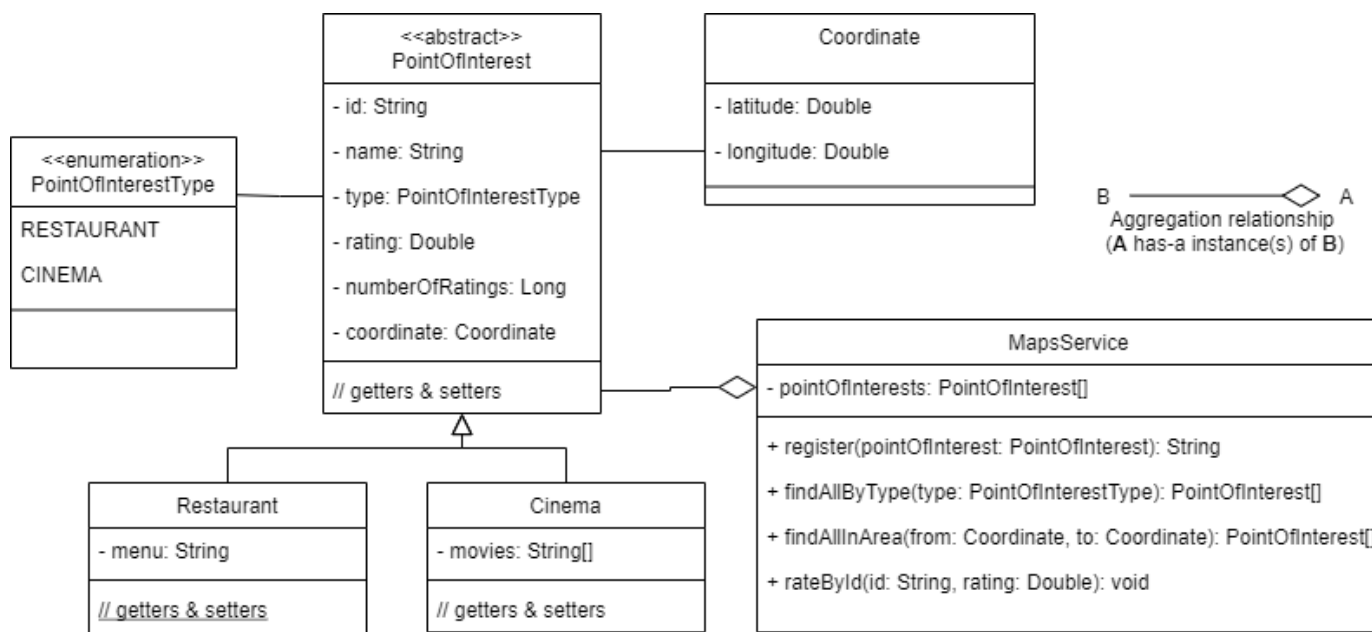


C. Maps

Aplicatia pe care trebuie sa o construiti astazi reprezinta un software de management al unor puncte de interes pe o harta. Aplicatia va lucra cu doua tipuri de puncte de interes: **Restaurant** si **Cinema**. Printre functionalitatile pe care trebuie sa le implementati se numara: inregistrarea unui nou punct de interes, cautarea tuturor punctelor de interes dupa tip, cautarea tuturor punctelor de interes dintr-o anumita zona sau evaluarea unui punct de interes cu o nota.

Figure 1: Digrana UML



Cerinte

1. Cititi si intelegeti diagrama UML mai sus prezentata, iar apoi creati structura de clase dupa cum va este prezentat in diagrama.

2. Implementati metodele din clasa **MapsService** dupa cum urmeaza:

- Metoda **register** trebuie sa genereze si sa atribuiască un **id** unic pentru fiecare obiect primit ca si parametru inainte ca acesta sa fie salvat in vectorul **pointOfInterests**. Acest **id** va fi si valoarea intoarsa din metoda;
- Daca obiectul primit ca si parametru are deja setat atributul **id** atunci se va intoarce mesajul **'Id already set'**. De asemenea, daca punctul de interes care urmeaza a fi salvat nu are oricare dintre attributele **name**, **type** sau **coordinate** setate atunci se va intoarce mesajul **'Invalid data'**;
- Daca nu mai este loc pentru o noua intrare in vectorul **pointOfInterests** atunci acesta se va redimensiona astfel incat sa se poata adauga cu succes noul obiect primit ca si parametru in metoda **register**.
- Metoda **remove** trebuie sa stearga toate punctele de interes primite ca si parametru din vectorul **pointOfInterests** (faceti o cautare dupa **id** aici);
- Metoda **remove** trebuie sa intoarca o lista de puncte de interes care nu au putut fi sterse sau o lista goala daca toate (cele primite ca si parametru) au fost sterse cu succes.

3. Creati o clasa **Main** unde sa se regaseasca metoda statica **main**. Creati o instanta a clasei **MapsService** si niste instance ale celorlalte clase dupa cum urmeaza:

- Creati o instanta a clasei **Restaurant** si doua instance a clasei **Cinema**;
- Inregistrati instancele create anterior folosind metoda **MapsService register**;
- Folositi metoda **MapsService remove** pentru a sterge una din instancele clasei **Cinema**.

4. Implementati metoda **findAllByType** astfel incat aceasta sa intoarca o lista de obiecte de tipul **PointOfInterest** care au acelasi **type** ca si cel primit ca si parametru. Implementati metoda **findAllInArea** astfel incat aceasta sa intoarca o lista de obiecte de tipul **PointOfInterest** care se afla in interiorul regiunii definita de cele doua puncte primite ca si parametri in metoda.

5. Implementati metoda `rateById` dupa cum urmeaza:

- Daca nu se gaseste nici un punct de interes cu id-ul primit ca si parametru metoda nu face nimic si intoarce valoarea minima a unui obiect de tipul `Double`
- Daca se gaseste punctul de interes cu id-ul primit ca si parametru atunci se va actualiza rating-ul punctului respectiv dupa formula $(oldRating + newRating) / (numberOfRatings + 1)$, iar mai apoi se va incrementa `numberOfRatings`