



**UNIVERSITATEA TEHNICĂ**  
DIN CLUJ-NAPOCA

# **Proiectare Software**

## **Colectare Deseuri**

**Profesor îndrumător:**

Anca Iordan

**Student:**

Gavra Anamaria

**Grupa:** 30236



# UNIVERSITATEA TEHNICĂ

## DIN CLUJ-NAPOCA

### Cuprins

Enuntul Problemei.....	2
Justificarea limbajului ales.....	3
Descrierea diagramelor UML.....	5
Diagrama de cazuri de utilizare.....	5
Diagrama de clase.....	6

## Enuntul Problemei

Dezvoltați (analiză, proiectare, implementare) o aplicație desktop care poate fi utilizată de către o firmă de colectare deșeuri. Aplicația va avea 3 tipuri de utilizatori: angajat, coordonator activitate și administrator. Utilizatorii de tip angajat pot efectua următoarele operații după autentificare:

- Vizualizarea listei cu pozițiile deșeurilor ce urmează a fi colectate;
- Vizualizarea traseului optim în funcție de poziția deșeurilor (se va modela prin utilizarea grafurilor).

Utilizatorii de tip coordonator activitate pot efectua următoarele operații după autentificare:

- toate operațiile permise utilizatorilor de tip angajat;
- Operații CRUD în ceea ce privește persistența locațiilor cu deșeuri;
- Alocarea unei locații cu deșeuri către un angajat în vederea colectării;
- Salvare rapoarte/liste cu informații despre liste cu pozițiile deșeurilor (csv,/json).

Utilizatorii de tip administrator pot efectua următoarele operații după autentificare:

- Operații CRUD pentru informațiile legate de utilizatori.

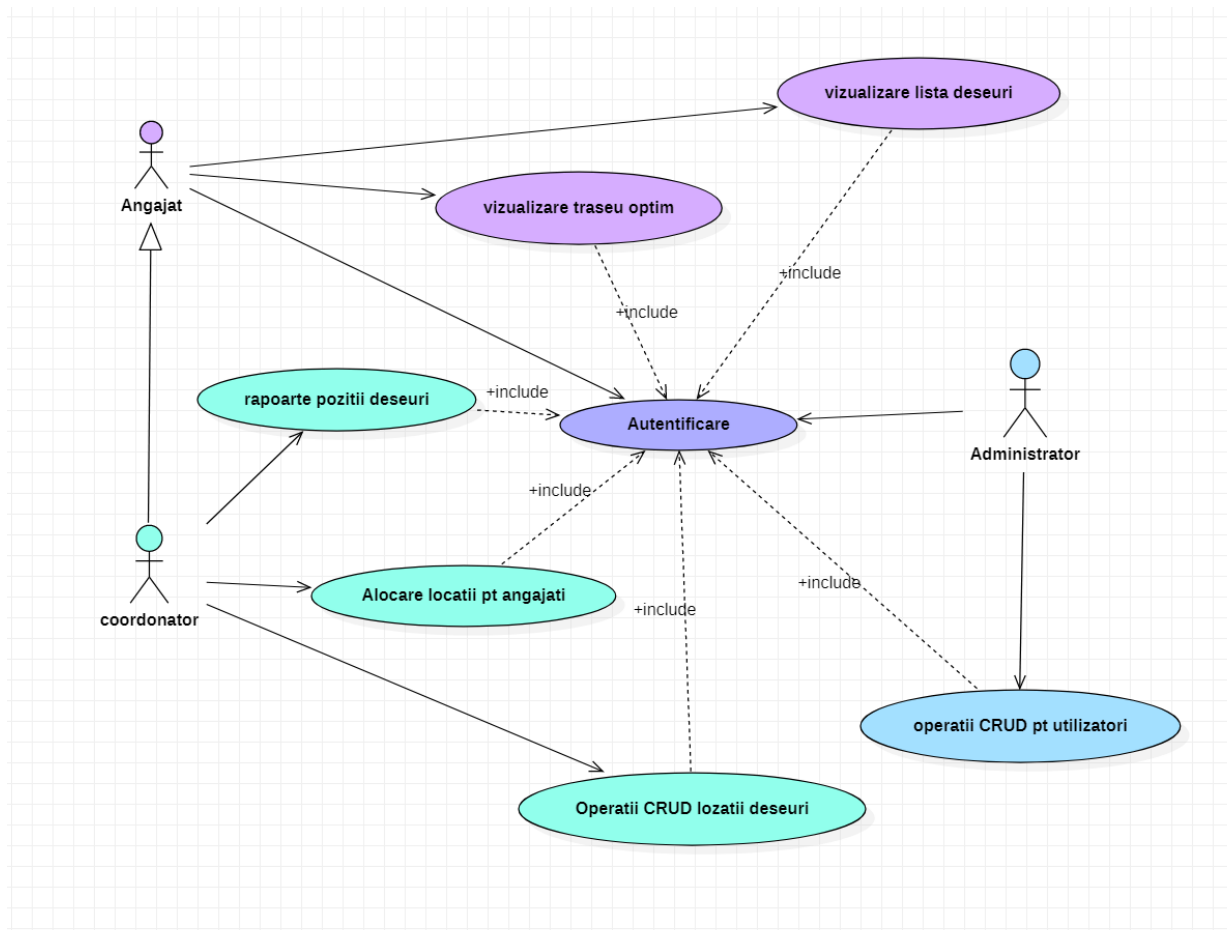
## Justificarea limbajului ales

Am ales sa realizez implementarea acestui proiect in limbajul „Java”, in primul rand, deoarece a fost unul dintre primele limbaje de nivel inalt pe care le-am studiat. Alte motive pentru care am ales acest limbaj sunt urmatoarele :

- Este un limbaj foarte folosit si cunoscut de toata lumea.
- Este usor de inteles
- Codul este organizat
- Se poate muta usor de pe un sistemul unui computer pe al altuia
- Functiile au denumiri sugestive si sunt usor de utilizat.

## Descrierea diagramelor UML

### Diagrama de cazuri de utilizare



**Figure 1**

In aceasta diagrama sunt prezentate actiunile pe care le poate face fiecare utilizator prin intermediul interfetei grafice aferente. Aceste functionalitati sunt prezentate si in prima sectiune (“Enuntul Problemei”). Utilizatorii au toti o functie comuna (cea de login), iar cazurile de utilizare specifice fiecarui utilizator sunt colorate in culorile corespunzatoare.

## Diagrama de clase

Aceasta prezinta clasele implementate in cadrul proiectului. Clasele sunt organizate in trei pachete principale sugestive pentru arhitectura utilizata:

### 1. Pachetul Model

Contine clasele care se ocupa de persistenta datelor si de implementarea functionalizatorilor. Acestea sunt:

- Clasa locatie care memoreaza strada si numarul fiecărei pozitii a deseurilor, dar si numele angajatului caruia ii este atribuita locatia spre colectare.
- Clasa PersistentaLocatii, care, dupa cum indica si numele, se ocupa de persistenta locatiilor, avand functii de citire si scriere in fisier sau de adaugare sau stergere a datelor din lista de locatii.
- Clasa Utilizator care contine numele de utilizator si parola pentru conectare, dar si tipul utilizatorului conectat.
- Clasa PersistentaUtilizatori care contine functiile prin intermediul carora se realizeaza persistenta utilizatorilor.
- Clasa Muchie care cuprinde doua locatii si distanta dintre acestea
- Clasa Oras care cuprinde strazile orasului, locatiile deseurilor si intersectiile, cu aceste informatii calculand traseul minim

### 2. Pachetul View

Contine interfetele grafice pentru fiecare tip de utilizator si cea pentru logarea utilizatorilor si interfete care contin metode pentru accesul la datee din fiecare view.

### 3. Pachetul Presenter



# **UNIVERSITATEA TEHNICĂ**

## **DIN CLUJ-NAPOCA**

Contine cate o clasa de tip Presenter pentru fiecare interfata grafica. Aceasta are rolul de a lega interfata grafica de functionalitatile implementate in pachetul Model.