

# PROIECT FINAL

PREDA OANA  
MARIA

20.11.2024



# Cuprins

## Partea 1 – Noțiuni teoretice

1. Ce sunt cerințele de business, la ce ne folosesc și cine le creează.
2. Care este diferența între un test condition și test case?
3. Care sunt etapele procesului de testare?
4. Care este diferența între retesting și regression testing?
5. Care este diferența între functional testing și non-functional testing?
6. Care este diferența între blackbox testing și whitebox testing?
7. Care sunt tehnicile de testare?
8. Care este diferența între verification și validation?
9. Care este diferența între positive testing și negative testing?
10. Care sunt nivelurile de testare?

## Partea II – Partea practică

1. Cerințele pentru website-ul ales pentru testare.
2. Condițiile de testare și cazurile de testare.
3. Matricea trasabilității.
4. Raportul generat din aplicația Jira.
5. Analiza de risc.
6. Raportul de defect.
7. Concluzii generale în urma testării.



# Partea I – Noțiuni teoretice



## 1. Q : Ce sunt cerințele de business, la ce ne folosesc și cine le creează.

A : Documente care conțin felul în care produsul ar trebui să funcționeze conform dorințelor clientului.

Astfel de documente fie sunt primite de la client, fie există un membru al echipei care adună informațiile și creează documentele.

Pe baza acestor documente, programatorii vor crea software-ul, iar testerii vor crea teste.

## 2. Q : Care este diferența între un test condition și test case.

A : Test case : Un set de condiții și pași specifici, inclusiv input data și expected results, folosite pentru a verifica dacă o caracteristică sau o funcționalitate a software-ului funcționează conform specificațiilor.

Test condition : O situație sau un set de criterii pe care un tester dorește să le verifice sau să le valideze.

Relația dintre test case și test condition : Un test case este specific și detaliat și este creat pentru a verifica un singur test condition.

## 3. Q : Etapele procesului de testare :

**Test Planning** – Definirea strategiei de testare, a resurselor și a calendarului

Scop : Stabilește scopul și obiectivele testării

Strategie : Definieste abordarea testării (manuală, automată, mixtă)

Resurse : Identifică resursele umane și tehnice necesare.

Calendar : Planifică activitățile de testare din timp.

**Test Analysis** – Înțelegerea detaliată a funcționalităților dorite.

Scop : Înțelegerea în detaliu a cerințelor de business și tehnice.

Activități : Analizarea documentației de cerințe.

Rezultate : Crearea unei liste de cerințe testabile și identificarea condițiilor de testare.

**Test Design** – Crearea de test cases

Scop : Elaborarea detaliată a test case-urilor

Activități : Crearea de test case-uri pe baza cerințelor analizate, identificarea datelor de test necesare, crearea scenariilor de test.

Rezultate : Test case-uri detaliate și pregătite pentru execuție.

### **3. Q : Etapele procesului de testare :**

#### **Test Implementation – Pregatirea mediului de testare si configurarea testelor**

Scop : Configurarea mediului de testare si pregatirea testelor pentru executie.

Activitati : Instalarea si configurarea instrumentelor de testare.

Rezultate : Mediu de testare functional si teste pregatite pentru executie.

#### **Test execution – Executarea manuala si/sau automata a testelor**

Scop : Executia testelor conform planificarii.

Activitati : Executarea test case-urilor manualee si automate. Compararea rezultatelor.

Rezultate : Raport de executie a testelor cu rezultate si observatii.

#### **Test monitoring and control – Supravegherea progresului testarii si ajustarea planurilor.**

Scop : Monitorizarea continua a progresului testarii si ajustarea planurilor in functie de necesitati.

Activitati : Monitorizarea indicatorilor de performanta.

Ajustarea planului de testare bazat pe rezultate curente.

Comunicarea progresului si a problemelor identificate cu echipa de proiect.

Rezultate : Status actualizat al testarii si ajustari facute la nevoie.

#### **Test completion – Concluzionarea activitatilor de testare, si incheierea procesului.**

Scop : Incheierea formala a activitatilor de testare.

Activitati : Evaluarea rezultatelor testarii in comparatie cu obiectivele initiale.

Documentarea lectiilor invatate si a recomandarilor pentru proiectele viitoare.

Rezultate : Raport final de testare si inchiderea procesului de testare.

### **4. Q : Care este diferența între retesting și regression testing.**

A : Retestarea este un proces prin care se verifică dacă defectele marcate ca și remediate au fost într-adevăr remediate.

Testarea de regresie e un process prin care se verifică programul sau o parte din program pentru a ne asigura că schimbările aduse asupra lui nu au cauzat/descoperit alte defecte.

### **5. Q : Care este diferența între functional testing și non-functional testing.**

A : Testare funcțională – Verifică dacă produsul își îndeplinește funcțiile, iar  
Testarea non-funcțională verifică attribute care descriu cât de bine își îndeplinește sistemul funcțiile.



**6. Q : Care este diferenta intre Blackbox testing si Whitebox testing.**

A : **Blackbox testing** : Testarea produsului ca o "cutie neagra", fara a cunoaste codul sursa.

**Whitebox testing** : Testarea produsului cu acces la codul sursa, permitand testarea detaliata a logicii interne.

**7. Tehnicile de testare se impart in mai multe categorii :**

**Black box**

X Partitionarea echivalenta x Analiza valorilor limita x Testarea tranzitiilor de stare x Tabel decisional x Use case

**White box**

x Statement coverage x Decision coverage

**Experience based :**

X Ad-hoc testing x Error guessing x Testarea exploratorie

**8. Q : Care este diferența între verification și validation?**

**Verificare:** Asigurarea că produsul este construit conform specificațiilor.

**Validare:** Asigurarea că produsul satisface nevoile utilizatorilor finali.

**9. Q : Care este diferența între positive testing și negative testing?**

**Testare pozitivă** - testarea sistemului cu valori pe care ar trebui să le poată procesa.

**Testare Negativă** - testare cu valori pe care sistemul nu ar trebui să le poată procesa în mod normal pentru a ne asigura ca aceste valori sunt într-adevăr respinse și că nu cauzează un crash al sistemului.



## 10. Q : Care sunt nivelurile de testare?

### **Unit testing** – Testarea individuala a componentelor software.

Caracteristici : Testeaza unitati individuale de cod, cum ar fi functii sau metode.

Este realizata, de obicei, de catre dezvoltatori.

Exemplu : Verificarea daca o functie de calculare a sumei returneaza rezultatul corect pentru doua numere date.

### **Integration testing** – Testarea interactiunii dintre componentele software.

Caracteristici : Se concentreaza pe modul in care modulele software interactioneaza intre ele. Poate testa combinatii de doua sau mai multe unitati. Identifica probleme care apar atunci cand componentele sunt combinate.

Exemplu : Verificarea daca un modul de login functioneaza corect impreuna cu un modul de validare a utilizatorilor.

### **System testing** – Testarea completa a sistemului ca un intreg.

Testeaza intregul sistem software pentru a se asigura ca toate componentele functioneaza impreuna.

Realizata intr-un mediu care imita conditiile de productie. Acopera scenarii complete de utilizare.

Exemplu : Verificarea intregului proces de cumparare pe un site de comert electronic, de la selectarea produsului pana la finalizarea platii.

### **Acceptance testing:** Testarea finala efectuata de catre utilizatori finali.

Caracteristici : Realizat de catre utilizatori sau echipe de QA pentru a valida cerintele de afaceri;

Verifica daca sistemul indeplineste cerintele si asteptarile utilizatorilor. Ultimul pas inainte de implementarea in productie.

Exemplu : Un client verifica daca un sistem de gestionare a inventarului indeplineste toate cerintele specificate in contractul de dezvoltare.

# Partea II – Partea practica



## CERINTELE PAGINII WEB ALEASA PENTRU TESTARE



Aplicatia aleasa pentru testare este <https://automationexercise.com/>, modulele User Registration si Cart.

Proiectul a fost realizat cu ajutorul aplicatiei Jira si al pluginului Zephyr Squad, unde am creat un epic si 2 story-uri, 3 task-uri, care la randul lor contin fiecare cate 2 subtask-uri.

Acestea se regasesc in imaginea de mai jos.

Nr.Crt	Title	Issue Type
1	As a user of the "Automation exercise" e-commerce website, i want to have a smooth registration and shopping cart experience	Epic
2	As a new user, i want that the system allows me to create and manage my accounts, including registration, login, and password recovery.	Story
3	Manage Shopping Cart Items	Story
4	Verify Shopping Cart Item Management Functionality	Task
5	Develop Quantity Adjustment Controls	Subtask
6	Verify Item Removal Functionality	Subtask
7	Enhance Shopping Cart User Interface and Experience	Task
8	Design and Implement Responsive Cart Layout	Subtask
9	Integrate Real-Time Updates and Visual Feedback	Subtask
10	Verify User Registration and Account Management	Task
11	Create Registration Form and Backend Integration	Subtask
12	Verify Email Verification and Account Activation	Subtask

## Detalierea cerintelor identificate pentru website-ul ales

TMTA20PO-13 / TMTA20PO-12

As a new user, i want that the system allows me to create and manage my accounts, including registration, login, and password recovery.

Attach

Create subtask

Link issue



### Description

As a new user, I want the system to provide a seamless and secure way for me to create and manage my account. This includes the ability to register a new account by providing necessary information like email, username, and password. Once registered, I want to be able to log in using my credentials and access my account. Additionally, if I forget my password, I need a simple process to recover or reset it, ensuring I can always regain access to my account while keeping my data secure.

TMTA20PO-13 / TMTA20PO-14

## Manage Shopping Cart Items

Attach

Create subtask

Link issue



### Description











As a user of the Automation Exercise platform, I want to be able to manage the items in my shopping cart easily so that I can effectively review, modify, and finalize my purchases without any hassle. This includes being able to add items, adjust quantities, remove products, and see real-time updates to ensure that my shopping experience is smooth and efficient.



# CONDITIILE SI CAZURILE DE TESTARE

**Test cases :** Am definit cazuri de testare pentru fiecare functionalitate cheie, inclusiv interactiunea utilizatorului cu cosul de cumparaturi si procesul de inregistrare si autentificare, pentru a verifica conformitatea cu cerintele specificate si pentru a identifica eventualele probleme.

**Test conditions :** Am definit conditii de testare in conformitate cu cerintele specificate, pentru a verifica interactiunea utilizatorului cu cosul de cumparaturi si procesul de inregistrare si autentificare, asigurand astfel, o experienta lipsita de bug-uri pentru utilizatori.

Type	Summary
	Verify that the login interface and authentication logic correctly authenticate users with valid credentials and grant them access to their accounts.
	Verify Email Verification Is Sent
	Verify that the registration form correctly validates user input, preventing the submission of incomplete or invalid data.
	Verify that the shopping cart updates in real-time when a user adds an item, without requiring a page reload.
	Verify that the shopping cart layout is correctly displayed on a desktop screen.
	Verify that a user can remove a single item from shopping Cart
	Verify that a user can successfully increase the quantity of an item in the shopping cart.
	Verify Password Recovery Workflow
	Verify Backend Handling of Registration Data
	Verify Successful User Registration

# Test cases

Verify that the login interface and authentication logic correctly authenticate users with valid credentials and grant them access to their accounts.

+ Add @ Apps

## Description

This test case verifies the functionality and effectiveness of the login interface and authentication logic implemented on the [Automation Exercise](#) website. The focus is on ensuring that the login system correctly authenticates users who provide valid credentials and grants them access to their accounts.

## Test Details

Detail ▾

Restore Default Size



Add Step ▾

Columns ▾

-	-	✖ Test Step	✖ Test Data	✖ Test Result	Action
⋮	1	Navigate to the login page on the Automation Exercise website.		Checked.	🗑️ ⚙️
⋮	2	Enter a valid registered email address in the email field.		Checked.	🗑️ ⚙️
⋮	3	Enter the correct password associated with the email address in the password field.		Checked.	🗑️ ⚙️
⋮	4	Click the "Login" button.		Checked.	🗑️ ⚙️

## Verify Email Verification Is Sent

+ Add @ Apps

## Description

This test case aims to validate that an email verification message is sent to the user's registered email address upon successful completion of the registration process on the [Automation Exercise](#) website.

-	-	✖ Test Step	✖ Test Data	✖ Test Result	Action
⋮	1	Navigate to the registration page on the Automation Exercise website.		Checked.	🗑️ ⚙️
⋮	2	Complete the registration form with valid information, including a valid email address.		Checked.	🗑️ ⚙️
⋮	3	Submit the registration form.		Checked.	🗑️ ⚙️
⋮	4	Check the email inbox of the registered email address for a verification email from Automation Exercise.		Failed.	🗑️ ⚙️

# Traceability Matrix – Matricea de trasabilitate

- Actualizarile regulate ale matricei de trasabilitate ajuta la mentinerea transparentei si a alinierii intre obiectivele de testare si cele de proiect pe tot parcursul ciclului de viata al testului.
- Matricea indica starea curenta a fiecarui caz de testare, ajutand la monitorizarea procesului de testare si la identificarea oricaror lacune sau lipsa de acoperire.
- Matricea poate fi utilizata in scopuri de conformitate si pentru a demonstra ca toate cerintele au fost testate si si verificate in mod adecvat.
  - In urma executarii a 10 teste, au fost identificate 4 defecte.

Traceability Matrix ? Report Type Export

Type: Story ▼ Versions: All ▼ Cycles: All ▼ Folders: All ▼ Execution Status: All ▼ Clear Filters

Requirements without tests Executions without defects Executions with defects

Issues Showing of 1-2 of 2 [Expand All](#)

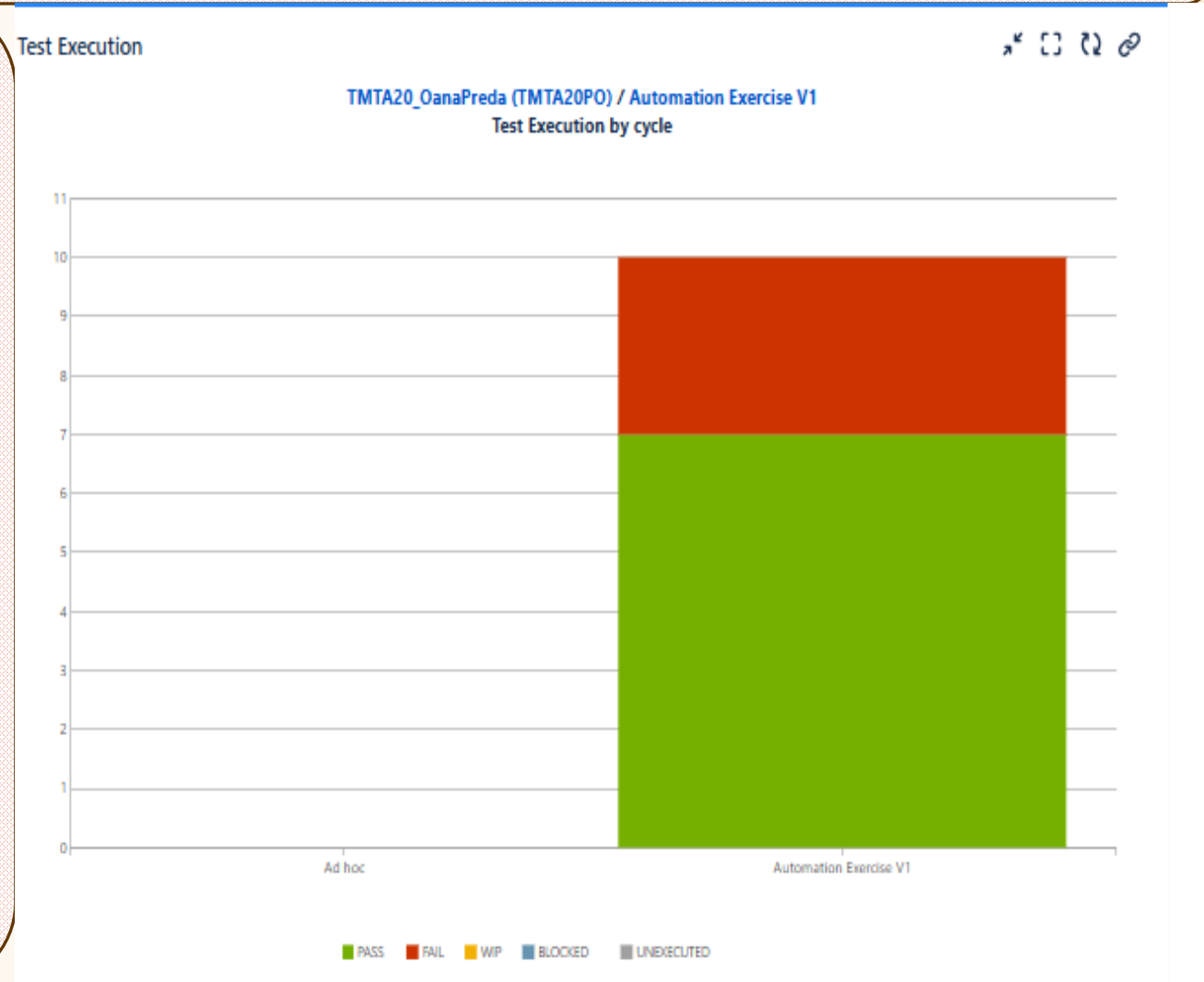
Requirements <span>▼</span>	Tests	Executions	Defects
<a href="#">TMTA20PO-14</a> Manage Shopping Cart Items	1 DEFECTS 4 Tests	4 Executions	<a href="#">TMTA20PO-34</a> A user can't increase or decrease the quantity of an item in... <a href="#">STEP</a> <a href="#">BACKLOG</a>
<a href="#">TMTA20PO-12</a> As a new user, i want that the system allows me to create a...	3 DEFECTS 6 Tests	6 Executions	3 Defects

View 10 20 30 < 1 >

# Raportul general generat in aplicatia Jira

Raportul de executie al testelor a fost generat pentru test cycle summary, creat pentru modulul testat.

Din raportul atasat, reiese ca dintr-un total de 10 de teste efectuate, 3 au esuat.









# Gravitatea bug-urilor identificate

**Gravitate mare :** Imposibilitatea de a primi un e-mail de verificare după înregistrare afectează direct capacitatea utilizatorului de a finaliza procesul de înregistrare. Fără verificare, utilizatorul nu se poate autentifica sau accesa sistemul, ceea ce duce la frustrarea utilizatorului și la potențialul abandon. De asemenea, împiedică afacerea să achiziționeze noi utilizatori în mod eficient, făcând această problemă o prioritate ridicată pentru remediere.

## Verification Email Not Received After Registration

 Attach  Create subtask  Link issue 

### Description

Description: After completing the registration process on the Automation Exercise website, users are not receiving the verification email in their inbox. The verification email, which contains a link to activate the user's account, is not appearing in the inbox, spam, or junk folders.

### Environment

Website/Application: URL: <https://automationexercise.com>


Registered Email Address

Browser Name: Mozilla Firefox

OS Name: Windows 10

### Linked issues

relates to

 TMTA20PO-41 Verify Email Verification Email Is Sent

  BACKLOG 

# Analiza de risc

## Riscuri de proiect :

- Lipsa experientei echipei de testare.
- Timpul scurt de fixare a defectelor.
- Riscul de a dezvolta un produs software slab calitativ si neconform din cauza specificatiilor neactualizate.

## Riscuri de produs :

- Functionalitati noi care pot afecta performanta aplicatiei.
- Complexitatea unor module ale aplicatiei pentru care exista riscul ca utilizatorului sa ii fie dificil sa le parcurga.

## Concluzii generale dupa testare

- In cadrul proiectului au fost executate toate cele 10 cazuri de testare planificate pentru executie.
  - Din cele 10 teste, au fost identificate 4 defecte, cu prioritate high.
- Defectele raportate au fost fixate si retestate, riscurile de produs fiind diminuate prin actualizarea aplicatiei.
- Pentru asigurarea calitatii continue a aplicatiei, se va efectua retestarea si testarea de regresie pentru noua versiune a acesteia.
  - Link-ul de GitHub poate fi accesat [aici](#).

Va multumesc pentru atentie!

