

Sistem de gestionare al garajelor și al autoturismelor

Albei Liviu-Andrei

BDTS – IF

Grupa: 405

Subgrupa Programare Web cu Java: 3

Business requirements pentru domeniul ales:

1. Gestionarea garajelor

- Sistemul trebuie să permită utilizatorilor să creeze, actualizeze și să șteargă garaje, incluzând detalii precum locația, capacitatea și locurile ocupate. Trebuie aplicate constrângeri pentru a preveni depășirea capacității maxime a garajelor (POJO Validation).

2. Alocarea vehiculelor în garaje

- Sistemul trebuie să permită adăugarea vehiculelor într-un garaj, respectând limita de locuri disponibile și să prevină adăugarea vehiculelor atunci când garajul este plin.

3. Gestionarea vehiculelor

- Sistemul trebuie să permită adăugarea, modificarea și ștergerea vehiculelor, inclusiv detalii precum marca, modelul, anul fabricației, tipul, capacitatea și statusul acestora.

4. Monitorizarea consumului de combustibil

- Sistemul trebuie să înregistreze jurnale de alimentare pentru fiecare vehicul, incluzând cantitatea de combustibil adăugată, costul per litru, costul total și data.

5. Planificarea și înregistrarea întreținerii

- Sistemul trebuie să permită înregistrarea lucrărilor de întreținere pentru vehicule, incluzând descrierea, costul, data curentă și următoarea dată planificată pentru întreținere.

6. Gestionarea cererilor de reparații

- Utilizatorii trebuie să poată adăuga cereri de reparații pentru vehicule, incluzând descrierea problemei, data solicitării, statusul cererii și costul total estimat.

7. Asocierea pieselor de schimb cu cererile de reparații

- Sistemul trebuie să permită asocierea pieselor de schimb cu cererile de reparații.

8. Posibilitate căutare în funcție de ID

- Utilizatorul poate găsi informațiile pentru fiecare entitate în parte după ID-ul asociat acesteia (ID ce se incrementează automat la adăugarea unui nou obiect).

9. Documentare API

- Documentația API a sistemului trebuie să fie făcută pe baza Swagger, care ajută utilizatorul să înțeleagă cum se utilizează API-urile fără o cunoaștere a documentației scrise manual.
- De asemenea, având Swagger la bază, testarea REST API-urilor este mai facilă și ușor de efectuat.

10. Interfață prietenoasă pentru gestionarea vehiculelor și garajelor

- Sistemul trebuie să ofere o interfață intuitivă și ușor de utilizat, care să permită utilizatorilor să vizualizeze și să adauge datele despre garaje, vehicule, alimentări, întrețineri, cereri de reparații și piese de schimb.

Descrierea a 5 funcționalități principale pentru faza MVP.

1. Gestionarea garajelor

- Utilizatorii trebuie să poată crea, actualiza și șterge garaje, incluzând detalii precum locația, capacitatea și numărul de locuri ocupate. Sistemul trebuie să prevină adăugarea vehiculelor dacă garajul este plin.

2. Gestionarea vehiculelor

- Sistemul trebuie să permită adăugarea, editarea și ștergerea vehiculelor, împreună cu detaliile acestora (marcă, model, an fabricație, tip, capacitate și status). Fiecare vehicul nu trebuie să fie asociat neapărat unui garaj.

3. Monitorizarea consumului de combustibil

- Utilizatorii trebuie să poată înregistra și vizualiza jurnalele de alimentare pentru fiecare vehicul, incluzând detalii precum cantitatea de combustibil, costul per litru, costul total și data alimentării.

4. Gestionarea cererilor de reparații

- Sistemul trebuie să permită utilizatorilor să creeze cereri de reparații pentru vehicule, incluzând descrierea problemei, data solicitării, statusul cererii și costul estimat. Cererile trebuie să fie asociate pieselor de schimb utilizate.

5. Istoric detaliat al vehiculelor

- Fiecare vehicul trebuie să aibă un istoric complet care să includă alimentările, întreținerile și cererile de reparații, oferind utilizatorilor o perspectivă clară asupra costurilor și stării fiecărui vehicul.

Baza de date pe care am lucrat.

