

eSTUD

Documentul de proiectare arhitecturala

Echipa BrigadaSenity:

Buzdugan Mihaela
Radu Sabina-Ioana
Gheorghe Adrian-Valentin

Coordonator:

Lambru Andrei Cristian

Contents

1. Introducere	3
1.1 Scopul sistemului	3
1.2 Definitii, acronime	3
2. Obiective de proiectare	4
3. Arhitectura propusa	5
3.1 Prezentare generala a arhitecturii sistemului.....	5
3.2 Decompozitia in subsisteme si responsabilitatile fiecarui subsistem	6
3.3 Distributia subsistemelor pe platforme hard/soft	7
3.4 Managementul datelor persistente	8
3.5 Controlul accesului utilizatorilor la sistem	8
3.6 Fluxul global al controlului	9
3.7 Conditiiile limita	10

1. Introducere

1.1 Scopul sistemului

Aplicația eSTUD poate fi folosită pentru centralizarea situației studenților dintr-o universitate. Astfel, folosind aplicația noastră, oricine deține un cont de student (care se află în baza de date a aplicației) își poate consulta foarte ușor situația școlară.

Studenții vor avea date personale, serie, grupă și catalog cu note. Vor exista și roluri de secretară, care poate adăuga note, studenți, serii sau grupe, cât și cel de administrator, care creează serii, grupe sau introduce secretarele în baza de date. Proiectul va avea ca și componentă principală o aplicație web care permite atât vizualizarea notelor de către studenți, cât și gestionarea situației acestora de către utilizarii privilegiați.

1.2 Definitii, acronime

- ❖ **Frontend** = se referă la fiecare parte a programului care interacționează în mod direct cu utilizatorul.
- ❖ **Backend** = componenta funcțională a unui program ce nu poate fi accesată de utilizatori.
- ❖ **DB (Baza de date)** = colecție organizată de informații.
- ❖ **SQL (Structured Query Language)** = limbaj de gestiune al bazelor de date.
- ❖ **MySQL Workbench** = instrument de proiectare al bazelor de date.
- ❖ **PyCharm** = mediu de dezvoltare integrat specializat pentru limbajul Python.
- ❖ **CSS (Cascading Style Sheet)** = standard pentru formatarea elementelor unui document HTML.
- ❖ **HTML (HyperText Markup Language)** = limbaj de marcare utilizat la crearea paginilor web.
- ❖ **JavaScript** = limbaj de programare orientat obiect bazat pe conceptul prototipurilor, folosit pentru logica sistemului și pentru interacțiunea elementelor din pagina web.

2. Obiective de proiectare

Calitatea produsului software va fi asigurata de cerintele bine specificate, aplicatia fiind intuitiva si usor de folosit, precum si facil de mentinut. Programul va fi capabil sa satisfaca principalele cerinte functionale: autentificare, gestionarea informatiilor studentilor de catre secretare si administrator si accesarea datelor personale de catre student.

1. Indeplinirea cerintelor de performanta: Serverul va fi capabil sa gestioneze numarul de utilizatori estimat si a raspunde intr-un timp destul de rapid cererilor acestora.

2. Indeplinirea cerintelor de calitate: Asigurarea aspectului intregii aplicatii. Aplicatia va avea un frontend placut vizual si intuitiv de folosit.

3. Indeplinirea cerintelor de incredere:

3.1 Fiabilitatea – atata timp cat conexiunea la internet este asigurata, programul va garanta o buna functionare.

3.2 Disponibilitatea – aplicatia va fi disponibila in permanenta

3.3 Toleranta la defecte – erorile cu care s-ar putea confrunta aplicatia sunt cele cauzate de retea. Nu sunt prevazute erori cauzate de software-ul aplicatiei.

4. Indeplinirea cerintelor de mentenanta: Aplicatia va putea fi folosita doar pe web. Toti cei care se vor ocupa de implementarea unor noi functii vor trebui sa rezolve si anumite probleme ce vor aparea pe parcurs dupa punerea in functiune a aplicatiei.

5. Indeplinirea cerintelor de planificare a proiectului: Termenele preconizate ale realizarii proiectului vor fi respectate, iar produsul livrat va respecta normele de calitate dorite.

3. Arhitectura propusa

3.1 Prezentare generala a arhitecturii sistemului

Avem ca scop separarea interfetei cu utilizatorul de datele si logica aplicatiei. Am ales sa folosim stilul arhitectural ierarhic inchis, pe 3 niveluri. Utilizatorii aplicatiei pot realiza urmatoarele actiuni, dupa cum urmeaza:

❖ **Utilizatorii neprivilegiati (conturile de studenti):**

1. pot sa isi acceseze datele personale.
2. pot sa isi acceseze notele de la fiecare materie

❖ **Utilizatorii privilegiati (administratorul si secretarele):**

1. pot sa vizualizeze, adauge sau sa stearga listele de serii
2. pot sa vizualizeze, adauge sau sa stearga listele de grupe
3. pot vizualiza informatiile unui student existent sau sa adauge unul nou
4. pot sa stearga un student sau sa ii modifice informatiile
5. pot sa editeze notele fiecarui student

❖ **O aritectura ierarhica inchisa pe 3 niveluri este alcatuita din 3 tipuri de subsisteme:**

1. **Subsistemul interfata utilizator:**

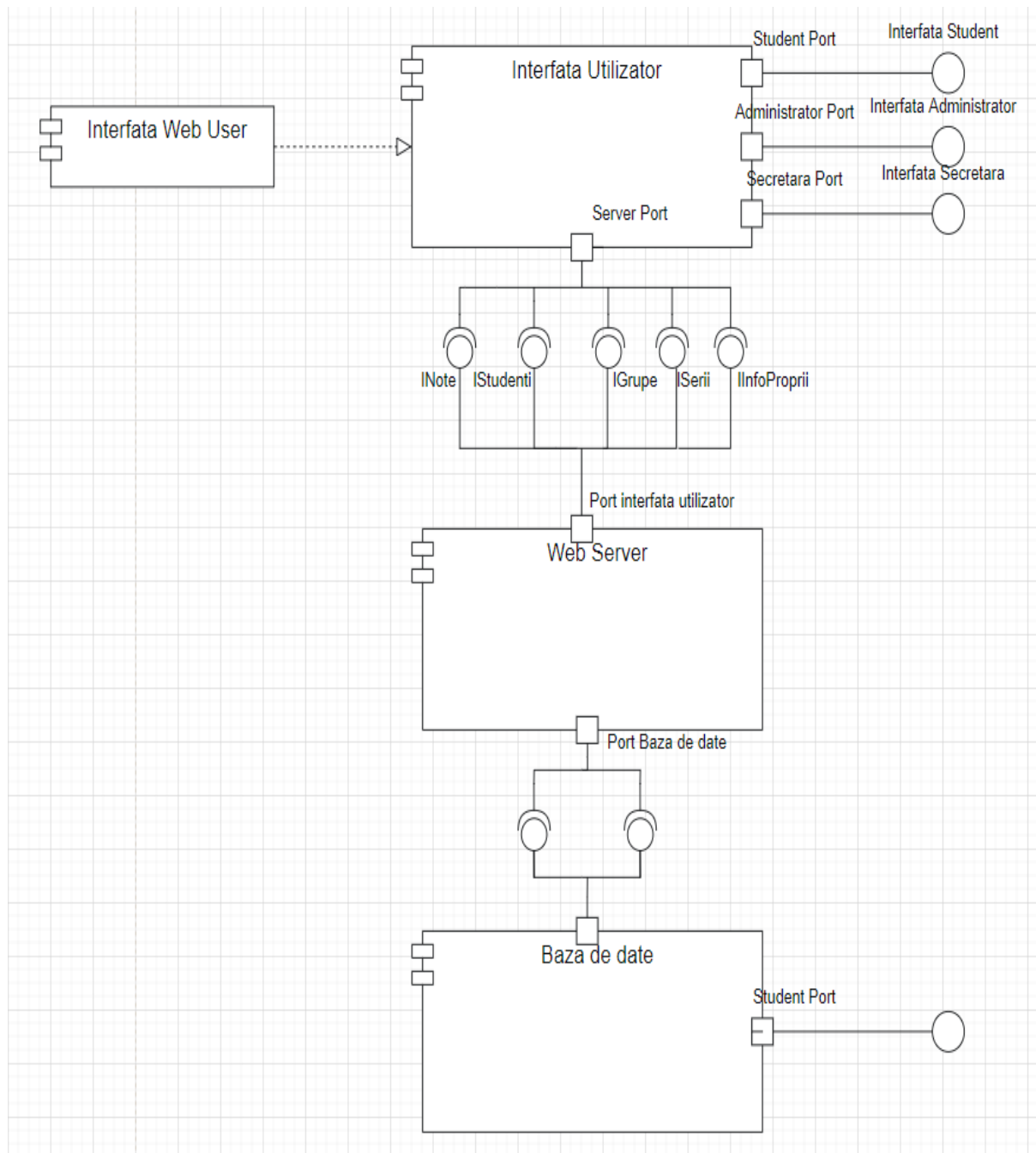
- a. Ecran prin care utilizatorul interactioneaza cu sistemul
- b. Aduce datele ce vor si afisate prin interactiunea cu serverul web
- c. Trimite cereri de creare si modificare a datelor catre server-ul web

2. **Subsistemul Web Server:**

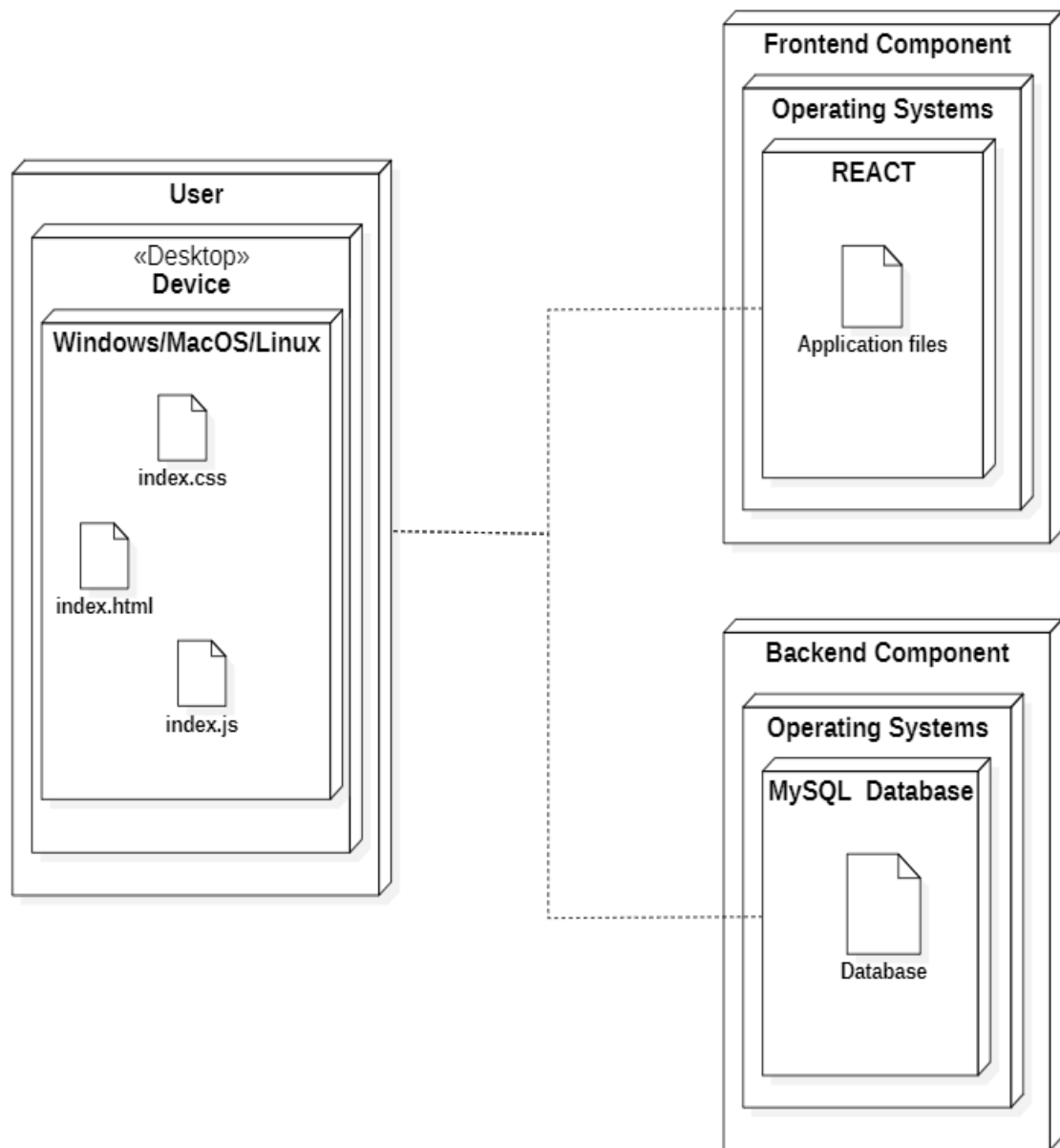
- a. Proceaseaza evenimentele generate de intera

3. **Subsistemul Baza de date:**

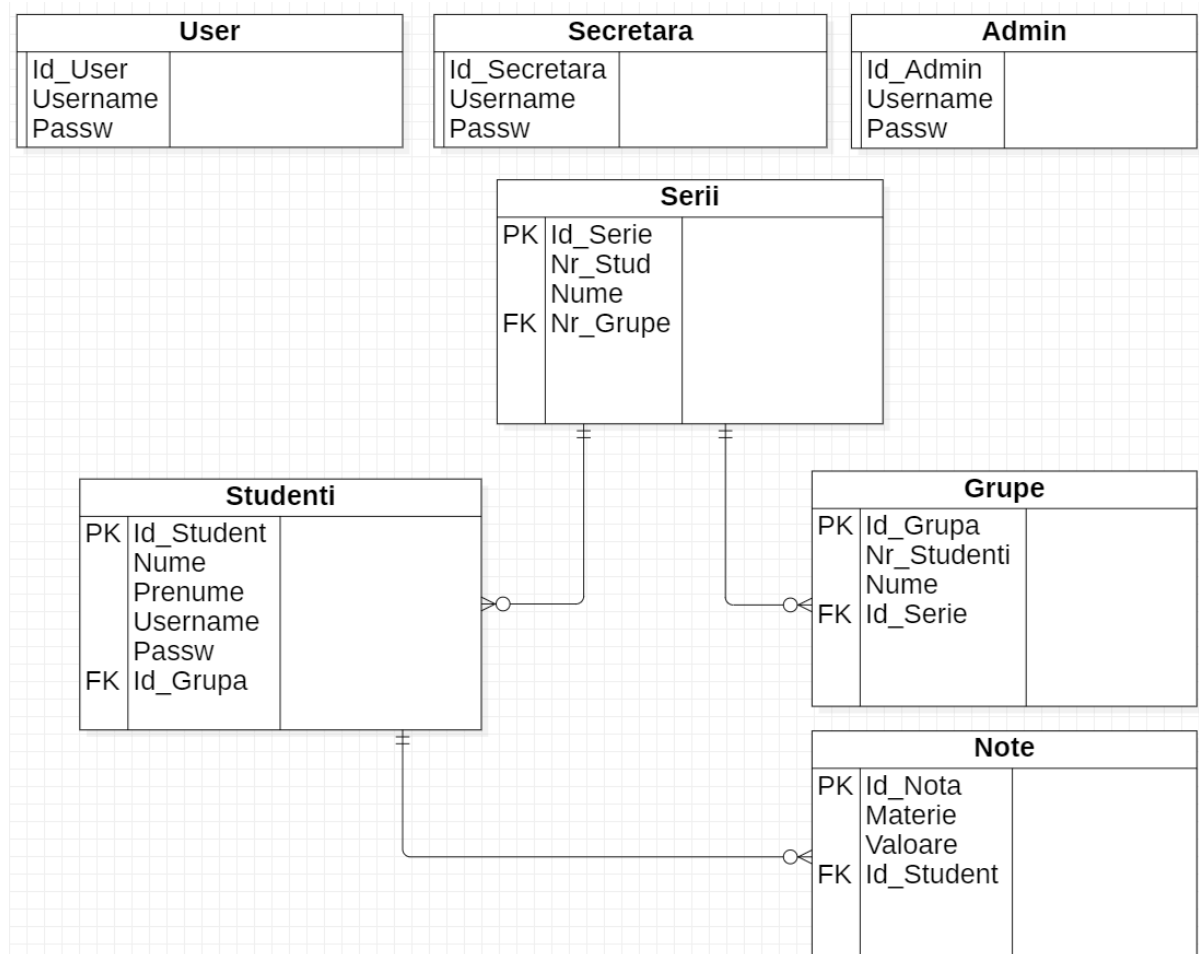
3.2 Decompozitia in subsisteme si responsabilitatile fiecarui subsistem (diagrama)



3.3 Distributia subsistemelor pe platforme hard/soft (diagrama)



3.4 Managementul datelor persistente (diagrama)



3.5 Controlul accesului utilizatorilor la sistem

Accesul utilizatorilor pe platforma se face prin accesarea contului folosind credentialele deja existente atat pentru studenti, cat si pentru administratori si secretare. Studentii primesc conturi oficiale, ce au drepturi limitate, in timp ce secretarele si administratorii sunt introdusi direct in baza de date. Autentificarea se face pe baza numelui de utilizator si a parolei.

Utilizatorul student:

- ❖ poate sa isi acceseze informatiile personale
- ❖ poate sa isi acceseze catalogul personal cu note

Utilizatorul secretara:

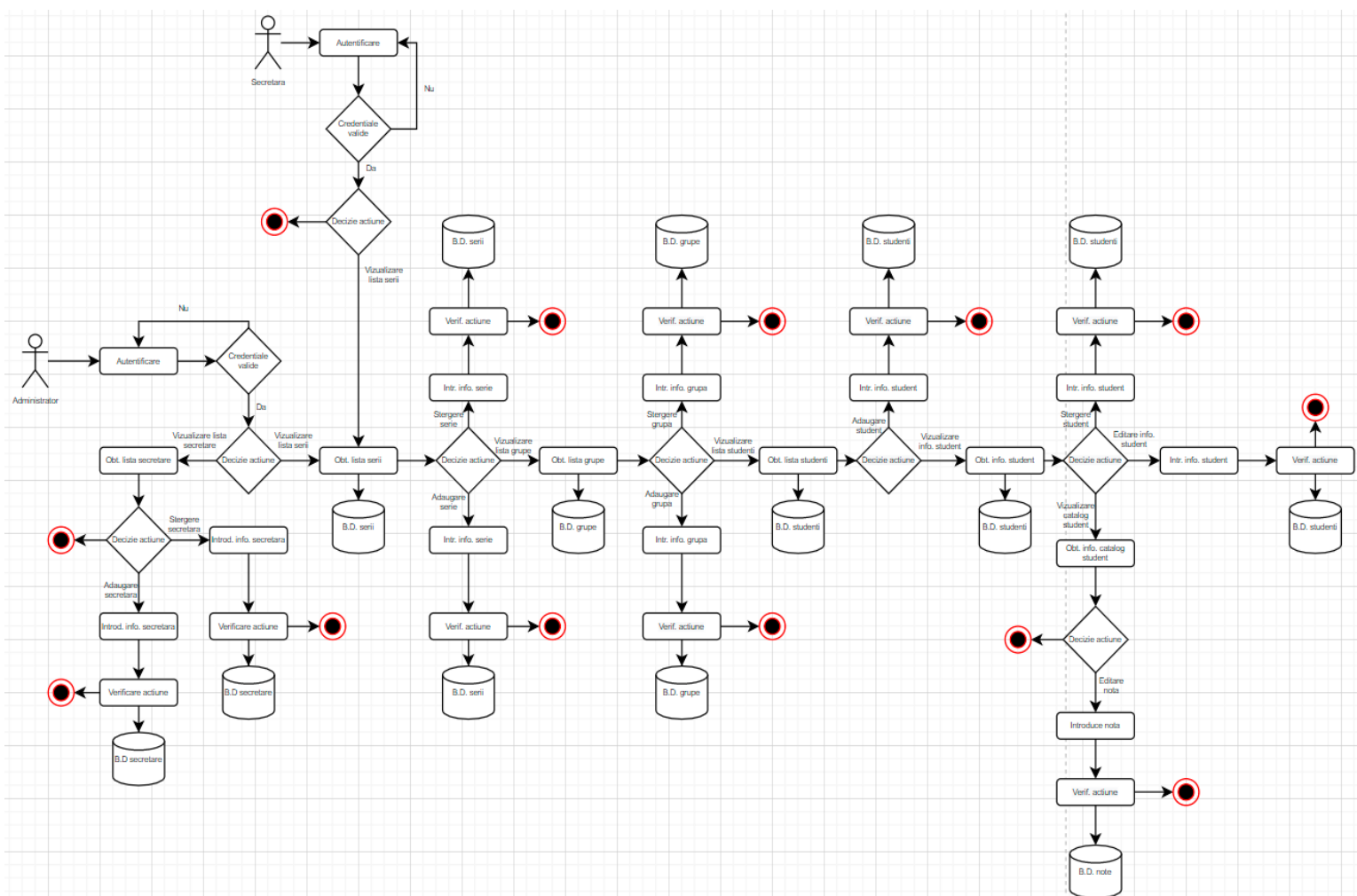
- ❖ poate gestiona lista de serii de studenti (sa o stearga, sa sa adauge una noua)

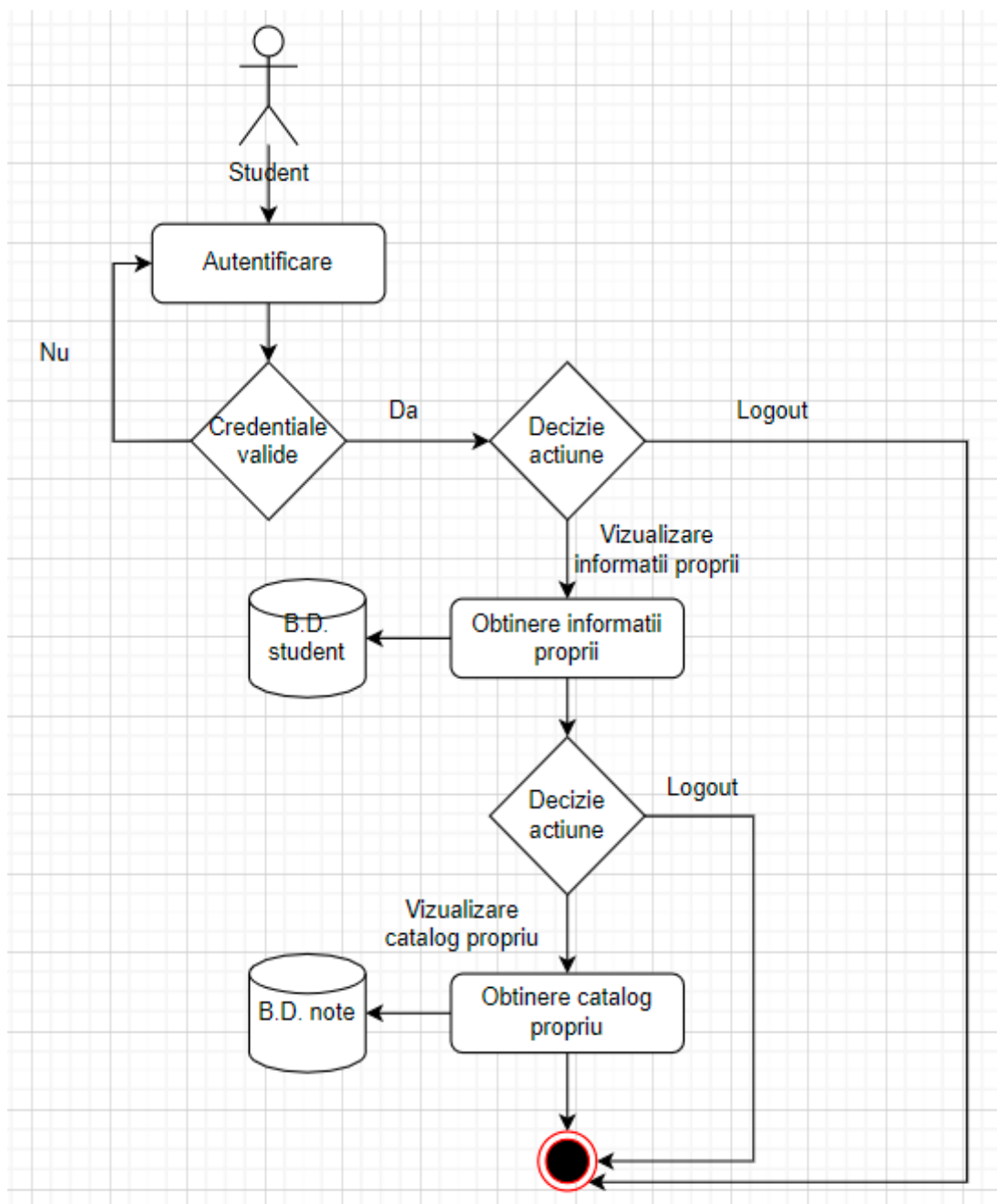
- ❖ poate gestiona lista de grupe de studenti(sa o stearga sau sa adauge una noua)
- ❖ poate gestiona lista de studenti ai unei grupe(sa adauge sau sa stearga studenti)
- ❖ poate modifica notele unui student

Utilizatorul administrator:

- ❖ poate gestiona lista de serii de studenti (...)
- ❖ poate gestiona lista de grupe de studenti(...)
- ❖ poate gestiona lista de studenti ai unei grupe(...)
- ❖ poate modifica notele unui student
- ❖ poate adauga secretare in baza de date

3.6 Fluxul global al controlului (diagrama)





3.7 Condițiile limita

Condițiile limita la care poate fi supus sistemul sunt:

- ❖ Dacă serverul/baza de date nu mai funcționează, aria de acoperire a serverului respectiv va fi decuplată, iar aplicația nu va funcționa.
- ❖ Sistemul poate pica pe o perioadă scurtă datorită traficului intens.
- ❖ Întârzieri în comunicarea cu utilizatorul datorită supraincercării sistemului