншими джерелами фінансових даних.

Пропозиції:

- Збільшення обсягу даних: Розширити обсяг даних, які програма може аналізувати та тестувати. Це дозволить користувачам отримувати більш повну та детальну інформацію про валютні курси.
- Оптимізація аналітичних інструментів: Розглянути можливість додавання аналітичних інструментів, які допоможуть користувачам робити більш глибокий та точний аналіз валютних даних.
- Мультиплатформенність: Розробити версії програми для різних платформ, таких як веб, мобільні пристрої та настільні комп'ютери, щоб забезпечити доступність для різних категорій користувачів.
- Підтримка різних мов: Додати можливість вибору мови користувачем для зручності використання програми.
- Інтерактивні графіки і графічні зображення: Розглянути можливість додавання інтерактивних графіків та графічних зображень, які допоможуть користувачам краще розуміти валютні тенденції.
- Розширені опції сповіщення: Додати розширені опції сповіщення для користувачів, що дозволять отримувати актуальну

						Аркуш
						39
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-9.МР	

інформацію про валютні зміни у вигляді повідомлень або електронних листів.

Покращення алгоритмів обробки даних: Розробити алгоритми для виявлення аномальних даних, автоматичного коригування

- дублікатів та відсутніх даних, щоб забезпечити більш точний аналіз.
- Інтерактивні навчальні матеріали: Додати навчальні матеріали та інтерактивні пояснення для користувачів, що допоможуть їм краще розуміти фінансовий аналіз та тестування.
- Підтримка різних географічних ринків: Розглянути можливість додавання підтримки для різних географічних ринків та бірж, щоб користувачі з різних країн могли використовувати програму.
- Користувацькі налаштування: Дозволити користувачам налаштовувати програму згідно зі своїми потребами, включаючи вигляд і функціональність.
- Забезпечення безпеки даних: Зробити акцент на забезпеченні конфіденційності та безпеки фінансових даних користувачів.
- Активна спільнота користувачів: Заснувати активну спільноту користувачів програми, де можна обмінюватися думками, досвідом та ідеями для подальшого розвитку.

Завдяки цим можливостям програма може стати ще більш цінним інструментом для фінансового аналізу та тестування валютних курсів, задовольняючи потреби різних користувачів та сприяючи їхньому успіху на фінансовому ринку.

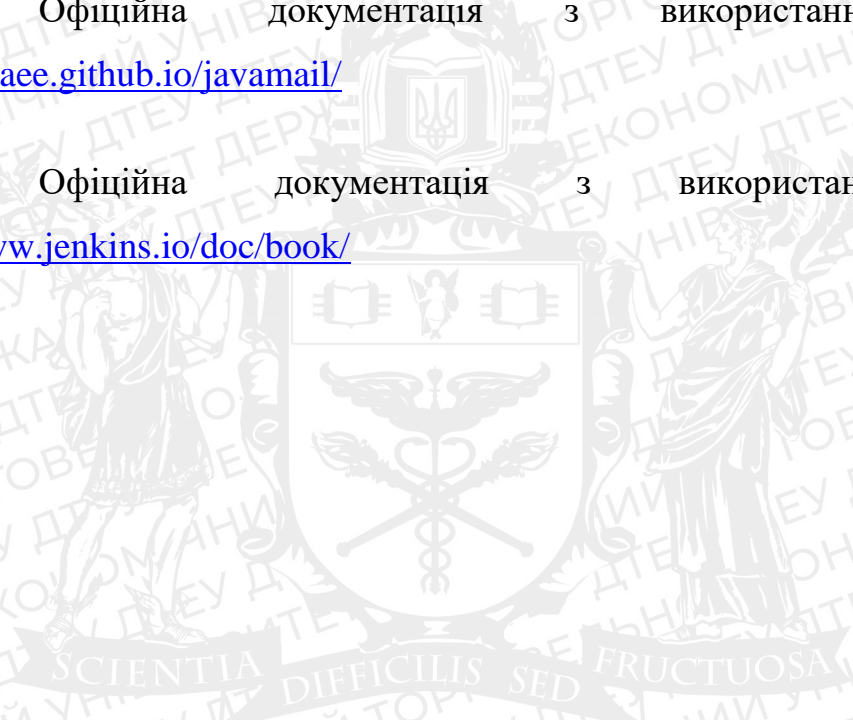
						ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			40

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Книга з автоматизованого тестування та мови програмування Test Automation using Selenium WebDriver with Java: Step by Step Guide автор Mr Navneesh Garg.
2. Книга з автоматизованого тестування Selenium Testing Tools Cookbook автор Gundecha Unmesh.
3. Книга з Rest api та Java Automating and Testing a REST API автора Mr Alan J Richardson.
4. Приклад налаштування JavaMail <https://www.baeldung.com/java-email>
5. Посібник з мови програмування Java. <https://metanit.com/java/tutorial/>
6. Курс програмування для початківців - Java 8-17
<https://www.udemy.com/course/learn-java-programming-java-beginners-course/>
7. Автоматизоване тестування з нуля, курс selenium + testing
<https://youtu.be/L2jMIJy0u90?si=WvMf4uWx7geyV2h4>
8. Курс Cucumber, Selenium & Java
<https://www.udemy.com/course/cucumber-selenium-java-develop-a-framework-in-25-hours/>
9. Офіційна документація з використання selenium
<https://www.selenium.dev/documentation/>
10. Офіційна документація з використання TestNG
<https://testng.org/doc/documentation-main.html>
11. Офіційна документація з використання RestAssured <https://rest-assured.io/>

					<i>ДТЕУ 121 02-9.МР</i>			
Зм.	Аркуш	№ докум.	Підпис	Дата				
Зав. каф.		Криворучко О.В.		01.11.23	<i>Програма автоматизованого тестування вебресурсів</i>	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		01.11.23		СВД	41	42
Гарант		Котенко Н.О.		01.11.23		<i>Факультет інформаційних технологій 2 курс, 2мз група</i>		
Розробив		Лавріненко В.С		01.11.23				
					<i>Список використаних джерел</i>			

12. Офіційна документація API для розробників Національного Банку
<https://bank.gov.ua/ua/open-data/api-dev>
13. Офіційна документація jdbc
<https://jdbc.postgresql.org/documentation>
14. Офіційна документація з використання IntelliJ IDEA
<https://www.jetbrains.com/help/idea/getting-started.html>
15. Офіційна документація з використання JavaMail
<https://javaee.github.io/javamail/>
16. Офіційна документація з використання Jenkins
<https://www.jenkins.io/doc/book/>



					ДТЕУ 121 023-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		42

ТЕХНІЧНЕ ЗАВДАННЯ

Технічне завдання на розробку програмного продукту "Система автоматизованого фінансового аналізу та тестування валютних курсів"

1. Мета проекту

1.1. Розробити додаток для аналізу та відстеження курсів валют.

2. Основні функції додатку

2.1. Отримання даних про курси валют:

Додаток повинен мати можливість отримувати дані про курси валют з відкритих джерел, таких як вебсайти організацій чи банків.

2.2. Аналіз та відображення даних:

Додаток повинен обробляти отримані дані і відображати користувачу актуальні курси валют.

2.3. Графічний інтерфейс:

Додаток повинен мати зручний графічний інтерфейс, який дозволить користувачеві легко взаємодіяти з програмою.

2.4. Відстеження змін:

Користувач повинен мати можливість відстежувати зміни курсів валют за певний період часу та отримувати сповіщення про них.

3. Технічні вимоги

3.1. Мова програмування:

Для розробки додатку рекомендується використовувати мову програмування Java.

3.2. База даних

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 02-9.МР			
Зав. каф.	Криворучко О.В.			06.09.23	Програма автоматизованого тестування вебресурсів Технічне завдання	Стадія	Аркуш	Аркушів
Керівник	Котенко Н.О.			06.09.23		ТЗ	43	42
Гарант	Котенко Н.О.			06.09.23		Факультет інформаційних технологій		
Розробив	Лавріненко В.С.			06.09.23		2 курс, 2мз група		

Для зберігання даних про курси валют можна використовувати базу даних MySQL.

3.3. Графічний інтерфейс:

Додаток повинен мати графічний інтерфейс, розроблений з використанням JavaFX або інших відповідних технологій.

3.4. Автоматизація:

Додаток повинен мати можливість автоматичного оновлення даних та відстеження їх змін.

4. План роботи

4.1. Розробка графічного інтерфейсу додатку.

4.2. Реалізація функціональності отримання даних про курси валют з відкритих джерел.

4.3. Реалізація функціональності аналізу та відображення даних.

4.4. Реалізація можливості відстеження змін курсів валют та сповіщення користувача.

4.5. Тестування та відлагодження додатку.

4.6. Документація та підготовка до випуску.

5. Завдання для виконання

5.1. Розробка графічного інтерфейсу.

5.2. Розробка функціональності отримання даних про курси валют.

5.3. Розробка функціональності аналізу та відображення даних.

5.4. Розробка функціональності відстеження змін курсів валют та сповіщення користувача.

5.5. Проведення тестування та відлагодження.

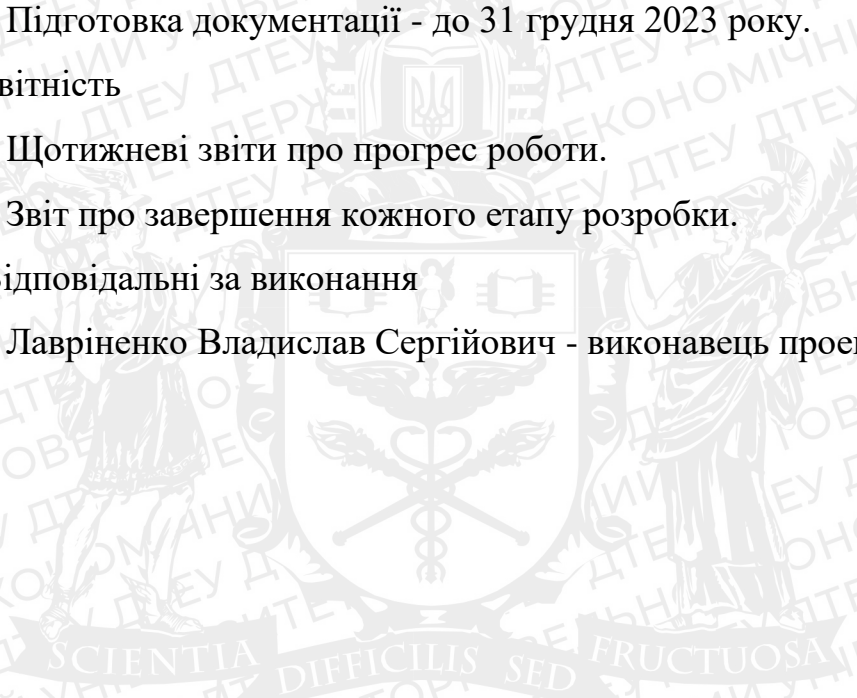
5.6. Підготовка документації.

6. Строки виконання

6.1. Розробка графічного інтерфейсу - до 15 жовтня 2023 року.

						ДТЕУ 121 023-9.МР	Аркуш
							44
Зм.	Аркуш	№ докум	Підпис	Дата			

- 6.2. Реалізація функціональності отримання даних - до 30 жовтня 2023 року.
- 6.3. Реалізація функціональності аналізу даних - до 15 листопада 2023 року.
- 6.4. Реалізація функціональності відстеження змін - до 30 листопада 2023 року.
- 6.5. Тестування та відлагодження - до 15 грудня 2023 року.
- 6.6. Підготовка документації - до 31 грудня 2023 року.
7. Звітність
- 7.1. Щотижневі звіти про прогрес роботи.
- 7.2. Звіт про завершення кожного етапу розробки.
9. Відповідальні за виконання
- 9.1. Лавріненко Владислав Сергійович - виконавець проекту.ІС



					ДТЕУ 121 02з-9.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		45

ПРОГРАМА ТА МЕТОДИКА ТЕСТУВАННЯ

Програма автоматизованого тестування розроблена для аналізу та відстеження курсів валют і має наступний функціонал:

- Отримання даних про курси валют з відкритих джерел.
- Аналіз та відображення даних для користувача.
- Відстеження змін курсів валют та сповіщення користувача.

1.2. Метою тестування є перевірка роботи програми, визначення її відповідності вимогам та виявлення можливих дефектів.

2. Огляд програми

2.1. Програма розроблена на мові програмування Java та використовує базу даних MySQL для зберігання даних про курси валют.

2.2. Програма має графічний інтерфейс, що дозволяє користувачу взаємодіяти з програмою.

3. Методика Тестування

3.1. Тестування функціональності отримання даних:

Запуск програми та спроба отримати дані про курси валют з відкритих джерел.

Перевірка, чи програма коректно обробляє та зберігає отримані дані.

3.2. Тестування функціональності аналізу та відображення даних:

Запуск програми та перевірка правильності відображення курсів валют на графічному інтерфейсі.

Перевірка відображення даних відповідно до отриманих даних з джерел

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 02з - 9.МР			
Зав. каф.		Криворучко О.В.		06.09.23	Програма автоматизованого тестування вебресурсів	Стадія	Аркуш	Аркушів
Керівник		Котенко Н.О.		06.09.23		ПМТ	46	42
Гарант		Котенко Н.О.		06.09.23	Програма та методика тестування	Факультет інформаційних технологій 2 курс, 2мз група		
Розробив		Лаврінченко В.С		06.09.23				

3.3. Тестування функціональності відстеження змін курсів валют та сповіщення користувача:

Запуск програми та встановлення відстеження змін курсу певної валюти.

Симуляція зміни курсу валюти та перевірка сповіщення користувача про цю зміну.

4. Висновки

4.1. Програма пройшла тестування та виявлені наступні результати:

Функціональність отримання даних працює коректно.

Функціональність аналізу та відображення даних також працює правильно.

Функціональність відстеження змін курсів валют та сповіщення користувача відпрацьовує належним чином.

4.2. Не було виявлено критичних дефектів або невідповідностей вимогам.

5. Пропозиції

5.1. Рекомендується провести додаткове тестування на реальних даних та в реальних умовах користування для підтвердження стабільності та надійності програми.

5.2. В разі виявлення нових функцій або змін у вимогах до програми, рекомендується проводити регресійне тестування для підтвердження її коректної роботи.

						Аркуш
					ДТЕУ 121 02з-9.МР	47
Зм.	Аркуш	№ докум	Підпис	Дата		

ДОДАТКИ

Додаток А

Посилання на роботу:

<https://github.com/VladyslavLavrinenko/SeleniumCurrencyTrackerMain>

Демонстрація коду програми:

```
// Імпорт необхідних бібліотек та класів
import com.aventstack.extentreports.reporter.ExtentSparkReporter;
import io.restassured.RestAssured;
import io.restassured.response.Response;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.Test;
import org.openqa.selenium.WebElement;
import org.testng.annotations.BeforeClass;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.concurrent.TimeUnit;
import org.testng.annotations.AfterClass;
import com.aventstack.extentreports.ExtentReports;
import com.aventstack.extentreports.ExtentTest;
import com.aventstack.extentreports.Status;

// Ініціалізація змінних та об'єктів
public class Main {
    private WebDriver driver;
    private ExtentReports extentReports;
    private ExtentTest extentTest;

    // Підготовчі дії перед тестами
    @BeforeClass
    public void setUp() {
        System.setProperty("webdriver.chrome.driver",
"C:\\Users\\Pamadmin\\Desktop\\Pop\\116\\chromedriver.exe");
        driver = new ChromeDriver();
    }
}
```

```

String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmss").format(new Date());

ExtentSparkReporter sparkReporter = new
ExtentSparkReporter("C:\\Users\\Pamadmin\\Desktop\\SparkReport" + timeStamp + ".html");
extentReports = new ExtentReports();
extentReports.attachReporter(sparkReporter);

}

@Test
public void testExchangeRate() {

String currency = "USD"; // Вкажіть код валюти наприклад: EUR CAD
String dateEx = "20230830"; // Дата курсу НБУ
String site = "https://minfin.com.ua/currency/nbu/"; // Сайт НБУ

System.setProperty("webdriver.chrome.driver", "C:\\Users\\Pamadmin\\Desktop\\Pop\\chromedriver-
win64\\chromedriver.exe");
driver.get(site);

// Отримання даних з вебсайту та API
WebElement element = driver.findElement(By.cssSelector(".sc-1x32wa2-9.fevpFL"));
String currencyText = element.getText().split("\n")[0].replace(" ", ".");
double webExchangeRate = Double.parseDouble(currencyText);
System.out.println("Курс " + currency + ": " + webExchangeRate + " з сайту " + site);

String apiUrl = "https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?valcode=" + currency +
"&date=" + dateEx + "&json";
Response response = RestAssured.get(apiUrl);
double apiExchangeRate = response.jsonPath().getDouble("[0].rate");
System.out.println("Курс " + currency + " з сайту НБУ " + apiExchangeRate);

driver.manage().timeouts().setScriptTimeout(30, TimeUnit.SECONDS);
driver.quit();

extentTest = extentReports.createTest("Test Exchange Rate");

// Порівняння курсів валют та проведення тесту
if (Math.abs(webExchangeRate - apiExchangeRate) < 0.01) {

```

```

System.out.println("Курс USD однакові.");

extentTest.log(Status.PASS, "Курс USD однакові." + " Це " + webExchangeRate + " за обрану дату " +
dateEx);

} else {

System.out.println("Курс USD не збігається");

extentTest.log(Status.FAIL, "Курс USD не збігається" + " з сайтом " + site + " курс на сайті " +
webExchangeRate + " за обрану дату " + dateEx );

}

}

// Завершальні дії після тестів

@AfterClass

public void tearDown() {

extentReports.flush();

if (driver != null) {

driver.quit();

}

}

```

Рис 1.4 Приклад коду

Використання GIT дозволить вносити оновлення, та контролювати зміни

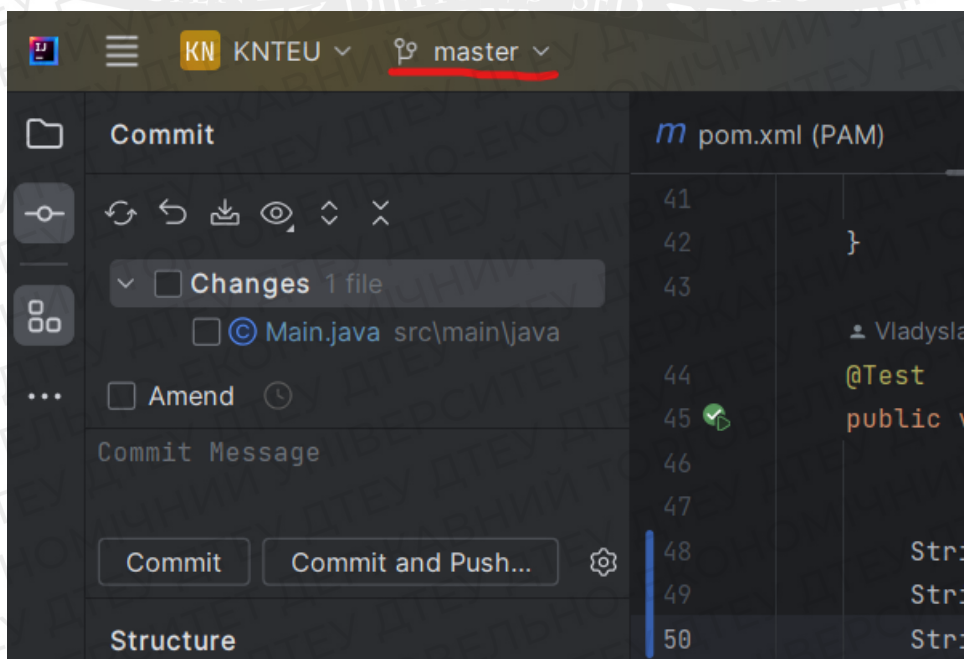


Рис 1.5 Демонстрація синхронізації з GIT

VladyslavLavrinenko / SeleniumCurrencyTrackerMain

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

SeleniumCurrencyTrackerMain Public

Pin Unwatch 1 Fork 0 Star 0

master 1 branch 0 tags

Go to file Add file Code

About

No description, website, or topics provided.

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Help people interested in this repository understand your project by adding a README.

Add a README

File	Version	Last Commit
.idea	1.0	last week
src	1.0	last week
.gitignore	1.0	last week
pom.xml	1.0	last week
testng.xml	1.0	last week

Рис 1.6 Демонстрація збереженої програми на GitHub