**Logo

Description automatically generated**

Proiect Proiectarea Sistemelor Informatice

-Aplicație Web pentru gestiunea unui service auto-

Student: Cupii Cezin

Grupa: 1079

Seria: C

Profesor coordonator: Floarea Alexandra-Maria-Ioana

Cuprins

# Prezentarea sistemului informatic

## Descrierea generală a sistemului informatic

Încă de când eram copil am avut o pasiune pentru mașini, ele fiind, jucăriile mele preferate, probabil ca majoritatea băieților de la acea vârstă. În schimb, odată cu varsta si cu dobandirea carnetului de conducere au inceput sa apara problemele, in general cele administrative, legate de intretinerea autovehiculului personal.

În anul II de facultate, am primit mașina părinților mei în București cu care am împărtășit o serie de peripeții într-un oraș relativ nou pentru mine, care m-am deplasat în mare parte în subteran, cu metroul. Prima întâmplare a fost când o curea s-a rupt și nu am mai putut mișca mașina din loc, așa că a trebuit să găsesc un serviciu de tractare. Cum suntem in era digitală, primul lucru pe care l-am făcut a fost să caut pe Google. Majoritatea serviciilor pe care le-am găsit au fost fie prea departe de locul unde eram, fie prețul era prea mare sau chiar necomunicat.

O altă întâmplare cu același deznodământ a fost în momentul în care a trebuit să îmi fac o programare la un service pentru un simplu schimb de ulei sau când mi-a expirat asigurarea și ITP ul fără ca eu să știu.

În urma acestor peripeții, m-am gândit că trebuie să fac ceva ca cei care ajung în situația mea, destul de mulți la număr, să nu mai piardă la fel de mult timp ca mine și să nu aibă aceleași probleme.

Aplicația eficientizează atât procesul de management al uneia sau a mai multor mașini, de înscriere a acestora pentru o anumită intervenție, cât și procesul de management al unui service, unde se pot vedea mult mai organizat lista mașinilor cât și statusul acestora.

Fiecare utilizator va găsi în cadrul aplicației opțiunea de a adaugă una sau mai multe mașini în contul personal, vizualizarea service-urilor grafic cu ajutorul unor hărți interactive, programarea la un service din lista celor disponibile pentru o anumită intervenție (Diagnosticare, ITP, schimb de ulei),Vizualizarea serviciilor de tractare prin intermediul unei hărți interactive.Primirea de notificări în momentul apropierii datei de expirare a anumitor acte (ITP,asigurare, rovinieta, etc).

Din perspectiva administratorului de service, el va avea acces la funcționalități cum ar fi Vizualizarea tuturor mașinilor din cadrul service-ului respectiv, vizualizarea statusului mașinilor din cadrul service-ului respectiv, vizualizarea profitului din ultimele n zile, mașinile care sunt în așteptare pentru a fi preluate, piesele de care este nevoie pentru fiecare lucrare și, în caz că se dorește, automatizarea comenzilor pentru asigurarea unui stoc constant, o pagină de statistici în care managerul își poate vizualiza vânzările pentru a putea modifică strategia în vederea maximizării profitului și minimizării pierderilor.

## Specificarea cerintelor

Conform celor menționate anterior, aplicația denumită CarCare își propune să ofere utilizatorilor un mijloc eficient de gestionare a mașinilor personale sau a service-urilor pe care le administrează.

### Diagrame ale cazurilor de utilizare

Cerințele funcționale avute în vedere în cadrul dezvoltării aplicației vor fi identificate și modelate prin intermediul diagramelor cazurilor de utilizare.

Diagram

Description automatically generated

**Figura 1 – Diagrama generală a cazurilor de utilizare**

Având în vedere Figura 1, sistemul informatic prezintă două tipuri de utilizatori: clienți și administratori service.

După logarea în aplicație, utilizatorul este clasificat în una din cele două categorii, având anumite drepturi și acces la anumite funcționalități din cadrul ei, astfel distingându-se mai multe cazuri de utilizare pentru fiecare tip.

Clientul va putea gestiona mașinile proprii în cadrul aplicației, acest lucru constând în adăugarea, editarea și ștergerea unei mașini. El poate vizualiza lista de service-uri autorizate prezentă, își poate programa un autovehicul pentru o anumită intervenție în cadrul unui service sau poate vizualiza și serviciile de tractare și locația lor.

Cea mai importantă funcționalitate de care va benefia un administrator de service este aceea de administrare de mașini care constă în vizualizarea programarilor și a statusurilor mașinilor, cât și vizualizarea statisticilor pentru a putea maximiza profitul

### Diagrama cazului de utilizare programare service

Diagram

Description automatically generated

**Figura 2 – Diagrama cazului de utilizare programare service**

|  |  |
| --- | --- |
| Element al cazului de utilizare | Descriere |
| Cod | CU02 |
| Stare | Schiță |
| Scop | Programarea unei masini in service |
| Nume | Programare service |
| Actor principal | Client |
| Descriere | Clientul realizeazaza o cerere pentru programarea masinii in service |
| Precondiții | Clientul trebuie sa aiba aplicatia instalata, conexiune la internet, sa aiba cont activ si cel putin o masina adaugata |
| Postcondiții | Programarea a fost realizata cu succes si clientul primeste o confirmare a acesteia |
| Declanșator | Clientul realizeaza programarea |
| Flux de bază | 1. Clientul se autentifica in aplicatie 2. Clientul selecteaza optiunea de programare a unei masini 3. Clientul completeaza formularul de adaugare 4. Se verifica datele 5. Administratorul de service confirma, daca totul este in regula, programarea |
| Fluxuri alternative | - |
| Relații | - |
| Frecvența utilizării | Frecvent |
| Reguli ale afacerii | Se poate realiza programarea daca toate conditiile sunt indeplinite |

### Diagrama cazului de utilizare administrare masini

Diagram

Description automatically generated

**Figura 3 – Diagrama cazului de utilizare administrare masini**

|  |  |
| --- | --- |
| Element al cazului de utilizare | Descriere |
| Cod | CU03 |
| Stare | Schiță |
| Scop | Administrarea masinilor din service |
| Nume | Administrare Masini |
| Actor principal | Adminstrator Service |
| Descriere | Administratorul poate gestiona toate informatiile legate de masinile din cadrul service-ului sau |
| Precondiții | Administratorul are aplicatia instalata, are acces la internet, este logat si are cel putin o masina in cadrul service-ului |
| Postcondiții | Administratorul a modificat tot ce isi dorea la masinile din cadrul service-ului |
| Declanșator | Administratorul acceseaza pagina cu masinile prezente in service |
| Flux de bază | 1. Administratorul este autentificat in aplicatie 2. Administratorul acceseaza pagina masinilor 3. Administratorul poate vizualiza lista masinilor 4. Administratorul poate cauta o anumita masina dupa anumite campuri 5. Administratorul poate filtra masinile dupa anumite criterii 6. Administratorul poate edita o anumita masina sau o poate sterge |
| Fluxuri alternative | - |
| Relații | - |
| Frecvența utilizării | Foarte frecvent |
| Reguli ale afacerii | Se poate realiza administrarea daca exista cel putin o masina in service |

# Analiza sistemului informatic

## Diagrame de activitate

* Diagrama de activitate „Adăugare mașină”

Diagram

Description automatically generated

**Figura 4 – Diagrama de activitate „Adăugare mașină”**

În figura 4 este reprezentată adăugarea unei mașini de către un client. Acesta va accesa opțiunea de adăugare mașină din meniul aplicației, urmând să fie redirecționat către completarea formularului cu date relevante despre mașină (Marca, an, capacitatea motorului, număr de înmatriculare). După completarea acesteia, sistemul va verifică dacă datele introduse sunt corecte. Dacă datele sunt incorecte, utilizatorul este întors la introducerea lor corectă cu ajutorul validarilor. În cazul contrar, sistemul introduce mașina in baza de date și este întrebat dacă dorește să introducă o altă mașină, in cazul afirmativ, tot procesul este reluat iar in cazul negativ, procesul se încheie.

* Diagrama de activitate „Programare service”

Diagram

Description automatically generated

**Figura 5 – Diagrama de activitate „Programare service”**

În figura 5 se regăsește diagrama de activitate pentru programarea unei mașini în service. Utilizatorul intră în această parte a aplicației unde este întâmpinat de opțiunea de a selecta mașina dorită, urmând să introducă o serie de date necesare programării ( selectarea serviciului, a datei, a service-ului și a pachetului de preț). După ce toate aceste date sunt introduse, acestea sunt trimise în sistem pentru validarea lor. Dacă ele sunt valide, programarea este adăugată în baza de date, administratorul confirmă programarea iar activitatea este încheiată. În cazul în care datele nu trec de validări, utilizatorul este întors in primul pas unde trebuie să introducă datele corecte.

* Diagrama de activitate „Vizualizare servicii tractare”

Diagram

Description automatically generated

**Figura 6 – Diagrama de activitate „Vizualizare servicii tractare”**

Figura 6 prezintă activitatea de vizualizare a serviciilor de tractare, acțiune realizată de un client din cadrul aplicației în momentul în care el accesează opțiunea de vizualizare a serviciilor de tractare, în urma căreia este afișată lista serviciilor de tractare în cadrul unei hărți interactive. În nodul decizional se hotărăște dacă utilizatorul vrea să vadă detalii legate de un anumit serviciu de tractare sau nu. Ambele variante duc la încheierea activității.

## Diagrame de clase

DE ADAUGAT AICI

## Diagrame de stare

* Diagrama de stare a unui utilizator

Diagram

Description automatically generated

**Figura 8 – Diagrama de stare a unui utilizator**

În diagrama de mai sus sunt prezentate stările prin care trece orice utilizator de când deschide aplicația până când termină de lucrat pe ea. Acesta trece prin procesul de autentificare iar dacă datele sunt corecte, el va fi trimis în aplicație și o poate folosi. Când acesta termină ce are de făcut, se deconectează.

* Diagrama de stare a unei programari

Diagram

Description automatically generated

**Figura 9 – Diagrama de stare a unei programari**

În figura 9 este prezentată diagrama de stare a unei programări înainte să fie creată. Această ete realizată de un client care dorește să își repare ceva la unul din autovehiculele personale. După completarea formularului de adăugare, datele sunt verificate de sistem, dacă acestea sunt valide, programarea se realizează și se adaugă în baza de date. În caz contrar, utilizatorul trebuie să refacă cererea.

## Diagrame de interactiune

* Diagrama de interactiune pentru logare

Diagram

Description automatically generated

**Figura 10 – Diagrama de interactiune pentru logare**

Diagrama de interacțiune pentru logare în cadrul aplicației este prezentată în figura 10. Cei 3 parteneri care interacționează sunt userul, sistemul și baza de date. În momentul în care userul dorește să se logheze, el trimite un mesaj sincron către sistem, care îi verifică existența în baza de date și trimite un răspuns. Dacă acesta este pozitiv, userul va fi logat în aplicație. Dacă este negativ, va fi trimis un mesaj de eroare și userul va trebui să reintroducă datele.

* Diagrama de interactiune pentru programarea unei masini

Diagram

Description automatically generated

**Figura 11 – Diagrama de interactiune pentru programarea unei masini**

În diagrama de mai sus este prezentată interacțiunea dintre client, sistem și administratorul de service în momentul în care se realizează o programare de mașină. Userul va selecta mașina pe care dorește să o programeze, primind un răspuns de la sistem cu lista de servicii disponibile pentru mașina selectată. Următorul pas este selectarea serviciului, a datei și a pachetului de preț. După ce fiecare tip de dată este introdus, sistemul le setează și la sfârșit trimite un rezumat al programării. Loop-ul are scopul de a arăta că procesul se poate repeta pentru câte mașini dorește userul.

* Diagrama de interactiune pentru adaugarea unei masini

Table

Description automatically generated

**Figura 12 – Diagrama de interactiune pentru adaugarea unei masini**

În figura de mai sus este reprezentată interacțiunea dintre client și sistem în momentul adăugării unei mașini. Fragmentul loop arată că pentru fiecare mașină, studentul va introduce date, iar in fragmentul alt se verifică dacă datele introduse de către utilizator sunt corecte.