



FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
DEPARTAMENTUL CALCULATOARE

Documentatie proiect

la disciplina

Programare orientata pe obiecte

Birou de turism



S P H E R A T R A V E L
DISCOVERING THE WORLD TOGETHER

Tirlea Maria-Cristina & Niculicioiu Stefania-Cristiana, grupa 30225

An academic: 2020 – 2021

Cuprins

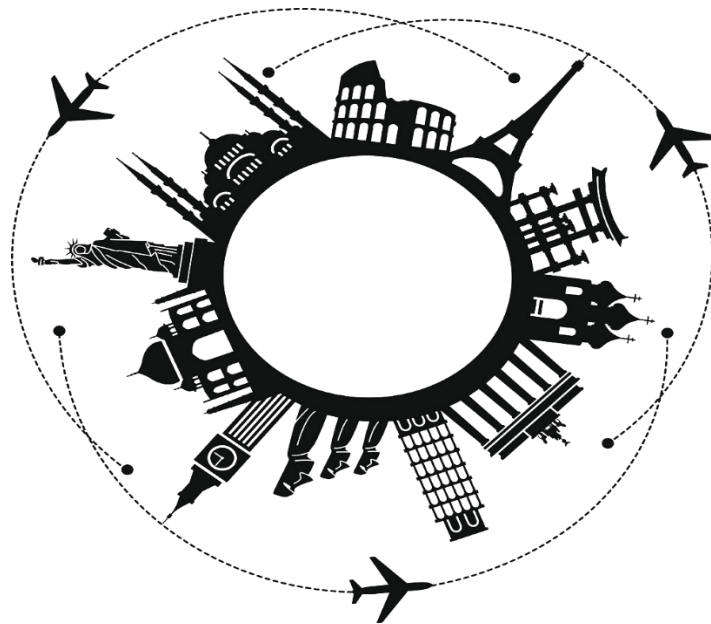
1. Introducere

2. Arhitectura propusa

Diagrama UML

3. Functionalitate

Explicatii pentru diagrama de clase



1. Introducere

Cerinta proiectului

Sa se realizeze o aplicație grafică Java pentru o agenție de turism.

Utilizatorul (angajat al agentiei) poate adauga date despre statiuni cum ar fi:

- nume stațiune
- perioada de sejur, specificata prin data de început si sfârșit a sejurului si preț/sejur.

Tot utilizatorul poate adauga informatii despre clienti cum ar fi:

Date personale, perioada de sejur dorita si stațiunea dorita.

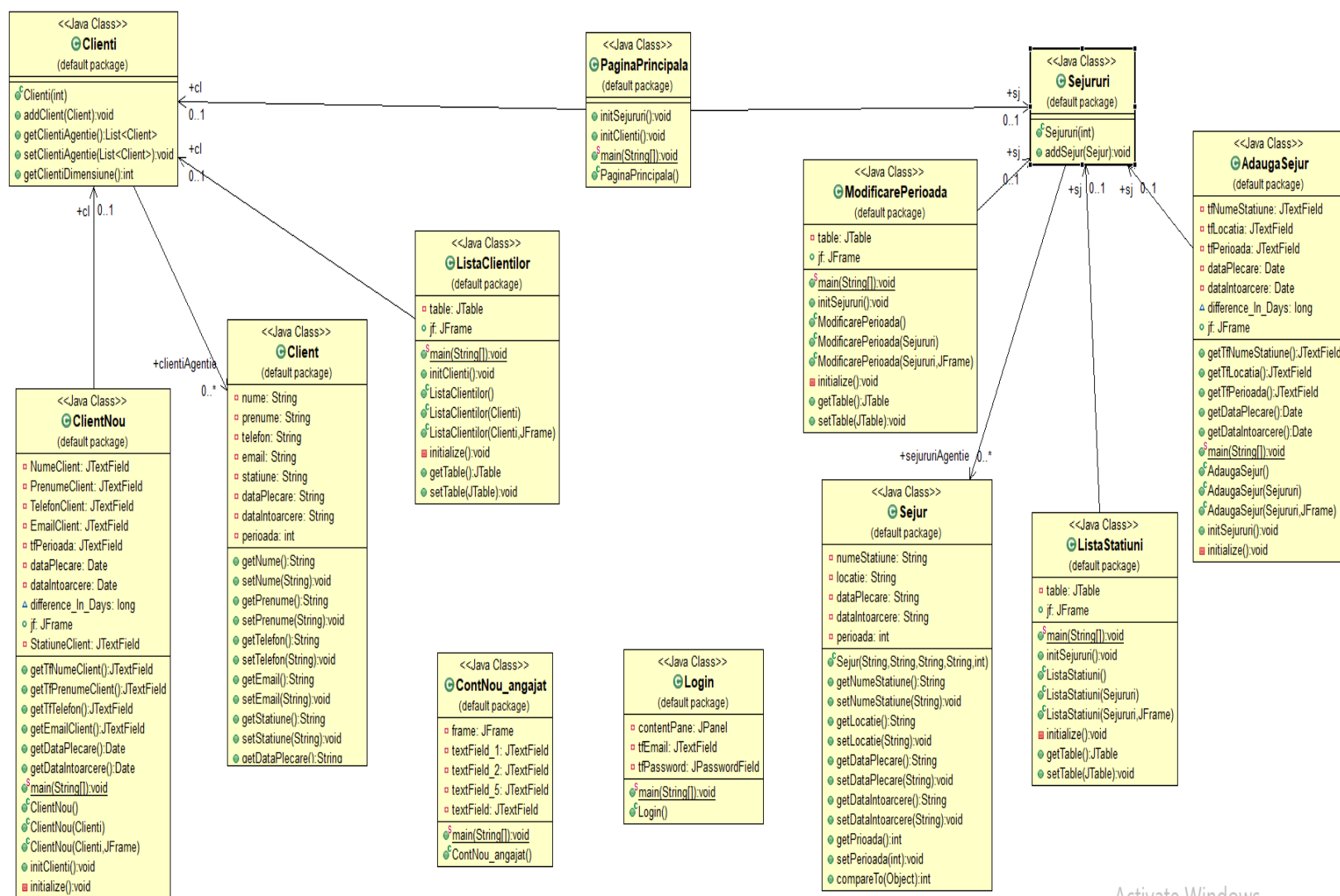
Adaugati aplicatiei butoane / componente care sa permita urmatoarele operatii:

- afișare stațiuni sortate după nume;
- adăugare a unei noi stațiuni cu perioadele aferente;
- adăugare a unei noi perioade la o stațiune;
- afișare clienți care doresc o stațiune.



2. Arhitectura propusa

Diagrama UML



Activate Windows
Go to Settings to activate Windows

3. Functionalitatea

Explicatii pentru diagrama de clase

Programul nostru incepe prin rularea fisierului „Login.java” care prezinta interfata de pornire a aplicatiei si care duce la „Meniul principal”. In aceasta fereasta, utilizatorul care este, de fapt, un angajat al biroului de turism se poate inregistra in aplicatie pentru a desfasura mai departe anumite activitati. De asemenea, am oferit si posibilitatea de o noua inregistrare a unui angajat care inca nu are un cont prin interfata „ContNou_angajat.java”.

Dupa inregistrarea in aplicatie se poate vizualiza interfata de „PaginaPrincipala.java” care face legatura intre toate celelalte interfete. In aceasta sunt puse cele 5 butoane ale aplicatiei. Butonul „Lista statiunilor turistice” va deschide o noua fereasta unde se pot vizualiza intr-un tabel toate statiunile adaugate, ele fiind sortate in ordine alfabetica. Butonul „Lista clientilor” deschide o fereasta in care, de data aceasta, se pot vizualiza toti clientii inregistrati, pe care ii putem sorta in functie de statiunile deja adaugate, pe care si le doresc. Butonul „Adaugare sejur” afiseaza o interfata in care utilizatorul are posibilitatea de a adauga toate datele pentru un nou sejur care se doreste a fi inregistrat. Butonul „Adaugare perioada” va deschide o pagina in care putem vizualiza toate sejururile disponibile si ofera posibilitatea utilizatorului de a modifica datele de inceput/sfarsit ale acestora. Butonul „Client nou” deschide o interfata in care utilizatorul poate adauga date despre un nou client. Atat pentru sejururi, cat si pentru clienti au fost adaugate un numar de date, cu posibilitatea de a mai adauga pe parcursul rularii programului.

Fisierul „ListaStatiuni.java” contine interfata prin care se pot vizualiza toate sejururile disponibile, printr-un tabel cu 5 coloane in care sunt stocate pe rand: numele statiunii, locatia acesteia, data plecarii, data intoarcerii si perioada in care aceasta este disponibila. In aceasta interfata este si un buton „Sort” care odata ce este apasat o sa ordoneze alfabetic in functie de numele statiunii, toate sejururile disponibile. De asemenea, mai exista si un buton „Back” care odata ce este apasat se intoarce la „Meniul Principal”, inchizand interfata curenta.

Fisierul „ListaClientilor.java” contine un tabel cu lista clientilor inregistrati care pot fi sortati in functie de statiunea pe care si-o doresc. Numele statiunilor

sunt pastrate intr-un ComboBox din care pot fi alese. De asemenea, si in aceasta interfata utilizatorului ii este disponibil un buton „Back” care ii permite revenirea la pagina principala, inchizand fereastra curenta.

Fisierul „AdaugaSejur.java” contine mai multe componente de tipul „JTextField” si, de asemenea, 2 campuri pentru a alege data dintr-un calendar specific. Acestea au putut fi realizate cu adaugarea unui jar numit „jcalendar”. Utilizatorul are posibilitatea de a completa campurile cu datele unui nou pachet de vacanta. Dupa ce toate cele necesare au fost completate, utilizatorul poate apasa butonul „Confirmare” care verifica daca toate campurile sunt completate corespunzator si care finalizeaza adugarea unui sejur in aplicatie, iar daca campurile nu sunt completate afiseaza mesajul de eroare: „Va rugam, completati toate campurile!”. De asemenea, si in aceasta interfata utilizatorului ii este disponibil un buton „Back” care ii permite revenirea la pagina principala, inchizand fereastra curenta.

Fisierul „ModificarePerioada.java” prezinta o interfata in care este vizibil din nou un tabel in care se pot vizualiza toate sejururile inregistrate cu posibilitatea de modificare a datei de plecare/intoarcere direct din coloana corespunzatoare. Aceasta modificare o sa fie vizibila dupa in toate celelalte interfete in care se afiseaza data sejururilor. De asemenea, si in aceasta interfata utilizatorului ii este disponibil un buton „Back” care ii permite revenirea la pagina principala, inchizand fereastra curenta.

Fisierul „ClientNou.java” deschide o noua interfata care are mai multe componente de tipul „JTextField” si, de asemenea, 2 campuri pentru a alege data dintr-un calendar specific. Utilizatorul are posibilitatea de a completa campurile cu datele unui nou client. Dupa ce toate cele necesare au fost completate, utilizatorul poate apasa butonul „Confirmare” care verifica daca toate campurile sunt completate corespunzator si care finalizeaza adugarea unui nou client in aplicatie, iar daca campurile nu sunt completate afiseaza mesajul de eroare: „Va rugam, completati toate campurile!”. De asemenea, si in aceasta interfata utilizatorului ii este disponibil un buton „Back” care ii permite revenirea la pagina principala, inchizand fereastra curenta.

Fisierul „Clienti” contine clasa publica clienti si clasa client ce detine urmatoarele attribute: nume, prenume, telefon, email, statiune, dataPlecare, dataIntoarcere si perioada. Aceasta clasa are atat un constructor ce initializeaza un obiect de tip client, cat si functii de tip set si get ce ajuta la setarea unui atribut al clasei sau la returnarea sa pentru a fi accesibil din alta clasa. De asemena, in

clasa Clienti se gaseste o lista cu obiecte de tip client. Functia “addClient” adauga un obiect de tipul client primit ca si parametru in lista clasei (clienti). Functia “getClientiDimensiune” returneaza numarul de clienti stocati in lista.

Fisierul „Sejur.java” contine clasa sejur ce detine urmatoarele atribute: numeStatiune, locatie, dataPlecare, dataIntoarcere si perioada. Aceasta clasa are atat un constructor ce initializeaza un obiect de tip sejur, cat si functii de tip set si get ce ajuta la setarea unui atribut al clasei sau la returnarea sa pentru a fi accesibil din alta clasa. De asemenea, in clasa Sejururi se gaseste o lista cu obiecte de tip sejur si, de asemenea, o functie de „addSejur” care adauga un obiect de tipul sejur primit ca si parametru in lista clasei (sejururi).