



UNIVERSITATEA DE VEST DIN TIMIȘOARA  
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ  
PROGRAMUL DE STUDII DE LICENȚĂ: Informatică

# LUCRARE DE LICENȚĂ

**COORDONATOR:**  
Conf. Dr. Monica Sancira

**ABSOLVENT:**  
Săndoiu Maria-Adina

TIMIȘOARA  
2023

UNIVERSITATEA DE VEST DIN TIMIȘOARA  
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ  
PROGRAMUL DE STUDII DE LICENȚĂ: Informatică

# Approval Dashboard

**COORDONATOR:**  
Conf. Dr. Monica Sancira

**ABSOLVENT:**  
Săndoiu Maria-Adina

TIMIȘOARA  
2023

# Abstract

The purpose of this project is to develop an application that facilitates the work of teachers in monitoring the status of each student's thesis work, from each specialization, while saving time for both the teaching staff and the students. The process of uploading, checking, and communicating is designed to be user-friendly and straightforward.

This application consists of two main components: the first part is the actual application itself, which utilizes a SQL database to store all the user-specific data, and the second part involves the automation of email communication between the student and their assigned thesis supervisor regarding the uploaded written work.

In the first part, React.js is used for both functionality and the graphical interface, along with a SQL database. JSX is an extension of the JavaScript syntax that provides a way to represent components using a syntax similar to HTML. This syntax makes the code writing process easier and makes the components more accessible for understanding. Within the React framework, JSX is used to define components, but it can also be combined with pure JavaScript. To convert JSX code into standard JavaScript objects, a preprocessor called Babel is used.

The second part of the application is implemented using the PHP language. With PHP, we establish a connection with the database to manage the necessary data and information for the application's operation. This connection allows us to efficiently and securely store and extract information from the database.

Additionally, PHP is used to implement the automation of email communication. This can be achieved through PHP's built-in functions and libraries for handling and sending email messages. By doing so, we can automatically send notifications or information to users based on specific events or actions within the application.

By combining the functionality provided by PHP for interacting with the database and automating email communication, we can create a powerful and dynamic web application that efficiently manages and processes data in an easily maintainable manner. [1]

# Rezumat

Scopul acestei lucrari este de a realiza o aplicație care să ușureze munca profesorilor în ceea ce privește monitorizarea statusului lucrării de licență, al fiecărui student, de la fiecare specializare, totodată economisind timp, atât al cadrelor didactice, cât și al studenților, modul de încărcare, verificare și comunicare fiind unul foarte simplu de utilizat.

Această aplicație are două componente majore: prima parte o prezintă aplicația propriu zisă care beneficiază de o bază de date, unde sunt stocate toate datele privind fiecare utilizator în parte, iar cea de a doua parte conține automatizarea în ceea ce privește emailul cu răspunsul profesorului coordonator la partea scrisă încărcată de student.

În prima parte se folosește limbajul React.js pentru funcționalitate și interfața grafică dar și baza de date SQL. JSX este o extensie a sintaxei limbajului JavaScript, care furnizează o modalitate de a reprezenta componentele utilizând o sintaxă similară cu HTML. Această sintaxă facilitează scrierea codului și face componentele mai accesibile pentru înțelegere. În cadrul framework-ului React, JSX este utilizat pentru a defini componentele, însă acesta poate fi combinat și cu JavaScript pur. Pentru a transforma codul JSX în obiecte JavaScript standard, se utilizează un pre-procesor denumit Babel.

A doua parte a aplicației este implementată utilizând limbajul PHP. Cu ajutorul PHP, vom realiza conexiunea cu baza de date pentru a gestiona datele și informațiile necesare în funcționarea aplicației. Această conexiune ne permite să stocăm și să extragem informații din baza de date în mod eficient și securizat.

De asemenea, folosind PHP, vom implementa funcționalitatea de automatizare a trimiterea de email-uri. Acest lucru poate fi realizat prin intermediul funcțiilor și librăriilor disponibile în PHP pentru manipularea și expedierea mesajelor de email. Astfel, putem trimite notificări sau informații către utilizatori în mod automat, în funcție de anumite evenimente sau acțiuni în aplicație.

Prin combinarea funcționalității oferite de PHP pentru interacțiunea cu baza de date și automatizarea email-urilor, putem crea o aplicație web puternică și dinamică, care poate gestiona și procesa datele într-un mod eficient și ușor de întreținut.

# Cuprins

<b>1</b>	<b>Introducere</b>	<b>6</b>
<b>2</b>	<b>Tehnologii utilizate</b>	<b>8</b>
2.1	React.js . . . . .	8
2.2	React Material UI . . . . .	8
2.3	PHP . . . . .	9
2.4	MySQL . . . . .	9
2.5	Visual Studio Code . . . . .	10
<b>3</b>	<b>Analiza stării de artă</b>	<b>12</b>
3.1	Aplicații similare . . . . .	12
3.2	Moodle.Prezentare generală . . . . .	12
3.2.1	Dezavantaje . . . . .	13
3.3	Turnitin.Prezentare generală . . . . .	13
3.3.1	Dezavantaje . . . . .	14
<b>4</b>	<b>Metodologia de proiectare și dezvoltare a aplicației</b>	<b>16</b>
4.1	Arhitectura sistemului . . . . .	16
4.2	Diagrama cazurilor de utilizare . . . . .	18
<b>5</b>	<b>Descrierea aplicației</b>	<b>20</b>
5.1	Facilități aplicație . . . . .	20
5.2	Detalii de implementare . . . . .	20
5.3	Interfața . . . . .	22
	<b>Bibliografie</b>	<b>25</b>

# Capitolul 1

## Introducere

### Motivația dezvoltării aplicației

Ideea de a crea această aplicație a apărut ca urmare a experienței personale de a avea lucrarea de licență și de a fi nevoit să participăm la întâlniri periodice cu profesorul coordonator. Situația era adesea provocatoare, deoarece trebuia să așteptăm unii după alții și timpul era limitat. Am realizat că există o nevoie evidentă de a simplifica și a eficientiza procesul de monitorizare și comunicare a progresului lucrării de licență.

Observând procesul tradițional de monitorizare a lucrărilor de licență și comunicare între studenți și profesorii coordonatori, am identificat nevoia evidentă de a simplifica și optimiza acest proces. În cadrul procesului tradițional, studenții și profesorii coordonatori trebuie să se întâlnească personal, sau prin intermediul platformelor deja existente, în mod regulat pentru a discuta despre progresul lucrărilor de licență. Aceste întâlniri pot fi dificil de programat, consumatoare de timp și pot crea neplăceri pentru ambii participanți. Prin intermediul aplicației, am vrut să simplific comunicarea, permițând studenților să încarce versiunile lucrărilor lor, iar profesorii să ofere feedback și să monitorizeze progresul în mod eficient, evitând astfel necesitatea întâlnirilor frecvente.

Un alt motiv important este economisirea timpului atât pentru studenți, cât și pentru profesori. Prin intermediul aplicației, studenții pot actualiza și încărca lucrările lor de licență în orice moment convenabil, fără a fi restricționați de programul întâlnirilor. Profesorii pot, de asemenea, să ofere feedback și să monitorizeze progresul în mod flexibil, având acces la lucrări și actualizări în timp real. Aceasta reduce semnificativ pierderea de timp și permite tuturor părților implicate să își gestioneze mai eficient programul și să se concentreze pe alte responsabilități academice sau profesionale.

O altă motivație pentru dezvoltarea acestei aplicații a fost crearea unui mediu simplu și intuitiv, ușor de utilizat atât pentru studenți, cât și pentru profesori. Am dorit să elimin complexitatea și barierele tehnice care pot apărea în utilizarea altor platforme sau metode tradiționale de comunicare și monitorizare a lucrărilor de licență. Astfel, am creat o interfață prietenoasă, care permite utilizatorilor să se orienteze rapid și să acceseze informațiile necesare fără dificultate.

## Descriere

Aplicația este o soluție completă și eficientă pentru monitorizarea lucrărilor de licență, oferind o platformă simplă și accesibilă pentru profesori și studenți. Aceasta facilitează comunicarea și colaborarea între aceștia, permițându-le să interacționeze și să colaboreze într-un mod mult mai eficient.

Cu ajutorul aplicației, profesorii au posibilitatea de a vizualiza și evalua lucrările încărcate de către studenți. Ei pot accesa informații detaliate despre fiecare student și pot urmări progresul individual al lucrărilor de licență. Prin intermediul acestei platforme, profesorii pot oferi feedback detaliat și relevant, ajutând studenții să își îmbunătățească lucrările.

## Problema și rezolvarea acesteia

Una dintre problemele majore întâmpinate în procesul de monitorizare a lucrărilor de licență este gestionarea eficientă a informațiilor și a comunicării între profesori și studenți. Metodele tradiționale, precum întâlniri fizice sau corespondență prin e-mail, pot fi ineficiente și pot duce la pierdere de timp și confuzie..

Soluția este să se evite această interacțiune, iar atunci când o lucrare de licență este trimisă către aprobare, aceasta să fie transmisă direct către profesorul coordonator, în vederea corectării și soluționării problemelor pe care le poate avea.

Astfel, timpul acordat întâlnirilor fizice/online este oferit altei activități, totodata feedback-ul transmis de către profesor poate primi durata necesară, pentru a putea fi mai ușor de înțeles de către student.

## Motivație

Soluționarea problemei în ceea ce privește timpul petrecut în așteptarea prezentării lucrării, așteptarea feedback-ului, dar și a întrebărilor constante.

# Capitolul 2

## Tehnologii utilizate

### 2.1 React.js

React.js se concentrează pe crearea de componente reutilizabile și modularizate. Componentele pot fi definite ca funcții sau clase și pot fi compuse pentru a crea interfețe complexe. Unul dintre principiile fundamentale ale React.js este conceptul de "virtual DOM" (Document Object Model). Virtual DOM este o reprezentare în memorie a structurii interfeței utilizator și se actualizează eficient în comparație cu DOM-ul real, ceea ce duce la o performanță mai bună.

React.js utilizează sintaxa JSX (JavaScript XML) pentru a defini structura interfeței utilizator. JSX combină HTML-ul și JavaScript-ul într-un mod intuitiv, permițând dezvoltatorilor să creeze componente vizuale într-un mod declarativ. React.js promovează un flux unidirecțional de date. Datele sunt transmise prin intermediul proprietăților (props) către componente și pot fi actualizate prin intermediul funcțiilor de stare (state). Aceasta duce la o gestionare eficientă și predictibilă a datelor aplicației. React.js suportă reactivitatea și reprezentarea condițională a componentelor. Aceasta înseamnă că componentele pot reacționa la schimbările datelor și pot actualiza automat interfața utilizator.

React.js are o comunitate puternică și un ecosistem bogat de biblioteci și instrumente adiacente, cum ar fi React Router pentru gestionarea rutelor, Redux pentru gestionarea stării aplicației și multe altele.

React.js este folosit într-o varietate de aplicații web, de la aplicații simple și site-uri web până la aplicații web complexe și scalabile. Este frecvent utilizat în dezvoltarea de aplicații de tip single-page (SPA) și în crearea interfețelor utilizator reactiv.

### 2.2 React Material UI

React Material-UI este o bibliotecă populară și puternică pentru dezvoltarea interfețelor de utilizator în aplicații React. Cu un set bogat de componente predefinite și stiluri inspirate din designul Material Design, React Material-UI facilitează crearea de aplicații moderne și atrăgătoare. Caracteristicile cheie ale React Material-UI includ design atractiv și coerent, componente reutilizabile, flexibilitate și personalizare. Biblioteca se bazează pe principiile designului Material Design, oferind o estetică plăcută și un aspect vizual coerent în întreaga aplicație. Componentele predefinite au stiluri bine definite și se potrivesc între ele pentru a crea o experiență vizuală armonioasă. React



Material-UI pune la dispoziție o colecție extinsă de componente reutilizabile, cum ar fi butoane, cărți, bare de navigare, casete de selectare și câmpuri de text. Aceste componente sunt bine documentate și ușor de utilizat, permițând dezvoltatorilor să creeze rapid interfețe complexe și interactive.

Un alt avantaj al React Material-UI este flexibilitatea și personalizarea. Dezvoltatorii pot modifica și personaliza aspectul și comportamentul componentelor în funcție de nevoile lor specifice. De la schimbarea culorilor și a stilurilor până la adăugarea de animații și efecte speciale, React Material-UI permite adaptarea interfeței la identitatea vizuală a aplicației și la cerințele de design.

## 2.3 PHP

PHP, un limbaj de programare open-source, este utilizat în special pentru dezvoltarea aplicațiilor web dinamice. Acest limbaj interpretat rulează pe server și generează cod HTML către clientul web, furnizând funcționalități interactive și dinamice.

PHP are o comunitate mare și activă de dezvoltatori, beneficiind de suport și contribuții constante. Aceasta a dus la dezvoltarea unei colecții bogate de biblioteci și cadre de lucru care facilitează construirea rapidă și eficientă a aplicațiilor web.

Unul dintre avantajele majore ale PHP este ușurința sa de învățare și utilizare. Sintaxa sa este asemănătoare cu cea a altor limbaje de programare, cum ar fi C și JavaScript, ceea ce facilitează tranziția pentru dezvoltatori. PHP oferă, de asemenea, flexibilitate, permițând programatorilor să-și aleagă modul de organizare a codului și să utilizeze diverse paradigme de programare.

PHP include suport nativ pentru interacțiunea cu baze de date, oferind posibilitatea dezvoltatorilor de a crea și gestiona baze de date MySQL sau alte sisteme de gestionare a bazelor de date. Acest lucru face PHP ideal pentru dezvoltarea aplicațiilor web care necesită stocare și manipulare a datelor.

Compatibilitatea PHP cu o gamă largă de servere web, precum Apache și Nginx, permite implementarea ușoară și rularea aplicațiilor PHP pe diferite platforme.

În rezumat, PHP este un limbaj de programare puternic și versatil pentru dezvoltarea aplicațiilor web. Datorită ușurinței de învățare, flexibilității și suportului extensiv, PHP rămâne o alegere populară pentru dezvoltatorii web în prezent.

## 2.4 MySQL

MySQL este un sistem de gestionare a bazelor de date relaționale (RDBMS) care a câștigat o popularitate considerabilă în comunitatea dezvoltatorilor și administratorilor de baze de date. Inițial dezvoltat de către compania suedeză MySQL AB, MySQL a fost lansat pentru prima dată în anul 1995 și, de atunci, a evoluat într-un RDBMS robust și fiabil.

Unul dintre motivele principale pentru popularitatea MySQL este faptul că este o soluție open-source. Aceasta înseamnă că codul sursă al sistemului este disponibil public și poate fi modificat și adaptat în funcție de nevoile individuale. Acest aspect a facilitat o comunitate activă și puternică de dezvoltatori care au contribuit la îmbunătățirea continuă a MySQL de-a lungul anilor.

MySQL oferă o gamă largă de funcționalități care acoperă toate aspectele esențiale ale

gestionării bazelor de date. Aceasta include crearea, modificarea și interogarea datelor, precum și gestionarea utilizatorilor și a drepturilor de acces. De asemenea, MySQL suportă multiple tipuri de stocare a datelor, cum ar fi tabelele InnoDB și MyISAM, oferind flexibilitate în funcție de cerințele proiectelor.

Un alt aspect esențial al MySQL este performanța sa ridicată. Datorită arhitecturii sale bine optimizate și tehnologiilor avansate de caching, MySQL poate gestiona un volum mare de date și poate efectua interogări complexe într-un timp scurt. Această caracteristică este extrem de valoroasă în aplicațiile care necesită o prelucrare rapidă a informațiilor, cum ar fi aplicațiile web sau sistemele de gestionare a stocurilor.

MySQL este, de asemenea, cunoscut pentru scalabilitatea sa. Aceasta înseamnă că poate gestiona cu ușurință creșterea volumului de date și a numărului de utilizatori. Acest lucru este esențial într-un mediu în care nevoile de stocare și procesare a datelor pot crește rapid în timp. MySQL oferă opțiuni de scalabilitate orizontală și verticală, permițând administrarea eficientă a creșterii sistemului.

## 2.5 Visual Studio Code

Visual Studio este unul dintre cele mai populare și puternice medii de dezvoltare integrate (IDE) disponibile pe piață. Creat de Microsoft, acesta oferă programatorilor o gamă largă de instrumente și funcționalități pentru a crea aplicații de înaltă calitate în diverse limbaje de programare.

Unul dintre aspectele remarcabile ale Visual Studio este suportul său extins pentru diferite limbaje. De la limbaje clasice precum C, C++ și Visual Basic, până la limbaje moderne precum Python, JavaScript și TypeScript, Visual Studio permite dezvoltatorilor să lucreze în mediul preferat. IDE-ul oferă completare automată de cod, evidențiere de sintaxă, depanare și multe alte caracteristici utile pentru a îmbunătăți eficiența programatorilor.

Un alt avantaj major al Visual Studio este integrarea puternică cu platformele și serviciile Microsoft. Dezvoltatorii pot crea aplicații pentru Windows, Azure, iOS, Android și multe alte platforme, utilizând instrumentele și bibliotecile specifice fiecărei platforme. De asemenea, Visual Studio permite integrarea cu servicii precum Azure DevOps, GitHub și Docker, facilitând colaborarea și gestionarea proiectelor de echipă. Visual Studio oferă, de asemenea, suport bogat pentru dezvoltarea de aplicații web și cloud. Dezvoltatorii pot crea site-uri web dinamice și aplicații web robuste utilizând ASP.NET și ASP.NET Core, beneficiind de unelte avansate pentru dezvoltarea front-end și back-end. De asemenea, Visual Studio permite crearea și implementarea ușoară a serviciilor și aplicațiilor în cloud utilizând platforma Azure.

Pentru dezvoltarea de aplicații mobile, Visual Studio furnizează șabloane și unelte puternice pentru dezvoltarea aplicațiilor iOS și Android. Dezvoltatorii pot crea aplicații mobile native sau cross-platform utilizând Xamarin, o platformă de dezvoltare mobile integrată în Visual Studio. Aceasta permite partajarea de cod între platforme și dezvoltarea rapidă a aplicațiilor mobile.

În plus, Visual Studio dispune de o gamă largă de extensii și plugin-uri create de comunitatea dezvoltatorilor. Acestea permit personalizarea IDE-ului și adăugarea de funcționalități suplimentare în funcție de nevoile fiecărui dezvoltator. De la instrumente de testare și depanare la teme și stiluri personalizate, există o varietate de extensii disponibile pentru a îmbunătăți experiența de dezvoltare.

Visual Studio este o platformă puternică și versatilă pentru dezvoltarea de software, oferind instrumente și funcționalități avansate pentru programatori. Indiferent de limbajul sau platforma de dezvoltare preferate, Visual Studio pune la dispoziție resursele necesare pentru a crea aplicații de înaltă calitate și performanță.

# Capitolul 3

## Analiza stării de artă

### 3.1 Aplicații similare

Există o varietate de aplicații disponibile pe piață care oferă funcționalități similare celei descrise în cadrul aplicației tale. Aceste aplicații au fost create pentru a satisface nevoile utilizatorilor în gestionarea lucrărilor de licență și pentru a îmbunătăți procesele academice asociate precum Moodle sau Turnitin .

### 3.2 Moodle.Prezentare generală

Moodle este o platformă de învățare online care facilitează procesul de predare și învățare într-un mediu virtual. Este una dintre cele mai populare și larg utilizate soluții de e-learning, fiind folosită în instituții de învățământ de toate nivelurile, de la școli și universități până la organizații de formare profesională și companii.

Platforma Moodle oferă o gamă variată de funcționalități și instrumente care permit profesorilor să creeze și să administreze cursuri online interactiv, iar studenților le oferă posibilitatea să acceseze materialele de curs, să participe la activități și să interacționeze cu colegii și profesorii lor.

Interfața intuitivă și ușor de utilizat a Moodle facilitează accesul la conținutul cursurilor și navigarea în cadrul platformei. Profesorii pot încărca resurse și materiale de studiu, cum ar fi documente, prezentări, videoclipuri sau linkuri către resurse externe. De asemenea, pot crea activități interactive, cum ar fi forumuri de discuții, teste, sarcini sau colaborarea la proiecte în grup. Studenții pot accesa materialele de curs, să participe la discuțiile online și să își completeze sarcinile și teste în mod interactiv. Moodle oferă, de asemenea, funcționalități de evaluare și feedback, permițând profesorilor să monitorizeze progresul studenților, să evalueze lucrările și să ofere feedback personalizat. Un aspect important al Moodle este flexibilitatea și personalizarea sa. Platforma permite personalizarea aspectului vizual și configurarea setărilor de curs în funcție de nevoile și preferințele fiecărei instituții sau a fiecărui curs. De asemenea, poate fi extinsă prin adăugarea de module suplimentare și integrarea cu alte sisteme sau instrumente de învățare. Un alt avantaj al Moodle este comunitatea vastă și activă de utilizatori și dezvoltatori. Există o mare varietate de resurse, forumuri și documentație disponibile online, care oferă suport și soluții pentru utilizatori. Moodle este o platformă puternică și versatilă de învățare online, care facilitează interacțiunea și colaborarea între pro-

fesori și studenți într-un mediu virtual. Cu ajutorul său, instituțiile de învățământ și organizațiile pot oferi cursuri și materiale de învățare interactiv, accesibile de oriunde și oricând, contribuind astfel la îmbunătățirea experienței de învățare și predare.

### 3.2.1 Dezavantaje

- Complexitatea initiala: Implementarea initiala a Moodle poate fi o provocare, mai ales pentru institutiile sau organizatiile mai mici, care pot avea resurse tehnice limitate. Este necesar sa se aloce timp si efort pentru configurarea, personalizarea si mentinerea platformei.
- Curba de invatare: Utilizarea Moodle poate necesita o perioada de familiarizare si invatare pentru utilizatori, mai ales pentru cei care nu sunt obisnuiti cu platformele de invatare online sau nu au experienta tehnica avansata. Acest lucru poate crea dificultati initiale si poate necesita formare suplimentara pentru cadrele didactice si studenti.
- Nevoia de resurse hardware si software: Pentru a rula Moodle eficient, este nevoie de resurse hardware si software adecvate. Acest lucru poate include un server puternic, spatiu de stocare suficient si o configuratie adecvata a serverului web. Aceste resurse pot necesita investitii suplimentare si expertiza tehnica pentru a le gestiona.
- Limitari ale functionalitatii: In ciuda oferirii unui set bogat de functionalitati, Moodle poate avea anumite limitari sau poate fi dificil de personalizat pentru a se potrivi exact nevoilor specifice ale unei institutii sau organizatii. Poate fi necesara implicarea dezvoltatorilor pentru a adauga functionalitati personalizate sau a imbunatati anumite aspecte ale platformei.

## 3.3 Turnitin.Prezentare generală

Turnitin este o platformă de evaluare academică și detectare a plagiatului utilizată pe scară largă în instituțiile de învățământ și în mediul academic. Scopul principal al Turnitin este de a promova integritatea academică și de a preveni actele de plagiat în lucrările academice. Unul dintre principalele avantaje ale Turnitin este capacitatea sa de a detecta și evidenția posibilele surse de plagiat într-un text. Prin compararea lucrărilor studenților cu o vastă bază de date care conține texte academice, pagini web și alte surse, Turnitin identifică porțiuni de text care sunt similare cu alte surse existente. Aceasta oferă profesorilor și instituțiilor de învățământ o unealtă puternică pentru a verifica originalitatea și autenticitatea lucrărilor academice. Platforma oferă și funcționalități de feedback și evaluare a lucrărilor, permițând profesorilor să ofere comentarii și sugestii direct în cadrul textului studenților. Aceasta facilitează comunicarea și interacțiunea între profesor și student, oferind posibilitatea de a îmbunătăți calitatea și coerența lucrărilor academice.

Un alt aspect important al Turnitin este posibilitatea de a crea rapoarte și analize detaliate cu privire la similitudinile găsite în lucrări și sursele detectate. Aceasta oferă o perspectivă clară asupra gradului de originalitate al lucrărilor și permite profesorilor să identifice și să abordeze problemele de plagiat într-un mod eficient. Turnitin este, de asemenea, integrabil cu alte platforme și sisteme de gestionare a învățământului,

permițând instituțiilor de învățământ să utilizeze această soluție în cadrul infrastructurii lor existente. Cu toate acestea, este important de menționat că utilizarea Turnitin are și câteva limitări și dezavantaje. De exemplu, algoritmul de detectare a plagiatului nu este perfect și poate genera uneori rezultate eronate sau interpretări incorecte. De asemenea, utilizarea Turnitin poate implica costuri suplimentare pentru instituțiile de învățământ, iar unele studii indică faptul că dependența excesivă de astfel de instrumente poate afecta creativitatea și gândirea critică a studenților. În concluzie, Turnitin este o platformă puternică și utilă pentru evaluarea academică și detectarea plagiatului. Cu ajutorul său, instituțiile de învățământ pot promova integritatea academică și asigura calitatea și originalitatea lucrărilor academice. Cu toate acestea, utilizarea sa trebuie să fie însoțită de o abordare echilibrată și de înțelegerea adecvată a limitărilor și implicațiilor etice ale utilizării acestui tip de instrumente.

### 3.3.1 Dezavantaje

- **Costul:** Utilizarea Turnitin poate implica costuri semnificative, mai ales pentru instituțiile de învățământ sau organizațiile care doresc să beneficieze de toate funcționalitățile oferite de platforma. Licențele pentru Turnitin pot fi costisitoare, iar aceasta poate reprezenta o barieră financiară pentru anumite instituții sau utilizatori individuali.
- **Limitări lingvistice:** Turnitin poate fi mai eficient în detectarea plagiatului în anumite limbi, precum limba engleză. Pentru alte limbi mai puțin răspândite sau cu structuri lingvistice diferite, eficacitatea sa poate fi redusă. Acest lucru poate afecta rezultatele de detectare a plagiatului pentru anumite lucrări sau texte.
- **Dependența de utilizatorii finali:** Eficiența și acuratețea Turnitin în detectarea plagiatului depind în mare măsură de utilizatorii finali, cum ar fi cadrele didactice, care trebuie să interpreteze și să analizeze rezultatele generate de platforma. Utilizatorii trebuie să aibă cunoștințe solide despre plagiat și să fie în măsură să evalueze corect rezultatele furnizate de Turnitin.

Dacă s-ar face o comparație între ce este deja pe piață (exemplul oferit mai sus) și ce este prezentat în această lucrare ar fi autenticitatea sugerată și în titlu. Am inclus o funcționalitate autentică și importantă pentru procesul de evaluare a lucrărilor de licență - "Approve" și "Decline". Această caracteristică are un impact semnificativ asupra rezultatului final al unei lucrări și asigură o evaluare justă și obiectivă.

Atunci când un student își finalizează lucrarea de licență și o încarcă în aplicație, profesorul coordonator are posibilitatea de a o evalua și de a decide dacă acesta primește aprobarea ("Approve") sau respingerea ("Decline"). Această etapă crucială are la bază un criteriu specific și transparent - suma notelor acordate de profesor studenților.

Pentru a exemplifica, să presupunem că sunt disponibile 10 teme de licență, iar nota maximă pe care un student o poate obține este 10. Înainte ca lucrarea să fie aprobată, profesorul trebuie să evalueze fiecare aspect al lucrării și să acorde note individuale

pentru fiecare student. Suma totală a acestor note acordate trebuie să depășească 6 pentru ca lucrarea să fie considerată pentru aprobare.

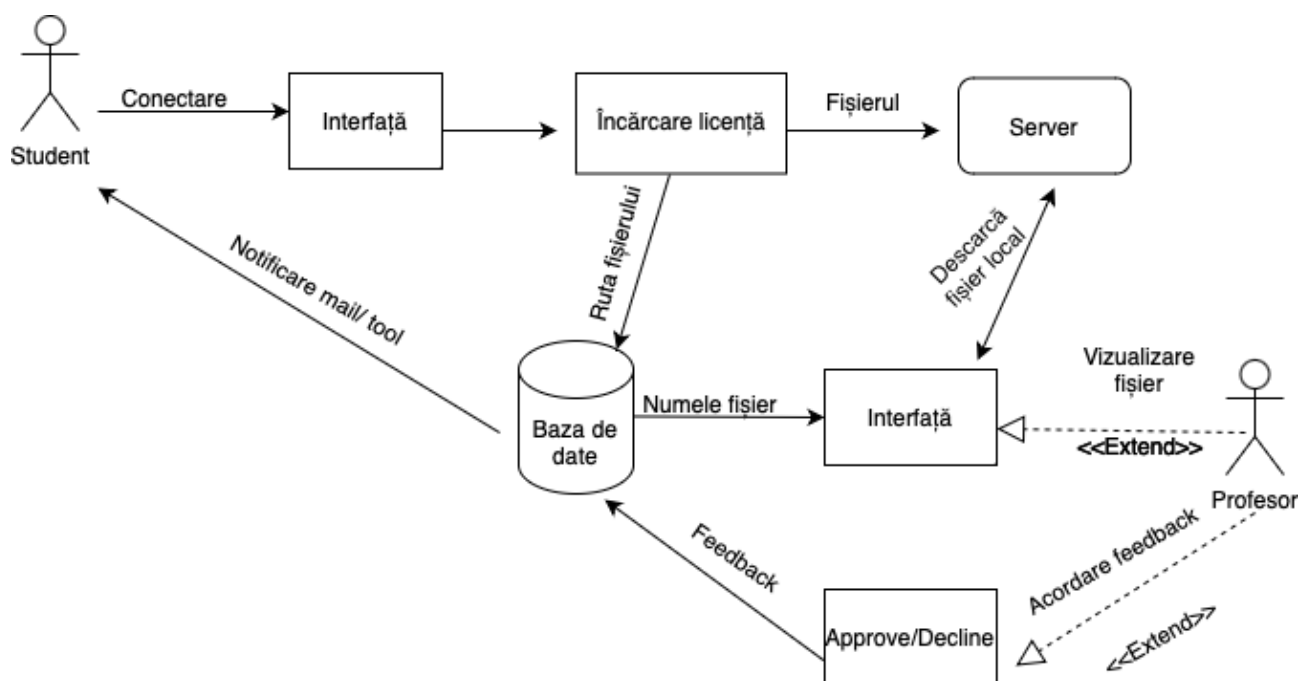
Această abordare se asigură că profesorii analizează și evaluează cu atenție fiecare lucrare și oferă o evaluare echilibrată. În același timp, se promovează standardele academice ridicate și se încurajează studenții să se străduiască să obțină rezultate bune în lucrările lor de licență.

Prin implementarea acestei funcționalități, aplicația oferă un cadru clar și coerent pentru evaluarea lucrărilor de licență și asigură o transparență în procesul de aprobare. Această abordare contribuie la menținerea integrității și calității lucrărilor academice și reprezintă o etapă crucială în parcursul academic al studenților.

## Capitolul 4

# Metodologia de proiectare și dezvoltare a aplicației

### 4.1 Arhitectura sistemului



În diagrama de mai sus se poate observa procesul de aprobare și refuzare a cererii de acces în aplicație, în contextul arhitecturii sistemului.

Primul pas este inițiat de către student, care are posibilitatea de a trimite lucrarea de licență pentru verificare. Acest lucru se realizează prin apăsarea unui buton special afișat în pagina principală a aplicației. Fișierul încărcat poate fi în format PDF, dar și o imagine sau un videoclip. După încărcare, fișierul este stocat în baza de date a aplicației și, în același timp, salvat pe serverul aplicației pentru accesibilitate și gestionare facilă.

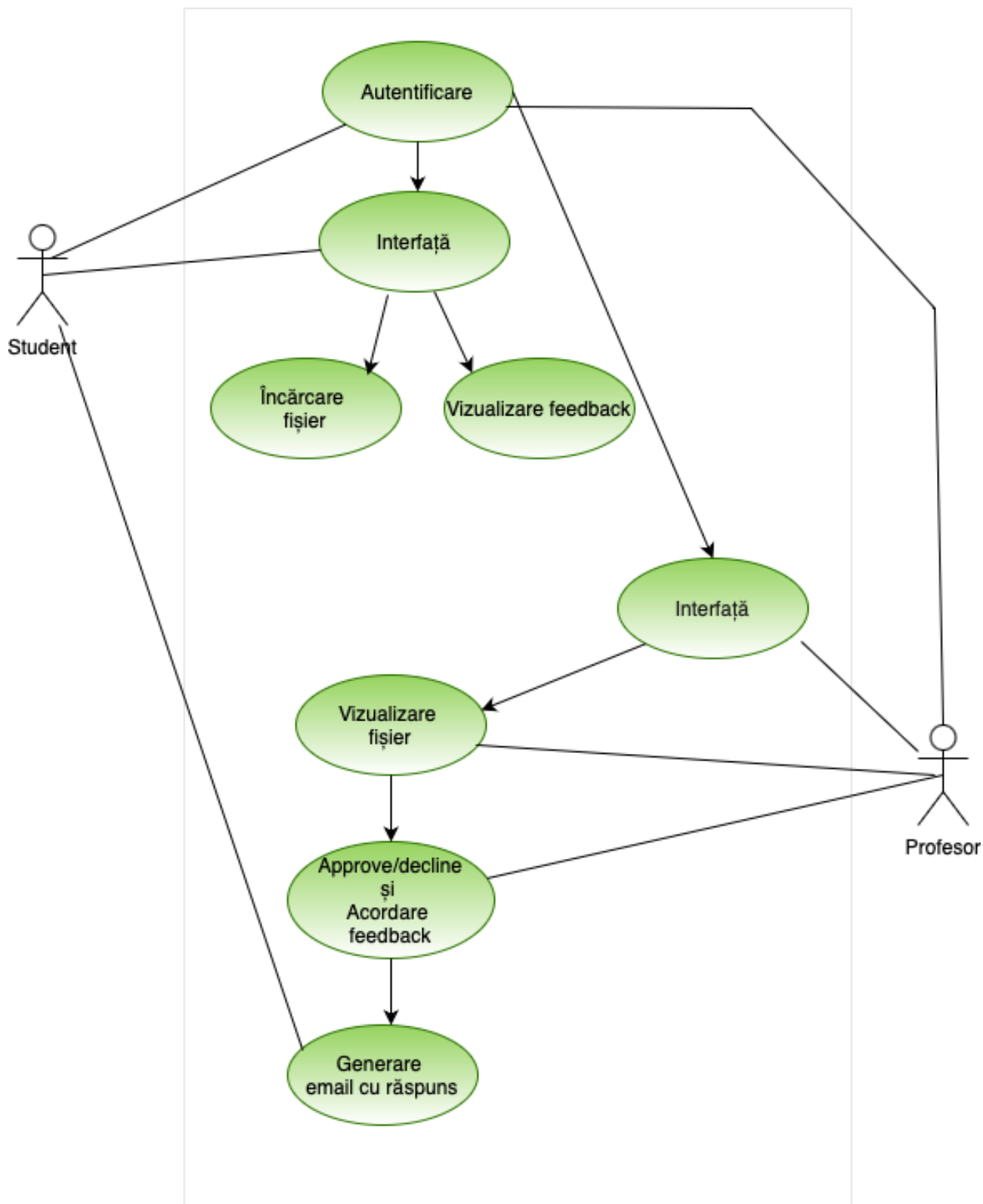


În partea dedicată profesorului coordonator, în interfața acestuia, lucrarea de licență încărcată de către student este afișată. Profesorul coordonator are posibilitatea de a vizualiza conținutul lucrării, dar și de a o descărca pentru o analiză mai detaliată. Această etapă permite profesorului să se familiarizeze cu conținutul lucrării și să evalueze corectitudinea și relevanța informațiilor prezentate.

Următorul pas este acordarea feedback-ului din partea profesorului coordonator către student. Acesta poate oferi observații, sugestii și recomandări cu privire la lucrare prin intermediul unei secțiuni de comentarii. Feedback-ul este important pentru îmbunătățirea lucrării și pentru a asigura că studenții primesc ghidare adecvată în procesul de finalizare a lucrării de licență.

După evaluarea și feedback-ul oferit, profesorul coordonator încadrează statusul lucrării de licență la "approve" sau "decline". Această încadrare a statusului reprezintă o decizie luată de profesorul coordonator în funcție de calitatea, coerența și conformitatea lucrării cu cerințele academice. Acest feedback este transmis prin email, dar poate fi observat și în aplicație.

## 4.2 Diagrama cazurilor de utilizare



În diagrama de mai sus observăm cazurile de utilizare prezente în sistem atât pentru profesor cât și pentru student. Pentru studenți, în dashboard, apare butonul de încărcare al unui fișier, dar și feedback-ul profesorului coordonator, iar profesorul, are

posibilitatea de a vizualiza, dar și descărca fișierul încărcat de student, ofera feedback și da approve/decline la lucrarea de licență.

Următorul caz este generarea unui email, care va fi trimis către student, ce va conține tot ceea ce profesorul coordonator a specificat în legătură cu lucrarea de licență.

# Capitolul 5

## Descrierea aplicației

### 5.1 Facilități aplicație

Interfața aplicației are un design minimalist și ușor de accesat și de gestionat, cu un timp de răspuns rapid la comanda acționată de utilizator.

Aplicația poate oferi o modalitate de autentificare și autorizare a utilizatorilor, permițându-le să acceseze funcționalitățile aplicației într-un mod securizat și personalizat, totodată poate oferi utilizatorilor posibilitatea de a căuta și filtra conținutul în aplicație în funcție de criteriile lor.

Pentru interfața utilizatorilor, când statusul este 'profesor', după conectare, aceștia pot comunica cu studenții prin intermediul aplicației. Aceasta poate include funcționalități precum trimiterea de mesaje individuale pentru a oferi feedback, notificări privind sarcini sau informații importante. Totodată, aceasta include un sistem de gestionare a notelor, evidențierea punctelor tari și a aspectelor de îmbunătățit, precum și vizualizarea progresului în funcție de notele obținute, printr-un barchart, în funcție de care se va putea da approve sau decline.

Pe lângă partea de interfață, există și partea de automatizare. Aceasta ușurează și mai mult utilizarea aplicației, având ca și element principal mail-ul, acesta trimițându-se automat atunci când profesorul a acordat nota și feedback-ul, dar și atunci când o temă a fost asignată.

Aceste facilități sunt doar câteva exemple de funcționalități pe care aplicația le poate oferi pentru a ușura și eficientiza activitățile profesorilor în monitorizarea lucrărilor de licență. În funcție de cerințele specifice ale proiectului, pot fi adăugate și alte funcționalități relevante pentru îmbunătățirea experienței utilizatorilor.

### 5.2 Detalii de implementare

Implementarea unei aplicații este un proces complex și captivant, care necesită o abordare atentă și bine planificată. Pentru început, am să definesc în mod clar cerințele aplicației și am stabilit funcționalitățile pe care am dorit să le includ. Aceste cerințe vor sta la baza întregului proces de dezvoltare.

Odată ce cerințele au fost stabilite, următorul pas a fost proiectarea arhitecturii aplicației. Aici m-am concentrat pe structurarea logică a aplicației și pe definirea modulelor și a interacțiunilor între acestea. Arhitectura trebuie să fie bine gândită și să ofere flexibilitate pentru viitoarele extensii și modificări ale aplicației.

După ce am stabilit arhitectura, m-am concentrat pe dezvoltarea interfeței utilizator. În acest stadiu, m-am asigurat că interfața aplicației este intuitivă, ușor de utilizat și atractivă din punct de vedere vizual.

Una dintre părțile centrale ale implementării este dezvoltarea funcționalităților aplicației. Aici am lucrat la codificarea logicii de bază care permite utilizatorilor să acceseze și să utilizeze diferitele funcționalități ale aplicației. Aceasta poate include conecatarea, vizualizarea statusului, comunicarea între profesori și studenți, evaluarea performanței studenților și multe altele. Am utilizat limbaje de programare precum PHP, JavaScript și tehnologii adecvate pentru a realiza aceste funcționalități.

Un alt aspect important a fost implementarea și gestionarea bazei de date a aplicației. Aceasta permite să stocăm și să gestionăm datele utilizatorilor și alte informații relevante. Am definit o schemă de baze de date care reflectă structura datelor și relațiile dintre ele. Pentru a interacționa cu baza de date, folosim limbaje precum SQL care ne permit să extragem și să actualizăm datele într-un mod eficient și securizat.

Pe măsură ce am avansat în procesul de implementare, a fost important să efectuez teste pentru a mă asigura că aplicația funcționează corect. Am realizat teste unitare, teste de integrare și teste de regresie pentru a identifica și rezolva eventualele erori sau probleme. Testarea și depanarea reprezintă un pas critic pentru a livra o aplicație de calitate utilizatorilor.

## 5.3 Interfața

Pentru partea de interfață, mai exact partea de front-end, construită folosind framework-ul de JavaScript, React, am început cu partea de autentificare folosind adresa instituțională UVT. Această adresă trebuie să fie înregistrată în baza de date, alături de parolă. Dacă aceste date nu sunt corect introduse, va fi afișat un mesaj de avertizare printr-un pop-up.

Pentru a realiza aceasta, am implementat un formular de login care solicită utilizatorului să introducă adresa de e-mail instituțională și parola asociată. După ce utilizatorul completează aceste câmpuri și apasă pe butonul de autentificare, datele sunt trimise către server pentru a fi verificate în baza de date.

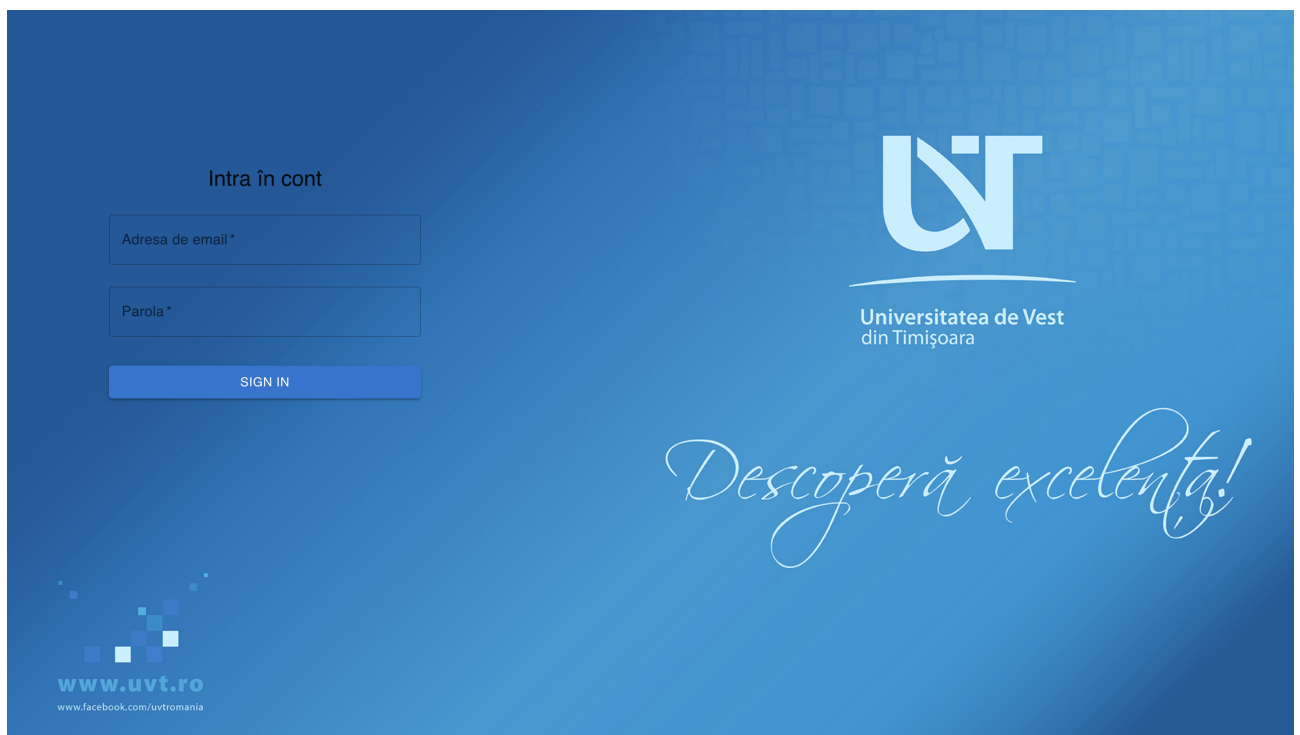


Figura 5.1: Pagina de Login

În cazul în care datele introduse nu corespund cu cele din baza de date, se afișează un mesaj de eroare în mod vizibil, prin intermediul unui pop-up sau a unei notificări. Mesajul de eroare poate indica că adresa de e-mail sau parola este incorectă, iar utilizatorul este încurajat să reîncece introducerea corectă a datelor de autentificare. Am verificat dacă ceea ce este introdus în textfield corespunde cu ceea ce este în baza de date. Am creat funcția `checkEmailExistsLog()`, este o funcție asincronă (async function) care folosește librăria Axios pentru a efectua o cerere GET către o adresă specificată. Cererea include adresa de email (`emailLog`) și parola (`parolaLog`) ca parametri în URL.

După ce serverul răspunde, rezultatul este stocat în variabila `response`. Apoi, se verifică dacă valoarea `response.data.exists` este false. Dacă este false, înseamnă că adresa de email și parola introduse nu există în baza de date, și prin urmare se setează variabila `setOpenModalLog` ca true pentru a deschide un modul de avertizare. În caz contrar, se presupune că autentificarea a fost reușită și utilizatorul este redirecționat

către pagina `"/Portal"` utilizând funcția `navigateLog()`. De asemenea, valoarea `response.data.exists` este asignată variabilei `setTot`.

```
async function checkEmailExistsLog() {
  try {
    const response = await axios.get('http://localhost:8080/approve_app');

    if (response.data.exists === false) {
      setOpenModalLog(true);
    } else {
      navigateLog("/Portal");
    }
    setTot(response.data.exists);
  } catch (error) {
    console.error(error);
  }
}

function handleClick() {
  if (emailLog === "" || parolaLog === "") {
    setOpenModalLog2(true);
  } else {
    checkEmailExistsLog();
  }
}
```

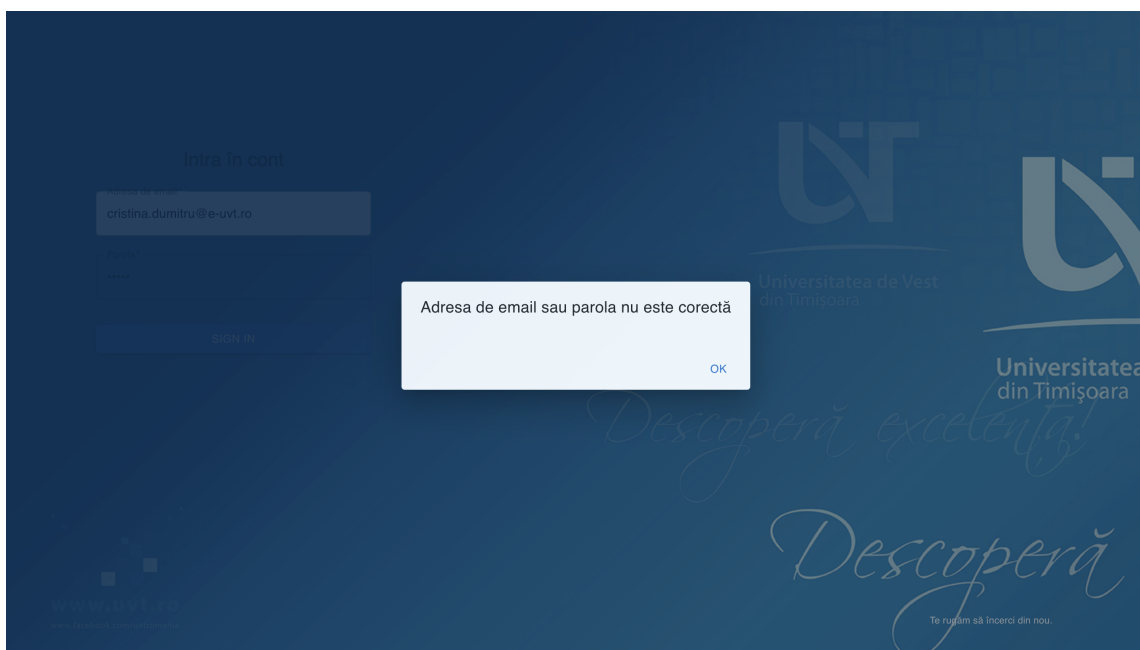


Figura 5.2: Pagina de Login cu eroare

După ce procesul de conectare este finalizat cu succes, utilizatorul este direcționat către pagina principală, unde are loc atribuirea contului la statutul de "student" sau "profesor". În funcție de statutul atribuit, utilizatorul va avea acces la o interfață specifică, adaptată nevoilor și funcționalităților relevante.

În ceea ce privește interfața studentului avem ca și buton principal, opțiunea "Tema nouă" care permite studenților să acceseze tema de licență nou asignată. În această secțiune, ei pot vedea detaliile temei, inclusiv cerințele specifice care trebuie îndeplinite. De asemenea, este disponibil și un buton "Încarcă" care permite studenților să încarce lucrarea lor în conformitate cu cerințele temei respective.

Această funcționalitate oferă studenților un mod simplu și eficient de a vedea tema lor de licență și de a începe procesul de elaborare a lucrării. Ei pot accesa cerințele și instrucțiunile detaliate pentru a-și orienta munca și pot încărca lucrarea finalizată atunci când sunt gata să o trimită pentru evaluare.

Prin intermediul acestui flux de lucru bine definit, se asigură că studenții au o înțelegere clară a temei lor și a cerințelor asociate, iar procesul de încărcare a lucrării se realizează într-un mod structurat și eficient. Acest lucru facilitează gestionarea și evaluarea lucrărilor de licență de către profesori, oferindu-le o viziune coerentă și ușor de urmărit asupra progresului și rezultatelor studenților.

În cadrul interfeței, se poate observa și afișarea datelor generale, cum ar fi informațiile despre profesorul coordonator și denumirea lucrării de licență. De asemenea, există o secțiune dedicată temelor care au fost trimise pentru notare, precum și temelor care au fost deja evaluate, afișându-se și nota obținută de către student.

Pentru a facilita urmărirea progresului și evaluarea lucrărilor, studenților le este pusă la dispoziție o bară de progres (progress bar) care indică nivelul de avans al lucrării de licență. Această bară de progres reflectă completitudinea lucrării și poate fi actualizată pe măsură ce studenții își finalizează diferitele etape sau secțiuni ale lucrării lor.



Cu ajutorul acestei bare de progres, studenții pot vedea în mod vizual cât de aproape sunt de finalizarea lucrării de licență și cât de mult le mai rămâne de lucru. Aceasta oferă o orientare clară și ajută la gestionarea timpului și a eforturilor în vederea finalizării cu succes a lucrării.

Astfel, studenții pot urmări progresul lor într-un mod simplu și eficient, având o imagine vizuală asupra stadiului lucrării lor de licență și putând planifica activitățile viitoare în consecință.

# Bibliografie

- [1] Conf. Dr. Cristina Mîndruță. *Inginerie Software - Curs*. Catedra de Inginerie Software a Facultății de Matematică și Informatică, Universitatea de Vest din Timișoara, 2021.