**Rezumatul proiectului de diplomă al studentului:** LUŢĂ I. Albert-Lucian, grupa: 444A

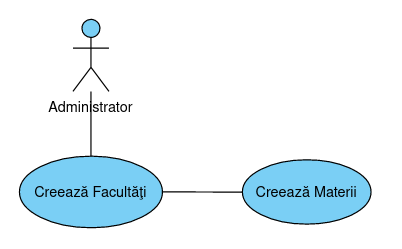
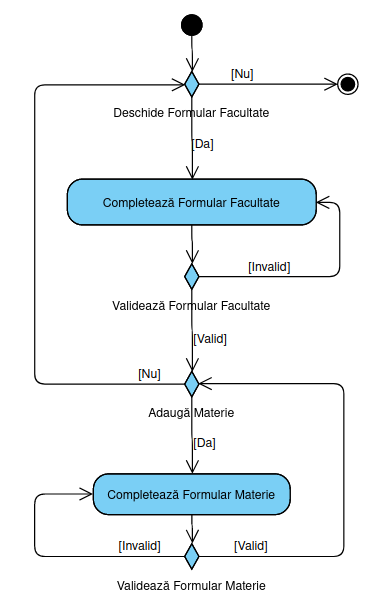
**Programul de studiu:** Ingineria Informaţiei, 2021

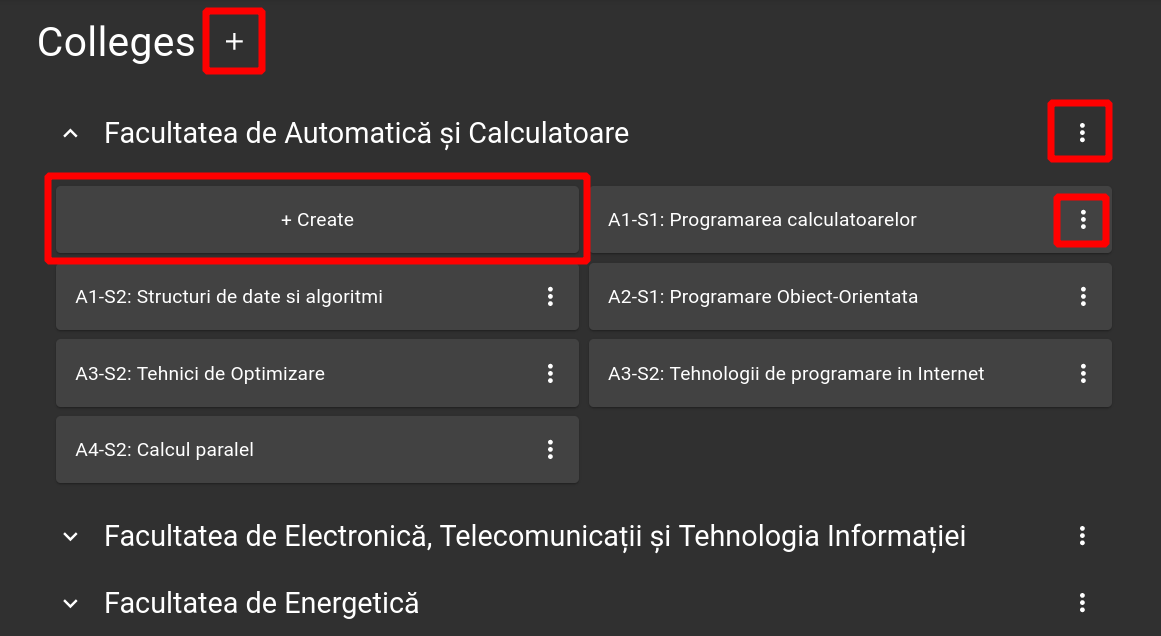
Includerea instrumentelor de predare-testare într-o aplicație e-learning.

**Conducător științific:** Ş.L. Dr. Ing. Elena Cristina STOICA

**Obiectivele proiectului:** S-a proiectat și dezvoltat o aplicaţie web de tip e-learning, care conține utilizatori de 3 tipuri: student, profesor și universitate. Fiecare rol enumerat are acces la diferite funcţionalităţi. Utilizatorii cu rol de administrator vor putea crea structura universităţii, facultăţi și materii, și manageria utilizatorii, rolul acestora și materiile la care sunt înrolaţi. Utilizatorii cu rol de profesor vor putea structura materia în secţiuni(ex. curs, seminar, laborator), crea/edita activităţi, nota studenți. Activităţile pot fi de 4 tipuri: materiale, temă, test și forum, fiecare dintre acestea având anumite particularităţi. Utilizatorii cu rol de student pot participa la activităţile create de către profesori, unele dintre acestea fiind notate. Orice tip de utilizator va putea participa la discuţiile de pe forum, dar doar administratorii și profesorii pot crea noi discuții.

**Realizarea proiectului și rezultatele obținute:** Înainte de a trece la implementarea propriu-zisă a proiectului, trebuie să se înţeleagă foarte bine cerinţele și metodele prin care acesta va fi construit. Aceasta treaptă preliminară implementării se numește proiectare. Pentru fiecare dintre funcţionalităţile prezentate în cadrul obiectivelor de mai sus, s-au creat 2 tipuri de diagrame, caz de utilizare și de activitate. Acestea ne vor descrie exact care sunt aşteptările când vine vorba de funcţionalităţile la implementare.

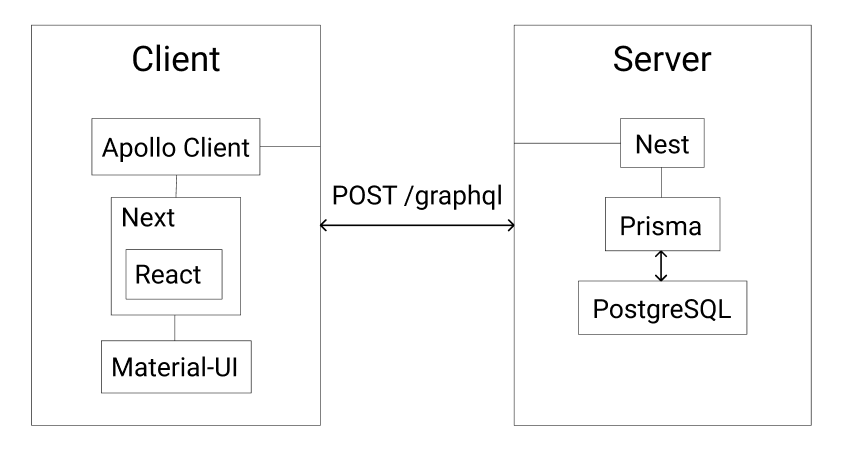
Fig. 1: Exemplu Proiectare - Diagramă Caz de Utilizare(stânga) și Activitate(dreapta) – Creează Structura Universităţii

Fig. 2: Exemplu Implementare – Tablou de bord Universitate - Facultăţi și Materii

Fiind o aplicație web, am folosit arhitectura standard de construire a acestora „client-server”. Aceasta împarte aplicaţia în două părţi, fiecare având un rol separat, comunicând între ele prin conexiuni TCP. Partea de client se referă la interfaţa vizuala cu care interacţioneaza utilizatorul, iar partea de server se referă la punctul în care se răspunde cererilor primite de la client.  
Limbajul de programare folosit atât pentru client, cât și pentru server este TypeScript, iar tehnologiile folosite sunt următoarele:

Client: React, Next, Material-ui și Apollo Client

Server: Node, Nest, PostgreSQL și Prisma

Fig. 3: Arhitectura aplicaţiei