

F. Companie de transport in comun - STB.

F.1. Trebuie implementat un modul care sa creeze obiecte de tipul MijlocTransport: Autobuz, Tramvai, Troleibuz. Modulul trebuie sa ajute la crearea de obiecte de familia de clase MijlocTransport. Tipurile de transport sunt salvate intr-un enum{Autobuz, Tramvai, Troleibuz}.

F.2. Pentru crearea de obiecte de tip Autobuz sunt consumate foarte multe resurse. Din acest motiv trebuie propusa o varianta prin care daca exista deja un obiect creat, sa fie folosit acest obiect pentru viitoarele obiecte, fara a mai fi nevoie de crearea de la 0 a obiectelor de tip Autobuz. Aceeasi regula se va aplica si pentru alte tipuri de mijloace de transport.

F.3. Pentru obiectele de tipul AutobuzLinie se doreste ca in momentul in care au fost create obiectele de acest tip sa nu mai poata fi modificate. De asemenea pentru crearea unui obiect de tipul AutobuzLinie trebuie precizat modelul de autobuz folosit, soferul care il va conduce, daca va avea opriri la capat de linie, daca deschide usile in fiecare statie fara ca pasagerii sa solicite acest lucru, textul afisat de ecranul derulator si alte elemente. Unele attribute sunt optionale avand o valoare prestabilita. Se doreste implementarea modulului care se va ocupa de initializarea acestor obiecte, nefiind necesara introducerea atributelor optionale, iar obiectul odata creat sa nu mai poata fi modificat.

F.4. Compania de transport in comun preia si infrastructura de Metrou a orasului si doreste ca biletele sau abonamentele achizitionate de catre clienti pentru transportul terestru sa poata fi folosite si pentru transportul subteran. Sisteme software ale celor doua moduri de transport sunt diferite si se doreste implementarea unui modul care sa permita utilizarea celor doua sisteme fara a le modifica. Modulul implementat trebuie sa se ocupe de validarea biletelor sau si a abonamentelor.

F.5. Autobuzele vin din fabrica cu 3 usi: usa din fata, usa de la mijloc si usa din spate. Pentru fiecare usa, soferul are cate un buton pentru punerea usilor in modul liber (sa poata fi deschise la solicitarea calatorilor) si un alt buton pentru deschiderea fortata a acestora. In total soferul are 6 butoane. Se doreste implementarea unui modul care sa simplifice procesul pentru sofer si sa aiba un buton pentru punerea tuturor usilor in modul liber si un alt buton pentru deschiderea fortata a tuturor usilor.

F.6. Pentru a gestiona garanția asigurată pentru fiecare autobuz disponibil în cadrul flotei companiei, se dorește implementarea unei soluții ce permite vizualizarea fiecărui autobuz în funcție de tipul acestuia și grupul pentru care este destinat. Structura definește o ierarhie între tipurile de autobuze din flotă, grupate pe dimensiunea acestora (grupuri mici – 10 locuri, grupuri medii – 30 locuri, grupuri mari – 50 locuri). Fiecare autobuz este descris de producător, model și număr de locuri.

F.7. Compania doreste ca modulul de printare al biletelor sa permita cu ocazia anumitor zile nationale sa printeze pe bilet un mesaj de "La multi ani". Se doreste implementarea acestui modul care sa adauge functionalitatea de printare mesaj customizat. Exista posibilitatea ca aceasta

functionalitate sa nu fie folosita, si din acest motiv se doreste sa fie optionala, fara modificarea codului existent.

F.8. Pentru autobuzele de noapte se doreste ca oprirea in statie sa se faca doar daca exista persoane in autobuz. In caz contrar autobuzul se retrage la autobaza fara sa mai realizeze opriri. Sa se implementeze modul care va permite oprirea in statie a autobuzelor doar daca exista calatori in autobuz.

F.9. Pentru fiecare AutobuzLinie sunt salvate in memorie, informatii precum: model autobuz, an fabricatie, numar locuri, numar linie, prima statie, ultima statie. Aceste informatii sunt salvate pentru fiecare AutobuzLinie din oras. Se doreste implementarea unui modul care sa asigure gestiunea in mod optim a memoriei, tinandu-se cont de faptul ca anumite informatii sunt redundante, deoarece pe o linie vor merge mai multe autobuze.

F.10. Un calator are posibilitatea sa plateasca cu cardul de calatorii, cardul bancar sau prin SMS. Trebuie implementat modulul dintr-un validator calatorie care sa permita plata calatoriei printruna din cele trei metode. Calatorul va decide modul de plata atunci cand se urca in mijlocul de transport.

F.11. Se doreste implementarea unei aplicatii prin care utilizatorii se pot inregistra intr-o lista de calatori care sa fie anuntati atunci cand autobuzul pleaca de la capat de linie. Autobuzul cand pleaca in cursa de la capatul de linie trebuie sa anunte toti acesti calatori că s-a pus in miscare. Calatorii vor putea in acest mod sa stie cand trebuie sa astepte autobuzul in statie.

F.12. Trebuie implementat un modul care sa ii spuna calatorului ce mijloc de transport trebuie sa foloseasca in functie de distanta pe care o are de parcurs. Astfel, daca un calator are de parcurs o distanta mai mica de 3 km, este recomandat sa mearga cu Troleibuzul. Daca are o distanta cuprinsa intre 3 si 5 km i se recomanda sa foloseasca autobuzul, iar daca are o distanta cuprinsa intre 5 km si 10 km, i se recomanda sa foloseasca Tramvaiul. In cazul in care distanta este mai mare decat 10 km i se recomanda sa foloseasca Metrroul. Sa se implementeze acest modul in cadrul aplicatiei.

F.13. Pentru fiecare Autobuz se solicita salvarea acestora in memorie pentru posibilitatea de a reveni la o forma anterioara a obiectului respectiv. Se doreste implementarea unui modul care sa asigure aceasta salvare cu consum optim de memorie in conditiile in care este cunoscut faptul ca atributele care se vor modifica cu o frecventa mai mare sunt cele legate de soferul care il conduce, consumul mediu, etc. Celelalte atribute precum model, an fabricatie, numar locuri nu se vor modifica frecvent.

F.14. Un tramvai circula de fiecare data pe aceeasi linie, astfel el are de parcurs confom unui patern bine stabilit statiile de pe acea linie: Statia1, Statia2, Statia3, Statia4, Statia5, Statia6 atunci cand merge intr-un sens si invers atunci cand parcurge traseul in sens invers. Sa se implementeze modulului care asigura oprirea in statiile stabilite de catre patern pentru Tramvai.

F.15. Se doreste implementarea unui modul care sa gestioneze autobuzele din cadrul flotei in functie de starea in care se afla fiecare autobuz. Starile posibile sunt: InCursa, LaCapatDeLinie, LaReparat. Un autobuz care se afla la capat de linie poate sa plece in cursa, insa un autobuz care este in cursa nu poate sa plece in cursa. Va putea pleca in cursa doar dupa ce ajunge la capat de

linie. De asemenea un autobuz care este la reparat va putea s aplece in cursa doar dupa ce este reparat. Un autobuz care nu este in cursa nu are cum sa ajunga la capat de linie. Modulul implementat trebuie sa tina cont de toate aceste conditii.

F.16. In cadrul unei autobaze este ceruta implementarea unui modul software prin care operatorul sa poata solicita plecările pentru fiecare autobuz de la capat de linie si pe ce linie va merge (numarul liniei). Comenzile de plecare pe o anumita linie vor fi salvate intr-o colectie, iar in momentul in care autobuzul este disponibil acesta va prelua comanda de plecare in traseu. Operatorul are posibilitatea sa stabileasca plecările de la prima ora a zilei, astfel nu mai este necesara interventia acestuia pe timpul zilei.