**Observatie: Acest fisier va fi modificat periodic. Va rog sa-l consultati frecvent!**

**Proiect Laborator MAP 2019-2020**

**Tema proiect: Evaluarea temelor de laborator**

Contents

[Enunțul pe scurt 1](#_Toc23752920)

[Gestiunea resurselor: student, tema, nota, profesor, structuraAnUniv, feedback ... 2](#_Toc23752921)

[Atributele necesare fiecărei resurse ar putea fi: 2](#_Toc23752922)

[Principalele funcționalități 2](#_Toc23752923)

[- F1, F2 (Iterația 1): 2](#_Toc23752924)

[- F3 (Iterația 4): 3](#_Toc23752925)

[- F4 (Iterația 2): 3](#_Toc23752926)

[- F5 (Iterația 2): 3](#_Toc23752927)

[- F6 - (Iterația 4): 4](#_Toc23752928)

[- F7 - (Iterația 3 – GUI): 4](#_Toc23752929)

[- F8 - (Iterația 3): 4](#_Toc23752930)

[- F9 (Iterația 4): 4](#_Toc23752931)

# Enunțul pe scurt

Dezvoltați o aplicație care ajute cadrele didactice să monitorizeze evaluarea temelor de laborator la disciplina MAP. Aplicația va permite înregistrarea unui feedback oferit *studentului* la adaugarea unei *note* la o anumita *tema* de laborator si notificarea acestuia prin email. De asemenea, aplicația va oferi studentului posibilitatea înregistrării unui feedback vis-a-vis de claritatea cerintelor formulate, dificultatea temei precum si a experienței de învățare avută la această disciplină. Filtrări si rapoarte cu privire la predarea si evaluarea temelor sunt de asemenea oferite utilizatorului.

Se va folosi *Metodologia de dezvoltare incrementală bazată pe funcționalități* feature-driven development process (an 1, FP+POO).

# Gestiunea resurselor: student, tema, nota, profesor, structuraAnUniv, feedback ...

Ideal ar fi ca informațiile legate de studenți sa poată sa fie importate dintr-un alt sistem. Nu ne permite timpul sa cream ecrane de introdus date. Astfel, inițial, vom crea CSV-ri cu aceste informații (Studenti, Profesori, Teme, StructuraAnUniv ...) si le vom importa in sistemul nostru.

Pentru fiecare resursa trebuie sa avem posibilitatea sa vizualizam lista, sa adăugam, sa editam si sa ștergem resursa (Funcțiile CRUD – Create, Read, Update, Delete).

## Atributele necesare fiecărei resurse ar putea fi:

Student: ID, nume, prenume, grupa, email, cadruDidacticIndrumatorLab (String)

Profesor: ID, nume, prenume, email – deocamdată nu se cere aceasta entitate, se va lucra cu numele profesorului (String)

Tema: ID, descriere, startWeek (Integer-nr saptamanii din cursul semetrului), deadlineWeek (Integer - numarul saptamanii din cursul sem reprezentand termenul de predare)

Nota: ID, data, profesor, valoare, unde ID=ID\_Student+ID\_Tema (un student la o anumita tema are o singura nota)

StructuraAnUniversitar: ID, anulUniversitar, sem1: StructuraSemestru, sem2: StructuraSemestru:

Observatie: Puteti veni cu propria proiectare privind structura anului universitar

Definiți o reprezentare pentru resursa StructuraSemestru care sa definească un semestru ca o reuniune de intervale calendaristice ce definesc cele 14 săptămâni. Apoi definiți StructuraAnUniv care cuprinde structura celor doua semestre. Aplicația ar trebui sa funcționeze si in sem 2, pentru o alta disciplina.

# Principalele funcționalități

Aplicația va oferi profesorului următoarele funcționalități:

* F1, F2 (Iterația 1): Implementarea operațiilor CRUD pentru entitățile Student (F1) si Tema (F2).

Constrangeri impuse pentru entitatea Tema (ID, descriere, startWeek, deadlineWeek):

* + startWeek si deadlineWeek sa aiba valori intregi de la 1 la 14, startWeek < deadlineWeek ;
  + startWeek – reprezinta saptamana cand a fost adaugata tema (se determina automat (nu se introduce de utilizator) avand data curenta si structura anului universitar;
  + la modificarea termenului de predare pt o temă existentă, deadlineWeek, nr săptămânii curente trebuie sa fie mai mic sau egal decât nr săptămânii cu termenul de predare.
* F3 (Iterația 4): *la adăugarea unei noi teme de laborator, precum si la modificare termenului de predare al unei teme, toți studenții vor fi notificați printr-un mail. Aplicația va oferi posibilitatea dezabonării de la aceste notificări.*
* F4 (Iterația 2): *Adăugarea unei note* pentru un anumit student la o temă de laborator va avea in vedere urmatoarele constrangeri:
  + Profesorul acordă o notă de la 1 la 10 pentru fiecare temă de laborator.
  + Predarea unei teme după săptămâna în care este stabilit termenul de predare al temei respective, conduce la diminuarea notei cu 1 punct pentru fiecare săptămână de întârziere.
  + Eventualele depunctări datorate întârzierilor în predarea unei teme se vor calcula automat, afișându-se nota maximă pe care o poate lua studentul pentru tema respectivă.
  + Un student, la o tema de laborator, are o singură notă.
  + După 2 săptămâni de întârziere în predarea unei teme, tema respectivă nu va mai putea fi predată, decât daca este o situație excepțională: studentul a lipsit motivat.
  + Implemetati si cazul când profesorul nu a introdus notele la timp.
  + Găsiți voi si alte scenarii posibile sau situatii de exceptie pentru adaugarea unei note.
* F5 (Iterația 2): la adăugarea unei note se vor adăuga în fișierul *NumeStudent (format XML sau JSON sau text)* următoarele informații:

1. o ”Tema:” numărul temei
2. o ”Nota:” nota acordată pe tema respectivă
3. o ”Predată în săptămâna:” nr Săptămână în care tema a fost Predată
4. o ”Deadline:” nr Săptămână cu termen limită de predare, Deadline
5. o ”Feedback:” aprecieri, sugestii, precizări în legătură cu depunctările efectuate.
6. Exemplu:

|  |
| --- |
| 1. Tema: 3 2. Tema: 3   Nota: 9.5   1. Predată în săptămâna: 5 2. Deadline: 5 3. Feedback: 4. - Vezi conceptul de Interfata - in proiectul tau, obiectele de tipul repository nu sunt declarate de tipul interfetei. 5. Tema: 4 6. Nota: 8 7. Predată în săptămâna: 5 8. Deadline: 5 9. Feedback: 10. - Vezi conceptul de Interfata - in proiectul tau, obiectele de tipul repository nu sunt declarate de tipul interfetei 11. - Cod duplicat la Repository – sugestie: definirea unei clase AbstractFileRepository |

* F6 - (Iterația 4): *Fișierul* NumeStudent *(sau conținutul lui) va fi trimis printr-un mail studentului respectiv, săptămânal, mail ce va avea subiectul “Feedback laborator MAP”.*
* F7 - (Iterația 3 – GUI): Se va completa aceasta cerinta.
* F8 - (Iterația 3): Filtrarea entităților în baza unor criterii. Criteriile se vor stabili ulterior.
  + 1: Toți studenții unei grupe
  + 2: Toți studenții care au predat o anumita tema
  + 3: Toți studenții care au predat o anumita tema unui profesor anume
  + 4: Toate notele la o anumita tema, dintr-o saptamana data
* F9 (Iterația 4): Rapoarte
  + 1: Nota la laborator pentru fiecare student (media ponderată a notelor de la temele de laborator; pondere tema=nr de săptămâni alocate temei).
  + 2: Cea mai grea tema: media notelor la tema respectivă este cea mai mică.
  + 3: Studenții care pot intra în examen (media mai mare sau egală cu 4).
  + 4: Studenții care au predat la timp toate temele.