

Milestone 1 – Alegerea proiectului Platformă distribuită tip LeetCode **Nume:** Denis Covei
Grupa: 342C1

1. Arhitectura soluției Platformă este implementată ca un stack Docker Swarm format din următoarele servicii: Keycloak – serviciu SSO pentru autentificare cu OAuth2/OIDC; User Service – componentă proprie pentru gestionarea utilizatorilor și a rolurilor; Database Service (PostgreSQL) – persistă utilizatori, probleme, submisiile și joburi; Problem Management Service – componentă proprie pentru managementul problemelor; Code Runner Service – container sandbox cu GCC pentru compilarea și rularea soluțiilor; Queue Service (RabbitMQ/Redis Streams) – gestionează execuția asincronă a joburilor; API Gateway (Traefik/Nginx) – expune accesul extern și rutează cererile către microservicii. 2. Funcționalități Autentificare cu Keycloak și preluarea metadatelor utilizatorului; Crearea automată a profilului utilizatorului și atribuirea rolurilor (admin/user); Stocarea problemelor în fișiere text și expunerea lor prin Problem Service; Upload, listare și rezolvare probleme printr-un REST API; Trimitere soluției la Code Runner Service prin intermediul unei cozi de mesaje; Execuția codului în containere izolate cu GCC și limitări de timp/memorie; Returnarea rezultatului (success, compilation error, runtime error, timeout); Rate limiting distribuit pentru a proteja sistemul de abuz. 3. Module avansate **Modul avansat 1 – Execuție de cod sandboxed cu replicare workers**

Soluțiile utilizatorilor sunt procesate de mai multe instanțe ale Code Runner Service, asigurând concurență și scalabilitate.

Modul avansat 2 – Sistem de rate-limiting distribuit

Lmitează numărul de submisiile ale unui utilizator într-un interval, folosind un mecanism distribuit (Redis).

4. Rezumat Arhitectura propusă respectă cerințele proiectului: microservicii, SSO, DB, containere Docker, stack Swarm, două module proprii și două funcționalități avansate specifice sistemelor distribuite.