Titlul lucrării

PROIECT DE DIPLOMĂ

Autor: **Ruxandra Mara IERIMA**

Conducător științific: **Prof.dr.ing. Honoriu VĂLEAN**

|  |  |
| --- | --- |
| DECAN  **Prof.dr.ing. Liviu MICLEA** | Vizat,  DIRECTOR DEPARTAMENT AUTOMATICĂ  **Prof.dr.ing. Honoriu VĂLEAN** |

Autor: **Prenume NUME**

Cuprins

[1. INTRODUCERE 2](#_Toc10148291)

[1.1 Context general 2](#_Toc10148292)

[1.2 Obiective 2](#_Toc10148293)

[1.3 Specificații 2](#_Toc10148294)

[2. STADIUL ACTUAL 3](#_Toc10148295)

[3. FUNDAMENTARE TEORETICĂ 5](#_Toc10148296)

[4.1 Pricipiile testării 5](#_Toc10148297)

[3.11 Ce este testarea 5](#_Toc10148298)

[3.12 Tipuri de testare 5](#_Toc10148299)

[3.13 Testarea automată (tipuri de teste automate exemple) 5](#_Toc10148300)

[3.14 Certificări( ISTQB) 5](#_Toc10148301)

[4.2 Selenium WebDriver 5](#_Toc10148302)

[3.21 Introducere (cand a aparut, ce face, definitii) 5](#_Toc10148303)

[3.22 Comenzi (descriere operatii si comenzi, implementarea pe un proiect-page object) 5](#_Toc10148304)

[4.3 Gherkin 5](#_Toc10148305)

[3.31 Introducere (benefici utilizare, fondare etc) 5](#_Toc10148306)

[3.32 Limbajul Gherkin (descriere cuvinte cheie, sintaxa, exemple) 5](#_Toc10148307)

[4. IMPLEMENTAREA SOLUȚIEI 5](#_Toc10148308)

[4.1 Descrierea aplicației (framework-ului) 5](#_Toc10148309)

[4.2 Configurarea programelor (setup imagini tooluri folosite) 5](#_Toc10148310)

[4.3 Implementarea soluției (cod) 5](#_Toc10148311)

[5. REZULTATE EXPERIMENTALE 5](#_Toc10148312)

[5.1 Rularea testelor (se ruleaza toate testele deodata) 5](#_Toc10148313)

[5.2 Generarea raportului de validare 5](#_Toc10148314)

[6. CONCLUZII 5](#_Toc10148315)

[7. BIBLIOGRAFIE 5](#_Toc10148316)

[8. ANEXE 5](#_Toc10148317)

# INTRODUCERE

## Context general

In introducere familiarizați cititorul cu motivația lucrării, plasați lucrarea într-un context care să permită cititorului să înțeleagă obiectivele.

Descrieți importanța lucrării, de ce merita să o faceți, plasati ideile într-un context larg.

Sustineți studiul: de ce exact aceasta aplicație/implementare. Comentați asupra aspectelor teoretice sau practice care v-au facut să o alegeți.

In funcție de natura lucrării, ar putea fi necesar să prezentați informații de fond asupra domeniului în care se încadrează aplicația, mai ales dacă aveti o lucrare într-un domeniu multidisciplinar. In acest caz, puteți introduce terminologia pe care o utilizați în continuare.

Descrieti pe scurt lucrarea: ce conține fiecare capitol.

## Obiective

Enumerati și explicati obiectivele lucrării: ce v-ați propus să realizați în contextul prezentat anterior.

Obiectivele pot fi prezentate sub formă de listă care să evidențieze precis orientarea lucrării, să identifice conceptele fundamentale pe care le studiati, să stabilească scopul aplicației pe care o realizați, sau enuntați întrebarile la care intentionați să răspundeți în lucrare.

## Specificații

In specificațiile lucrării detaliați cerințele. Descrieți ce intentionați să obtineți. Vă puteți referi la funcțiile aplicației, interfață, nivele de performanță, structuri de date, elemente, securitate, fiabilitate, calitate, limitări, etc.

# STADIUL ACTUAL

Testarea software are ca scop oferirea de informații cu privire la calitatea unui produs sau a unui serviciu. În investigarea produsului supus testării se va lua în considerare contextul în care acesta va fi utilizat. Pentru a putea înțelege și estima riscurile pe care le reprezintă implementarea produsului în dezvoltare, testarea software oferă o perspectivă obiectivă.

Această testare reprezintă procesul de identificare a defectelor și a erorilor unui produs prin observarea diferențelor dintre un comportament dorit și un comportament observat cu ajutorul diferitelor tehnici de testare. Un alt aspect al testării software reprezintă validarea produsului finit prin verificarea faptului că acesta corespunde așteptărilor.

Testarea este foarte importantă deoarece chiar și unele erori minore dintr-o aplicație pot avea efecte negative precum costuri suplimentare sau desconsiderarea calității acelui produs de către client.

Acest concept nu este unul nou apărut însă dezvoltarea și identificarea de noi tehnologii este in continuă creștere datorită standardelor și așteptările ridicate ale clienților.

Clasificarea fazelor și obiectivelor testării realizată de către Gelperin și W.C. Hetzel[1-WIKI]:

**Orientarea spre depanare (până în 1956)**

Această perioadă a fost orientată mai mult spre partea hardware, unde testarea a fost adesea asociată cu depanarea: neexistând o diferență clară între teste și depanare. Pe partea de software, defectele nu erau considerate atât de importante, în general persoanele care se ocupau cu implementarea codului făceau si partea de testare.

Prima referire la termenul de “bug” în domeniul informatic când pe un circuit electronic defectuos s-a descoperit o molie.

**Orientarea spre demonstrație (1957-1918)**

Diferența dintre depanare și testare începea să se distingă în această perioadă. Accentul pe testare și pe remedierea defectelor a crescut, persoanele implicate in dezvoltarea de programe informatice fiind mult mai conștiente de această problemă. Scopul sistemelor software era acela de a îndeplini cerințele.

**Orientarea spre defectare (1979-1982)**

În această perioadă se propune, pe lângă procesul de testare, o serie de activități de analiză și control cu scopul creșterii calității produsului. Această abordare vine ca o rezolvare a testării bazate pe demonstrație unde setul de date introdus pentru validarea procesului putea fi ales involuntar in favoarea produsului.

**Orientarea spre evaluare (1983-1987)**

Principalul scop a fost evaluarea produsului în timpul ciclului de viață al acestuia, cât și alegerea diferitelor metode de testare și validare în funcție de caracteristicele sistemului in ideea creșterii calității. În această perioadă au apărut și primele locuri de muncă dedicate de tester.

**Orientarea spre prevenire (1988-prezent)**

Pentru a reduce costurile de dezvoltare și de întreținere a unui sistem software, testarea a fost introdusă încă din primele faze, în paralel cu implementarea produsului. Încep să fie definite primele activități principale ale testării precum: planificarea, analiza, proiectarea, implementarea, execuția și întreținerea. Prin respectarea acestor metode, defectele apărute în urma implementării aplicației scad considerabil.

**Metodologii utilizate in procesul de testare:**

* **Testarea Agile -** este un proces de testare care urmează principiile Agile ale dezvoltării software, unde cerințele sunt specificate treptat de către client. Procesul este unu continuu în care exista o integrare intre implementare și testare, scopul comun fiind obținerea unei calități ridicate.
* **Metodologia Scrum –** activitățile de testare din cadrul acestui proces implică estimarea efortului de către tester cu privire la timpul necesar pe care îl dedica pentru testarea unei anumite secțiuni din aplicație. O altă activitate importantă din acest proces este aceea de a prioritiza defectele după gradul de severitate pe care acestea le prezintă.

# FUNDAMENTARE TEORETICĂ

## Pricipiile testării

### Ce este testarea

### Tipuri de testare

### Testarea automată (tipuri de teste automate exemple)

### Certificări( ISTQB)

## Selenium WebDriver

### Introducere (cand a aparut, ce face, definitii)

### Comenzi (descriere operatii si comenzi, implementarea pe un proiect-page object)

## Gherkin

### Introducere (benefici utilizare, fondare etc)

### Limbajul Gherkin (descriere cuvinte cheie, sintaxa, exemple)

# IMPLEMENTAREA SOLUȚIEI

## Descrierea aplicației (framework-ului)

## Configurarea programelor (setup imagini tooluri folosite)

## Implementarea soluției (cod)

# REZULTATE EXPERIMENTALE

## Rularea testelor (se ruleaza toate testele deodata)

## Generarea raportului de validare

# CONCLUZII

# BIBLIOGRAFIE

# ANEXE