* *Echipa rachetă *

= Plan de testare al unui site de e-commerce =

*Echipa este formată din:*

1. Oprea Alexandru – team lead
2. Purice Claudiu - dev
3. Panait Costin - dev
4. Druta Teodora - dev
5. Niculescu Diana - dev
6. Petre Adelin - tester

Introducere:

Acest plan de testare are urmatoarele obiective:

1. Sa defineasca componentele folosite in procesul de testare
2. Sa comunice tertilor responsabili elementele ce trebuiesc testate, sa stabileasca obiective in functie de un program definit si sa defineasca nevoile de mediu
3. Sa stabileasca cum vor fi efectuate testele

Elementele de test

Sistemele ce vor fi testate includ partea de front-end a site-ului impreuna cu partea de back-end a platformei. Aceste sisteme vor fi testate in Chrome 78.0, FireFox 45.0, Safari 9.1, Internet Explorer 10 si 11.

Testele vor fi facute de pe un sistem de operare Windows 10.

Proprietati ce vor fi testate:

Introducerea in browser a URL-ului ce se doreste a fi accesat

Verificarea daca acesta este disponibil pe Server si returnarea paginii sau trimiterea mesajelor dupa caz. Astfel, exista mai multe scenarii ce trebuie testate:

**Scenariul 1) – succes „200 OK”**

- clientul specifică o resursă existentă și disponibila

-serverul va trimite un răspuns cu acea resursă. Resursa cerută poate fi; pagină web ce poate avea conținut static și/sau dinamic, sau imagini. Toate resursele cerute cor fi afișsate in browser-ul unde s-a efectuat cererea

**Scenariul 2) – format request invalid „400 bad request”**

**Scenariul 3) – acces interzis „401 Unauthorized”**

-credentiale gresite sau inexistente

-server-ul raspunde cu o pagina web ce va contine mesajul ’’Access interzis!’’

**Scenariul 4) – resursă inexistenă „404 Not Found’’**

**-**clientul dorește sa acceseze o resursă care nu există în nicio locație începând cu directorul site-ului setat ca rădăcină în fișierul de configurare

-serverul trimite ca răspuns o pagină web având afisat mesajul ’’Resursă inexistentă!’’

**Scenariul 5) – eroare internă „500 Internal Server Error”**

-cererea clientului nu a putut fi satisfăcută din cauza unei erori interne a serverului web

-serverul va trimite un mesaj corespunzaror, în brwser-ul cientului fiind afișat mesajul ’’Eroare internă!’’

**Scenariul 6) – neimplementat „501 Not Implemented”**

-clientul specifică o altă metodă din cadrul protocolului HTTP decât cele implementate în cadrul serverului

-acesta va primi un răspuns specific acestei erori, o pagină web cu mesajul ’’Metoda HTTP neimplementată!’’

**Scenariul 7) – versiunea HTTP nu este compatibila „505 HTTP Version Not Supported”**

**-**clientul foloseste o versiune HTTP neacceptata

-serverul va informa sugestiv clientul

**Scenariu 8) -pagina a fost mutata permanent „301 Moved Permanently”**

-clientul cere o pagina care a fost la un moment dat pe server, dar care a fost indepartata permanent

-serverul instiinteaza clientul trimitandu-i un mesaj specific

De asemnea, se vor verifica functionalitatile parser-ului de limbaj php in html.

Capabilitati ce nu vor fi testate

Scenariul achizitionarii de pe mobil nu va fi testat. Doar functionalitatile de windows vor fi testate.

**Abordare**

Testele vor fi efectuate folosind Postman.

Pe parcursul efectuarii testelor, rezultatul testelor vor fi salvate ca screen-shoturi intr-un raport de testare.

Cand testele se vor finaliza, raportul de testare va fi intocmit si trimis managerului de proiect pentru a aproba aplicatia sau nu.

Criterii de validare

Toate functionalitatile principale ale sistemului ar trebui sa functioneze in urma testelor individuale. Nu trebuie sa existe defecte majore in functionarea sistemului si un end-user ar trebui sa poate efectua un ciclu complet de plata si sa initieze o cerere de retur fara erori. 95% din toate cazurile de test ar trebui sa treaca si niciunul din testele picate nu trebuie sa fie crucial pentru buna functionare a site-ului.

Nevoi de mediu

Site-ul trebuie sa fie populat cu date potrivite pentru testare, asta insemnand produse multiple cu preturi diferite.

Criterii de suspendare

Testarea trebuie sa fie intrerupta imediat daca sistemul prezinta probleme de uatentificare sau esec in orice actiuni de baza CRUD(Creare, Citire, Actualizare, Stergere).

Sarcini de testare

Urmatoarele activitati trebuie finalizate:

* Planul de testare pregatit
* Specifiicatii functionale scrise
* Pregatirea mediului de testare(date de testare, autentificari de testare, informatii privind plata testelor etc.)
* Efectuarea testelor
* Un raport de testare pregatit

Nevoile de mediu

Serverul trebuie populat cu diferitele tipuri de pagini web, sortate pe categorii in fisiere separate(imagini jepg, png, html, php, etc.). Scrierea in fisierul de configurare a caracteristicilor fiecarei pagini.

Responsabilitati

Testerul trebuie sa inteleaga aplicatia si asteptarile avute in urma efectuarii testelor, trebuie sa stie riscurile cu privire la aplicatie.

Nevoile de personal si instruire

Testarea va fi efectuata de un singur tester. Acesta trebuie sa stie cum functioneaza aplicatia data spre testare si sa o inteleaga.

Program

Testarea va dura o zi de la terminarea aplicatiei.

Riscuri si Contingente

Daca testarea nu va fi completa intr-o zi, ar putea intarziea remedierea erorilor si testarea finala. Daca se intampla acest lucru, se poate sa fie afectata data stabilita pentru finalizarea aplicatiei.

In cazul in care testerul nu intelege suficient aplicatia, testarea ar putea fi afectata si efectutata gresit.

Aprobari

Managerul proiectului trebuie sa fie de acord cu finalizarea proiectului de testare si sa stabileasca momentul in care aplicatia este pregatita sa treaca la urmatorul pas.