

2.4 Proceduri și Funcții

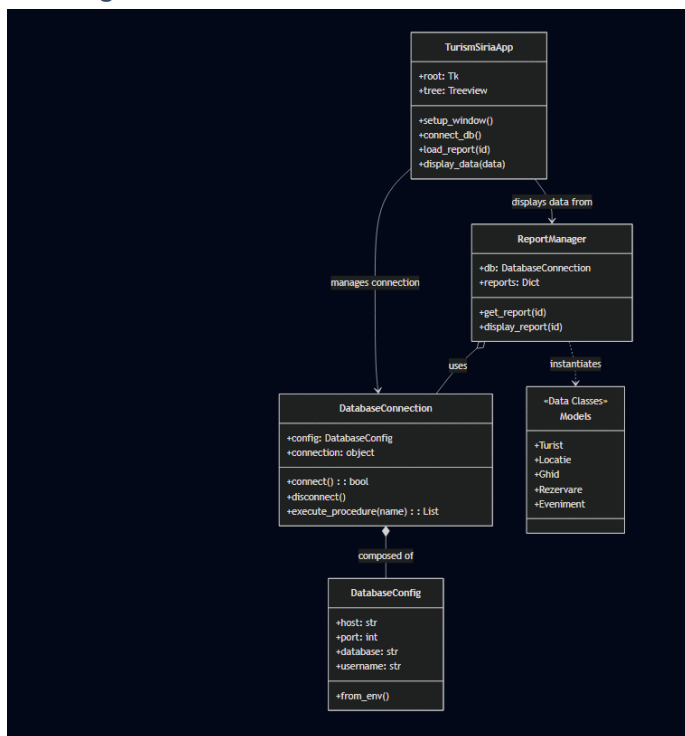
Procedură	Complexitate	Descriere
sp_top_locatii_rating	4	Top locații ordonate după rating mediu
sp_statistici_tururi_regiune	6	Statistici tururi agregate pe regiuni
sp_analiza_turisti_activi	7+	Analiză completă turiști cu vizite și cheltuieli
sp_performanta_ghizi	7	Evaluare ghizi după venituri și rating
sp_popularitate_unesco	7	Popularitate situri UNESCO
sp_evenimente_participari	5	Evenimente cu participări și ocupare
sp_dashboard_sumar	4	Dashboard cu metrice generale

Triggere implementate:

- trg_update_nr_vizite - Actualizare automată număr vizite turist
- trg_validare_rezervare - Validare dată tur în viitor
- trg_update_participanti - Actualizare număr participanți eveniment

3. Descrierea Aplicației

3.1 Diagrama de Clase



3.2 Arhitectura și Structura Claselor

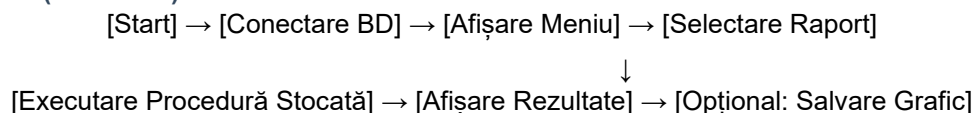
Aplicația este dezvoltată în Python și folosește următoarele module:

- main.py - Aplicația principală (consolă) cu meniu interactiv
- gui.py - Interfața grafică Tkinter
- models.py - Modele de date (dataclasses)

Clase principale:

- DatabaseConfig - Configurația conexiunii la BD
- DatabaseConnection - Gestionarea conexiunii Oracle/SQL Server
- ReportManager - Gestionarea și afișarea rapoartelor
- ReportVisualizer - Vizualizări grafice cu matplotlib

3.3 Fluxul de Lucru (Workflow)



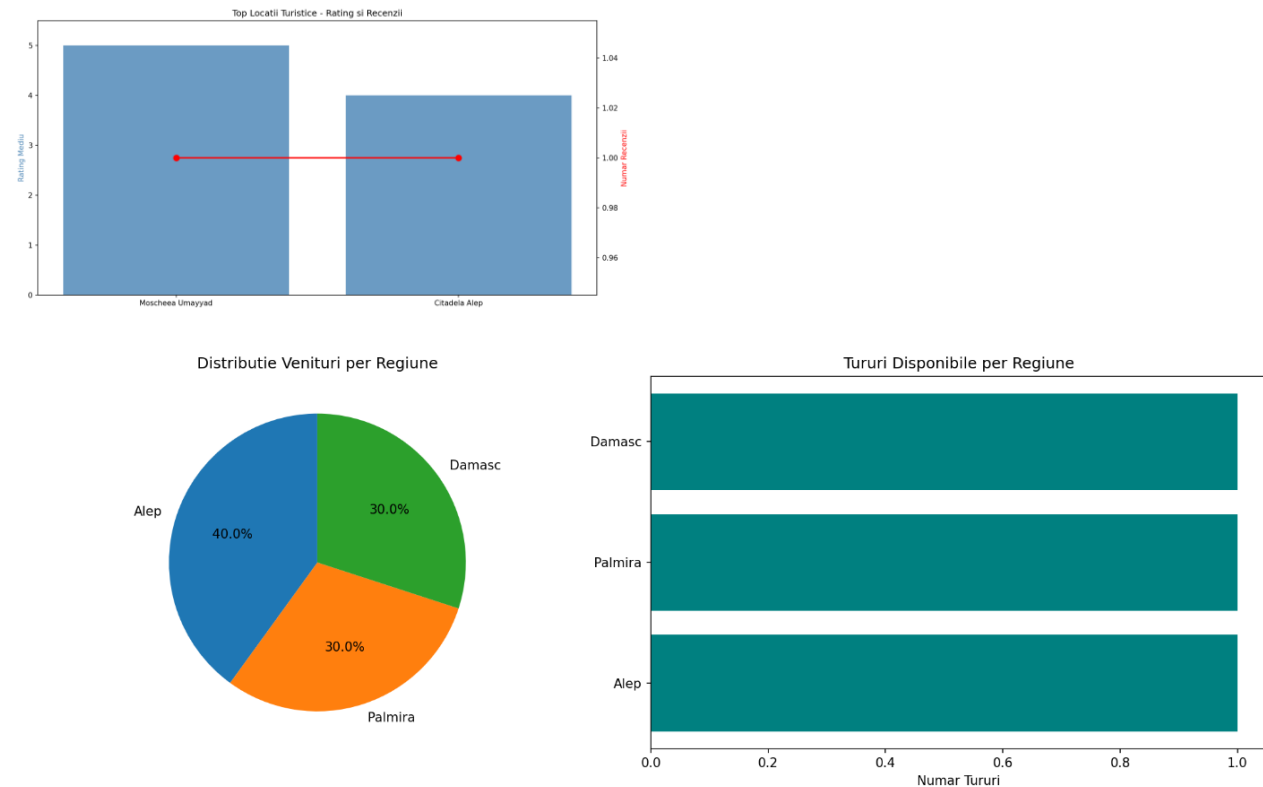
3.4 Conexiunea la Baza de Date

Conexiunea se realizează folosind biblioteca oracledb pentru Oracle. Configurația se încarcă din variabile de mediu (DB_HOST, DB_PORT, DB_NAME, DB_USER, DB_PASSWORD). Aplicația apelează exclusiv proceduri stocate, conform cerinței proiectului.

Exemplu conexiune:

```
dsn = f"{host}:{port}/{database}"  
connection = oracledb.connect(user=user, password=pwd, dsn=dsn)
```

3.5 Capturi de Ecran



Exemple de rapoarte afișate:

- Meniul principal cu cele 7 opțiuni de rapoarte
- Raport Top Locații cu rating și număr recenzii
- Dashboard sumar cu metrice generale

4. Concluzii

Proiectul implementează cu succes un sistem complet de gestiune pentru turism cultural. Baza de date este normalizată, cu 11 tabele și constrângeri de integritate corespunzătoare. Aplicația Python oferă o interfață intuitivă pentru accesarea rapoartelor, toate generate prin proceduri stocate.

Punctele forte ale proiectului includ: arhitectura modulară, separarea logicii de business în proceduri stocate, triggere pentru menținerea consistenței datelor, și rapoarte cu complexitate variată (4-7+) care oferă informații valoroase despre activitatea turistică.

5. Bibliografie

Curs Baze de Date 2 - Platforma Moodle

Oracle Database Documentation - <https://docs.oracle.com>

Python oracledb Documentation - <https://python-oracledb.readthedocs.io/>

Matplotlib User Guide - <https://matplotlib.org/>