



**UNIVERSITATEA DIN BUCUREȘTI**

**FACULTATEA DE  
MATEMATICĂ ȘI INFORMATICĂ**



**SPECIALIZAREA INFORMATICĂ**

**Lucrare de licență**

**SNAKE ACADEMY - PLATFORMĂ  
EDUCAȚIONALĂ**

**Absolvent**

**Ion Alexandra**

**Coordonator științific**

**Prof. Dr. Florentin Eugen Ipate**

**București, iunie 2023**

## Rezumat

Platforma educațională „Snake Academy” urmărește introducerea utilizatorilor de toate vârstele, dar în special a elevilor, în limbajul de programare Python într-un mod interactiv care sporește dorința de cunoaștere. Studenții înrolați în aplicația web au la dispoziție cursuri pentru învățarea limbajului Python, însoțite de teorie, exemple, dar și video-uri care asigură creșterea perseverenței utilizatorilor de a învăța pe cont propriu. În cadrul lecțiilor se poate testa cod python pentru o înțelegere cât mai bună a conceptelor explicate.

Un alt obiectiv pe care platforma și-l propune este stimularea competiției sănătoase prin intermediul sistemului de ranking al platformei. În funcție de progresul fiecărui student, acesta poate obține puncte pentru a-și crește rank-ul și pentru a debloca avataruri pentru profil. De asemenea, aplicația pune la dispoziție un forum cu ajutorul căruia își propune să dezvolte o comunitate unită și menită ajutorului reciproc în educație.

## Abstract

The educational platform "Snake Academy" aims to introduce users of all ages, but especially students, to the Python programming language, in an interactive way that increases the desire for knowledge. Students enrolled in the web application can explore various Python courses, accompanied by theory, examples, but also videos that ensure the increase of users perseverance to learn on their own. During the lessons, you can test python code for a better understanding of the explained concepts.

One additional goal of the platform is to stimulate healthy competition through the platform's ranking system. Depending on the progress of each student, he can get points to increase his rank and unlock profile avatars. The application also provides a forum with the help of which it aims to develop a united community in education.

## CUPRINS

I.	INTRODUCERE .....	5
1.1.	Motivație.....	5
1.2.	Preliminarii .....	5
1.3.	Platforme similare vs Snake Academy .....	7
1.4.	Structura lucrării .....	7
II.	PREZENTAREA TEHNOLOGIILOR .....	8
2.1.	ASP.NET 6 Core Web API .....	8
2.2.	Entity Framework .....	9
2.3.	JSON Web Token .....	10
2.4.	HTML, CSS și JavaScript .....	11
2.5.	Angular 15 .....	12
2.6.	Facebook Login .....	13
2.7.	Syncfusion – Rich Text Editor .....	13
III.	ARHITECTURA APLICAȚIEI.....	15
3.1.	Backend .....	15
3.1.1.	Baza de date .....	15
3.1.2.	Arhitectura de bază și DTOs .....	17
3.1.3.	Repository Pattern .....	18
3.2.	Frontend .....	19
3.2.1.	Autentificarea în aplicație .....	19
3.2.2.	Componentele și rutele aplicației .....	20
IV.	PREZENTAREA APLICAȚIEI .....	21
4.1.	UI și UX.....	21
4.2.	Perspectiva studentului .....	22

4.2.1.	Meniul aplicației.....	22
4.2.2.	Paginile „Register” și „Login” .....	23
4.2.3.	Pagina „Profile”.....	23
4.2.4.	Pagina „Rank Tops” .....	25
4.2.5.	Pagina „Forum” .....	26
4.2.6.	Paginile „Chapters”, „Lesson” și „Quiz” .....	27
4.3.	Perspectiva admin-ului .....	29
4.3.1.	Cursurile .....	30
4.3.2.	Comunitatea .....	32
V.	CONCLUZII ȘI PERSPECTIVE .....	33
5.1.	Concluzii.....	33
5.2.	Perspective.....	33
VI.	LISTĂ DE FIGURI.....	35
VII.	BIBLIOGRAFIE .....	37

# **I. INTRODUCERE**

## **1.1. Motivație**

Încă din copilărie am fost pasionată de profesia de profesor, căutând mereu să îmi ajut colegii. În ciclul gimnazial mi-am descoperit pasiunea pentru programare, iar la liceu urmând profilul real de matematică și informatică, intensiv informatică, mi-am cultivat această pasiune, începând să ofer meditații în acest domeniu. Mergând pe această cale, am efectuat practica profesională la școala de programare Logiscool, pe parcursul căreia am identificat nevoile elevilor mei. Urmărind îndeaproape elevii cu vârste cuprinse între 7 și 18 ani, dar și pe părinții acestora, am putut observa care sunt problemele pe care le întâmpină în procesul de învățare.

Prin această lucrare de licență mi-am dorit să satisfac nevoile tinerilor dornici de cunoaștere, de a studia pe cont propriu, într-o manieră potrivită vârstei lor, ușor de înțeles și atractivă vizual. În cadrul cercetării mele, am constatat că există mai multe platforme online care oferă cursuri pentru Python, dar niciuna dintre ele nu este suficient de atractivă pentru elevi și nu oferă un mediu stimulant de învățare. Principalul scop al „Snake Academy” este să atragă și să mențină studenții într-o comunitate educativă și potrivită vârstei lor.

## **1.2. Preliminarii**

Pentru a înțelege mai bine aplicația se vor prezenta trei componente ale acesteia care ajută la îndeplinirea obiectivelor propuse:

- Cursurile de Python
- Sistemul de stabilire al rank-urilor
- Comunitatea: prieteni și forum

### **Cursurile de Python**

După înregistrarea utilizatorului în platformă, acesta are la dispoziție mai multe lecții, structurate pe capitole, care conțin atât informație teoretică, cât și exemple rezolvate, probleme propuse și video-uri explicative. În pagina fiecărei lecții există un interpretor de python pentru a exersa teoria sau exercițiile într-un mod rapid, fără a fi nevoie de un IDE separat pentru parcurgerea cursurilor. Codul scris în acest editor se poate descărca ulterior în calculator.

De asemenea, pentru a stimula elevii să înțeleagă lecțiile, nu doar să le parcurgă, la finalul fiecărui capitol utilizatorul are de rezolvat un quiz de 100 de puncte, pe baza cărora se stabilește dacă poate sau nu să treacă mai departe. Punctele quiz-urilor rezolvate se însumează și sunt publice pe pagina de profil a utilizatorului.

### **Sistemul de stabilire al rank-urilor**

Pentru a stimula dorința de învățare, am creat un sistem de rank-uri care menține tinerii interesați de performanța proprie. În cadrul platformei există trei rankuri ce pot fi obținute: *Baby Snake*, *Super Snake* și *Famous Snake*. Fiecare rank deblochează noi avataruri pentru profilul utilizatorului și îl încadrează într-un clasament al punctelor cu cei de același rank. Punctele obținute din rezolvarea quiz-urilor determină în ce rank se va încadra elevul, astfel: dacă punctele reprezintă maxim 30% din totalul punctelor disponibile, utilizatorul se va încadra în rankul Baby, între 31 și 80% reprezintă rank-ul Super, iar peste 80% se încadrează la rankul Famous Snake. Deoarece aplicația este în continuă actualizare, pot apărea capitole noi, iar totalul punctelor disponibile să crească, astfel, dacă utilizatorul nu parcurge noile quiz-uri va scădea în rank. Acest aspect poate determina utilizatorii să fie mai activi pe platformă și să fie la curent cu noile actualizări.

### **Comunitatea: prieteni și forum**

„Snake Academy” își propune să creeze o comunitate în care elevii își pot satisface nevoia de cunoaștere și care să le ofere susținere și prietenie. Astfel, utilizatorii au posibilitatea de a urmări alți studenți din aplicație și de a fi la curent cu progresul acestora. Prin vizitarea unui profil poți vedea numele și avatarul studentului, dar și descrierea acestuia, rank-ul, numărul de puncte și lecția curentă la care se află. Opțiunea de urmărire a profilului oferă utilizatorilor șansa de a fi la curent cu progresul prietenilor atunci când aceștia trec la alt rank, prin intermediul unei notificări.

Mai mult, pentru a se putea susține reciproc, utilizatorii pot comunica prin intermediul forumului. Acesta este un sistem de postări vizibile doar utilizatorilor autentificați în aplicație, la care orice student poate răspunde în mod anonim. Pentru a împiedica orice fel de răspunsuri inadecvate, admin-ul poate gestiona comentariile pentru a menține o comunitate prietenoasă.

### **1.3. Platforme similare vs Snake Academy**

În cadrul cercetării mele pentru această temă am testat împreună cu clasele de elevi de la Logiscool, mai multe platforme, încurajându-i să lucreze individual în afara orelor de curs, dar și ca suport teoretic folosit pe parcursul lecțiilor. Deși informațiile oferite de aceste platforme erau actualizate și extrem de utile, studenții se plictiseau repede și nu aveau nicio motivație pentru a parcurge cursurile pe cont propriu. Interfețele platformelor sunt destinate majoritar adulților, iar funcționalitățile oferite nu promovează competiția și nici nu oferă stimulente utilizatorilor la parcurgerea cursurilor. Astfel, „Snake Academy” a fost concepută pentru a satisface nevoia tinerilor de a interacționa cu o comunitate cu valori similare, în timp ce studiază într-un mediu atractiv și sunt stimulați prin competiție și recompense.

### **1.4. Structura lucrării**

Această lucrare a fost realizată utilizând tehnologii flexibile ce se pretează standardelor actuale, oferind posibilitatea actualizării aplicației la nevoie. Partea de backend a fost creată prin intermediul framework-ului ASP.NET Core Web API, iar cea de frontend a fost implementată utilizând framework-ul Angular 15. De asemenea, au fost utilizate și alte pachete care ajută la îmbunătățirea aplicației și care vor fi prezentate pe larg, în contextul aplicației, împreună cu tehnologiile principale, în capitolul 2.

Utilizarea tehnologiilor alese poate fi înțeleasă mai bine în capitolul 3, unde se descrie arhitectura aplicației. Subcapitolele detaliază atât implementarea, cât și funcționalitatea sistemului.

În capitolul 4 este explicată detaliat întreaga funcționalitate a aplicației din perspectiva utilizatorului, dar și din cea a admin-ului aplicației, pentru a crea o perspectivă mai amplă asupra funcționalității platformei.

Ultimul capitol este dedicat concluziilor, dar și perspectivelor pentru dezvoltarea viitoare a platformei, specificându-se avantajele unui astfel de proiect pe termen lung.

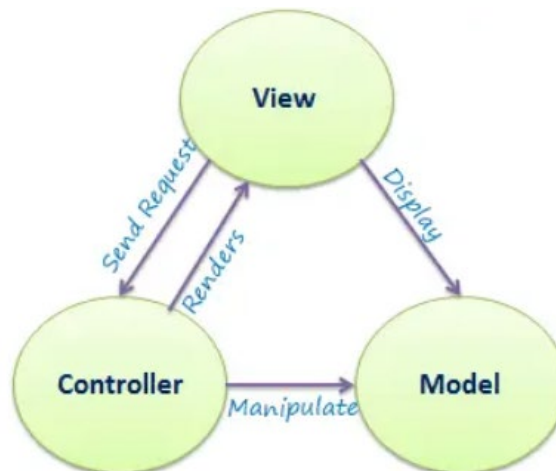
## II. PREZENTAREA TEHNOLOGIILOR

### 2.1. ASP.NET 6 Core Web API

.NET reprezintă o platformă open-source pentru dezvoltare, creată de Microsoft, destinată construirii mai multor tipuri de aplicații, precum aplicații web, pentru desktop, pentru mobil sau jocuri.<sup>[15]</sup>

*ASP.NET Web API* este un cadru pentru construirea de servicii HTTP care pot fi accesate de către orice client, inclusiv browsere și dispozitive mobile și este ideal pentru dezvoltarea API-urilor flexibile care aderă la principiile RESTful. De asemenea, acest framework permite utilizarea arhitecturii MVC pentru *Views* și *Controllers*.<sup>[21]</sup>

În cadrul proiectului, pentru partea de backend, s-a utilizat o arhitectură de tipul MVC (*Model-View-Controller*), care separă aplicația în trei componente. *Modelul* reprezintă forma pe care o iau datele aplicației (entitățile din baza de date) și este descris în C# sub forma unei clase. Componenta *View* face referire la interfața cu utilizatorul și este reprezentată în acest proiect de partea de frontend construită în Angular. *Controller-ul* gestionează cererile venite din partea utilizatorului, fiind responsabil de prelucrarea acestora și de întoarcerea unui răspuns după consultarea bazei de date, folosind componenta *Model*. Datele provenite din răspunsul dat de *Controller* sunt afișate în mod corespunzător de către componenta *View*. Aceasta din urmă nu are acces direct la *Model*, ci doar prin intermediul cererilor trimise către *Controller*.



Figură 1- Arhitectura MVC

Sursă imagine: <https://www.tutorialsteacher.com/mvc/mvc-architecture>



Acest standard de organizare ajută la obținerea principiului de separarea sarcinilor (*Separation of concerns*), care păstrează aplicația modulară și facilitează procesul de mentenanță. De asemenea, atât componenta *View*, cât și *Controller*-ul depind de *Model*, dar acesta din urmă nu depinde de celelalte componente. Acest aspect este unul dintre beneficiile cheie ale separării, permițând ca modelul să fie construit și testat independent de partea de frontend.<sup>[20]</sup>

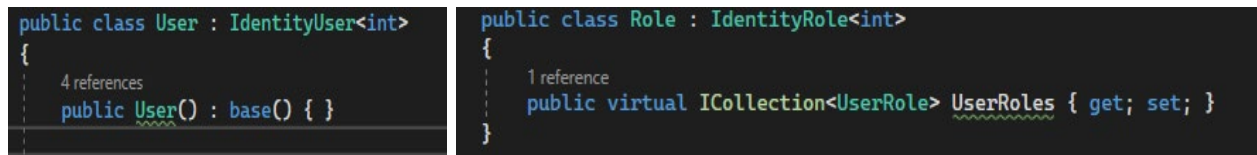
*API RESTful* este un stil arhitectural care permite accesarea datelor prin intermediul unor cereri standard HTTP. Datele pot fi accesate pentru operații de tipul CRUD (*Create, Read, Update, Delete*), utilizând metodologiile HTTP existente definite de protocolul RFC 2616:

- GET – pentru a prelua resursele
- PUT – pentru actualizarea resurselor
- POST – pentru a crea resursele
- DELETE – pentru a elimina resursele.

Formatele de date acceptate de REST API includ JSON, XML și HTML.<sup>[1]</sup>

## 2.2. Entity Framework

**Entity Framework Core** este o tehnologie ORM (*Object Reational Mapping*), extensibilă și open-source care permite dezvoltatorilor să lucreze cu date folosind obiecte .NET. Printre beneficiile acestui framework se numără posibilitatea de simplificare a codului. Dezvoltatorii pot defini baza de date utilizând metoda *Code-First*, care permite generarea și actualizarea entităților și relațiilor dintre acestea pe baza claselor C#. În cadrul aplicației „Snake Academy”, Entity Framework a fost utilizat în definirea entității „User”. De asemenea, acesta ajută în cadrul aplicației la eficientizarea definirii rolurilor pentru useri, prin extinderea claselor *IdentityUser* și *IdentityRole*. Mai mult, parolele nu sunt stocate în clar, acest framework punând la dispoziție un algoritm de *hashing* împreună cu un *salt*, care asigură securitatea și reduce vulnerabilitățile la atacul de tip *brute-force*.<sup>[13]</sup>



```
public class User : IdentityUser<int>
{
    4 references
    public User() : base() { }
}

public class Role : IdentityRole<int>
{
    1 reference
    public virtual ICollection<UserRole> UserRoles { get; set; }
}
```

Figură 2 - Entitățile User și Role

## 2.3. JSON Web Token

**JSON Web Token (JWT)** este un standard (RFC 7519) care definește o modalitate sigură și autonomă de schimb de informație între server și client, asigurând autentificarea și autorizarea. JWT este codificat în base 64 (UTF-8) și are trei componente, separate prin caracterul punct: *Header*, *Payload* și *Signature*. Componenta *Header* este reprezentată de algoritmul de criptare utilizat, precum HMAC sau RSA, în timp ce *Payload*-ul conține informații despre entitatea la care se face referință, în cazul curent, tabela User, iar semnatura (*Signature*) are rolul de validare a token-ului. În cadrul aplicației „Snake Academy”, token-ul are un rol important în procesul autentificare și autorizare al unui user, printre informațiile conținute de *Payload* numărându-se email-ul utilizatorului, numele, id-ul și rolul.<sup>[11]</sup>

The image shows a web interface for decoding a JWT token. On the left, under the 'Encoded' tab, a long base64-encoded string is pasted. On the right, under the 'Decoded' tab, the token's structure is displayed. The 'HEADER' section shows the algorithm as 'HS256' and the token type as 'JWT'. The 'PAYLOAD' section contains a JSON object with user details: email, unique name, name ID, JWT ID, role, not before, expiration, and issued at times. The 'VERIFY SIGNATURE' section shows the HMACSHA256 algorithm and provides a text input for the secret key, with a checkbox for 'secret base64 encoded'.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlbWFpbCI6ImFkbWluQGZkbWluLmNvbSIsInVuaXF1ZV9uYW1lIjoiaWRTaW4gQWNjb3VudCIsIm5hbnVpZCI6IjciLCJqdGkiOiIzZjEwYmYiOjE2ODQ0MzA2ODgsImV4cCI6MTY4NDkxNzA4OiwiaWF0IjoxNjg0ODMwNjg4fQ.SnKH_cZesUPd40_AaP6QYep7C0YtDKWZWi2j4xbnAOM
```

**Decoded**

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "email": "admin@admin.com",  "unique_name": "Admin Account",  "nameid": "7",  "jti": "3f10abd2-04e6-4ce0-b9d6-2247ad534607",  "role": "Admin",  "nbf": 1684830688,  "exp": 1684917088,  "iat": 1684830688}
```

VERIFY SIGNATURE

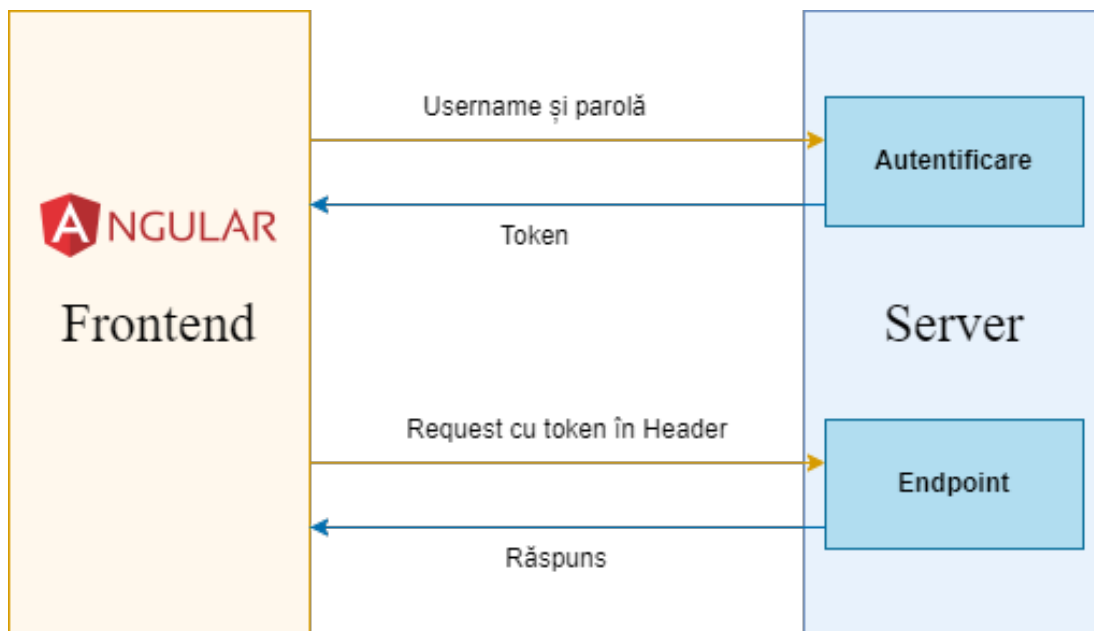
HMACSHA256(

```
base64UrlEncode(header) + "." +
base64UrlEncode(payload),
your-256-bit-secret
) secret base64 encoded
```

Figură 3- Componenta token-ului pentru contul admin-ului

**Jwt Bearer Framework** este un framework ASP.NET care implementează mecanismul de autentificare și autorizare, folosind JSON Web Token. Jwt Bearer ajută la securizarea *endpoint*-urilor prin validarea token-urilor furnizate în antetul cererilor, asigurându-se de autenticitatea și integritatea acestora. Dacă token-ul este valid, atunci clientul primește acces la informațiile protejate de respectivul *endpoint*. Token-ul este obținut de client în urma autentificării în aplicație,

prin furnizarea credențialelor (email și parolă). În cazul în care datele există în baza de date se generează și returnează JWT-ul, iar în caz contrar, se va returna un string null și un cod de eroare.



Figură 4 - Schema de funcționare a JWT în aplicație

## 2.4. HTML, CSS și JavaScript

**HyperText Markup Language** (HTML) este limbajul standard folosit pentru scrierea paginilor web. Acesta permite definirea secțiunilor paginii prin intermediul elementelor HTML cum ar fi etichetele și atributele. Etichetele informează browser-ul web unde începe și unde se termină un element, în timp ce atributele descriu caracteristicile acestuia, în pagina web afișându-se doar conținutul elementului.

În timp ce HTML-ul este folosit pentru crearea structurii paginii, *Cascading Style Sheets* (CSS) este responsabil pentru stilizarea ei, iar limbajul **JavaScript** este utilizat pentru funcționalitățile dinamice. Cele trei componente (HTML, CSS și JavaScript) sunt fundamentale pentru realizarea unei interfețe. <sup>[6]</sup>

**TypeScript** este un limbaj care definește un set de tipuri pentru JavaScript și care ajută utilizatorul să scrie cod mai ușor de înțeles. Acesta este dezvoltat de Microsoft și rulează în orice browser în care rulează și JavaScript. <sup>[23]</sup>

## 2.5. Angular 15

**Angular** este o platformă de dezvoltare a aplicațiilor web scalabilă, open-source, bazată pe TypeScript, care conține o suită bogată de biblioteci, incluzând librării pentru rutare, gestionarea formularelor și comunicare client-server, dar și alte instrumente care sporesc productivitatea. De asemenea, Angular are un sistem puternic bazat pe *dependency injection*, care îi permite gestionarea ușoară a dependențelor aplicației, promovând codul modular și reutilizabil. Angular 15 a fost lansat pe 16 noiembrie 2022, aducând îmbunătățiri versiunilor anterioare, fiind mai stabil și rezolvând multe dintre bug-urile avute de Angular 14. [5][8][18]

Framework-ul folosește DOM-ul (Document Object Model) obișnuit, actualizând tot arborele etichetelor HTML pentru orice modificare adusă paginii. Unul dintre beneficiile utilizării Angular 15 este posibilitatea structurării aplicației în funcție de componente. O componentă este definită de o clasă și reprezintă o parte a interfeței cu utilizatorul (UI). De asemenea, serviciile ajută la gestionarea logicii aplicației, care nu trebuie afișată, dar este nevoie să fie împărțită între mai multe componente.

Astfel, acest framework este ideal pentru realizarea părții de frontend a platformei „Snake Academy”, deoarece este modular, scalabil, oferă posibilitatea creării unor componente custom și este cross-platform, fiind compatibil cu mai multe browsere web.



Figură 5 - Pagina „Home” a platformei

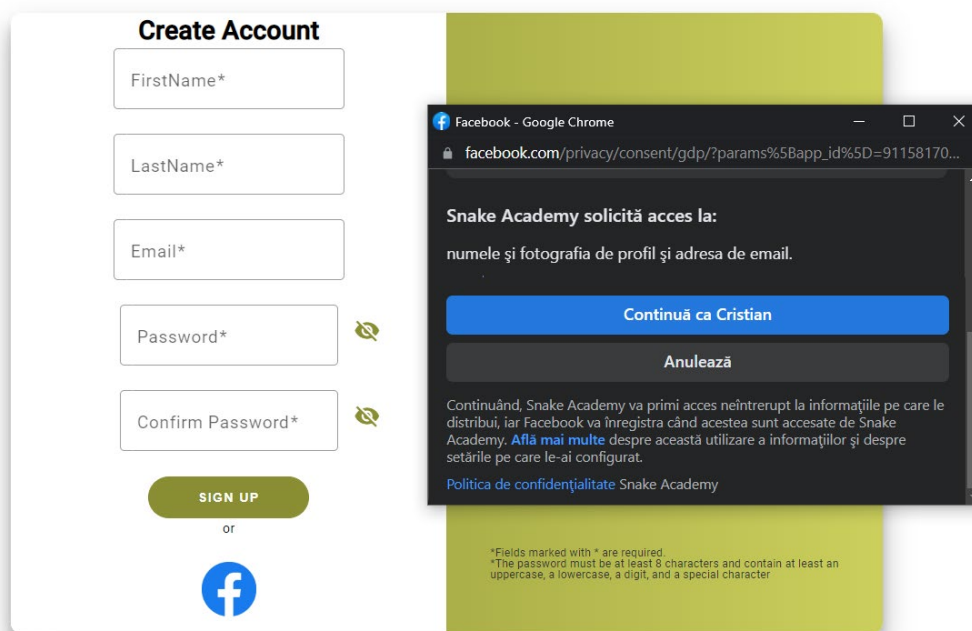
## 2.6. Facebook Login

„@abacritt/angularx-social-login” este o librărie care oferă dezvoltatorilor posibilitatea să implementeze autentificarea în aplicație printr-o rețea socială precum Facebook, Google, Amazon sau Microsoft.<sup>[12][16]</sup>

```
//social login
import { SocialLoginModule, SocialAuthServiceConfig } from '@abacritt/angularx-social-login';
import {
  FacebookLoginProvider
} from '@abacritt/angularx-social-login';
```

Figură 6 - Implementarea librăriei pentru login

În cadrul platformei „Snake Academy” este disponibilă opțiunea de înregistrare/autentificare cu Facebook.



Figură 7- Înregistrare user utilizând Facebook

## 2.7. Syncfusion – Rich Text Editor

**Rich Text Editor** este o componentă software pentru formatare de text, creată de Syncfusion, care poate fi integrată cu ușurință în orice browser. Editorul poate fi adaptat cerințelor aplicației, modificând bara de unelte cu o gamă largă de opțiuni. Rich Text Editor a fost integrat în

aplicația „Snake Academy” pentru a putea scrie și edita ușor din pagina de admin toate lecțiile din suita de cursuri, bara de unelte conținând o selecție de *tool-uri* potrivite pentru scrierea acestora.<sup>[19]</sup>

```
<ejs-richtexteditor #exemplerTE [toolbarSettings]="customToolbar">
  <ng-template #valueTemplate>
    <p id="lesson-content">    </p>
  </ng-template>
</ejs-richtexteditor>
```

```
@ViewChild('exemplerTE')
public componentObject!: RichTextEditorComponent;

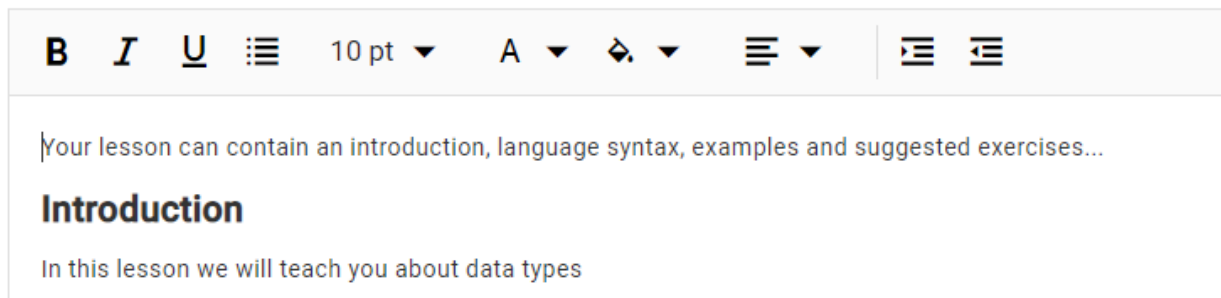
public customToolbar: Object={
  items: ['Bold','Italic','Underline','UnorderedList','FontSize','FontColor','BackgroundColor', 'Alignments', '|','Indent','Outdent']
};
```

Figură 8 - Integrarea componentei în pagina HTML și editarea bării de instrumente

În *Figura 8* este prezentată modalitatea de încorporare a editorului de text în aplicație și stabilirea instrumentelor care vor face parte din bara de unelte, iar în *Figura 9* se poate observa componenta din perspectiva utilizatorului.

Un alt avantaj pe care îl oferă Rich Text Editor este modalitatea ușoară de preluare a conținutului. Acesta este returnat în format HTML și poate fi salvat direct în baza de date și încărcat în pagina pentru cursuri, păstrându-se toate caracteristicile formătărilor.

Syncfusion oferă licență pentru acest produs contra-cost. În lipsa unei chei de activare valide, Rich Text Editor poate fi folosit în varianta *demo*, dar pagina este obstrucționată de un avertisment pentru activare. Pentru a putea utiliza la capacitate maximă Rich Text Editor, am depus o cerere pentru obținerea unei licențe gratuite, ce poate fi utilizată pe termen nelimitat în cadrul platformei „Snake Academy”.



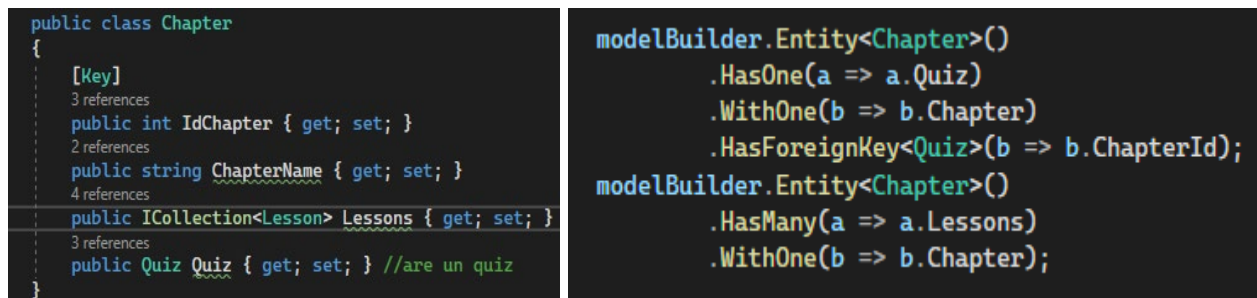
Figură 9 - Rich Text Editor

## III. ARHITECTURA APLICAȚIEI

### 3.1. Backend

#### 3.1.1. Baza de date

Baza de date a fost generată utilizând metoda *Code-First*, care permite definirea bazei de date și actualizarea acesteia pe baza claselor C#. Aceasta este stocată cu ajutorul SQL Server Management Studio (SSMS) și păstrează atât datele referitoare la cursurile platformei, cât și cele pentru utilizatori și comunitate. În *Figura 10* se poate observa codul C# pe baza căruia a fost generată entitatea *Capitol* și relațiile dintre aceasta și tabelele *Lecții* și *Quiz*.



```
public class Chapter
{
    [Key]
    3 references
    public int IdChapter { get; set; }
    2 references
    public string ChapterName { get; set; }
    4 references
    public ICollection<Lesson> Lessons { get; set; }
    3 references
    public Quiz Quiz { get; set; } //are un quiz
}

modelBuilder.Entity<Chapter>()
    .HasOne(a => a.Quiz)
    .WithOne(b => b.Chapter)
    .HasForeignKey<Quiz>(b => b.ChapterId);
modelBuilder.Entity<Chapter>()
    .HasMany(a => a.Lessons)
    .WithOne(b => b.Chapter);
```

Figură 10 - Clasa C# „Capitol” și generarea relațiilor cu alte tabele

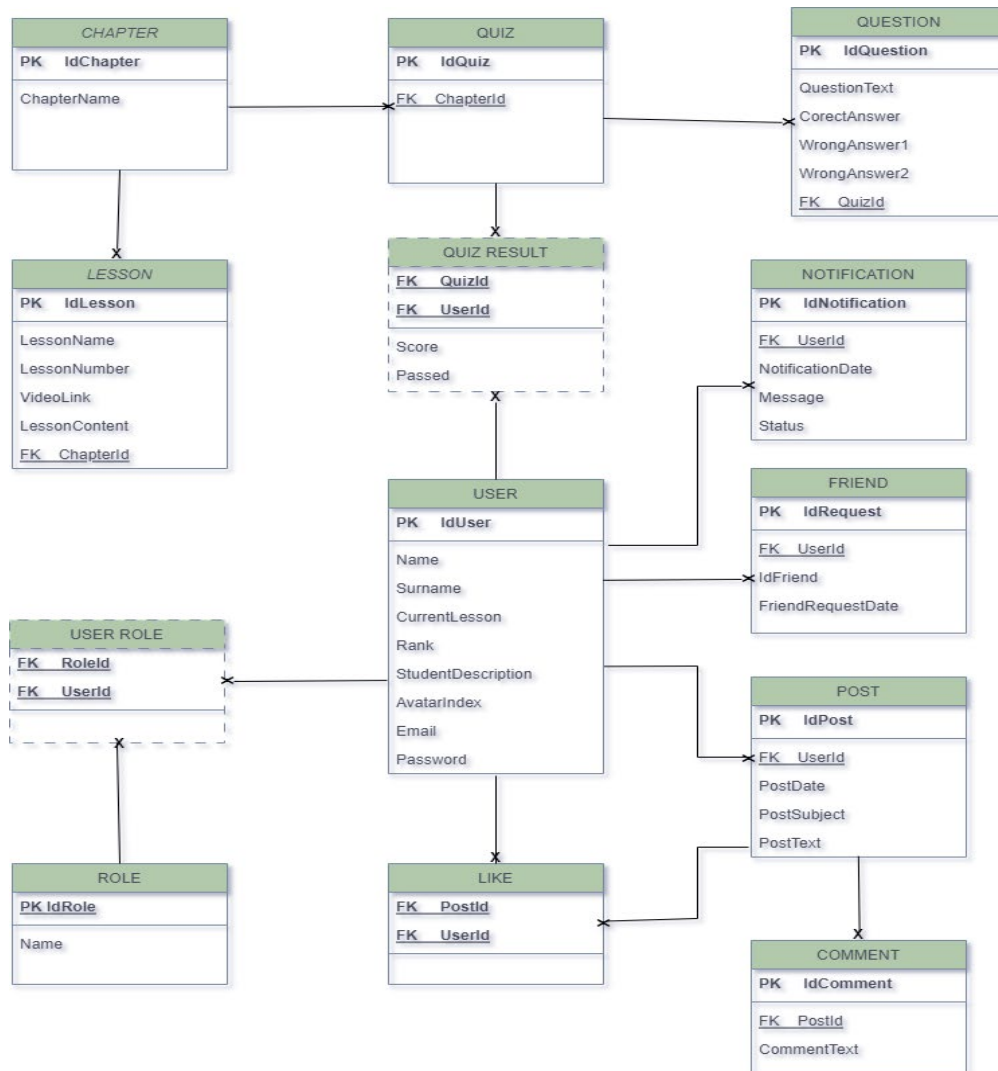
*Figura 11* reprezintă diagrama conceptuală a bazei de date, ilustrând entitățile și relațiile dintre acestea. Tabelele fac parte din 3 categorii: utilizator, cursuri și comunitate.

Datele **utilizatorului** sunt stocate în tabela *User*, dintre acestea făcând parte atât informații personale precum nume, prenume, email și parolă, cât și câmpuri care fac referire la activitatea studentului pe platformă (lecția la care se află, rank-ul, descrierea profilului, avatarul). De tabela principală *User* se leagă 2 tabele, *Notificare*, care menține notificările primite de student, și *Rol*, care stochează rolurile pe care le poate avea un utilizator. Rolurile posibile sunt cele de *Admin* și *Student*, acesta din urmă fiind atribuit automat la înregistrarea în platformă.

Entitățile *Capitol*, *Lecție*, *Quiz* și *Întrebare* constituie structura **cursurilor**, oferind posibilitatea de a gestiona ușor fiecare componentă. Cursurile sunt structurate pe capitole, fiecare dintre acestea fiind format din mai multe lecții și având repartizat un quiz final, conținând cinci întrebări de tip grilă. Admin-ul poate gestiona toate componentele cursurilor din pagina dedicată în platformă, la crearea unui capitol, fiind obligat să compună și quiz-ul cu toate cele 5 întrebări. La inserarea noului capitol în baza de date, se va crea și un quiz dedicat, dar și înregistrările aferente

întrebărilor, având 2 răspunsuri greșite și unul corect. Tabela *Lecție* cuprinde mai multe câmpuri, dintre care unul destinat conținutului teoretic, stocat în format HTML.

Partea de **comunitate** este reprezentată de Forum, datele aferente acestuia fiind stocate în tabelele *Postare*, *Comentariu* și *Like*. Un utilizator poate lăsa mai multe postări pe forum, cuprinzând un subiect, textul propriu-zis și data postării. De asemenea, acesta poate aprecia postările altor utilizatori sau poate lăsa comentarii în mod anonim.



Figură 11 - Diagrama conceptuală a bazei de date



### 3.1.2. Arhitectura de bază și DTOs

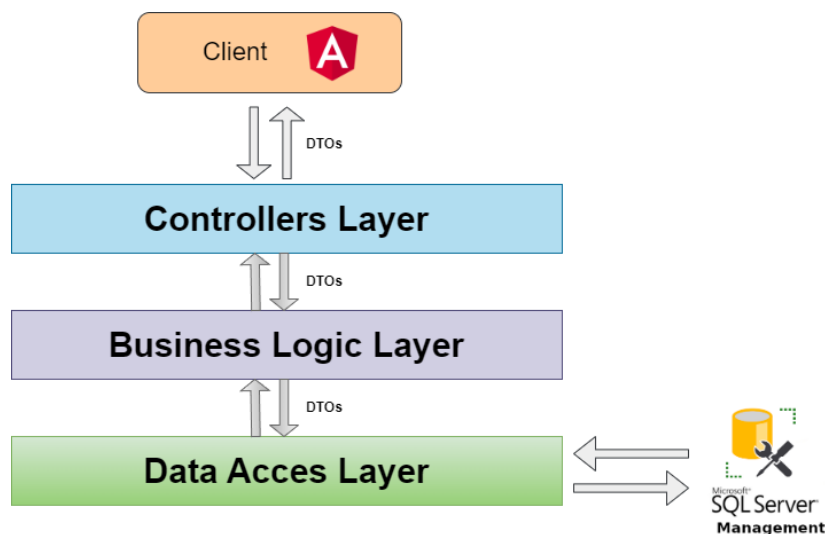
#### *Data Transfer Object*

*Data Transfer Objects* (DTOs) sunt obiecte care transportă date între procese pentru a reduce numărul de apeluri la metode, grupând mai mulți parametri într-un singur apel. Acest aspect este benefic, deoarece reduce supraîncărcarea rețelei. Un alt avantaj este încapsularea logicii serializării. Astfel, DTO-urile ne oferă posibilitatea să selectăm doar anumite atribute ale unei entități, cât și să combinăm câmpurile mai multor tabele într-un singur obiect, eficientizând traficul de date.<sup>[7]</sup>

#### Arhitectura de bază

Partea de server este structurată pe trei *layere*: *Controller Layer*, *Business Login Layer* (BLL) și *Data Access Layer* (DAL). ***Controller Layer-ul*** este reprezentat de setul de controlere și are scopul de preluare a cererilor venite din partea clientului și transmiterea mai departe a datelor către *BLL*.

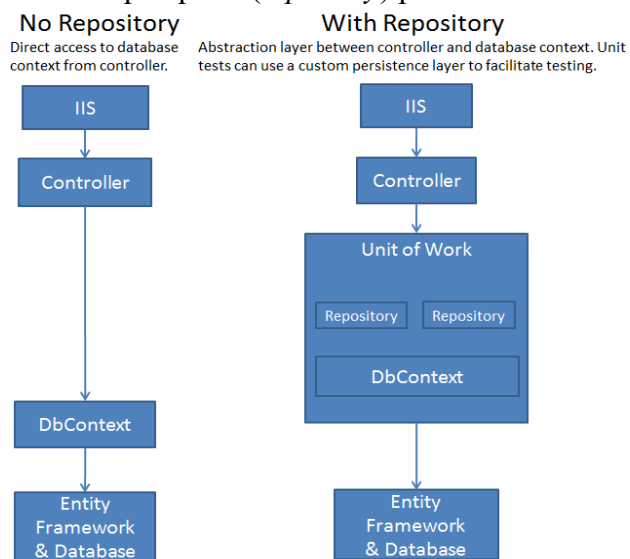
***Business Login Layer*** gestionează logica aplicației, definind toate operațiile care pot fi făcute asupra datelor în cadrul aplicației. Stratul este constituit din servicii, acestea fiind responsabile pentru manipularea și procesarea datelor stocate în bază. Printre sarcinile serviciilor se numără validarea datelor, efectuarea de calcule și transformări și gestionarea erorilor și excepțiilor. ***Data Access Layer*** este componenta responsabilă de comunicarea cu baza de date. Printre sarcinile acesteia se numără conectarea la sistemul de stocare, executarea instrucțiunilor SQL și maparea datelor din baza de date la obiectele de date ale aplicației.<sup>[9][10]</sup>



Figură 12 - Schemă a comunicării între layere

### 3.1.3. Repository Pattern

**Repository Pattern** este un model de design al aplicației care adaugă un layer de abstractizare pentru *Data Access Layer*, ascunzând detaliile exacte ale manipulării bazei de date. Acesta facilitează testarea automată și permite mentenanța mai ușoară. Modelul presupune crearea unei interfețe și a unei clase de tip depozit (*repoitory*) pentru fiecare entitate.<sup>[14]</sup>



Figură 13 - Schemă de comparare între utilizarea și neutilizarea modelului  
sursă imagine: <https://acesse.one/microsoft-repository-pattern>

În cadrul aplicației, toate clasele *repository* au fost injectate în clasa comună *RepositoryWrapper*, acest unic depozit fiind instanțiat în fiecare Controller, oferind acces către orice *repository* este necesar. În Figura 14 se poate observa exemplul Controller-ului entității *Capitol*.

```

public class RepositoryWrapper : IRepositoryWrapper
{
    private readonly Context _context;
    private IUserRepository _user;
    private IFriendRepository _friend;
    private ISessionTokenRepository _sessionToken;
    private IChapterRepository _chapter;
    private ILessonRepository _lesson;
    private IQuizRepository _quiz;
    private IQuizResultRepository _quizResult;
    private INotificationRepository _notification;
    private ICommentRepository _comment;
    private IPostRepository _post;
    private ILikeRepository _like;
    0 references
    public RepositoryWrapper(Context context)
    {
        _context = context;
    }
}

```

```

public class ChapterController : ControllerBase
{
    private readonly IRepositoryWrapper _repository;

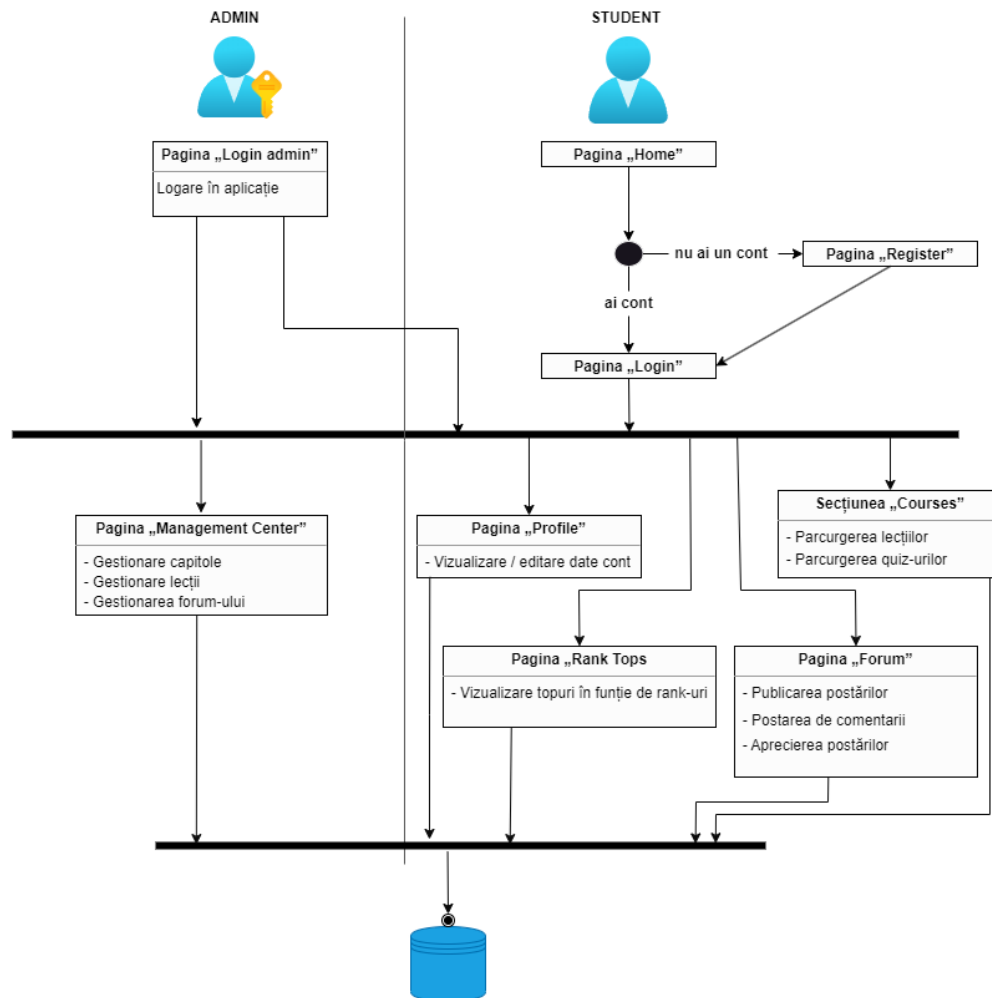
    0 references
    public ChapterController(IRepositoryWrapper repository)
    {
        _repository = repository;
    }
}

```

Figură 14 - Instanțierea clasei *RepositoryWrapper* în controller-ul *Capitol*

## 3.2. Frontend

### 3.2.1. Autentificarea în aplicație



Figură 15 - Fluxul de funcționare al aplicației.

Aplicația este împărțită în două mari interfețe: interfața pentru *student* (utilizatorul normal) și cea pentru *admin*. Logarea în platforma destinată studentului se face din pagina de logare normală, ce este accesibilă din pagina „Home”. Această formă de logare nu este destinată admin-ului și nu oferă beneficiile acestui rol. User-ul de tip *admin* se poate autentifica doar din pagina de login „Admin”, primind după logare acces la centrul de management. Prin autentificarea admin-ului, acesta are acces și la contul de student personal, beneficiind de toate posibilitățile unui student. În *Figura 16* se pot observa cele două formulare de logare diferite în funcție de rolul utilizatorului, în partea stângă fiind formularul pentru student, iar în cea dreaptă, pentru admin.



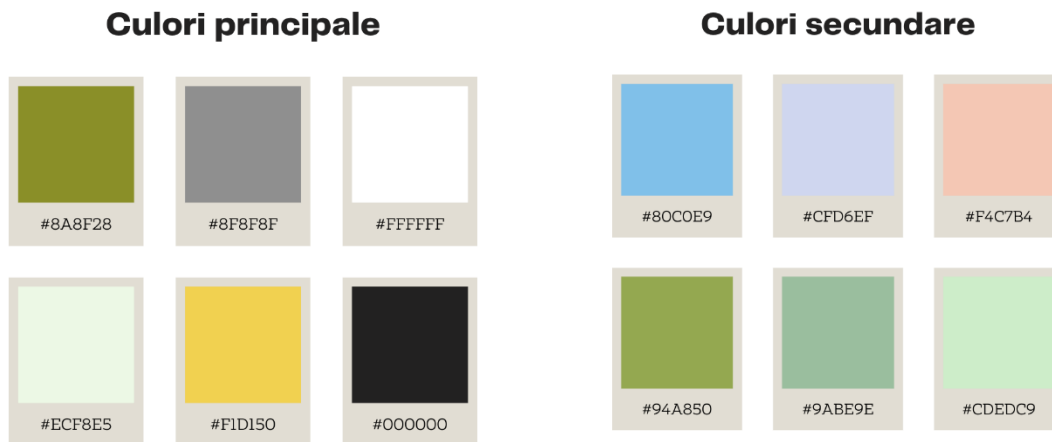
## IV. PREZENTAREA APLICAȚIEI

### 4.1. UI și UX



Figură 18 - Logo-ul aplicației

„Snake Academy” își propune să ofere utilizatorilor un mediu cât mai prietenos, care să îi motiveze, fiind interactiv și ușor de înțeles. Design-ul aplicației a fost conceput cu scopul de a fi atractiv pentru toți utilizatorii, dar în special pentru elevi. Estetica platformei se bazează pe o schemă de culori bine definită, plăcută ochiului uman, iar tematica se concentrează asupra șarpelui, făcându-se legătura cu limbajul Python. Paleta de culori este formată din șase culori principale, utilizate în toată componenta aplicației, dar există și mai multe culori secundare, utilizate local, într-o anumită componentă sau în elemente grafice (logo, avatar). În *Figura 19*, sunt reprezentate culorile principale, dar și unele dintre culorile secundare.



Figură 19 - Paleta de culori

Tema „șarpelui” asupra căreia se concentrează interfața grafică se poate observa atât în declinările culorii verde, cât și în imaginile utilizate pentru avatarul profilului sau pentru rank-uri. În *Figura 20* sunt reprezentate mai multe exemple pentru utilizarea simbolului asociat limbajului Python.



Figură 20 - Grafică utilizată pentru tema „șarpelui”

De asemenea, platforma este intuitivă, fiind ușor de navigat între toate componentele ei, dar și de utilizat toate caracteristicile pe care le pune la dispoziție. Paginile principale se pot accesa utilizând meniul disponibil pe tot parcursul navigării în aplicație.

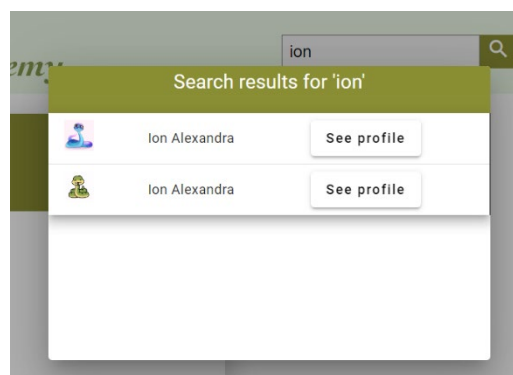


Figură 21 - Meniul aplicației

## 4.2. Perspectiva studentului

### 4.2.1. Meniul aplicației

Meniul este alcătuit din 3 butoane ce redirecționează utilizatorul către paginile „Home”, „Courses” și „Profile”, și un *dropdown* pentru paginile de comunitate, „Forum” și „RankTops”. Tot în cadrul meniului este afișat logo-ul platformei, cât și o bară de căutare pentru prieteni. Prin căutarea numelui unui utilizator se vor afișa toți studenții cu acel nume. Din lista de rezultate se poate naviga ușor către profilul selectat.



Figură 22 - Rezultatele utilizării bării de căutare

#### 4.2.2. Paginile „Register” și „Login”

Pentru a putea naviga prin aplicație și pentru a beneficia de toate posibilitățile acesteia, utilizatorii au nevoie de un cont. Înregistrarea în aplicație se face din pagina „Register”, prin completarea formularului sau prin utilizarea opțiunii de înregistrare cu Facebook. Prin înregistrare, în baza de date se vor stoca numele, prenumele și email-ul studentului, cât și parola aleasă. În urma înrolării, utilizatorul va fi redirecționat către pagina de „Login”, unde trebuie să completeze email-ul și parola setate anterior, ori poate utiliza metoda autentificării cu Facebook, dacă aceasta a fost folosită și la înregistrare. De asemenea, pagina de autentificare se poate accesa oricând în mod independent de cea de înregistrare. În partea stânga a *Figurii 23* se află formularul de înregistrare, iar în dreapta, cel de autentificare.

Figură 23 - Paginile „Register” și „Login”

#### 4.2.3. Pagina „Profile”

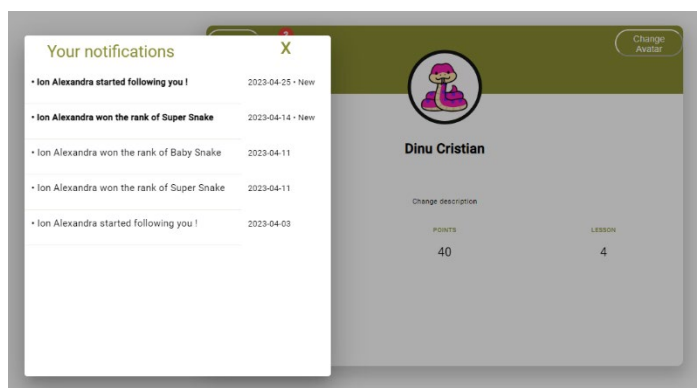
Fiecare user înregistrat are acces la pagina de profil, unde își poate vizualiza și gestiona activitatea în platformă. Profilul unui student este alcătuit din șase componente, acestea fiind vizibile tuturor utilizatorilor aplicației:

- Numele: alcătuit din numele și prenumele completate la înregistrare
- Descrierea: aceasta este goală inițial, dar poate fi oricând modificată acționând butonul „Change description”
- Punctele: acestea reprezintă suma punctelor obținute la quiz-uri

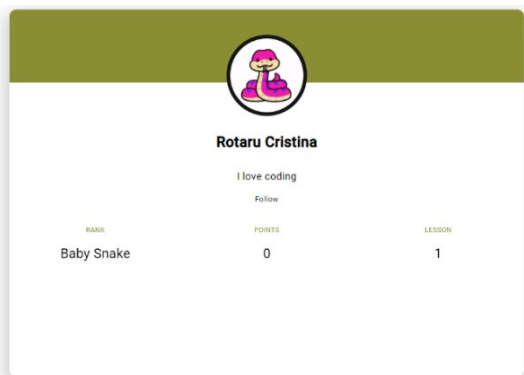
Figură 24 - Fereastra pentru schimbarea descrierii

- Rank-ul: este stabilit automat în funcție de puncte
- Lecția: lecția curentă la care se află studentul
- Avatar-ul: reprezentat de imaginea unui șarpe

Deși profilul personal și cel al altui utilizator conțin aceleași informații, diferența se poate observa la butonul „*Change description*”. La vizitarea unui profil, acest buton este înlocuit de cel pentru *follow/unfollow* care permite urmărirea sau anularea urmăririi aceluia student. Prin urmărirea unui alt student, utilizatorul este notificat atunci când prietenul urmărit își schimbă rank-ul. De asemenea, persoana urmărită primește o notificare prin care este înștiințată de noul prieten (Figura 25). Urmărirea nu necesită acordul persoanei către care a fost trimisă cererea.

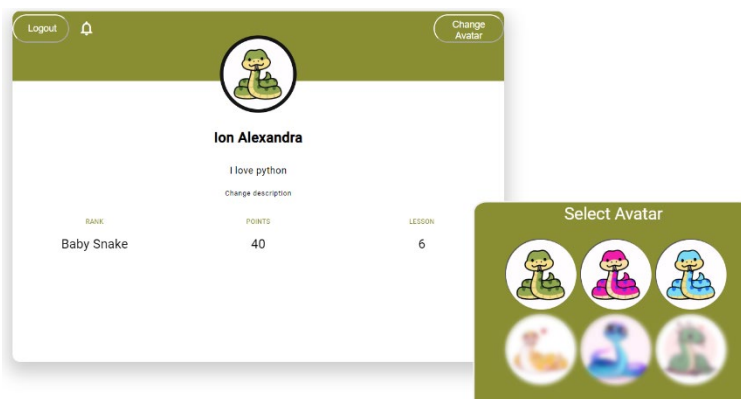


Figură 25 - Fereastra de notificări



Figură 26 – Pagina publică a unui prieten

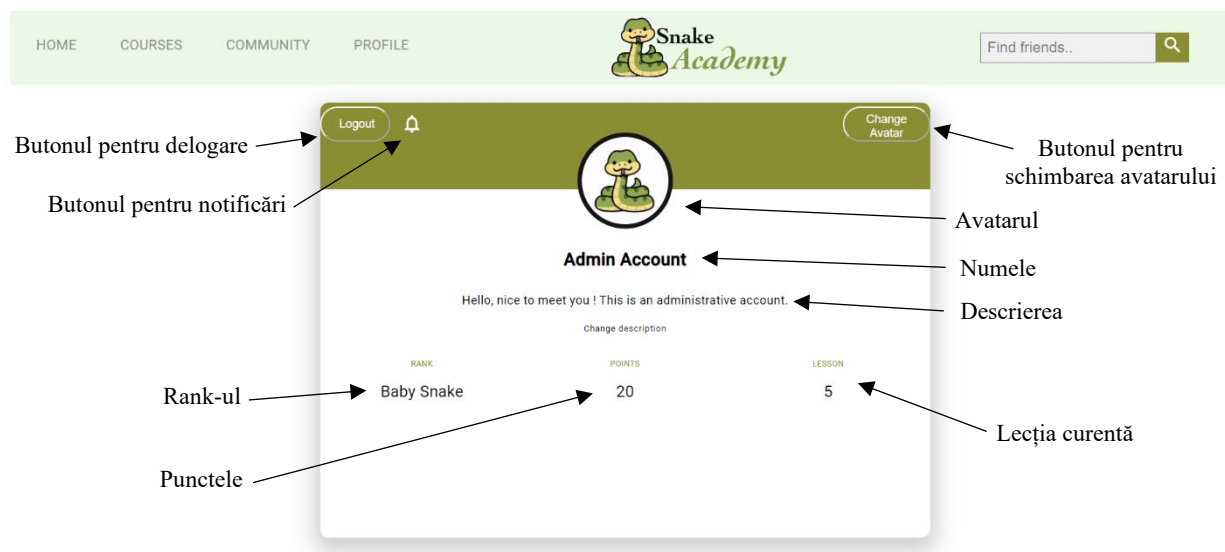
În plus, din cadrul profilului personal, utilizatorul se poate deloga din aplicație prin butonul „*Logout*”, fiind redirectionat către pagina de logare și pierzând accesul la restul paginilor. Tot din profil se pot vizualiza notificările primite acționând clopoțelul din stânga sus al paginii. Atunci când există notificări noi, clopoțelul este alb și afișează numărul de notificări necitite. Prin butonul „*Change Avatar*”, din colțul stânga sus, se deschide o fereastră de unde se poate schimba avatarul. În Figura 27, studentul are rank-ul „*Baby Snake*” și nu a deblocat toate avatarurile. Acestea sunt încadrate la un anumit rank și devin disponibile pentru utilizatorii care au atins acele praguri.



Figură 27 - Fereastra de schimbare a avatarului



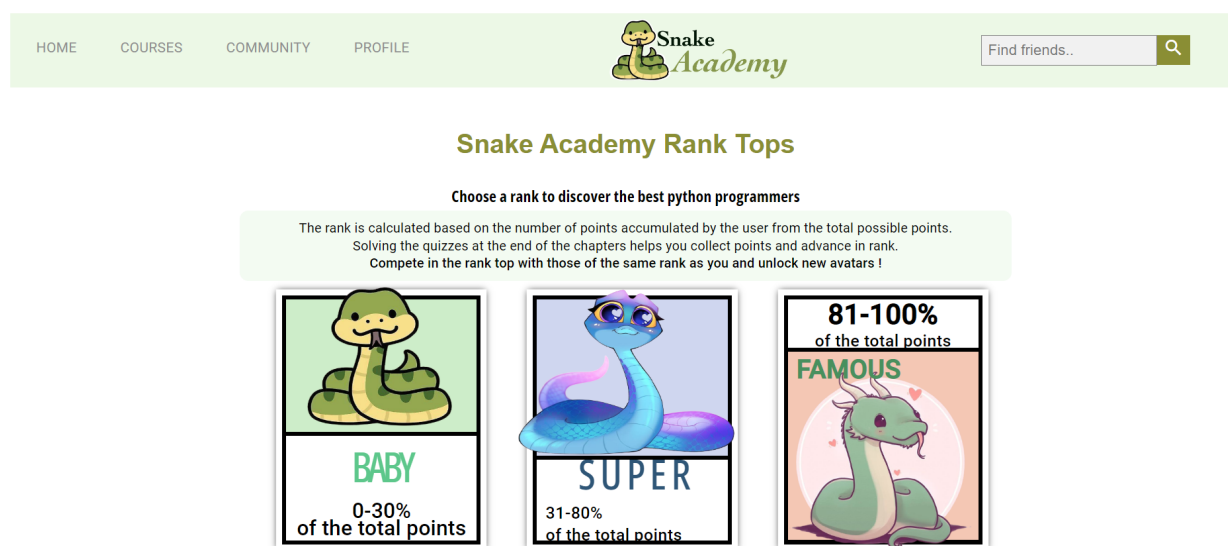
În *Figura 28* se pot observa toate componentele paginii de profil personal, prezentate anterior.

















*Figură 28 - Pagina „Profile” pentru contul de admin*

#### 4.2.4. Pagina „Rank Tops”

Pagina „Rank Tops” este o pagină destinată comunității, conținând trei topuri, unul pentru fiecare rank (*Baby Snake*, *Super Snake* și *Famous Snake*), în care sunt încadrați toți utilizatorii, descrescător după numărul punctelor. Din cadrul fiecărui tabel se poate naviga către profilul unui utilizator, excepție făcând user-ul autentificat, acesta fiind redirectionat către profilul propriu editabil. Alături de cele trei topuri se află instrucțiuni cu privire la sistemul de ranking.



*Figură 29 - Pagina „Rank Tops”*

Baby Snake		Super Snake		Famous Snake	
<div><div></div><div></div><div></div><div></div></div>		Baby Snake			
No.		Name	Rank	Score	
1		Dinu Cristian	Baby Snake	40	
2		Ion Alexandra	Baby Snake	40	
3		Admin Account	Baby Snake	20	
4		Dinu Cristian	Baby Snake	0	
5		Postolache Miruna	Baby Snake	0	
6		Rotaru Cristina	Baby Snake	0	
7		Andreea Icob	Baby Snake	0	

Figură 30 - Tabelul pentru rank-ul „Baby Snake”

#### 4.2.5. Pagina „Forum”


Pagina „Forum” este, de asemenea, o pagină destinată comunității, conținând postări nominale create de utilizatori. Orice utilizator poate posta pe forum în nume propriu, dar poate lăsa și comentarii anonime la postările existente. De asemenea, utilizatorii se pot susține reciproc prin utilizarea butonului „Like”.

**Forum**

Crearea unei noi postări



Rubrica de postări

**Subject: My rank**

 **Andreea Icob**  
Posted on 2023-04-03

Do you know how I can increase my rank?

Comments • All comments are anonymus

 Read the rank tops page  
 Solve quizzes

Butonul pentru postarea unui comentariu

Butonul pentru like

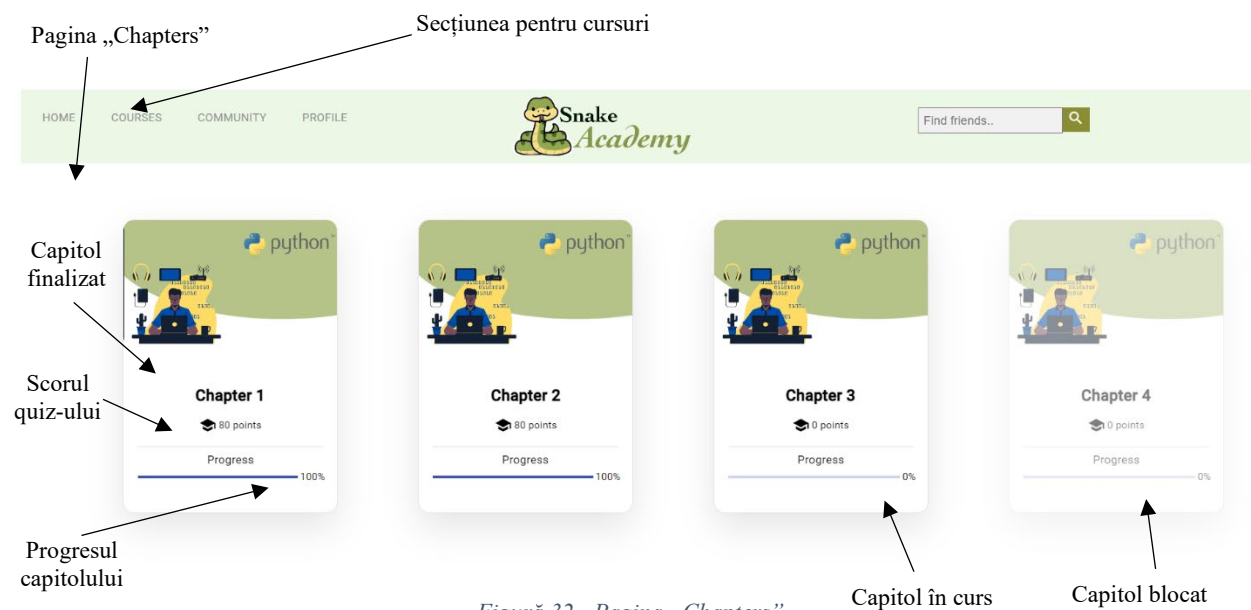
Rubrica de comentarii

Figură 31- Pagina „Forum”

## 4.2.6. Paginile „Chapters”, „Lesson” și „Quiz”

### 4.2.6.1. Capitole

Pagina „Chapters” se poate accesa din secțiunea „Courses” a meniului și este pagina principală a cursurilor. Aceasta conține o listă a tuturor capitolelor disponibile, împreună cu progresul studentului în cadrul fiecăruia. De asemenea, fiecare capitol are un punctaj atribuit, reprezentând scorul obținut de către elev la quiz-ul final. Pentru promovarea unui quiz, respectiv a unui capitol, este necesar un scor minim de 50 de puncte. Din pagina capitolelor se poate accesa orice capitol deblocat, împreună cu lecțiile aferente, un capitol fiind deblocat doar prin promovarea capitolului anterior.



Figură 32 - Pagina „Chapters”

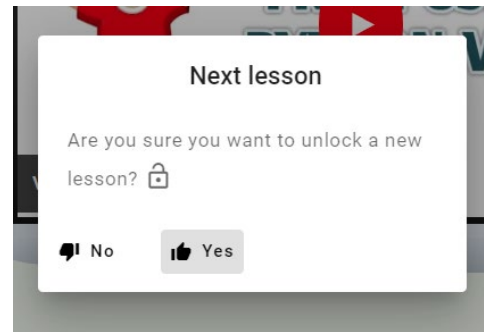
### 4.2.3.2. Lecții

Pagina „Lessons” se poate accesa din cadrul fiecărui capitol și cuprinde în partea stângă un meniu cu toate lecțiile și quiz-ul respectivului capitol, iar în partea dreaptă conținutul propriu-zis al lecției accesate.

### Meniul

Meniul ajută la navigarea între lecțiile deblocate din cadrul capitolului ales. O lecție se poate debloca dacă studentul apasă butonul „Next” după finalizarea ultimei lecții deblocate. Atunci

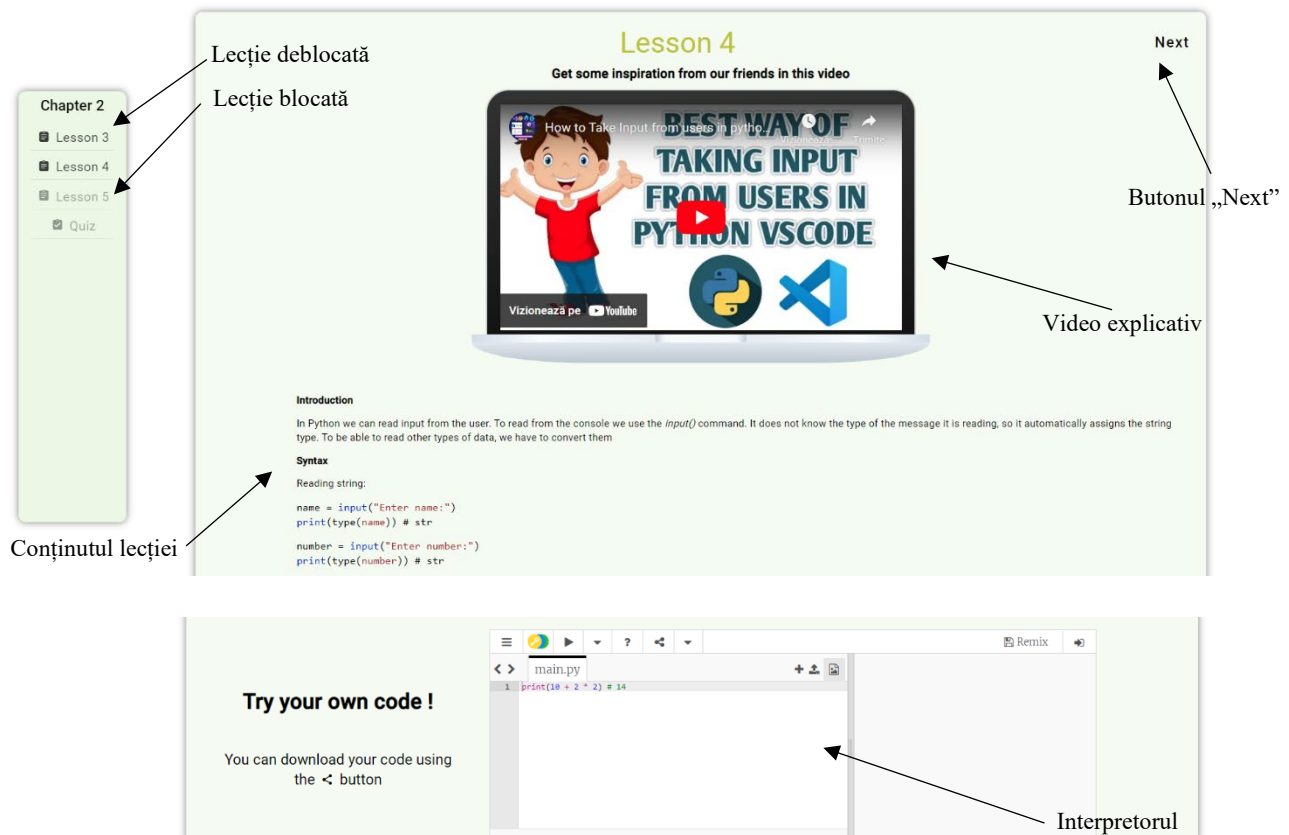
când se vrea deblocarea unei noi lecții, user-ul este întrebat dacă a terminat lecția precedentă și dorește să o deblocheze pe următoarea. Lecția curentă a elevului este reprezentată de ultima lecție deblocată, iar progresul unui capitol este calculat pe baza ultimei lecții terminate.



Figură 33 - Căsuța de dialog pentru deblocarea unei lecții

## Structura unei lecții

O lecție conține atât teorie, cât și video-uri explicative. La finalul fiecărei lecții există un interpretor care oferă studentului posibilitatea de a scrie cod python direct în pagină pentru a înțelege mai bine exemplele.



Figură 34 - Pagina „Lesson”

### 4.2.3.3. Quiz-uri

Quiz-ul este format din 5 întrebări tip grilă, cu răspuns unic, fiecare valorând 20 de puncte. Pentru a promova un quiz, implicit un capitol, este nevoie de minim 50% din punctajul total (trei

răspunsuri corecte). Următoarea lecție este deblocată doar prin promovarea quiz-ului, altfel studentul rămâne la ultima lecție din capitolul deblocat. Toate punctele adunate în urma unui quiz, fie el trecut sau nu, se adună la punctajul total al studentului.

## Quiz

This is the final quiz of the chapter. To pass this chapter, it is necessary to get at least 50 points.

### Data types in python

☐ text, float, bool

☐ int, float, str


☐ number, text, int

### How to read a message

☐ get("Message")

☐ read("Enter message")

☐ input("Enter message")



**Congratulations, you passed the quiz !**

Your score: 80 points

Done

Figură 35 - Pagina „Quiz” și rezultatul unui test

### 4.3. Perspectiva admin-ului

Deși admin-ul are acces la toate paginile destinate studentului, iar acest cont se comportă ca al unui user normal, are mai multe facilități, având acces la centrul de management. În interfața pentru administrarea platformei se poate autentifica doar un utilizator cu rol de *admin* prin intermediul formularului de logare special. Din cadrul centrului de management se poate gestiona întreaga platformă, atât partea de forum, cât și cea de cursuri.

Logout

## Management center

Chapters	See all chapters data	▼
Lessons	See all lessons data	▼
Forum	Manage posts and comments	▼

Figură 36 – Centrul de management

### 4.3.1. Cursurile

Toate cele trei elemente ale cursurilor (capitole, lecții și quiz-uri) se pot gestiona din tab-urile „Chapters” și „Lessons”. Din tab-ul „Chapters” se pot vizualiza toate capitolele existente în platformă, dar se pot crea și capitole noi.

Crearea unui capitol

Lista capitolelor existente

Chapter Id	Chapter Name	Number of Lessons	Quiz Id
1	Chapter 1	2	1
2	Chapter 2	3	2
3	Chapter 3	1	3
4	Chapter 4	2	4

Figură 37 - Tab-ul pentru capitole

Testele sunt strict legate de capitole, fiecare capitol având un quiz final care stabilește promovarea acestuia. Așadar, la crearea unui capitol nou este obligatorie și adăugarea unui quiz format din cinci întrebări de tip grilă, cu trei variante și răspuns unic. Prima lecție a unui capitol este obligatoriu să existe, deoarece un capitol nu poate fi publicat fără cursuri aferente. După adăugarea unei lecții se pot crea ulterior și altele.

Create a chapter

Create a new chapter and its quiz

The chapter cannot exist without lessons. Please don't forget to add the first lesson! You can add others later.

Chapter Name\*

Question 1

Question\* Corrent Answer\* Wrong Answer 1\* Wrong Answer 2\*

Question 2

Question\* Corrent Answer\* Wrong Answer 1\* Wrong Answer 2\*

Question 3

Question\* Corrent Answer\* Wrong Answer 1\* Wrong Answer 2\*

Question 4

Question\* Corrent Answer\* Wrong Answer 1\* Wrong Answer 2\*

Question 5

Question\* Corrent Answer\* Wrong Answer 1\* Wrong Answer 2\*

Create

Figură 38 - Crearea unui capitol nou

Din tab-ul „*Lessons*” se pot vizualiza toate lecțiile, împreună cu datele acestora. În dreptul fiecărei lecții din tabel există butonul *edit*, care permite editarea individuală a conținutului fiecărui curs. De asemenea, tot în această secțiune este posibilă crearea unor lecții noi și atribuirea acestora unor capitole deja existente. Pagina pentru editarea unei lecții este similară celei de creare, exceptând posibilitatea atribuirii lecției altui capitol sau schimbarea numărului sau numelui acesteia.

Lessons See all lessons data

Create a lesson Create a new lesson for an existing chapter

All lessons

Lesson Id	Lesson Number	Lesson Name	Chapter Id	Edit
1	1	Lesson 1	1	
5	2	Lesson 2	1	
6	3	Lesson 3	2	
7	4	Lesson 4	2	
8	5	Lesson 5	2	
9	6	Lesson 6	3	
10	7	Lesson 7	4	
11	8	Lesson 8	4	

Figură 39 - Tab-ul pentru lecții

Create a lesson Create a new lesson for an existing chapter

\*The video link must be embeddable

Lesson Number\* Lesson Name\* Embed Video Link\* Chapter\*

You can write the content of your lesson here:

**B I U** 10 pt **A**

Your lesson can contain an introduction, language syntax, examples and suggested exercises...

**Introduction**

In this lesson we will teach you about data types

- Integer
- Float
- Boolean
- String

**Syntax**

We can convert a string into an integer:

```
x = '10'
x = int(10)
```

**Exercises**

Reads 3 numbers from the console and displays their sum

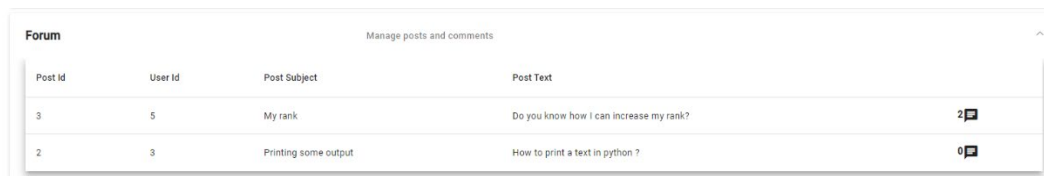
This is the content of your lesson, as it will appear on the page:

**CREATE**

Figură 40 - Crearea unei lecții noi

### 4.3.2. Comunitatea

Pentru gestionarea forumului admin-ul are acces la postări și la comentariile aferente acestora din tab-ul „*Forum*”. Toate postările pot fi vizualizate într-un tabel, iar comentariile pot fi șterse în cazul în care acestea încalcă regulile comunității.



The screenshot shows a web interface for managing forum posts and comments. At the top, there's a header with the title "Forum" and a subtitle "Manage posts and comments". Below this is a table with four columns: "Post Id", "User Id", "Post Subject", and "Post Text". The table contains two rows of data. The first row has Post Id 3, User Id 5, Post Subject "My rank", and Post Text "Do you know how I can increase my rank?". The second row has Post Id 2, User Id 3, Post Subject "Printing some output", and Post Text "How to print a text in python ?". To the right of each row, there are icons for deleting and editing the post, along with a count of comments (2 for the first post, 0 for the second).

Post Id	User Id	Post Subject	Post Text	
3	5	My rank	Do you know how I can increase my rank?	2
2	3	Printing some output	How to print a text in python ?	0

*Figură 41 - Gestionarea postărilor și a comentariilor*



## V. CONCLUZII ȘI PERSPECTIVE

### 5.1. Concluzii

Prin prezenta lucrare am expus conceptul, scopul, dar și procesul de realizare al platformei educaționale „Snake Academy”. Destinată în special tinerilor, aplicația își propune ajutarea acestora în procesul de învățare al limbajului Python, printr-o interfață atractivă, dar și intuitivă, fiind ușor de folosit. De asemenea, structura lecțiilor este un alt factor motivant pentru elevi, deoarece are atât teorie scrisă, cât și în format video.

Un alt aspect larg dezbătut a fost strategia de motivare a cursanților prin sistemul bazat pe comunitate: prietenie și competiție. Punând la dispoziție pagina de forum, aplicația încurajează elevii să se ajute reciproc, dar și să caute răspunsuri la întrebări într-un mediu accesibil. Sistemul de ranking sporește competiția și încurajează utilizatorii să nu renunțe la cursuri și să fie mai perseverenți pe termen lung. Mai mult, fiind în limba engleză, platforma își extinde comunitatea internațional, putând fi utilizată de oricine, fără constrângeri.

Din perspectivă tehnică, aplicația este modulară și ușor scalabilă, fiind o bază perfectă pentru dezvoltări ulterioare sau încorporarea ei într-un proiect mai amplu. De asemenea, pe plan local, aplicația este ușor de gestionat de către admin prin intermediul interfeței de management, cursurile putând fi actualizate în mod dinamic.

Așadar, platforma „Snake Academy” reușește să atingă obiectivele propuse, atât prin funcționalitățile ei, cât și prin interfața prietenoasă, susținând o comunitate unită, perseverentă și educată.

### 5.2. Perspective

Aplicația a fost construită pentru a permite mai multe posibile direcții de dezvoltare:

- **Înrolarea școlilor în aplicație**

Sistemul poate fi extins pentru a încadra utilizatorii în școli, iar sistemul de ranking să fie realizat intern pentru fiecare școală. Admin-ul fiecărei școli înrolate în platformă poate gestiona fiecare elev, vizualizând scorurile acestuia, dar și răspunsurile date la fiecare quiz.

- **Aplicația poate fi achiziționată de școli private sau publice.**

Aceasta este o posibilă dezvoltare mai amplă a celei anterioare, deoarece presupune oferirea accesului gestionării lecțiilor către administratorii școlilor. Fiecare școală își poate dezvolta propriile cursuri, schimbând doar conținutul lecțiilor, dar fără a se altera structura sau obiectivele aplicației.

- **Implementarea unui chat între prieteni**

Pentru utilizatorii care se urmăresc reciproc se poate pune la dispoziție un chat privat pentru a întări relațiile între studenți.

- **Salvarea în baza de date a exercițiilor rezolvate**

Pentru accesibilitate crescută, exercițiile rezolvate de studenți cu ajutorul interpretorului încorporat în fiecare lecție, se pot salva în baza de date. Acestea ar putea fi salvate pentru vizualizare sau editare.

Prin posibilele dezvoltări menționate mai sus se demonstrează ca „Snake Academy” are un potențial ridicat pentru extindere, mai ales datorită structurii sale, fiind o aplicație promițătoare pentru educația tinerilor și nu numai.

## VI. LISTĂ DE FIGURI

Figură 1- Arhitectura MVC .....	8
Figură 2 - Entitățile User și Role.....	9
Figură 3- Componenta token-ului pentru contul admin-ului .....	10
Figură 4 - Schema de funcționare a JWT în aplicație .....	11
Figură 5 - Pagina „Home” a platformei .....	12
Figură 6 - Implementarea librăriei pentru login .....	13
Figură 7- Înregistrare user utilizând Facebook .....	13
Figură 8 - Integrarea componentei în pagina HTML și editarea bării de instrumente.....	14
Figură 9 - Rich Text Editor .....	14
Figură 10 - Clasa C# „Capitol” și generarea relațiilor cu alte tabele .....	15
Figură 11 - Diagrama conceptuală a bazei de date.....	16
Figură 12 - Schemă a comunicării între layere .....	18
Figură 13 - Schemă de comparare între utilizarea și neutilizarea modelului .....	18
Figură 14 - Instanțierea clasei RepositoryWrapper în controller-ul Capitol .....	18
Figură 15 - Fluxul de funcționare al aplicației. ....	19
Figură 16 - Formularele pentru autentificare în funcție de rol .....	20
Figură 17 - Eroarea aruncată după încercarea de accesare a unei rute restricționate.....	20
Figură 18 - Logo-ul aplicației.....	21
Figură 19 - Paleta de culori .....	21
Figură 20 - Grafica utilizată pentru tema „șarpelui” .....	22
Figură 21 - Meniul aplicației.....	22
Figură 22 - Rezultatele utilizării bării de căutare .....	22
Figură 23 - Paginile „Register” și „Login” .....	23
Figură 24 - Fereastra pentru schimbarea descrierii .....	23
Figură 25 - Fereastra de notificări .....	24
Figură 26 – Pagina publică a unui prieten.....	24
Figură 27 - Fereastra de schimbare a avatarului .....	24
Figură 28 - Pagina „Profile” pentru contul de admin.....	25
Figură 29 - Pagina „Rank Tops” .....	25

Figură 30 - Tabelul pentru rank-ul „Baby Snake” .....	26
Figură 31- Pagina „Forum” .....	26
Figură 32 - Pagina „Chapters” .....	27
Figură 33 - Căsuța de dialog pentru deblocarea unei lecții .....	28
Figură 34 - Pagina „Lesson” .....	28
Figură 35 - Pagina „Quiz” și rezultatul unui test .....	29
Figură 36 – Centrul de management .....	29
Figură 37 - Tab-ul pentru capitole.....	30
Figură 38 - Crearea unui capitol nou.....	30
Figură 39 - Tab-ul pentru lecții .....	31
Figură 40 - Crearea unei lecții noi.....	31
Figură 41 - Gestionarea postărilor și a comentariilor.....	32

## VII. BIBLIOGRAFIE

1. Alexander S. Gillis - TechTarget, „REST API (RESTful API)” [Online].  
Available: <https://www.techtarget.com/searchapparchitecture/definition/RESTful-API>  
[Ultima accesare: 22 Mai 2023]
2. Angular Material, „Components” [Online].  
Available: <https://material.angular.io/components/categories> [Ultima accesare: 23 Mai 2023]
3. Angular Material, „Guides” [Online]. Available: <https://material.angular.io/guides> [Ultima accesare: 23 Mai 2023]
4. Angular, „Angular components overview” [Online].  
Available: <https://angular.io/guide/component-overview> [Ultima accesare: 24 Mai 2023]
5. Angular, „What is Angular?” [Online]. Available: <https://angular.io/guide/what-is-angular>  
[Ultima accesare: 23 Mai 2023]
6. Astari S. - Hostinger Tutorials, „What is HTML?”, 2 Mai 2023 [Online]. Available: <https://www.hostinger.com/tutorials/what-is-html> [Ultima accesare: 23 Mai 2023]
7. Baeldung, „The DTO Pattern” [Online]. Available: <https://www.baeldung.com/java-dto-pattern> [Ultima accesare: 24 Mai 2023]
8. Chinmayee Deshpande - Simplilearn, „What is Angular?: Arhitecture, Features and Advantages”, 24 Februarie 2023 [Online].  
Available: <https://www.simplilearn.com/tutorials/angular-tutorial/what-is-angular> [Ultima accesare: 23 Mai 2023]
9. Geeksforgeeks, „Business-Logic Layer” [Online].  
Available: <https://www.geeksforgeeks.org/business-logic-layer/> [Ultima accesare: 24 Mai 2023]
10. Geeksforgeeks, „Data-Access Layer” [Online].  
Available: <https://www.geeksforgeeks.org/data-access-layer/> [Ultima accesare: 24 Mai 2023]
11. JWT, „Introduction to JSON Web Tokens” [Online]. Available: <https://jwt.io/introduction>  
[Ultima accesare: 23 Mai 2023]
12. Meta, „Facebook Login” [Online].

- Available: <https://developers.facebook.com/docs/facebook-login/> [Ultima accesare: 23 Mai 2023]
13. Microsoft, „Entity Framework Core” 2021 [Online].  
Available: <https://learn.microsoft.com/en-us/ef/core> [Ultima accesare: 22 Mai 2023]
14. Microsoft, „Implementing the Repository and Unit of Work Patterns in an ASP.NET MVC Application” 7 Ianuarie 2022 [Online]. Available: <https://learn.microsoft.com/en-us/aspnet/mvc/overview/older-versions/getting-started-with-ef-5-using-mvc-4/implementing-the-repository-and-unit-of-work-patterns-in-an-asp-net-mvc-application> [Ultima accesare: 24 Mai 2023]
15. Microsoft, „What is .NET?” [Online]. Available: <https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet> [Ultima accesare: 22 Mai 2023]
16. npm, „Angular Social Login”, 8 Mai 2023 [Online].  
Available: <https://www.npmjs.com/package/@abacritt/angularx-social-login> [Ultima accesare: 23 Mai 2023]
17. Nwose L. Victor - Telerik, „Angular basics: CanActivate – Introduction to Routing Guards” [Online]. Available: <https://www.telerik.com/blogs/angular-basics-canactivate-introduction-routing-guards> 8 Martie 2022 [Ultima accesare: 24 Mai 2023]
18. Pangea, „Angular 15 features: what is new in the update?”, 30 Martie 2023 [Online].  
Available: <https://www.pangea.ai/angular-resources/angular-15-features-what-is-new-in-the-update/> [Ultima accesare: 23 Mai 2023]
19. Syncfusion, „Blazor Rich Text Editor” [Online]. Available: <https://www.syncfusion.com/blazor-components/blazor-wysiwyg-rich-text-editor> [Ultima accesare: 23 Mai 2023]
20. Tutorial Teacher, „ASP.NET MVC Arhitecture” [Online]. Available: <https://www.tutorialsteacher.com/mvc/mvc-architecture> [Ultima accesare: 22 Mai 2023]
21. Tutorial Teacher, „Learn ASP.NET Web API: What is Web API?” [Online]. Available: <https://www.tutorialsteacher.com/webapi/what-is-web-api> [Ultima accesare: 22 Mai 2023]
22. W3schools, „Python Tutorial [Online].  
Available: <https://www.w3schools.com/python/default.asp> [Ultima accesare: 25 Mai 2023]
23. Wikipedia, „TypeScript”, 18 Mai 2023 [Online]. Available: <https://en.wikipedia.org/wiki/TypeScript> [Ultima accesare: 23 Mai 2023]