Catalog universitar

Context

În contextul provocărilor din ce în ce mai aspre ale educației sec XXI și impinși de nevoia continuă de a evolua, ne-am propus să digitalizam modalitatea de gestiune existentă pentru un catalog universitar pentru a reduce în primul rând timpul extrem de mare pe care l-am pierde folosind birocrația clasică(hârtii inutile, pierderea acestora, necesitatea deplasării într-un anumit loc special pentru a modifica/verifica notele și absențele studenților, a accesa orarul, în ceea ce privește profesorii, iar pentru studenți greutatea cu care se confruntă privind situația proprie). Ca urmare propunem crearea unei aplicați moderne de gestionarea datelor care sa permită accesul la multiple facilitați.

Cerințe funcționale:

* Centralizarea catalogului

Descriere: catalogul propus va aduna toate datele necesare pentru acumularea tuturor entităților ce țin de o universitate(studenți, profesori, grupe, săli de clasa, facultăți, specializări, etc.).

* Aplicația să fie de tip client-server

Descriere: aplicația în sine va avea la bază 3 componente:

1. Clientul: aplicație bază pe o interfață grafică, care permite gestionarea rapidă și eficientă a entitaților catalogului în funcție de ce utilizator e logat. Acesta trimite cereri la server în scopul primirii unui răspuns la întrebările puse(query SQL).
2. Baza de date: stochează toate informațiile aferente entităților catalogului și trimite la cerere, dacă există, informații referitoare la acestea. De asemenea ca orice bază de date implementează operațiile de bază read, update, delete, create.
3. Server: face legătura între client și baza de date, primind cererea de la user și trimite un răspuns de la baza de date înapoi acestuia.

* Existența conturilor de utilizator

Descriere: conturile vor fi de 3 tipuri: student, profesor, admin

1. Student: are cel mai mic nivel de permisiuni, poate să își vadă propriile materii, note, absențe, restanțe, media, orarul(cu sălile de clasa), grupa, posibilitatea de a primi notificări la eventuale modificări.
2. Profesor: are un nivel mediu de permisiuni, poate să își vadă grupele de studenți, orarul(cu sălile de clasa), poate sa modifice notele unor studenți.
3. Admin: are un nivel ridicat de permisiuni, poate să modifice studenții și profesorii cât și conturile acestora, să adauge studenți/profesori, să publice anunțuri.

* Grupa de studii

Descriere: va avea o denumire, va reține studenții, profesorii cunrenți, eliberează o fișa de note. Vor fi împărțite pe ani de studii, facultate, specializare.

* Fișa de note

Descriere: conține notele unei grupe, poate returna media generală, numărul de restanțieri pe grupă, restanțele(studentul care are restanța, materia).

* Anul de studiu

Descriere: compus din toate grupele specializării, toți profesorii de la toate grupele, precum și materiile.

* Materia

Descriere: asociată unui profesor, unui student, an de studiu, unei facultați.

* Funcție

Descriere: funcții didactice(asistent universitar, lector universitar, conferențiar universitar, profesor universitar, doctor) și de conducere(șef laborator, șef departament, secretar, decan).

* Specializarea

Descriere: conține toți anii de studiu(folosită cel mai des ca un criteriu de sortare al studenților).

* Facultatea

Descriere: conține specializările(de asemenea folosită ca un criteriu de sortare al studenților dar și al profesorilor).

Cerințe nefuncționale:

* Detaliere server

Descriere: server creat în limbaj de programare C++, acesta gestionează socket-urile(mecanismele prin care asigurăm conexiunea dintre client și server prin internet, utilizând o adresa IP, un port și un protocol), execută operațiile de binding, listen, accept, receive și send, aflându-se în stransă legatură cu baza de date.

* Baza de date

Descriere:

1. Pentru student: procesează cerea acestuia de vizualizare(a notelor, colegilor, etc.) operație de tip SELECT
2. Pentru profesor: procesează cererea acestuia de vizualizare(a studenților, a notelor, etc.) operație de tip SELECT, cererea de modificare situație(note, absente) operație de tip INSERT/UPDATE
3. Pentru admin: procesează cererea de editare/ștergere/adăugare/vizualizare(a grupelor, a profesorilor, a fișelor de note, a orarului etc.).