



Universitatea Tehnică a Moldovei

Framework de testare automată a aplicațiilor web

Student: Cazac Daniela

Conducător: lector univ. Lisnic Inga

Chișinău, 2020

ADNOTARE

Structura tezei: teza cuprinde introducere, 2 capitole, 12 subcapitole, o totalitate de 28 figuri proprii și din surse, concluzii, bibliografie cu 13 titluri, este expusă pe 45 pagini text de bază.

Cuvintele-cheie: Framework, Geb, Behavior Driven Development, Gherkin, Cucumber.

Scopul lucrării constă în alegerea unui set de instrumente, integrarea și configurarea lor într-un mediu de testare automată a aplicațiilor web.

Obiectivele generale:

- analiza și documentarea conceptului de framework;
- analiza tipologiei framework-urilor;
- analiza avantajelor utilizării unui framework în procesul de testare;
- analiza necesității utilizării unui framework de testare;
- analiza componentelor unui framework de testare;
- cercetarea etapelor alegerii strategiei de automatizare;
- cercetarea etapelor proiectării mediului de automatizare;
- descrierea soluției Geb și integrarea acestuia în mediu de testare;
- analiza, documentarea și utilizarea limbajului Gherkin;
- analiza, documentarea și utilizarea unei tehnici de testare;
- analiza, documentarea și utilizarea paternului Page Object Model;
- adăugarea scenariilor în cadrul mediului de testare;
- rularea scenariilor și analiza rezultatelor.

Metodele aplicate la elaborarea proiectului (lucrării)

Pe parcursul elaborării acestui proiect s-au utilizat și aplicat diverse metode precum:

- analiza informațiilor și surselor de informare;
- generalizarea informațiilor din sursele electronice;
- documentarea prin analiză și argumentare;
- integrarea și configurarea instrumentelor alese.

Documentarea fiind prima etapă al acestui proiect a fost prima metodă aplicată. Prin documentarea teoretică, dar și analiza surselor informaționale a fost posibilă aprofundarea în conceptul de framework, dar și în etapele proceselor decizionale de testare din cadrul unui ciclu de dezvoltare a aplicațiilor. De asemenea, documentarea instrumentelor existente și utilizate pe piața IT a avut un rol și impact pozitiv dat fiind că înțelegerea scopurilor ce le urmează aceste instrumente este mult mai clară, respectiv poate fi înțeles dacă instrumentul respectiv este ceea ce se încadrează în proiectul dat și aportul, valoarea care le aduce.

Sursele bibliografice au adus un plus valoare procesului de documentare și analiză, pe baza lor s-au aflat și identificat detalii interesante, procese și etape cheie în procesul de dezvoltare.

În urma documentării s-au ales un set de instrumente care ulterior s-au configurat și integrat corespunzător în proiect, formând și rezultând într-un mediu de testare.

Rezultatele concrete obținute

În urma integrării instrumentelor și configurării acestora s-a îndeplinit scopul care s-a urmat și anume de a avea un mediu de testare automată a site-urilor web ce are posibilitatea de a rula testele și a prezenta rezultatele.

ANNOTATION

The thesis consists of introduction, 2 chapters, 12 subchapters, a sum of 28 figures both my own and taken from sources, conclusions, bibliography with 13 titles and it's related on 45 pages.

Keywords: Framework, Geb, Behavior Driven Development, Gherkin, Cucumber.

The purpose of this thesis is to choose a set of tools, integrate and configure them as part of test automation framework for testing web applications.

General objectives:

- analysis and documentation of framework concept;
- analysis of types of frameworks;
- analysis of advantages of using a testing framework in testing process;
- analysis of the need using a testing framework;
- analysis of a framework components;
- research of stages of choosing an automation strategy;
- research of stages designing of automation framework;
- research, analysis and description of Geb solution and it's integration in a framework;
- analysis, documentation and usage of Gherkin language;
- analysis, documentation and usage Page Object Model pattern;
- add test scenarios in framework;
- run test scenarios and analysis of results.

Methods used during elaboration of thesis.

During elaboration of this project have been used various methods like:

- analysis of sources of information;
- generalization of information from various sources of information;
- integration and configuration of selected tools.

Documentation was the first method used as it's the most important one at the beginning of every project. Using this method as well as analysis of sources of information was possible to go deeply and understand better the concept of framework, stages of testing decision-making processes from a whole life-cycle process of product development. Also documentation of existing and most used tools in IT is another important point to take into consideration and had an positive impact of a better undestanting if selected tools are suitable for a specific project or application and the value they bring to project.

Sources of informations brought value of the process of documentation and analysis because based on them there was found out a lot of interesting details, processes and main stages of the development process.

Following documentation, was chosen a set of tools that were configured and integrated in a project resulting in an automation testing framework.

Achieved results

Following integration and documentation of choosen tools has been achieved the scope of this thesis, namely to bring up an automation testing framework for testing web applications and which has possibility to run test scenarios.

Cuprins

INTRODUCERE	9
1 ABORDAREA TEORETICĂ ȘI ANALIZA CONCEPTULUI DE FRAMEWORK	10
1.1 Beneficiile unui framework de testare automatizată	10
1.2 Tipuri de framework-uri de testare automatizată	11
1.3 Componentele unui mediu de testare automatizată	18
1.4 Framework-uri existente	19
1.5 Etapele alegerii strategiei de automatizare	19
1.6 Etapele proiectării mediului de automatizare	21
2 TEHNOLOGII ȘI INSTRUMENTE UTILIZATE ÎN CREAREA MEDIULUI DE TESTARE	24
2.1 Geb	25
2.2 Configurarea și integrarea soluției Geb în cadrul proiectului:	27
2.3 Behavior-Driven Development	31
2.3 Limbajul Gherkin	32
2.4 Page Object Model (POM)	43
2.5 Integrarea paternului POM în cadrul proiectului	45
2.6 Rularea testelor	47
Concluzii	51
Bibliografie	52

INTRODUCERE

Importanța și actualitatea temei cercetate

În procesul de dezvoltare a unei aplicații un rol important îi revine activității de testare și verificare a produsului ca acesta să corespundă cerințelor și specificațiilor clientului. Un plus valoare acestui proces îl aduce testarea automatizată care are scopul de a facilita procesul de testare manuală. În acest context este nevoie de analizat specificul aplicației, ce tehnici și instrumente de testare ar fi binevenite și potrivite pentru aplicație dată. În urma acestei analize se obține o imagine mai clară despre viitorul mediu de testare. Un mediu de testare automată trebuie setat corespunzător cerințelor și tipului de testare. Mediul de testare automată are rolul și urmează scopul de a facilita procesul de testare manual, de asemenea de a economisi timp și efort. Astfel prin rularea testelor automatizate se urmărește facilitarea procesului de testare, de a se obține în scurt timp rapoartele cu rezultatele rulării.

Obiectivele generale:

- analiza și documentarea conceptului de framework;
- analiza tipologiei framework-urilor;
- analiza avantajelor utilizării unui framework în procesul de testare;
- analiza necesității utilizării unui framework de testare;
- analiza componentelor unui framework de testare;
- cercetarea etapelor alegerii strategiei de automatizare;
- cercetarea etapelor proiectării mediului de automatizare;
- descrierea soluției Geb și integrarea acestuia în mediu de testare;
- analiza, documentarea și utilizarea limbajului Gherkin;
- analiza, documentarea și utilizarea unei tehnici de testare;
- analiza, documentarea și utilizarea paternului Page Object Model;
- adăugarea scenariilor în cadrul mediului de testare;
- rularea scenariilor și analiza rezultatelor.

Metodele aplicate la elaborarea proiectului (lucrării)

Pe parcursul elaborării acestui proiect s-au utilizat și aplicat diverse metode precum:

- analiza informațiilor și surselor de informare;
- generalizarea informațiilor din sursele electronice;
- documentarea prin analiză și argumentare;
- integrarea și configurarea instrumentelor alese.

Bibliografie

1. Beneficiile unui framework de automatizare [citat 12.09.2020]. Disponibil: <https://smartbear.com/learn/automated-testing/test-automation-frameworks/>
2. Tipuri de framework-uri [citat 12.09.2020]. Disponibil: <https://www.softwaretestinghelp.com/test-automation-frameworks-selenium-tutorial-20/>
3. Cele mai utilizate instrumente de automatizare [citat 15.09.2020]. Disponibil: <https://dzone.com/articles/top-10-test-automation-frameworks-in-2020>
4. Etapele alegerii strategiei de automatizare [citat 16.09.2020]. Disponibil: <https://blog.gurock.com/set-up-automation-strategy/>
5. Etapele proiectării mediului de automatizare [citat 20.09.2020]. Disponibil: <https://smartbear.com/resources/ebooks/steps-for-building-a-successful-ui-automated-test/>
6. Browser automation testing folosind instrumentul Geb [citat 10.10.2020]. Disponibil: <https://www.softwaretestinghelp.com/geb-tutorial-browser-automation-testing-using-geb-tool/>
7. Utilizarea plugin-urilor [citat 11.10.2020]. Disponibil: https://docs.gradle.org/current/userguide/application_plugin.html
8. Folosirea dependențelor Geb în Gradle [citat 10.10.2020]. Disponibil: <https://gebish.org/manual/current/#gradle>
9. Proces de dezvoltare BDD [citat 12.10.2020]. Disponibil: <https://automationpanda.com/2017/01/25/bdd-101-introducing-bdd/>
10. Limbajul Gherkin [citat 13.10.2020]. Disponibil: <https://automationpanda.com/2017/01/26/bdd-101-the-gherkin-language>
11. Exemple Gherkin [citat 15.10.2020]. Disponibil: <https://automationpanda.com/2017/01/27/bdd-101-gherkin-by-example/>
12. Paternul Page Object Model [citat 20.10.2020]. Disponibil: <https://medium.com/tech-tajawal/page-object-model-pom-design-pattern-f9588630800b>
13. Paternul POM în Selenium [citat 22.10.2020]. Disponibil: <https://www.edureka.co/blog/page-object-model-in-selenium/>