**Numele şi prenumele absolventului:**

**Radu Ionel Bogdan**

**Tema lucrării:**

**Automated Acceptance Testing Framework utilizând Serenity BDD**

Descrierea lucrării :

FastTrackIT este o aplicaţie web utilizată pentru vânzarea de produse. Am construit o arhitectura de testare automată in Java, care ruleaza scripturi automate pentru a rula testele de regresie.

Tehnologii folosite :

Java, Selenium Webdriver, JUnit, Serenity

**Testarea automată** reprezintă o testare dinamică și analitică a unui produs [software](https://ro.wikipedia.org/wiki/Software), care presupune utilizarea unui program pentru executarea procedurilor (*test case*) sau a întregilor scenarii de testare.

Pentru a livra o soluţie software, de calitate, testarea şi automatizarea sunt extrem de importante. Există o mulțime de motive pentru care automatizarea testelor este benefică și, respectând cele mai bune practici de testare automată, te poţi asigura că nivelul de testare oferă rentabilitatea maximă a investiției. Testarea automată îţi va scurta ciclurile de dezvoltare, va evita sarcinile repetitive greoaie și va ajuta la îmbunătățirea calității software-ului.

Testarea automată crește profunzimea și diversitatea testelor si  contribuie la îmbunătățirea calității software-ului. Testele lungi, care sunt adesea evitate în timpul testării manuale, pot fi efectuate nesupravegheat in modul automat. Pot fi rulate chiar și pe mai multe tipuri de computer, cu diferite configurații. Testarea automată a software-ului poate “privi”  în interiorul unei aplicații și poate vedea conținutul memoriei, tabelele de date, conținutul fișierului și stările interne ale programului pentru a determina dacă produsul se comportă așa cum era de așteptat. Cu automatizarea testelor, se pot executa cu ușurință mii de cazuri de testare complexe si diferite, în timpul fiecărei testări, ceea ce ofera o acoperire imposibil de realizat, cu testele manuale.

**Java** este un limbaj de programare de nivel înalt, dezvoltat de JavaSoft, companie în cadrul firmei Sun Microsystems. Dintre caracteristicile principale ale limbajului amintim:

* simplitate, elimina supraîncarcarea operatorilor, mostenirea multipla si toate "facilitatile" ce pot provoca scrierea unui cod confuz.
* robustete, elimina sursele frecvente de erori ce apar in programare prin eliminarea pointerilor, administrarea automata a memoriei si eliminarea fisurilor de memorie printr-o procedura de colectare a 'gunoiului' care ruleaza în fundal. Un program Java care a trecut de compilare are proprietatea ca la executia sa nu "crapa sistemul".
* complet orientat pe obiecte - elimina complet stilul de programare procedural
* usurinta in ceea ce priveste programarea in retea
* securitate, este cel mai sigur limbaj de programare disponibil în acest moment, asigurând mecanisme stricte de securitate a programelor concretizate prin: verificarea dinamica a codului pentru detectarea secventelor periculoase, impunerea unor reguli stricte pentru rularea programelor lansate pe calculatoare aflate la distanta, etc
* este neutru din punct de vedere arhitectural
* portabililtate, cu alte cuvinte Java este un limbaj independent de platforma de lucru, aceeasi aplicatie ruland, fara nici o modificare, pe sisteme diferite cum ar fi Windows, UNIX sau Macintosh, lucru care aduce economii substantiale firmelor care dezvolta aplicatii pentru Internet.
* compilat si interpretat
* asigura o performanta ridicata a codului de octeti
* permite programarea cu fire de executie (multitheaded)
* dinamicitate
* este modelat dupa C si C++, trecerea de la C, C++ la Java facându-se foarte usor.
* permite creearea unor documente Web îmbunatatite cu animatie si multimedia.

[**Selenium** este o tehnologie sau un cadru open source](https://ro.lambdageeks.com/selenium-design-patterns-and-best-practices/) prin care putem automatiza aplicații bazate pe web. Selenium acceptă testarea cu mai multe browsere, mai multe limbi precum Java, Ruby, Perl, Python sunt de asemenea acceptate. Acceptă diferite platforme multiple, cum ar fi Windows, Linux, macOS etc., ca parte a platformei.

**JUnit** este un [cadru de](https://koaha.org/wiki/Framework) [testare unitară](https://koaha.org/wiki/Unit_testing) pentru [limbajul de](https://koaha.org/wiki/Java_(linguaggio_di_programmazione)) [programare](https://koaha.org/wiki/Linguaggio_di_programmazione) [Java](https://koaha.org/wiki/Java_(linguaggio_di_programmazione)) . Experiența cu JUnit a fost importantă în creșterea ideii de [Test Driven Development](https://koaha.org/wiki/Test_Driven_Development) și face parte dintr-o familie de cadre de testare unitară cunoscute colectiv sub numele de [xUnit](https://koaha.org/wiki/XUnit) .

**Serenity BDD** (Behavior Driven Development) este un “framework” si “open source library” pentru crearea de testare automata a software-ului. Ca si alte instrumente automate de testare, Serenity BDD este conceput pentru a usura dezvoltarea prin gasirea erorilor mai devreme si ajuta la asigurarea lansarii unui produs software final stabil si complet functional.

**Testarea prin regresie** este un tip de testare care se face pentru a verifica dacă o modificare a codului din software nu are impact asupra funcționalității existente a produsului. Aceasta este pentru a vă asigura că produsul funcționează bine cu funcționalități noi, remedieri de erori sau orice modificare a funcției existente. Cazurile de testare executate anterior sunt reexecutate pentru a verifica impactul schimbării. Testarea de regresie este un tip de testare software în care cazurile de testare sunt reexecutate pentru a verifica dacă funcționalitatea anterioară a aplicației funcționează bine și noile modificări nu au introdus erori noi. Acest test poate fi efectuat pe o nouă versiune atunci când există o schimbare semnificativă în funcționalitatea originală, chiar și într-o singură soluție de eroare. Regresia înseamnă retestarea părților neschimbate ale aplicației.

Arhitectura de testare automata creata verifica diversi pasi din aplicatie pentru a vedea daca au erori.