

Metode Avansate de Programare, sectia Informatică 2021–2022, LABORATOR 5

DEADLINE: Săptămâna 9

Organizare: TEAM WORK (VCS: GIT, Repository in Cloud – bitbucket)

A. Cerințe funcționale (– vezi fișierul **Enunt Proiect Lab - Retea de socializare.pdf**):

1. (1p) Să se afișeze toate relațiile de prietenie ale unui utilizator citit de la tastatură. Se va afișa:
Nume Prieten | Prenume Prieten | Data de la care sunt prieteni
2. (1p) Să se afișeze toate relațiile de prietenie ale unui utilizator, create într-o anumită luna a anului, utilizatorul și luna din an se citesc de la tastatură. Se va afișa:
Nume Prieten | Prenume Prieten | Data de la care sunt prieteni
3. (2p) Adăugați clasa Message având următoarele attribute: id, from: Utilizator, to: List<Utilizator>, message: String, data: LocalDateTime; Derivați ReplyMessage (Collaboration) din clasa Message și adăugați un atribut message: Message (care specifică la ce mesaj răspunde obiectul de tipul ReplyMessage). (Sau puteți avea în clasa Message un atribut reply: Message, care să fie null pentru un mesaj nou creat).
Afișați (cronologic) conversațiile a doi utilizatori, citiți de la tastatură.
4. (1p) Simulați trimiterea unei invitații de prietenie și adăugați relația doar dacă utilizatorul invitat o acceptă. Salvați cererile de prietenie și actualizați statusul lor (pending, approved, rejected).

B. Cerințe non-funcționale: 4p

1. (1p) Pentru cerințele 1 și 2 folosiți operații pe stream-uri (filter, map, reduce)
2. (1p) *Persistența datelor: baza de date.*
3. (2p) Interfața cu utilizatorul, să permită simularea trimiterii mesajelor, a conversației și a invitației de prietenie.