

Initial am plecat de la clasele Information, Education si Experience. Am incercat sa iau tema pe pasi logici si sa implementez fiecare clasa avand nevoie doar de clasele implemantate antecetend. In paralele am facut si design patterns pentru a nu mai avea dupa de modificat foarte mult.

Tema mi-a luat o perioada lunga de timp(m-am apucat la inceputul vacantei si am terminat-o la sfarsitul ei). Mi s-a parut o tema utila, la care am lucrat cu drag, insa faptul ca tema a fost modificata si remodificata de minim 5 ori m-a facut sa obosesc (as prefera ca tema sa nu mai fie modificata atat de des pe parcursul ei).

Functii mai greu de inteles:

1.getDegreeInFriendship

Pentru a stii la ce nivel ma aflu, atunci cand parcurgeam toti consumatori de pe nivelul curent ii adaugam intr-o coada si la final puneam si null pentru a stii cand se trece la nivelul urmator. Apoi scoteam din coada, faceam acelasi algoritm verificand daca consumerul a fost vizitat si continuam pana coada se golea.

2.getRecruiter

Am apelat functia getDegreeInFriendship si am bagat intr- o lista toti recruteri de pe nivelul cel ma departat. Am parcurs acesta lista si am luat recruterul cu ratingul cel mai mare.

3.process

Am parcurs lista de requesturi care era deja ordonata deoarece am facut-o Comparable si le selectam doar pe acelea care corespund jobului. Apoi verificam daca jobul este inca activ, iar daca mai este verificam si daca userul din request a fost sau nu angajat deja(daca nu il convertam in employee, ii setam salariu si il adaugam la departamentul potrivit-adica IT.

Daca numarul de candidati ajunge la zero, atunci insemna ca jobul s-a inchis si ca trebuie sa anuntam toti ceilalti useri ca au fost respinsi.