TITLU PROIECT : **Programarea sustinerii licentei si disertatiei**

Realizatori: Studenții grupei B7 – Modul Back-End

Leonte Codrin, Florescu Razvan, Dima Rares, Laur Ciucanu, Andreia Fedeles,Cristea Cdristian,Dimitrie Popescu

**FIȘA CERINȚELOR - BACK-END**

**Cuprins**

1. Descrierea problemei
2. Domenii
3. Actori si Obiective
4. Scenarii de utilizare

**1. Descriere**

Proiectul consta intro aplicatie web care are scopul de a realiza un management al notelor si repartizarii studentilor pe comisii de examinare a lucrarilor de licenta si disertatie in cadrul facultatii.

**2.Domenii**

Vom descrie scenarii de utilizare corespunzatoare folosirii aplicatiei , pentru activitatile suportate.

**3.Actori si obiective**

**Student:**

- se poate loga/deloga;

- poate vizualiza nota sa finala cat si a celorlalti studenti;

**Profesor:**

- se poate loga/deloga

- isi poate alege din ce comisie sa faca parte, atat timp cat toti studentii lui pot fi evaluati de acea comisie( o comisie avand un numar maxim cat si unul minim de evaluari ).

- trece nota, pe care pus-o, intr-o tabela;

- poate vizualiza toate notele studentului evaluat de el; ( studentul fiind evaluat de 3/4 profesori (licenta/master));

-poate vizualiza studentii pe care ii indruma;

-poate adauga sau sterge studenti pe care ii indruma;

**Secretar:**

- se poate loga/deloga;

- poate vizualiza notele finale ale studentilor;

- alege comisia pentru un profesor daca acesta nu si-a ales deja;

- gestioneaza notele puse de profesori;

- propune aranjarea studentilor pe comisii

- modifica lista studentilor examinati de comisia lor.

- hotaraste zilele de examinare, intervalul orar si pauzele

- decide asupra ordinii studentilor la sustinerea examinarii - genereaza rapoarte cu : -> programarea studentilor

-> lista studentilor cu medii pentru o zi de examinare

->lista totala cu studenti, medii si profesori examinatori

**Administrator:**

- se poate loga/deloga;

- are acces la toate informatiile pe care le ofera aplicatia;

- se ocupa cu intretinerea aplicatiei ;

- creeaza o sesiune de examinare;

- incheie sesiunile de inscriere si de examinare;

**Controller:**

- legatura dintre interfata grafica si Baza de Date

- preia comenzile venite din partea de View, acceseaza Baza de Date, face calcule si ofera un raspuns interfetei .

**Baze de date:**

- stocheaza date despre utilizatori, note etc. si realizeaza calcule pentru a oferi raspunsul asteptat de Controller.

**4.Scenarii de uilizare :**

**4.1. Logarea utilizatorului in sistem**

**4.1.1. Obiectiv/Context**

Utilizator este deconectat si doreste sa se logheze in sistem.

**4.1.2 Scenariu/pasi**

Controllerul primeste datele utilizatorului .

Se face o cere la baza de date, unde se face verificarea existentei contului .

Se construiește răspunsul pentru client. (logare sau invalidarea existentei contului corespunzator datelor introduse).

**4.1.3 Extensii**

Daca datele primite de la client sunt invalide, se va trimite un mesaj de eroare.

**4.2. Autentificarea in sistem**

**4.2.1. Obiectiv/Context**

Utilizator este deconectat si doreste sa se autentifice in sistem.

**4.2.2 Scenariu/pasi**

Controllerul primeste datele utilizatorului .

Serverul parsează datele primite de la client.

Se face o cere la baza de date, unde se face verificarea existentei contului .

Se construiește răspunsul pentru client. (autentifiicare reusita sau invalidarea contului corespunzator datelor introduse – contul exista deja in baza de date ).

**4.2.3 Extensii**

Daca datele primite de la client sunt invalide, se va trimite un mesaj de eroare.

**4.3.Administratorul creeaza o sesiune de examinare a licentei/dezertatiei**

**4.3.1. Obiectiv/Context**

Administratirul este logat.

El doreste doreste sa creeaze o noua sesiune de examinarei a lucrarilor de licenta /dizertatie .

**4.3.2 Scenariu/pasi**

Cotrollerul primeste datele de creeare a sesiunii ( numar comisii + numar membrii comisie + interval de inscriere ) si trimite cererea catre server.

Serverul parsează datele primite de la client.

Se face o cere la baza de date, unde se face editarea tabelei corespunzatoare sesiunilor de examinare.

**4.3.3 Extensii**

Daca datele primite de la client sunt invalide, se va trimite un mesaj de eroare.

**4.4.Administratorul editeaza perioada unei examinari**

**4.4.1. Obiectiv/Context**

Administratirul este logat.

El doreste doreste sa editeze perioada in care o comisiile vor examina studentii.

**4.4.2 Scenariu/pasi**

Cotrollerul primeste datele de editare ( ziua in care incep examinarile si ziua in care se sfarseste perioada aferenta) si trimite cererea catre server.

Serverul parsează datele primite de la client.

Se face o cere la baza de date, unde se face editarea tabelei corespunzatoare peroaderi de examinari.

**4.4.3 Extensii**

Daca datele primite de la client sunt invalide, se va trimite un mesaj de eroare.

**4.5.Administratorul editeaza data unei examinari**

**4.5.1. Obiectiv/Context**

Administratirul este logat.

El doreste doreste sa editeze data la care o comisie va examina studentii.

**4.5.2 Scenariu/pasi**

Cotrollerul primeste datele de editare si trimite cererea catre server.

Serverul parsează datele primite de la client.

Se face o cere la baza de date, unde se face editarea tabelei corespunzatoare examinarilor.

**4.5.3 Extensii**

Daca datele primite de la client sunt invalide, se va trimite un mesaj de eroare.

**4.6 Secrtarul gestioneaza notele puse de catre profesori**

**4.6.1 Obiectiv/Context**

Secretarul introduce notele finale puse de catre profesor in baza de date cu ajutorul clientului.

**4.6.2 Scenariu/Pasi**

1. Secretarul analizeaza notele (le verifica sa vada daca sunt bune)

2. Daca notele sunt in regula secretarul daca notele sunt in regula secretarul , controllerul va gestiona comada de validare a notelor ( cerere care care va introduce notele in baza de date).

**4.6.3 Extensii/Observatii**

Secretarul nu are inca ce sa gestioneze astfel incat notele nu au fost in ca trecute.

**4.7 Secretarul propune aranjarea pe comisii a studentilor**

**4.7.1 Obiectiv/Context**

Secretarul doreste sa propuna o repartizare preliminara a studentilor pe comisii .

**4.7.2 Scenariu/Pasi**

Controllerul primeste comanda de repartizare , face calcule si ofera ca raspuns o repartizare cat mai echilibrata a studentilor pe comisii.

**4.7.3 Extensii/Observatii**

Dezechilibrarea numarului de studenti repartizati din cauza inexistentei unei balante a numarului de studenti indrumati de catre profesorii din comisii.

**4.8 Secretarul editeaza zilele de examinare, intervalul orar si pauzele**

**4.8.1 Obiectiv/Context**

Secretarul doreste sa editeaze zilele, intervalul orar si pauzele din cadrun unei sesiuni de examinare.

**4.8.2 Scenariu/Pasi**

Controllerul primeste datele de editare si trimite cererea catre server..

Serverul parsează datele primite de la client.

Se face o cere la baza de date, unde se face editarea tabelei corespunzatoare examinarilor

**4.8.3 Extensii/Observatii**

Daca datele primite de la client sunt invalide(zilele nu corespund perioadei de examinare, intervalul orar se suprapune cu cel al unei alte comisii ,etc. se va trimite un mesaj de eroare.)

**4.9 Secretarul genereaza rapoarte**

**4.9.1 Obiectiv/Context**

Secretarul doreste sa genereze rapoarte cu :

-> programarea studentilor

-> lista studentilor cu medii pentru o zi de examinare

->lista totala cu studenti, medii si profesori examinatori

**4.9.2 Scenariu/Pasi**

Controllerul primeste cererea de gererare a rapoartelor si face la randul sau o cererea de prelevare a datelor necesare catre server.

Serverul cererea primite de la client.

Se face o cere la baza de date, de unde extrage datele necesare din tabelele corespunzatoare studentilor,notelor,etc si ofera un raspuns clientului.

Controllerul primeste datele si genereaza rapoartele necesare.

**4.9.3 Extensii/Observatii**

Se doreste generarea rapoartelor finale inainte de incheierea sesiunii de examinare ,etc.

**4.10 Studentul doreste sa vizualizele nota sa si/sau a alot studenti**

**4.10.1 Obiectiv/Context**

Studentul doreste sa vada notele finale.

**4.10.2 Scenariu/Pasi**

Controllerul primeste cererea de vizualizare a notei finale.

Ofera ca raspuns un raportul generat de catre secretar cu notele finale ale studentilor

**4.10.3 Extensii/Observatii**

Se doreste sa vada nota finala inaintea incheierii sesiunii de examinare.

**4.11 Profesorul alege comisia**

**4.11.1 Obiectiv/Context**

Profesorul doreste sa isi aleaga comisia din care face parte .

**4.11.2 Scenariu/Pasi**

1. Controller-ul va primii de la view informatii legate de component unei comisii (doar cea care a fost modificata)

2. Acesta va trimite pe rand informatiile preluate catre baza de date, unde se va face update la tabela cu comisia respectiva.

**4.11.3 Extensii/Observatii**

Se depaseste numarul maxim de studenti care pot fi evaluati de catre o comisie. Controller-ul va trimite un mesaj de esec catre view.

**4.12 Profesorul alege component studentilor pe care ii indruma**

**4.12.1 Obiectiv/Context**

Profesorul poate sterge sau adauga un student ca fiind sub indrumarea lui.

**4.12.2 Scenariu/Pasi**

1.Controller-ul primeste de la view niste date (nume, prenume, grupa din care face parte studentul).

2.Acesta va trimite catre baza de date informatiile primite, unde se va face update la tabela studentilor care sunt indrumati de acel profesor .

**4.12.3 Extensii/Observatii**

Profesorul a adaugat prea multi studenti ca fiind sub indrumarea lui. View-ul va primii un mesaj specific de eroare de la Controller.

**4.13 Profesorul vizualizeaza notele studentilor sai**

**4.12.1 Obiectiv/Context**

Profesorul poate sa vada toate notele pe care studentii lor le-au luat la evaluare.

**4.12.2 Scenariu/Pasi**

1.Controller-ul primeste de la view o cerere de a prelua date legate de notele pe care un student le-a obtinut la evaluare.

2.Acesta face la randul sau o cerere la baza de date pentru a obtine informatiile. Le obtine si le trimite catre view.

**4.12.3 Extensii/Observatii**

Studentul nu exista in baza de date sau nu este sub indrumarea acelui profesor. Controller-ul va trimite catre view un mesaj specific de eroare.