

Ingineria Programării - Proiect

Se vor forma echipe de 3-5 persoane. Fiecare echipă trebuie să construiască o aplicație (la alegerea echipei) folosind metologia SCRUM.

Proiect

Tema proiectelor constă într-o aplicație mobile/desktop/web complexă care pe lângă funcționalitățile de bază precum login/baza de date/etc va conține concepte de AI (de exemplu: sisteme de recomandare, sisteme de OCR, voice synthesizes, traduceri texte etc.) În cazul utilizării unor servicii directe oferite de provideri (e.g. serviciul de traduceri texte de la Google), se dorește o implementare anexă a unui concept de AI ce folosește rezultatul obținut de la provider.

Precizări:

- Nota 5: aplicația funcționează cap-coadă dar feature-urile de AI sunt minimale, nu s-a implementat nimic propriu ci doar s-a folosit un serviciu online.
- Nota 8: aplicația funcționează cap-coadă dar feature-urile sunt doar de bază(o regresie lineară, un SVM(Support-Vector Machine) sau NN(Neuronal Network) pentru o problemă simplă.
- Nota 10: aplicații cu feature-uri complexe la care echipa aduce o contribuție serioasă în ceea ce privește modulul de AI.

Exemplu 1. O aplicație mobile pentru recomandare de restaurante care folosește serviciile de localizare de la Google și rating-urile userilor din aplicația proprie.

Exemplu 2. Joc de tenis virtual care folosește Computer Vision(OpenCV) pentru a detecta pozițiile mâinii.

Exemplu 3. O aplicație web de recomandare pentru useri (gen Netflix) folosind un database online de filme cu descriere succinte. Se vor lega feature-urile/notele date de useri.

Exemplu 4. Estimarea cu un multiperceptron/regresie lineara a prețurilor caselor de la o agenție imobiliară.

Exemplu 5. Joc Unity ce folosește Unity ML(spre exemplu pentru învățarea anumitor behavior-uri automat).

Exemplu 6. Aplicații mobile ce folosesc AR sau ARCore.

Exemplu 7. Aplicații de robotică ce au la bază concepte de ML/Computer Vision etc.

Fiecare echipă va avea un repo pe Github/Bitbucket/Gitlab unde vor construi proiectul. Codul va trebui să respecte codestyle-ul specific limbajului/limbajelor de programare ales/alese. Se recomandă documentarea codului, acolo unde este cazul.

Fiecare echipă va folosi un tool de management al taskurilor. Pentru fiecare task vor fi asignate 2 persoane: persoana care îl implementează și persoana care îl testează.

Profesorul de laborator va avea acces atât la repo cât și la board-ul în care se regăsește gestiunea task-urilor.

Roluri

Unul dintre membrii echipei își va asuma rolul de Technical Lead, ceilalți fiind dezvoltatori. În echipă nu vor exista testerii dedicați, generarea de teste fiind inclusă în procesul de implementare, iar testarea de acceptanță va fi făcută încrucișat (de către un alt dezvoltator decât cel care a realizat implementarea).

Planificare

1. În săptămâna 2 toate echipele vor prezenta în cadrul laboratorului o descriere completă a proiectului. Eventual se vor negocia feature-urile cu profesorul de laborator în cazul în care dificultatea proiectului ales este foarte scăzută. De asemenea toate echipele vor prezenta un plan de implementare/task-uri/plan de testare, descrise într-un tool de project management precum Trello/Jira.
2. În săptămâna 4 toate echipele vor prezenta în cadrul laboratorului diagramele specifice arhitecturii aplicației. (Diagrama de clase, Diagrama bazei de date, Diagrama de stări). Se vor prezenta și Mockup-urile aplicației. Se recomandă aplicarea practicilor de UI/UX specifice aplicației construite și utilizarea unui număr minim de 2 design patterns (acestea pot fi unele deja implementate de framework-ul în care se dezvoltă aplicația).
3. În săptămâna 8 va avea loc o prezentare intermediară a aplicației. În cadrul acesteia se va obține un punctaj cuprins între 0-2 în funcție de stadiul aplicației.
4. În sesiune va avea loc prezentarea finală.