**1.Abstract**

Deoarece din ce in ce mai multi romani prezinta interes pentru domeniul auto, site-ul AutoStat vine cu o serie de informatii care redau intr-o maniera bine-structurata preferintele din tara noastra si genereaza statistici utile pentru cei care doresc sa-si achizitioneze un autovehicul, sau doar sa studieze piata in scopuri personale/profesionale.

Pentru ca numarul masinilor creste de la an la an, este interesant si totodata substantial, sa cunoastem directia in care se indreapta parcul auto din Romania, urmarind treptat evolutia prin intermediul rapoartelor oferite de site-ul nostru.

**2.Introducere**

AutoStat este o aplicatie web care permite vizualizarea de statistici din domeniul auto, pe baza informatiilor furnizate de Guvernul Romaniei, din ultimii ani. Utilizatorii pot accesa situatia parcului auto din tara noastra, statisticile fiind structurate pe judete, iar informatiile fiind centralizate pe baza criteriilor din documentele oficiale ( categoria autovehiculelor, marca, descrierea, valoarea si numarul de autovehicule de tipul respective).

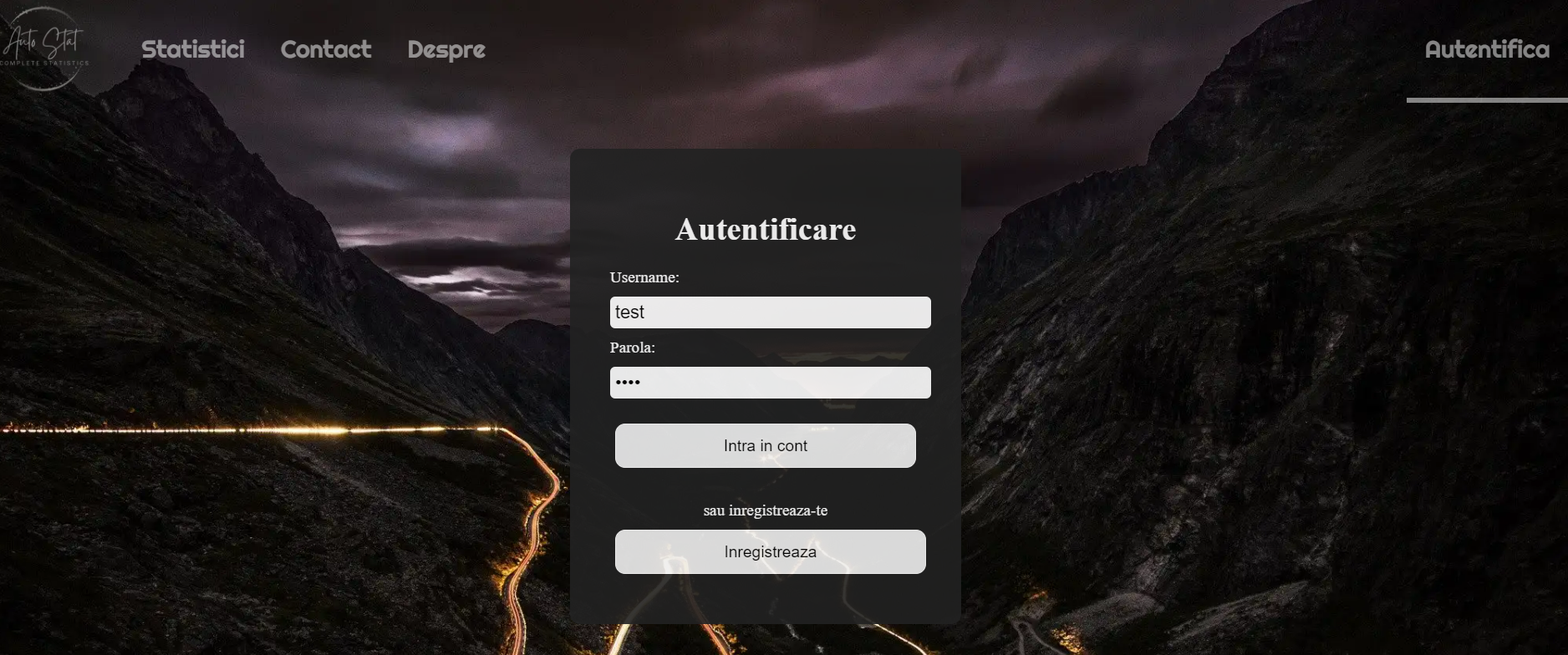
Aplicatia este dezvoltata utilizand Node.JS pentru serviciile API, si tehnologii web clasice (HTML, CSS si JavaScript) pentru interfata utilizatorului. Pentru imbunatatirea procesului de dezvoltare si exploatare a acesteia, am implmentat principia moderne, cum ar fi CI/CD, aplicatia fiind automat actualizata si pe serverele Azure cu fiecare modificare in codebase-ul de pe GitHub.

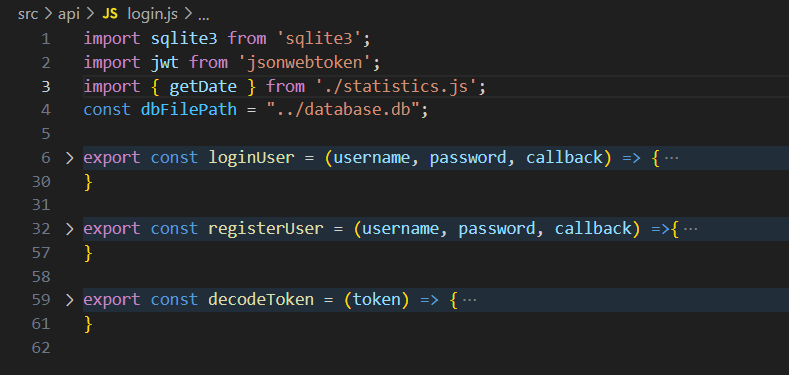
**3.Implementare**

3.1 Inregistrare si Login

Aplicatia permite creearea unui cont de utilizator, urmata de logarea propriu-zisa, care ofera acces catre pagina principala, de unde user-ul isi poate selecta urmatoarea actiune.

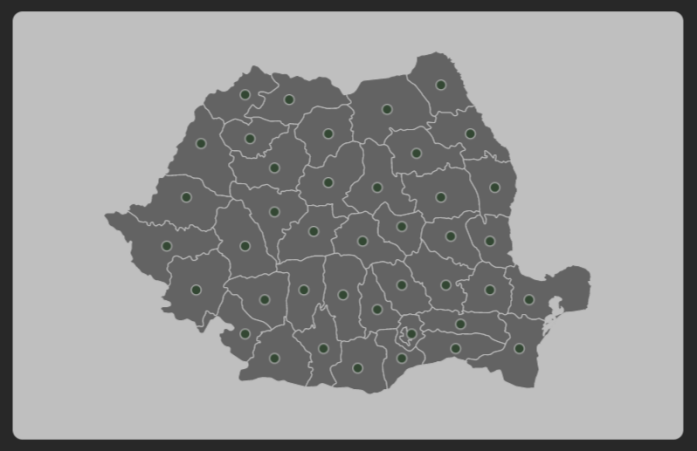
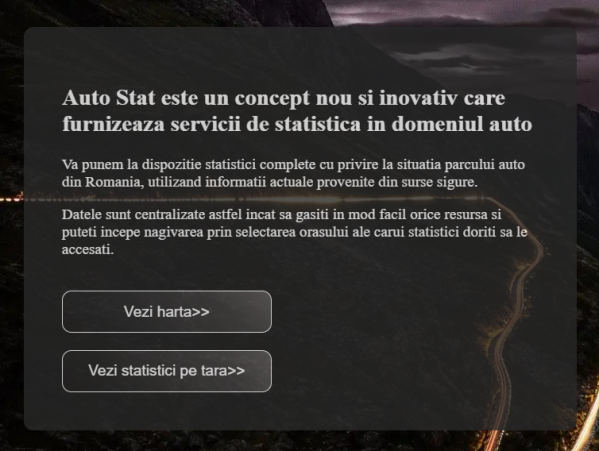
Daca utilizatorul nu are inca un cont, acesta se poate inregistra, iar datele sale vor fi trimise catre Backend si inserate, apoi, in baza de date. La autentificare, daca datele introduse sunt corecte si se gaesesc in baza de date, se genereaza automat un token gwt ( are 12 ore timp pana la expirare), care este mai apoi salvat de Frontent in LocalStorage. Ulterior, acest token este trimis, in cadrul fiecarui request, intr-un header “Authorization”, folosit de backend pentru validarea identitatii utilizatorului, dupa decodificare.



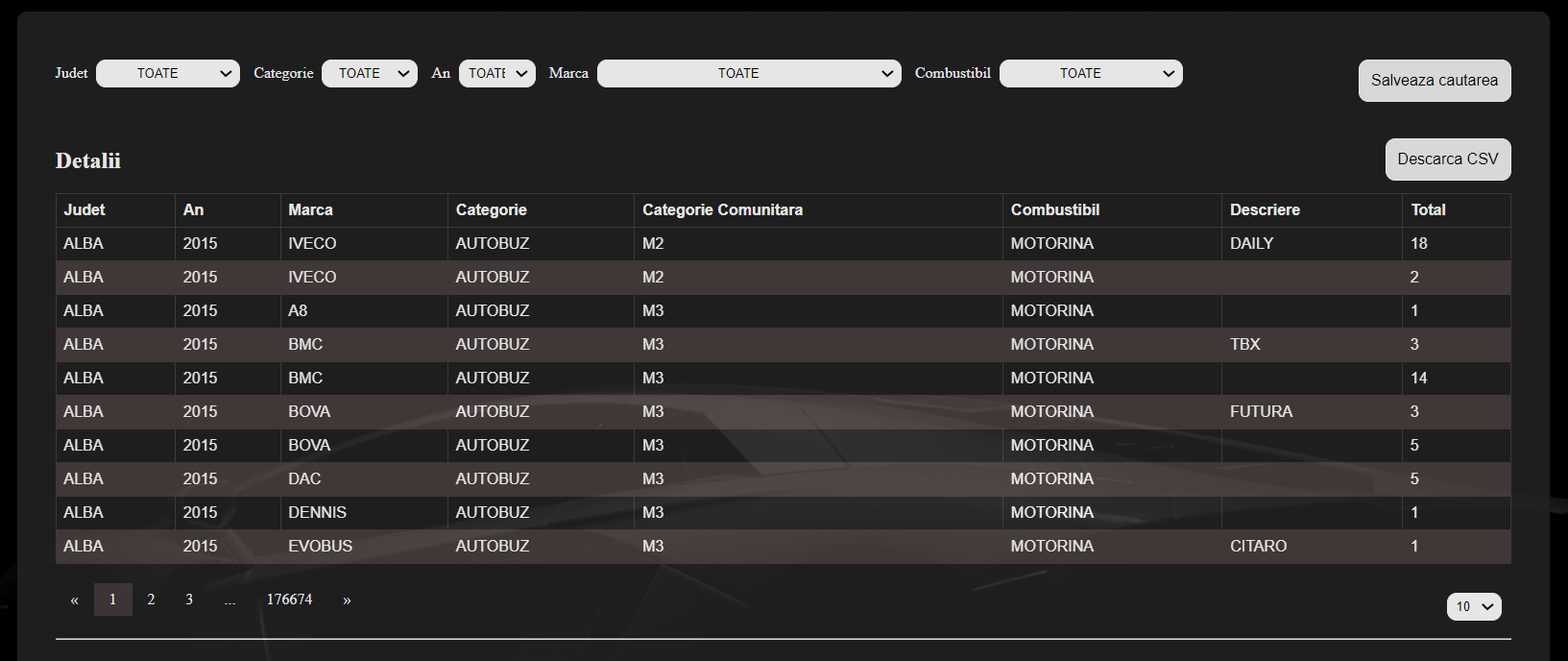


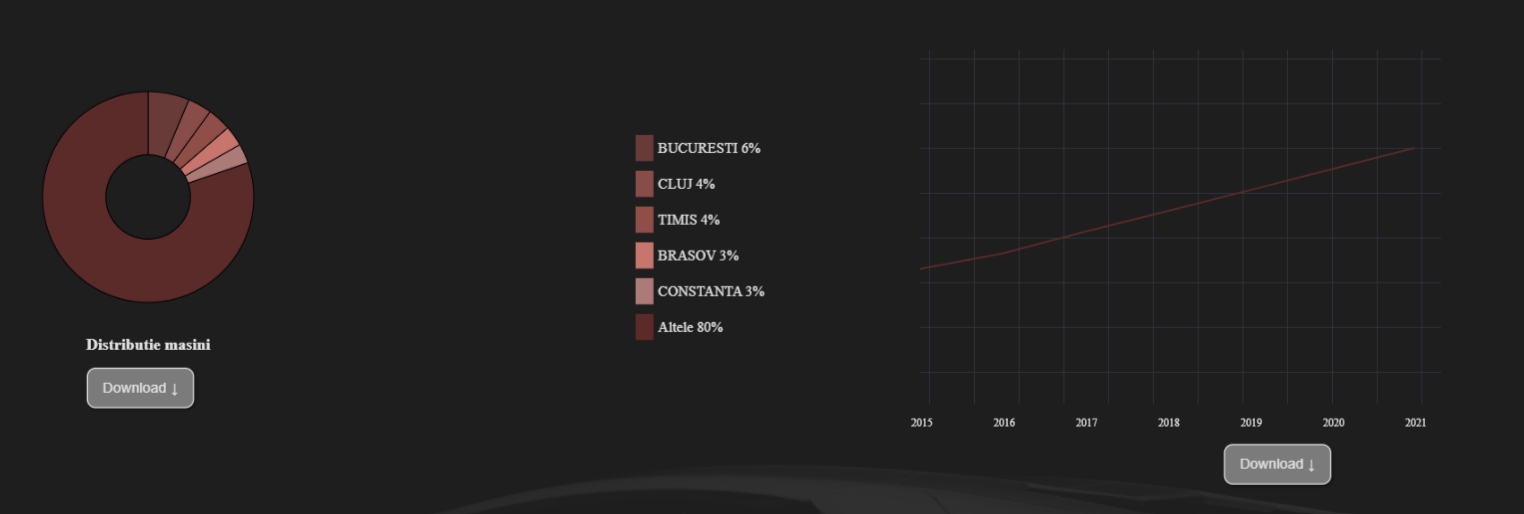
3.2 Vizualizare Statistici

Dupa logare, utilizatorul are acces pagina principal, care contine o sectiune de text informative, iar partea de sus se afla butoanele corespunzatoare pentru navigarea catre sectiunile importante. Butonul “Vezi harta”, aflat in ultimul paragraf, trimite utilizatorul direct catre pagina destinata statisticilor pe judete, in timp ce “Vezi statistici pe tara” va deschide pagina cu statistici corespunzatoare pentru intreg teritoriul Romaniei. Accesand harta Romaniei si apasand pe bulina verde corespunzatoare unui judet, userul este redirectat catre aceeasi pagina de statistici, cu filtrul aplicat pentru judetul selectat.



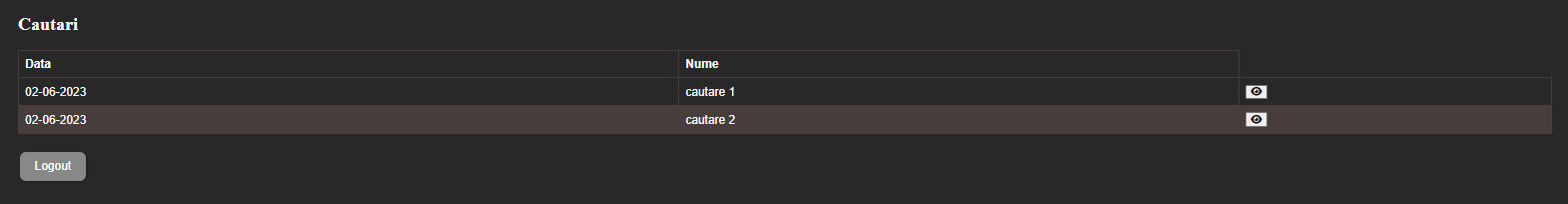
Pagina cu cel mai mare impact este cea destinata vizualizarii de statistici unde, pentru a accesa datele dorite,utilizatorul poate aplica o serie de filtre si poate opta pentru numarul de rezultatate pe care doreste sa le vada pe o pagina. Daca rezultatele cautarii prezinta interes sporit, pot fi descarcate in format CSV. Pentru a oferi utilizatorului o imagine cat mai clara asupra infrmatiilor solicitate, pagina pune la dispozitie, in seciunea de statistici, grafice care arata distributia procentuala a rezultatelor si care pot fi cu usurinta salvate in format Webp. Constructia lor a fost realizata cu ajutorul unei librarii interne, imparite in trei clase, cate una pentru fiecare tip de graphic (Piechart, Linechart, Barchart), plus un fiser “utils” ce contine metode ajutatoare folosite in afisarea pe ecran, utilizand tehnoloigia canvas.





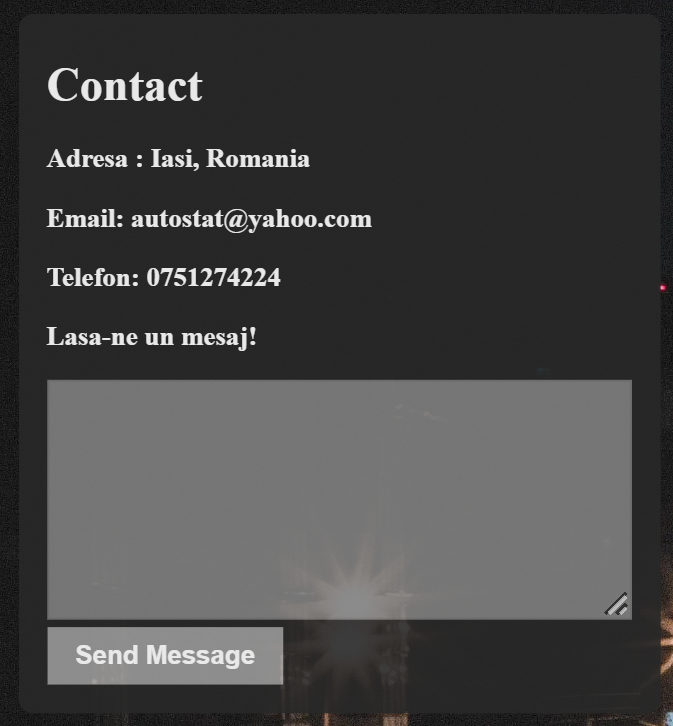


Logarea in pagina ofera posibilitatea de a salva un filtru de cautare aplicat, acesta putand fi ulterior vizualizat din pagina contului personal. Din pagina principala poate fi accesata harta Romaniei unde, apasand pe bulina verde corespunzatoare unui judet, userul este redirectat catre aceeasi pagina de statistici, cu filtrul aplicat pentru judetul selectat.

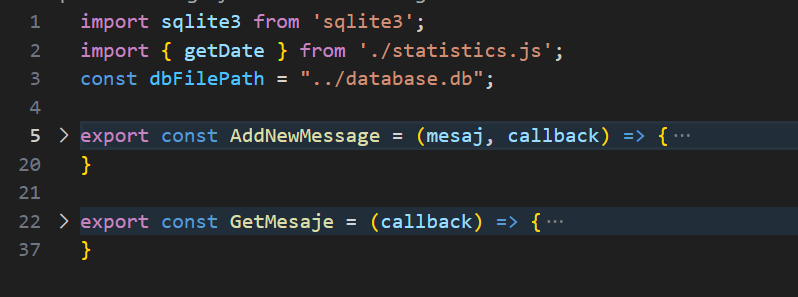


3.3 Pagina de Administrare

Aceesand pagina de Contact, se deschide sectiunea in care se afla datele de contact ale companiei Auto Stat si un textbox in care clientii pot trimite mesaje directe catre administrator. Dupa tastarea textului dorit si apasarea butonul “Trimite”, pe ecran se va afisa un mesaj care sa confirme realizarea cu success a actiunii.



In contextul acestui proces, backendul se ocupa de adaugarea mesajului in baza de date, impreuna cu data adaugarii prin intermediul functiei “AddNewMessage”, obtinerea ulterioara a mesajelor in scopul afisarii fiind asigurata de functia “GetMesaje”.



La logarea in aplicatie cu date corespunzatoare administratorului, in sectiunea “Contul meu” vor putea fi vizualizare mesajele primite de la utilizatori. Pentru obtinerea acestora si afisarea lor intr-un tabel, se face un apel AJAX de tip GET catre ruta corespunzatoare. Tot in pagina de adminstrare va putea fi vizualizat un grafic care arata numarul de utilizatori noi inregistrati in ultimele zile.

3.4 Prelucrarea datelor

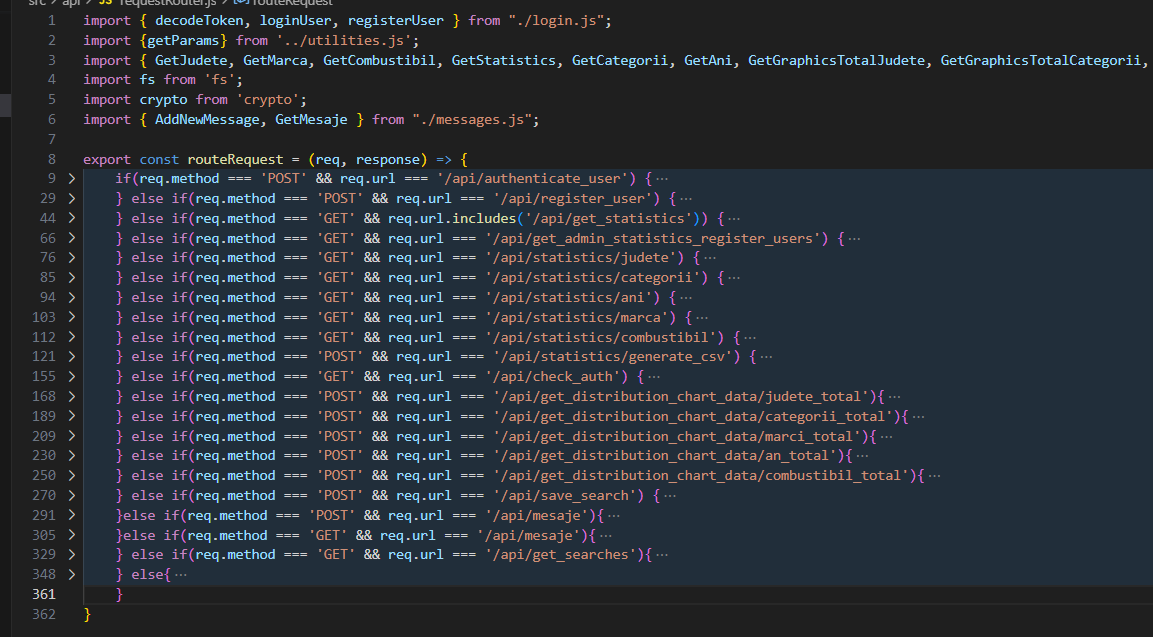
In urma descarcarii CSV-urilor cu informatii de pe site-ul guvernului, datele sunt introduse intr-un folder specific, apoi sunt preluate cu ajutorul unui microseriviciu si inserate intr-o baza de date SQLite, care contine un tabel avand coloane corespunzatoare tuturor specifiicatiilor pentru masini. In aceeasi baza de date, este creat un tabel care contine mesajele trimise de utilizator catre administrator, cel din urma avand posibilitatea de a le vizualiza din pagina de admin, dar si un tabel care permite stocarea cautarilor salvate, in scopul afisarii ulterioare. Tot in baza de date, exista un tabel specific utilizatorilor, care contine username-ul si parola.



Prelucrarea datelor are loc intr-un serviciu separate, pentru a nu afecta performanta aplicatiei. O prima executie are loc la pornirea aplicatiei pe server, ulterior serviciul ruland automat in fiecare zi la miezul noptii prin intermediul unui cronjob, realizat cu ajutorul librariei “node-cron”.

3.5 Rute

In fisierul request router sunt definite toate rutele API utilizate in constructia site-ului. Acestea sunt utilizate pentru autentificare, inregistrare, obtinerea de statistici, generarea de fisiere CSV, gestionarea mesajelor si alte operatiuni relevante pentru aplicatia web. La inceputul fisierului, exista o sectiune pentru importuri, in care sunt incluse modulele si functiile necesare pentru functionalitatea rutelor, dupa care este definite functia pricipala “routeRequest”, care gestioneaza toate cererile primite de la client. Aceasta verifica metoda HTTP impreuna cu URL-ul fiecarei cereri si apeleaza functiile corespunzatoare pentru a trata aceste cereri. Sunt utilizate metode HTTP specific REST, pentru manipularea resureselor si comunicarea intre server si client. In ceea ce priveste resuresele, acestea sunt reprezentate de utilizatorii din baza de date, iar informatiile despre acestia sunt transferate intre client si server sub forma unor obiecte JSON. Comunicarea intre clientsi server este stateless, ceea ce inseamna ca fiecare cerere contine toate informatiile necesare pentru a fi procesata, iar serverul nu mentine informatii desprea starea clientului intre cereri.



3.6 Compresare imagini

Pentru a optimiza scorul dat de Google Page Speed, am recurs la comprimarea resurselor utilizate, prin comprimarea scripturilor css si a imaginilor. Aceste actiuni sunt parte a pipeline-ului de compilare si deployment. Codul SCSS este compilat in CSS si minimizat, iar toate fisierele din folderul /images sunt trecute printr-un script de comprimare ce are la baza libraria npm “compress-images”, ceea ce a dus la un scor de 99.

In pagina Despre, se gasesc informatii utile legate de istoricul AutoStat, care atesta autenticitatea platformei, a statisticilor si siguranta utilizatorilor.

3.2 Design

Pe fiecare dintre paginile aplicatiei web, domeniul auto este sugerat fie de imaginile cu drumuri de noapte, dinamice, fie de unele cu vehicule in nuante inchise, totul fiind asamblat intr-un cadru dark, rece.

Cromatica paginii are la baza nuante neutre, imbinate pe alocuri cu elemente contrastante. Zonele de text sunt construite folosind efecte de transparenta, pentru un design modern si minimalist. Textul de culoare alba, folosit pe majoritatea paginilor, vine in opozitie cu fundalurile gri, oferind utilizatorilor o experienta placuta din punct de vedere vizual.

Logo-ul foloseste atat un font dinamic, pentru numele site-ului, cat si unul clasic, pentru motto, ambele zone de text fiind incadrate de forma ciculara, care foloseste aceleasi nuante neutre.

3.3 Tehnologii Frontend

Pentru realizarea interfetei web am folosit HTML, SCSS, compilat in CSS, si implicit JavaScript.

3.4 Tehnologii Backend

Pe partea de backend utilizam NodeJs, fara alte librarii.