TEMA 1

Aplicatia reuseste sa realizeze gestionarea tuturor locurilor de parcare dintr-un oras, precum si parcarea unei masini intr-un loc liber.

Entitati de baza:

1. Car

Atribute: ID( unic pentru fiecare masina din oras), numar de inmatriculare, lungime, latime, inaltime

1. Bus

Este derivat din clasa Car, numai ca necesita 5 locuri libere pentru a fi parcat.

1. ParkingLocation

Atribute: ID( unic pentru fiecare loc de parcare din oras), tip, zona, capacitate, rezervari din acest loc de parcare.

1. Reservation – descrie interactiunea dintre o masina si un loc de parcare

Atribute: ID, ID-ul masinii, ID-ul locului de parcare, numarul orelor de stationare, numarul minutelor de stationare, data sosirii, data plecarii

In tot orasul se afla mai multe locuri de parcare( cea de la hotelul din cerinta initiala fiind una dintre acestea), fiecare avand un ID unic si o anumita capacitate de locuri de parcare. O masina poate parca intr-un loc de parcare daca mai este loc liber.

Sunt 3 tipuri de locuri de parcare in oras: portocalii( unde se poate parca doar pe termen scurt: 15 min- 1h), mov( unde se poate parca intre 15 min – 3 h) si albastre( unde se poate parca intre 14 min – 24 h).

Preturile difera in functie de locatia locului de parcare: central, comercial sau muncitoreasc. Unele locuri de parcare pot avea un pret mai mare, iar altele un pret mai mic. Pentru toate locurile de parcare, se poate calcula pretul fix in functie de numarul de ore si minute de stationare. De asemenea, preturile fluctueaza pe timpul zilei, fiind mai scump un loc de parcare in mijlocul zilei.

Posibilitati de folosire:

1. Administrator

Pentru a te loga ca si administrator, trebuie stiuta parola: pass!

Posibilitati de folosire:

* adauga un loc de parcare in sistem
* adauga o masina in sistem
* adauga un autobuz in sistem
* face update la locurile de parcare
* face update la o masina
* sterge un loc de parcare din sistem
* sterge o masina din sistem dupa ID
* sterge o masina din sistem dupa numarul de inregistrare
* vede toate masinile
* vede toate autobuzele
* vede toate locurile de parcare
* vede toate rezervarile din tot orasul
* vede toate rezervarile pentru o singura parcare
* vede parcarile sortate dupa capacitate

1. Posesor de masina

Posibilitati de folosire:

* parcheaza o masina( poate face asta doar daca masina a fost introdusa in aplicatie deja) si vede suma pe care o are de plata
* sterge o rezervare
* adauga timp la rezervarea curenta
* vede toate parcarile din oras care au locuri libere
* vede toate parcarile dintr-o anumita zona care are toate locurile libere
* adauga o masina in sistem
* adauga un autobuz
* parcheaza un autobuz

Posesorul de masina trebuie sa isi stie ID-ul masinii pentru ca el functioneaza ca o parola si pe baza sa se fac rezervarile.

Cu 15 minute inainte ca o rezervare sa expire, utilizatorul primeste o notificare si ramane la decizia sa daca vrea sa mai adauge timp la rezervare sau sa plece in 15 minute.

Posibilitati de lucru cu fisiere:

1. txt
2. csv
3. json( ramane de implementat)

In functia main, se poate alege tipul de fisier cu care programul sa lucreze.

Se foloseste un repository template, clasa abstracta, din care se pot deriva repository-uri corespunzatoare diferitelor tipuri de fisiere.

Se folosesc clase de exceptii si validatori.

Diagrama UML:

