

PROIECT la DISCIPLINA
INTRODUCERE IN BAZE DE DATE
Punctaj – 50/100 puncte
Termen predare: saptamana 14/semI

În cadrul acestui proiect, se va realiza o aplicație cu interfață grafică care va oferi utilizatorilor funcționalități ,corespunzător cu drepturile pe care aceștia le dețin în sistem, descrise prin intermediul unor roluri.

După implementarea proiectului, studentul va fi capabil să:

- proiecteze o bază de date relațională cu mai multe tabele normalizate, între care există relații de dependență, gestionând informațiile din cadrul acesteia prin intermediul unei aplicații;
- dezvolte un sistem de control al drepturilor de acces al utilizatorilor în cadrul sistemului informatic pe baza rolurilor (RBAC – role-based access control);
- realizeze o interfață grafică pentru a facilita interacțiunea utilizatorului cu aplicația, procesând evenimentele pe care acesta le generează.

Cunoștințele necesare sunt:

- programarea în limbaj OOP;
- manipularea bazelor de date folosind MySQL;
- folosirea JDBC - API pentru SQL Connection;

Termene

Săptămâna10 – stabilirea echipei de studenți, team leader și responsabilități, modelul UML și implementarea bazei de date, eventuale întrebări referitor la cerințele proiectului.

Săptămâna11 – schiță de meniu/interfață, funcționalități pentru un utilizator (administrator/profesor/student)

Săptămâna12 – logare în aplicație, implementare roluri

Săptămâna 14 – predare

Echipa de proiect

Echipa va fi formată din 2 (max3)studenți. Este acceptabil ca din echipă să facă parte studenți din grupe diferite, proiectul va fi prezentat de către team-lead, în cadrul grupei din care acesta face parte, dar este de dorit să se limiteze aceste situații.

Enunț

Se dorește implementarea unui sistem informatic destinat gestiunii unei platforme de studiu.

Aplicația va folosi un sistem de gestiune pentru baze de date MySQL, iar interacțiunea cu aceasta va fi realizată doar prin interfața grafică. Funcționalitățile pe care le va oferi programul vizează operații ce țin de gestiunea studenților, profesorilor și administrarea operațiilor curente din cadrul unor programe de studiu.

Aplicația va putea fi accesată, pe baza unui proces de autentificare, de către mai multe tipuri de utilizatori: *studenți, profesori, administratori*. Pentru fiecare tip de utilizator se vor reține informații precum CNP, nume, prenume, adresa, număr de telefon, email, cont IBAN, numărul de contract. Fiecare utilizator își va putea vizualiza datele personale imediat după ce va accesa sistemul informatic, fără a avea însă posibilitatea de a le modifica. Totodată, programul trebuie să ofere și o funcționalitate pentru deautentificare, prin care se revine la fereastra care solicită datele de acces, astfel încât și un alt utilizator să îl poată folosi ulterior, fără a fi necesară repornirea sa.

Utilizatorul de tip **administrator** poate adăuga, modifica și șterge informații în baza de date, informații legate de utilizatori. De asemenea, va exista și un rol de **super-administrator** care poate opera inclusiv asupra utilizatorilor de tip administrator.

Administratorii pot să caute utilizatorii după nume și îi pot filtra după tip, pot asigna profesorii la cursuri și pot face căutare după numele cursului. La căutarea unui curs se afișează și numele profesorilor de la acel curs și un buton care permite vizualizarea tuturor studenților înscriși la cursul respectiv.

Pentru un utilizator de tip **profesor** se vor reține și cursurile predate, numărul minim și numărul maxim de ore pe care le poate predă și departamentul din care face parte.

Pentru un utilizator de tip **student** se va reține și anul de studiu și numărul de ore pe care trebuie să le sustină.

Aplicația va permite gestiunea cu ușurință a activităților didactice și astfel a interacțiunilor dintre studenți și profesori. Cursurile sunt predate de mai mulți profesori și au una sau mai multe tipuri de activități (curs, seminar, laborator), o descriere, și un număr maxim de studenți participanți. Studenții se pot înscrie la cursuri și sunt asignați profesorului cu cei mai puțini studenți la data înscrierii. Aceștia sunt evaluați cu note pentru fiecare tip de activitate și primesc o notă finală ca medie ponderată între tipurile de activități. Profesorul stabilește din interfața grafică împărțirea procentuală pe tipurile de activități (ex. 20% seminar, 35% laborator, 45% curs/examenul de la curs).

Fiecare activitate se desfășoară recursiv între două date, pe o anumită perioadă de timp. La asignarea unui profesor la un curs se vor alege tipurile de activități. De exemplu, profesorul X predă cursul Y cu activitățile: curs – săptămânal, laborator – săptămânal. Ulterior, profesorul poate programa activitățile (curs, seminar, laborator, colocviu, examen) într-un calendar, pe zile și ore, specificând și numărul maxim de participanți. Activitățile pot fi programate doar în viitor. Profesorii pot accesa un catalog, unde pot filtra studenții după cursuri și le pot adăuga note. Cataloagele pot fi descărcate sub formă de fișier.

La logare, studenții și profesorii pot să își vada activitățile din ziua curentă sau pot accesa o pagină cu toate activitățile la care sunt asignați / înscriși. Aceste liste pot fi descărcate din sistem sub formă unor fișiere.

Studenții se pot înscrie la cursuri, pot renunța la cursuri și își pot vedea notele. Aceștia trebuie să aleagă activitățile la care vor să participe și pot participa la ele doar dacă mai sunt locuri sau nu există o suprapunere cu o altă activitate (de exemplu, studentul dorește să participe la laboratorul de BD marți la ora 10. Se înscrie la acea activitate, iar înscrierea este validă doar dacă nu are deja o altă activitate marți la ora 10 sau dacă mai sunt locuri disponibile. În caz contrar, se afișează un mesaj de eroare).

Totodată, studenții se pot înscrie în grupuri de studiu pentru o anumită materie, dacă sunt înscriși la materia respectivă. Aceștia pot să vada toți membrii grupului și să lase mesaje. Pe grup, studenții pot adăuga activități și să definească un număr minim de participanți și o perioadă în care ceilalți studenți pot să anunțe participarea (de exemplu, un student adăuga o activitate de aprofundare a cursului pentru data de 12.12.2020, ora 16:00, 2 ore, cu număr minim de participanți 5 și timp de expirare 2 ore). Dacă numărul minim nu este atins, activitatea se anulează, iar studenții înscriși la ea primesc un mesaj de informare.

Precizări suplimentare

Popularea bazei de date cu informații trebuie să fie corespunzătoare pentru a putea ilustra funcționalitățile implementate.

Orice specificație care nu este menționată mai sus reprezintă decizie de implementare. Puteți considera orice simplificare în condițiile în care enunțul nu precizează altfel.

Barem de corectare și notare

Punctaj (p)	Criterii de acordare
10 p	proiectarea tabelor în baza de date <ul style="list-style-type: none">conformitatea structurii conceptuale a bazei de date cu o formă normalădefinirea de chei primare, chei străine (legături între toate tabelor bazei de date), constrângeri de integritate pentru formatul atributelor: populare corespunzătoare
5 p	gestiunea utilizatorilor <ul style="list-style-type: none">autentificare, deautentificareoperații utilizator de tip administrator / super-administratorimplementare sistem de control al drepturilor de acces al utilizatorilor în cadrul sistemului informatic pe baza rolurilorcautare cursuricautare utilizatori
5 p	funcționalități pentru utilizatorul de tip student <ul style="list-style-type: none">căutare cursinscriere la cursvizualizare notevizualizare grupuri si membrimesaje pe grupvizualizare / descarcare activitati curente / din viitor
5 p	funcționalități pentru utilizatorul de tip profesor <ul style="list-style-type: none">adaugare si programare activitatigestionare ponderi notenotare studentivizualizare liste studenti si descarcare cataloagevizualizare / descarcare activitati curente / din viitor
5 p	funcționalități pentru gestiunea cursurilor si a grupurilor de studiu <ul style="list-style-type: none">operatii CRUD pe cursurioperatii CRUD pe grupurile de studiuinscriere in / parasire grup de studiuadaugare activitate grup (realizare constrangeri de timp si numar de persoane)
5 p	modularizare :structura aplicației, lizibilitatea codului,comentarii, README
15 p	Realizarea interfetei grafice

BONUS

Se pot obține punctaje suplimentare, astfel:

- 10 puncte – elaborarea unui algoritm de repartizare a studentilor la toate tipurile de activitati ale cursurilor la care sunt inscrisi, fara sa existe suprapuneri de orar si sugerarea unor ferestre potrivite pentru activitatile din grupurile de studiu;
- 5 puncte – afisarea unor sugestii de participanti la grupurile de studiu;
- 5 puncte – posibilitatea adaugarii unui cadru didactic la o activitate din grupul de studiu;