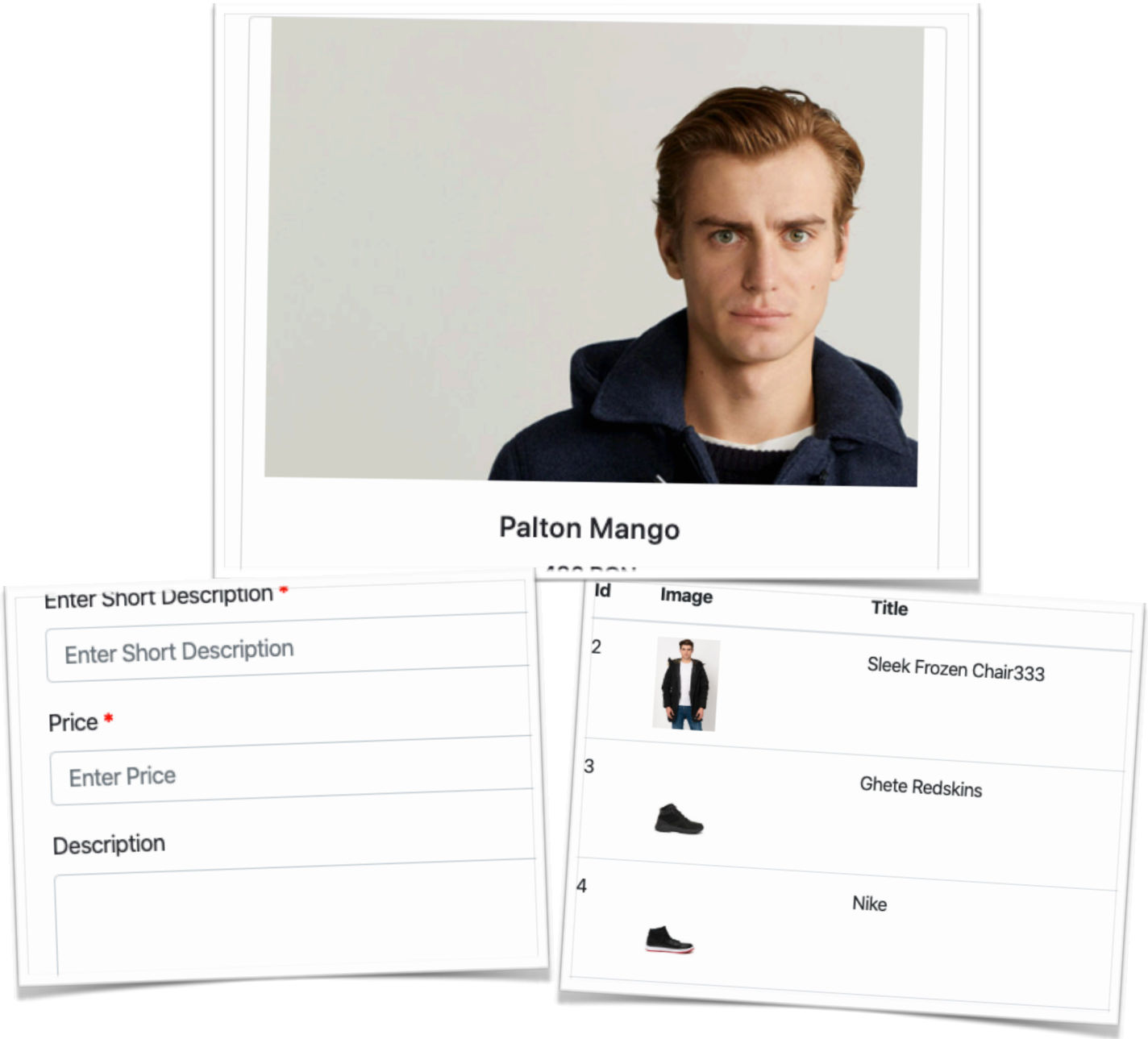
Iosifescu Dorin
Ase Cibernetica
Master Informatica Economica

Proiect afaceri Electronice



Iosifescu Dorin
Ase Cibernetica
Master Informatica Economica

Teste performanta pentru aplicatii web

Aplicatiile web sunt baza oricarei afaceri de comert electronic. In ultimii 5 ani volumul vanzarilor prin intermediul site-urilor de comert electronic a fost in crestere, la fel ca si numarul de comercianti vizibili on-line, competitia fiind din ce in ce mai dura.

Pentru a putea fi competitivi pe aceasta zona trebuie sa oferim o foarte mare atentie asupra urmatoarelor aspecte:

- serverele pe care este gazduita aplicatia trebuie sa suporte un numar mare de utilizatori, disponibilitate 24/7, back-up pentru date
- site-ul sa functioneze pe toate tipurile de dispozitive, atat mobile cat si desktop si sa existe compatibilitate pe toate browserele.
- timp de raspuns foarte bun pentru incarcarea continutului site-ului.
- usurinta la folosire
- securitate
- actualizarea informatiilor din site
- plati integrate cu diversi procesatori de plati electronice

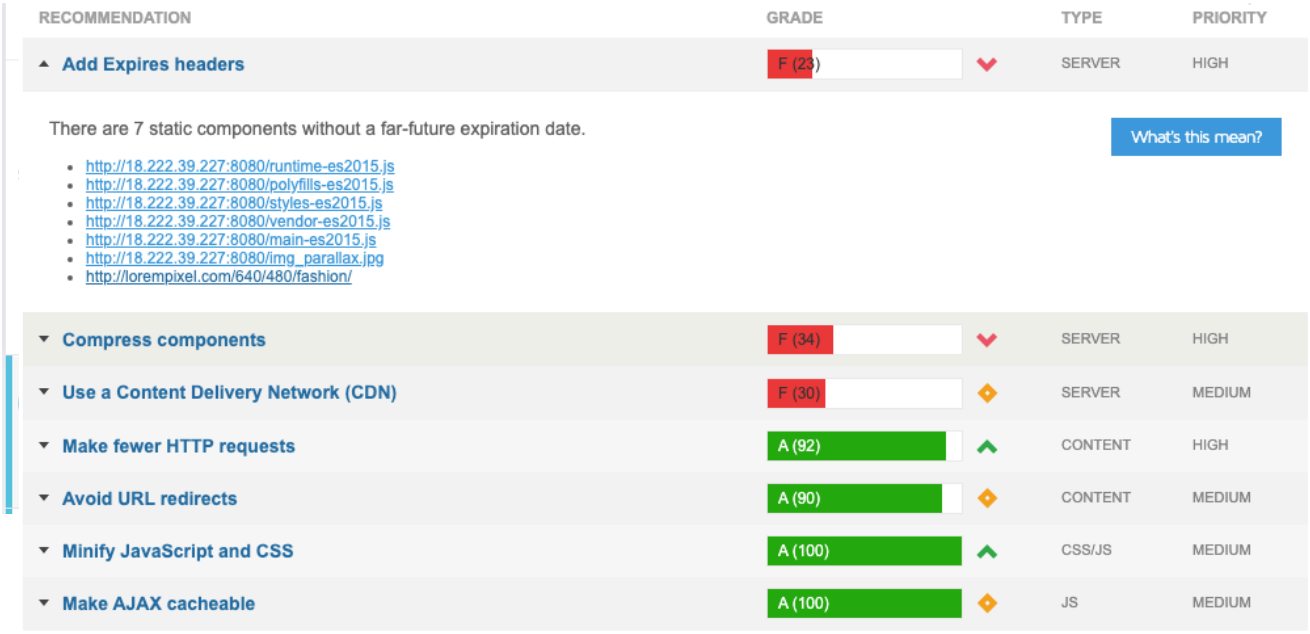
Toti pasii de mai sus trebuie sa intre in sectiunea de testare a produsului si ne ajuta la identificarea problemelor si defectelor de implementare si dezvoltare. Testarea trebuie sa continue pe tot parcursul dezvoltarii aplicatiei.

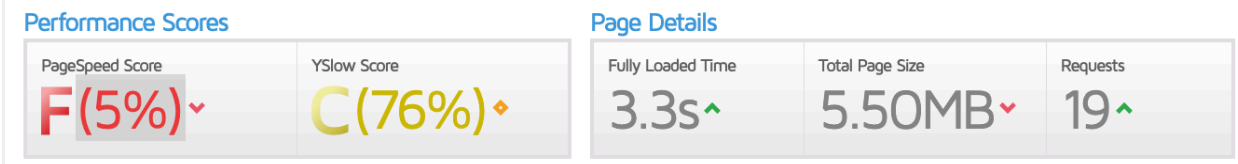
Iosifescu Dorin
Ase Cibernetica
Master Informatica Economica

Aplicare teste de performanta pentru partea de front-end a aplicatie din zona de fashion

Pentru testarea aplicatie am folosit tool-urile Load impact si GTmetrix din acestea rezultand o viteza de incarcare redusa, media fiind de 136 ms.

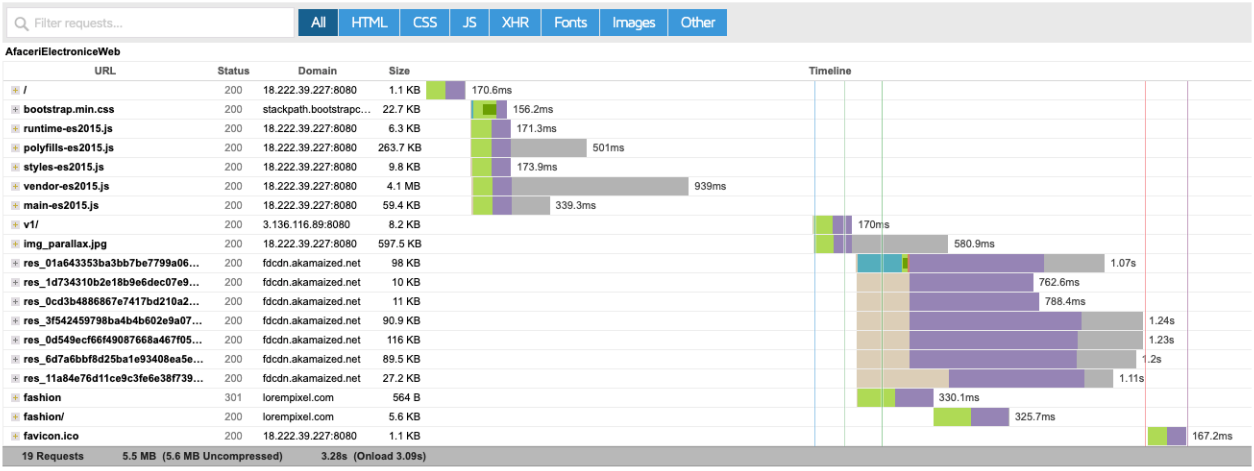
Serverul pe care ruleaza este un server de la Amazon Cloud 9 cu 2 gb ram cu sistem de operare Ubuntu. Aplicatia a fost dezvoltata in framework-ul Angular 8





Iosifescu Dorin
Ase Cibernetica
Master Informatica Economica

RECOMMENDATION	GRADE	TYPE	PRIORITY
Enable compression	F (0)	SERVER	HIGH
Minify JavaScript	F (0)	JS	HIGH
Enable Keep-Alive	F (38)	SERVER	HIGH
Leverage browser caching	D (67)	SERVER	HIGH
Optimize images	C (72)	IMAGES	HIGH
Specify a cache validator	B (85)	SERVER	HIGH
Minimize redirects	A (92)	CONTENT	HIGH
Minify CSS	A (99)	CSS	HIGH
Minify HTML	A (99)	CONTENT	LOW
Specify a Vary: Accept-Encoding header	A (94)	SERVER	LOW



Se observa ca nu este activa compresia pentru javascript, nu exista un mecanism de cache-ing si imaginile nu au fost supuse unui proces de compressie pentru o dimensiune mai redusa.

Conform auditului realizat pentru aplicatia de web trebuie:

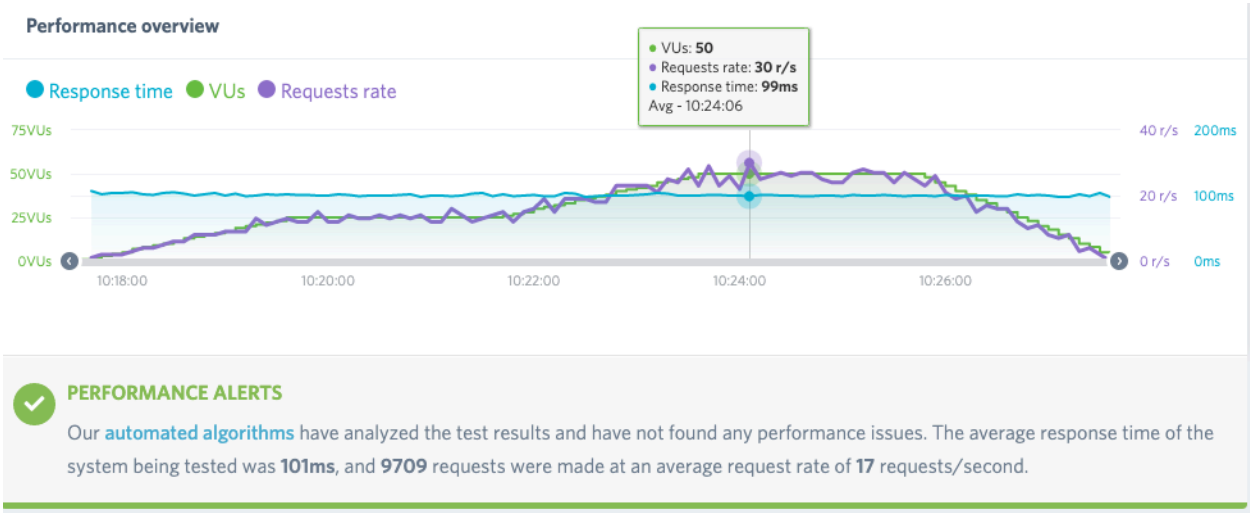
- sa oferim resurse mai mari pentru server
- sa se implementeze un sistem de caching care nu ar mai solicita serverul asa mult

- - sa actualizam pachetele din protect pentru ca nu sunt actualizate si pot genera probleme de securitate
- - sa nu mai folosim librarii din surse externe (Ex: <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

Aplicare teste de performanta back-end aplicatie fashion

Pentru testarea aplicatie de back-end am folosit tool-urile Load impact si GTmetrix din acestea rezultand o viteza de incarcare redusa, media fiind de 101 ms.

Serverul pe care ruleaza este un server de la Amazon Cloud 9 cu 1 gb ram cu sistem de operare Ubuntu. Aplicatie dezvoltata in node js cu framework express



Iosifescu Dorin
Ase Cibernetica
Master Informatica Economica

RECOMMENDATION	GRADE	TYPE	PRIORITY
Avoid bad requests	A (92)	CONTENT	HIGH
Enable compression	A (99)	SERVER	HIGH
Minify CSS	A (99)	CSS	HIGH
Minify HTML	A (99)	CONTENT	LOW
Avoid landing page redirects	A (100)	SERVER	HIGH
Defer parsing of JavaScript	A (100)	JS	HIGH
Enable Keep-Alive	A (100)	SERVER	HIGH
Inline small CSS	A (100)	CSS	HIGH
Inline small JavaScript	A (100)	JS	HIGH
Leverage browser caching	A (100)	SERVER	HIGH
Minify JavaScript	A (100)	JS	HIGH
Minimize redirects	A (100)	CONTENT	HIGH
Minimize request size	A (100)	CONTENT	HIGH

Se observa ca rata de request este mult mai mare pe serverul de la aplicatia web(vezi fig. 1) in comparatie cu cel de api, bazat pe node js.

La cel de web aveam un request rate de 198 per secunda pe un server cu 2 gb ram si pe cel api avem request/rate 30 per secunda la 1 gb ram, dar cu un response time mult mai bun de numai 99 ms pentru ca acesta nu face procesari foarte complexe, doar intoarce text sau json din baza de date.

Singurele modificari ce pot fi aduse aplicatie de api este sa adaugam un favicon, deoarece acesta nu este gasit.