Magazin Online de

Tancuri

INTRODUCERE:

Acest proiect, pe care l-am denumit “Magazin Online de Tancuri” reprezinta un magazin unde utilizatorii pot sa achizitioneze doar un singur tip de tank a cate 10 bucati maxim. Tancurile ce sunt prezentate in acest proiect sunt tancuri din SUA, Germania, China si Franta, cumprizand mai multe categorii, aceelea find Light Tank, Heavy Tank, Medium Tank, SPAA si Tank Destroyers.

Am ales sa sa fac un magazine despre tancuri deoarece sunt un pasionat al tuturor vehiculelor militare de la avioane pana la submarine, etc. si sunt un mare fan al jocului WarThunder, unde ai sansa de a incerca fiecare vehicul, de a vedea informatii referitoare despre acesta, cum functioneaza fiecare tip de proiectil, alegand o tara unde vei debloca aceste vehicule militare, dar trebuie sa dispui de un Battle Pass ca sa le deblochezi mult mai repede.

Din cate am putut sa observ, in Romania nu exista astfel de site-uri unde se pot achizitiona tank-uri, posibil ca legislatia din Romania permite doar unele tank-uri, cum ar fi cele Light Tanks cum ar fi TAB-71 si posibil ca pretul de achizitionare, dar si de intretinere a unui tank este costisitoare.

Consider ca este benefic un astfel de site, deoarece oamenii pot sa afle pretul a unui tank foarte usor, pot dispune de mai multe informatii referitoare despre acesta si poate Armata Romana isi schimba floata de vehicule blindate cu unele mult mai bune fata de cele actuale.

FUNDAMENTE TEORETICE:

Tancurile au aparut ca rapsuns la o necessitate in razboaiele din trecut dar si in cele din present. Primul tanc modern a fost construit de francezi si de britanici, acesta numindu-se “Little Willie” in timpul primului razboi mondial, care a fost un conflict militar de dimensiuni mondiale, un razboi global declansat in Europa , care a inceput in 28 iulie 1914 – 11 noiembrie 1918, unde au participat peste 70 de milioane de militari, unde au peste 9 milioane de combatant si 7 milioane de civili au murit ca urmare a razboiului. Istoria ne specifica ca primele tancuri erau niste tractoare actionate cu abur, unde au ajuns sa performeze in tancuri de necombatut in fronturile de razboi.

SOLUTIA PROPUSA PENTRU REZOLVARE:

Proiectul “Magazin Online de Tancuri”, reprezinta crearea unui site web, care ofera utilizatorului posibilitatea de a interactiona cu site-ul. Acest site cumprinde o gama variata de tancuri asa cum am spus in capitolul Introducere. Utilizatorii pot sa vizualizeze informatii despre fiecare model de tanc, cum ar fi informatii despre blindaj, capacitatea echipajului si despre armamentul disponibil.

Pentru a putea realiza acest proiect am folosit tehnologii precum Java, React si CSS. Pentru a realiza partea de frontend, unde am realizat interfata acestui site am folosit React si CSS cu curmatoarele tehnologii :

* **Axios** face cereri HTTPS catre server
* **Yup** ce se ocupa cu validarea datelor din form

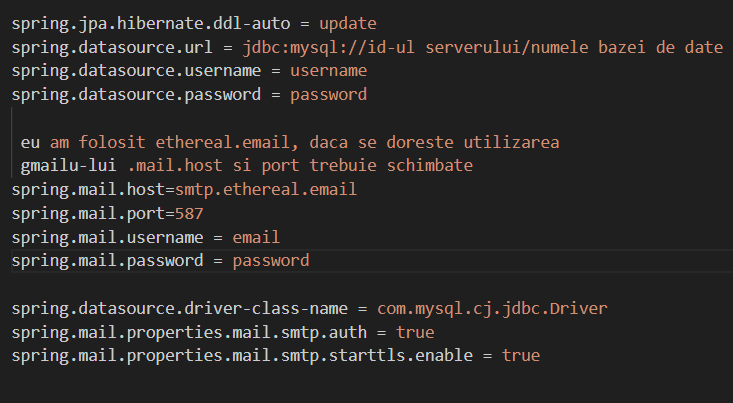
Pentru partea de backend am folosit limbajul de programare Java, unde am adaugat cateva dependency-uri ca si : MySQL Driver, Spring Web, Spring Data JPA si Java Mail Sender, ce imi permit sa adaug datele de cumparare in baza de date, sa se conecteze cu baza de date sis a trimiti un mail cu datele ce sunt introduce de utilizator.

PREZENTAREA APLICATIEI:

Pentru a putea folosi aceasta aplicatie trebuie descarcat sau se poate face un git clone https://github.com/Andrei6700/Online-shop-using-Java

Pentru al putea porni trebuie deschis un terminal unde se adauga cd Front-end , care reprezinta folderu-ul unde am realizat partea de UI, introducem comanda node install in terminal si apoi npm start si acuma am pornit partea de UI al site-ului, acuma trebuie selectat folderul de backend cd Back-end.

Pana sa pornim “serverul” trebuie schimbata configuratia fisierului application.properties din folderul resources, unde vom schimba urmatoarele :



Iar dup ace sa modificat fisierul application.properties, ne ducem la fisierul TankshopAppliaction.java unde il rulam si vom observa daca am facut bine configuratia pentru application.properties ca serverul va porni, dar sa nu uitam sa pornim XAMPP.

STRUCTURA APLICATIEI:

* Front-end:

In folderul de frontend am realizat 2 foldere :

* + UI-Form

In acest folder am am realizat intrega functionalitate a formului, ce este destinat adaugarii de date. Totodata in acest folder pe langa functionalitatea formului, despre care vom vorbi mai tarziu cumprinde si pagina secundara, pagina unde se afla interfata formului.

In codul din pagina-main.js am importat Filters, Content si Footer le-am importat din page-layout in pagina-main pentru a face codul cat se poate mai curat, astfel incat utilizatorul sa poata sa sa is de-a seama unde sunt fiecare componente.

In folderul page-layout impreuna cu CSS-ul am creat 3 fisiere : content.js ce cuprinde totalitate produselor ce sunt afisate pe prima pagaina; filters.js acesta cuprinde filtrele ce sunt afisate pe partea stanga iar footer.js cuprinde un link pentru a-mi accesa adresa de email.

In folderul page-setup este prezentat codul pentru crearea filtrului ce pare in stanga; fisierul ContentChildrens contine niste prop-uri ca de exemplu “{tank.name}” ce il ia de la component parinte, aceea find TankData.js unde sunt afisate toate informatiile referitoare despre tank.

* + UI

In acest folder am realizat intregul design al paginii principale, unde contine imaginiile tancurilor afisate pe site, codul filtrelor din stanga, care momentan sunt nefunctionale, codul pentru afisarea elementelor produsului (descriere, imagine) si layput-ul paginii pentru a putea fi pozitionate asa cum mi-am propus.

* Back-end:

- **EmailSender.java**: Are ca scop trimiterea de emailuri, unde utilizam JavaMailSender pentru a trimite mailuri iar metoda sendSimpleEmail est responsabila cu cu trimiterea unui mail catre adresa specificata in message.setForm(“example@gmail.com”).

- **FormHTTPS.java:** aici sunt gestionate toate cererile HTTPS care vin din hostul setat, ce preia datele din form cand este completat. In SendFormDBAndEmail primeste datele prin cererea de la POST si apoi trimite un mail cu datele introduse.

- **Form.java:** Aceasta clasa contine toate datele ce trebuie completate, adica name, email,tara,telefon, etc.

- **FormRepository.java:** Aici toate operatiile sunt definite pentru accesarea obiectelor din Form in baza de date.

- **FormService.java:** Ea are ca scop salvarea formularului si obtinearea tuturor obiectelor stocate .

- **FormServiceImpl.java**: are ca scop salvarea si obtinerea formului din baza de date folosind FormRepository.

- **TankshopApplication.java:** aici se apeleaza SpringBoot si prin intermediul clasei main, aplicatie este pornita si EmailSender are ca scop trimiterea unui email de notificare catre o adresa de email.

BIBLIOGRAFIE

* <https://www.springboottutorial.com/spring-boot-with-mysql-and-oracle>
* <https://spring.io/guides/gs/accessing-data-mysql/>
* <https://react.dev/learn>
* <https://axiom.org/docs/>
* <https://www.baeldung.com/java-email>
* <https://www.concretepage.com/hibernate/generatedvalue-strategy_generationtype-identity-hibernate>

ANEXE

Aici este prezentat un mic demo al acestui proiect : https://military-tank-shop-oxg6qq21u-andrei6700.vercel.app/ , din pacate nu are functionalitatea de a trimite mail-uri sau sa adauge utilizatorul in baza de date prin intermediul acestui demo.