Tehnologii QA

PANAINTE ALEXANDRU

**Nume tehnologie: JMeter**

Descriere: Apache JMeter este o aplicatie open source destinata sa masoare performanta si comportamentul aplicatiilor web.

Avantaje:

-Aplicatia este gratis

-Pe pagina acesteia sunt prezente tutoriale pentru a o utiliza

-Este usor de instalat

-Sunt multiple plugin-uri disponibile si pot fi instalate pentru diferite necesitati

Dezavantaje:

-Aplicatia are un learning curve ridicat

-Consumul de memorie este mare in modul GUI, acest lucru rezultand in erori pentru un numar mare de utilizatori

Raport: Apache JMeter este un instrument de testare a încărcării la nivel de protocol. Poate fi utilizat pentru testarea timpilor de încărcare pentru elemente statice și dinamice într-o aplicație web. Un tester poate simula o sarcină grea pe un server, grup de servere sau o retea pentru a le testa caracteristicele.

Link tutoriale: <https://jmeter.apache.org/>

**Nume tehnologie: Mocha**

Descriere: Mocha este un cadru de testare JavaScript pentru programele Node.js, care oferă suport pentru browser, testare asincronă, rapoarte de acoperire a testelor și utilizarea oricărei biblioteci de afirmații.

Avantaje:

-Testarile pe NodeJS sunt extrem de simple deoarece si Mocha ruleaza pe NodeJS

-Se poate alege dintre diferite formate de testing

-Poate rula si in browser

Dezavantaje:

-Testele nu pot fi rulate in ordine random

-Pentru cineva care abia incepe sa invete cum sa faca teste poate fi putin greu deoarece trebuie alese si invatate si librariile care sunt necesare

Raport: Mocha este o aplicatie de testing gratis si relativ usor de invtat. Deoarece aplicatia in sine ruleaza pe NodeJS va face testingul mult mai facil. Sunt prezente multiple tutoriale ce vor facilita invatarea si utilizarea aplicatiei.

Link tutoriale: <https://www.youtube.com/watch?v=MLTRHc5dk6s>

Codrin Donciu si Sabina Petrisor

**Nume tehnologie: Jasmine**

Descriere: Jasmine este varianta upgraded de la JsUnit, care era framework-ul de testare pentru JavaScript

la începutul anilor 2000. Practic, e un BDD (Behavior Driven Development) testing framework pentru JS și e foarte

influențat de alte framework-uri de testare ca ScrewUnit, JSSpec, Jspec și RSpec.

Avantaje :

-este o tehnologie independentă de browser, DOM (Document Object Model) sau orice alt framework JavaScript

-este open-source

-rulează pe orice platformă JS, și nu depinde nici de aplicație sau IDE fiindcă are inbuilt test runner

-are o sintaxă „easy-to-read” după cum se laudă dezvoltatorii de la Pivotal Labs

-suportă front-end testing prin extensia Jasmine-jQuery

Dezavantaje:

-e mai mult axat pe Behavior Driven Development, deși e complet capabil de test driven development

Raport: Pare foarte facil de utilizat, mai ales că sintaxa e de așa natură încât poate fi înțeleasă de oricine,

e foarte high-level și orice script apare ca o „poveste”.

Codrin Donciu

**Nume tehnologie: Karma**

Descriere: Test runner pentru JS care rulează pe Node.js.

Construit pentru testarea AngularJS sau orice alt proiect JavaScript.

Avantaje:

-compatibil cu Jasmine, Mocha, QUnit etc;

-poate rula teste nu doar în orice browser, dar și pe orice dispozitiv: desktop, smartphone, tabletă etc.

-se integrează și cu alte pachete;

-are suport excelent pentru plugin-uri;

Dezavantaje:

-desi tot ce este necesar pentru a rula Karma se află pe site-ul lor oficial, nu au foarte multă documentație și chiar dezvoltatorii

te trimit către Google sau Stack Overflow;

-dacă nu folosești AngularJS, nu e chiar cea mai de preferat tehnlogie;

Raport: E posibil să putem face uz de această tehnologie, având în vedere că este

foarte recomandată în cadrul proiectelor AGILE sau pentru alte medii de integrare continuă, cum este și proiectul de față.

**Nume tehnologie: Selenium**

Descriere: Este un framework portabil pentru testarea de web applications,

e scris in Java și vine din partea celor de la Apache.

Avantaje:

-open source;

-scrierea de teste funcționale se poate face fără a învăța limbaj specific de testing, și se face în

propriul mediu de dezvoltare, numit Selenium IDE;

-propune Selenese, un domain-specific language în care se pot scrie teste în orice limbaj popular

de programare: C#, Java, Groovy, PHP, Scala, Python...

-Selenium IDE vine ca extensie pentru Google Chrome și Firefox

Dezavantaje:

-pentru a folosi aceasta tehnologie de testare în limbajul preferat, tester-ul trebuie totuși să fie

pregătit adecvat în acel limbaj, deci nu e destul de beginner friendly

-nu există suport dedicat din partea dezvoltatorilor, deci tot comunitatea va oferi cel mai mult

ajutor cu întrebări sau dificulăți în folosire;

-nu are unelte inbuilt de a genera rapoarte, lucru care ar trebui să fie făcut manual; bănuiesc că în cadrul

proiectului, rapoartele sunt necesare pentru a putea comunica eficient între departamente și între membrii QA;

dacă alegem această tehnologie, putem totuși să o folosim combinând funcționalități din plug-in-urile Junit și TestNG

-nu e foarte user-friendly („the testers must use Selenium properly to overcome some of its prominent limitations”)

**Tehnologii selectate pentru utilizare:**

In final in urma sedintei de pe 29-03-2020 am decis sa ne axam pe Jasmine pentru testing-in cadrul aplicatiei.

<https://www.tutorialspoint.com/jasminejs/index.htm>

<https://www.youtube.com/watch?v=h2eWfvcAOTI>

Motivatie: Utilizare simpla, compatibilitate mare.

Powershell/cmd/bash pentru scripting si mici automatizari si testare manuala (ex: ping).

Firefox/Chrome/Opera tetare manuala look-ul aplicatiei, stabilitate, user experience.