

Metode Avansate de Programare – Laborator 8

Tema: DuckSocialNetwork

DEADLINE: Săptămâna 11

Cerințe funcționale

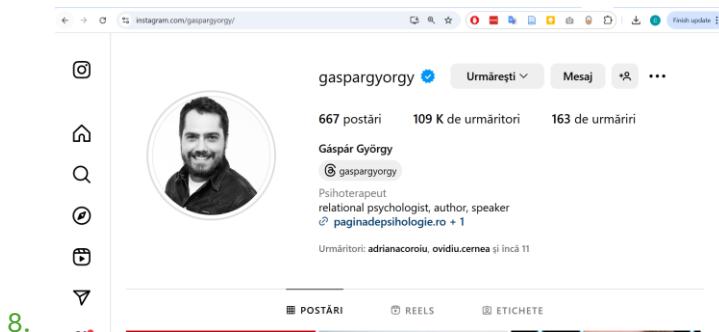
1. Adaugati clasa Message avand urmatoarele attribute: id, from: Utilizator, to:List<Utilizator>, message: String, data:LocalDateTime; Derivati clasa ReplyMessage (Collaboration) din clasa Message si adaugati un atribut message:Message (care specifica la ce mesaj raspunde obiectul de tipul ReplyMessage). (Sau puteti avea in clasa Message un atribut reply: Message, care sa fie null pentru un mesaj nou creat).
2. **Implementati trimiterea de mesaje printr-o interfata grafica (chat) astfel încât** sa permita simularea trimiterii mesajelor si vizualizarea conversatiei.
3. Autentificare utilizatori (Login). Important: parolele sa fie criptate.
4. Lab 9 - Simulati trimiterea unei *invitatii de prietenie* si adaugati relatia doar daca utilizatorul invitat o accepta. Salvati cererile de prietenie si actualizati statusul lor (*pending, approved, rejected*). **Deadline S12**
5. Lab 9 - Notificarea utilizatorului autentificat, la o noua cerere de prietenie.

Deadline S12

6. Lab 10 – Race Event (inregistrare la eveniment, notificari, derulare eveniment , trimitere mesaje - *se va adauga o cerinta de programare concurrenta*

Deadline S13

7. Definirea unui obiect de tip Page care sa defineasca un GUI care include pagina de profil a unui utilizator. Vezi imaginea urmatoare ca exemplu:



8.

Deadline S13

B. Cerințe non-funcționale:

1. **GUI: MVC**
2. Folositi operatii pe stream-uri (filter, map, reduce)

C. Persistența datelor: baza de date.