

Raport de dezvoltare

Andrei Mihai-Cosmin, Cazanov Veaceslav, Girnet Andrei, Mihalcenco David, Szocs Mihaela-Felici

- Planul de risc al dezvoltării proiectului:

În abordarea aleatoare pentru construcția arborilor, riscul principal este dat de imprevizibilitatea rezultatelor și variabilitatea complexității generate. Această metodă poate produce atât soluții eficiente ($O(1)$), cât și soluții cu complexitate ridicată ($O(n)$), în funcție de alegerile aleatoare ale algoritmului. Există posibilitatea ca, datorită caracterului aleatoriu, arborii generați să nu atingă nivelul de eficiență dorit sau să prezinte performanțe variabile. Astfel, riscul derivă din imposibilitatea de a controla în mod direct rezultatele și complexitatea soluțiilor generate de algoritmul bazat pe alegeri aleatoare.

- Progres al echipei în dezvoltarea soluției:

Echipa a avansat semnificativ în dezvoltarea soluției. Cei trei baieti de la software au făcut progrese notabile, parcurgând detaliile problemei, împărțind task-urile în sarcini mai mici și implementând algoritmul conform descrierii din raportul tehnic. S-a pus accent pe colaborare și coordonare prin înscrierea echipei în Jira, pentru a asigura o înțelegere clară a cerințelor și o implementare eficientă a algoritmului propus.

În paralel, QA Engineer-ul și-a început activitatea, concentrându-se pe elaborarea testelor manuale. Acest pas este esențial pentru a evalua și asigura corectitudinea funcționalității algoritmului și a soluției în ansamblu.

- Impactul metodologiei de dezvoltare folosite:

Metodologia de dezvoltare adoptată pare a fi flexibilă și potrivită, exemplificată prin implementarea unui algoritm Monte Carlo pentru construcția arborilor. Această abordare aleatoare aduce avantaje prin generarea de soluții în $O(1)$ sau dezavantaje prin generarea acestora în $O(n)$. Cu toate acestea, imprevizibilitatea rezultatelor poate constitui un risc, evidențiind necesitatea testelor manuale și automate pentru asigurarea calității - pastrarea fiabilității algoritmului.

- Monitorizare, evaluare și controlul evoluției proiectului:

Asa cum am menționat, proiectul este monitorizat pe Jira la adresa mpsproiect.atlassian.net unde sunt împartite taskurile și rezolvate. După rezolvarea fiecărui task de către developer, acesta este trecut în faza de testare, unde QA se ocupă de validarea tichetului. Monitorizarea de software este făcută în github, unde se pot vedea commit-urile fiecăruia. Evaluarea task-urilor se face în storypoint pe jira pentru a determina complexitatea. De asemenea pe github toate modificările sunt trecute prin Pull Request unde e necesar cel puțin un approve ca să poți merge în main. Astfel, commit-urile direct în main sunt interzise. Controlul evoluției proiectului poate

fi văzut pe jira, în prezent având un demo functional și 2 arbori care au fost cei mai buni. Pentru milestone 3 se va implementa salvarea tree-urilor în fișiere și compararea lor. De asemenea se vor rula mai multe teste.