CodeQuiz

Alexandru Grecu 342C3

1 Descrierea temei

Aplicatia **CodeQuiz** este o platforma web interactiva destinata testarii cunostintelor de programare si concepte tehnice IT. Scopul principal este de a oferi utilizatorilor un mediu structurat pentru a raspunde la chestionare (quiz-uri) urmarind imbunatatirea cunostintelor, a-si urmari progresul si a contribui cu continut nou.

Utilizatorii pot parcurge teste pe diverse topicuri tehnice precum Sisteme de Operare, Arhitectura Software, Baze de Date, Algoritmi si Dezvoltare Web. Platforma ofera feedback imediat dupa finalizarea fiecarui quiz, afisand scorul obtinut si permitand review-ul raspunsurilor date.

Functionalitatile principale includ:

- Autentificare si inregistrare utilizatori
- Parcurgerea quiz-urilor cu intrebari cu raspuns multiplu
- Vizualizarea istoricului si progresului personal
- Contribuirea cu intrebari noi (supuse aprobarii administratorului)
- Oferirea de review-uri pentru intrebari
- Panou de administrare cu rapoarte grafice si statistici

2 Descrierea bazei de date

2.1 Diagrama bazei de date

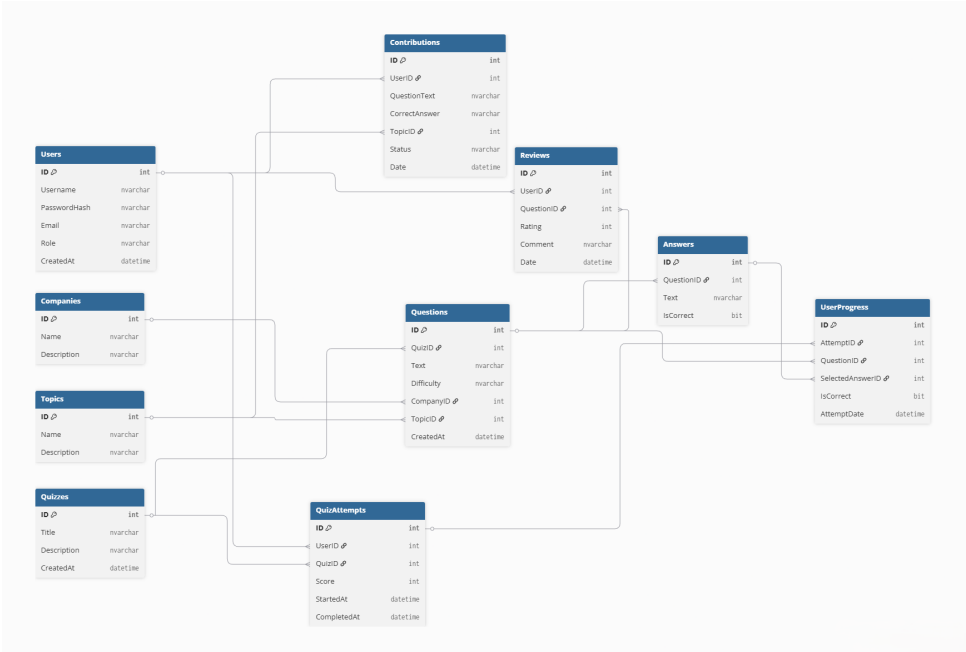


Figure 1: Diagrama Bazei de Date

2.2 Structura tabelelor

Baza de date este compusa din **10 tabele** principale:

Tabel	Descriere	Coloane principale
Users	Utilizatorii platformei	ID, Username, PasswordHash, Email, Role, CreatedAt
Companies	Companiile asociate intrebarilor	ID, Name, Description
Topics	Subiectele testelor	ID, Name, Description
Quizzes	Quiz-urile disponibile	ID, Title, Description, CreatedAt
Questions	Intrebarile din quiz-uri	ID, QuizID, Text, Difficulty, CompanyID, TopicID
Answers	Variantele de raspuns	ID, QuestionID, Text, IsCorrect
QuizAttempts	Sesiunile de testare	ID, UserID, QuizID, Score, StartedAt, CompletedAt
UserProgress	Raspunsurile date	ID, AttemptID, QuestionID, SelectedAnswerID, IsCorrect
Reviews	Feedback utilizatori	ID, UserID, QuestionID, Rating, Comment, Date
Contributions	Intrebari propuse	ID, UserID, QuestionText, CorrectAnswer, Status

Table 1: Structura tabelelor din baza de date

2.3 Descrierea constrangerilor de integritate

Chei Primare (PK): Fiecare tabel are coloana ID de tip INT IDENTITY(1,1) PRIMARY KEY.
Chei Externe (FK):

- Questions.QuizID $\rightarrow$ Quizzes.ID
- Questions.CompanyID $\rightarrow$ Companies.ID
- Questions.TopicID $\rightarrow$ Topics.ID
- Answers.QuestionID $\rightarrow$ Questions.ID (ON DELETE CASCADE)
- QuizAttempts.UserID $\rightarrow$ Users.ID
- QuizAttempts.QuizID $\rightarrow$ Quizzes.ID
- UserProgress.AttemptID $\rightarrow$ QuizAttempts.ID (ON DELETE CASCADE)
- Reviews.UserID $\rightarrow$ Users.ID
- Contributions.UserID $\rightarrow$ Users.ID

Constrangeri CHECK:

- Users.Role IN ('User', 'Admin')
- Questions.Difficulty IN ('Easy', 'Medium', 'Hard')
- Reviews.Rating $>=$ 1 AND Rating $<=$ 5
- Contributions.Status IN ('Pending', 'Approved', 'Rejected')

Unicitate: Users.Username si Users.Email sunt UNIQUE.

2.4 Descrierea procedurilor si functiilor

Aplicatia utilizeaza exclusiv **proceduri stocate** pentru toate operatiunile cu baza de date:
Gestionare Utilizatori: `RegisterUser`, `LoginUser`, `GetUserByID`
Flux Quiz: `GetQuizzes`, `GetQuestionsByQuiz`, `GetAnswersByQuestion`, `StartQuizAttempt`, `RecordAttempt`, `CompleteQuizAttempt`, `GetQuizHistory`, `GetPendingQuizzes`, `GetQuizAttemptDetails`
Rapoarte Admin (Complexitate 4-7):

- `GetCompanyStats` (Complexitate 4) - Top companii dupa rata de esec
- `GetTopicStats` (Complexitate 6) - Rata de succes pe topicuri
- `GetUserLeaderboard` (Complexitate 7) - Clasament utilizatori dupa engagement
- `GetTopicDifficultyAnalysis` (Complexitate 7) - Analiza succes pe dificultate

Contributii si Review-uri: `AddContribution`, `GetPendingContributions`, `ApproveContribution` (foloseste TRANSACTION), `RejectContribution`, `AddReview`, `GetAllReviews`

3 Descrierea aplicatiei

3.1 Structura claselor

Aplicatia este dezvoltata in **Python** cu **Flask**. Clasa principala `MSSQLConnection` gestioneaza conexiunea la baza de date prin metodele `openConnection()`, `closeConnection()` si `execute_query()`. Modulul `app.py` defineste decoratorii `@login_required` si `@admin_required`, functia `create_plot()` pentru grafice, si 19 route-uri Flask pentru functionalitatile aplicatiei.

3.2 Diagrama de clase

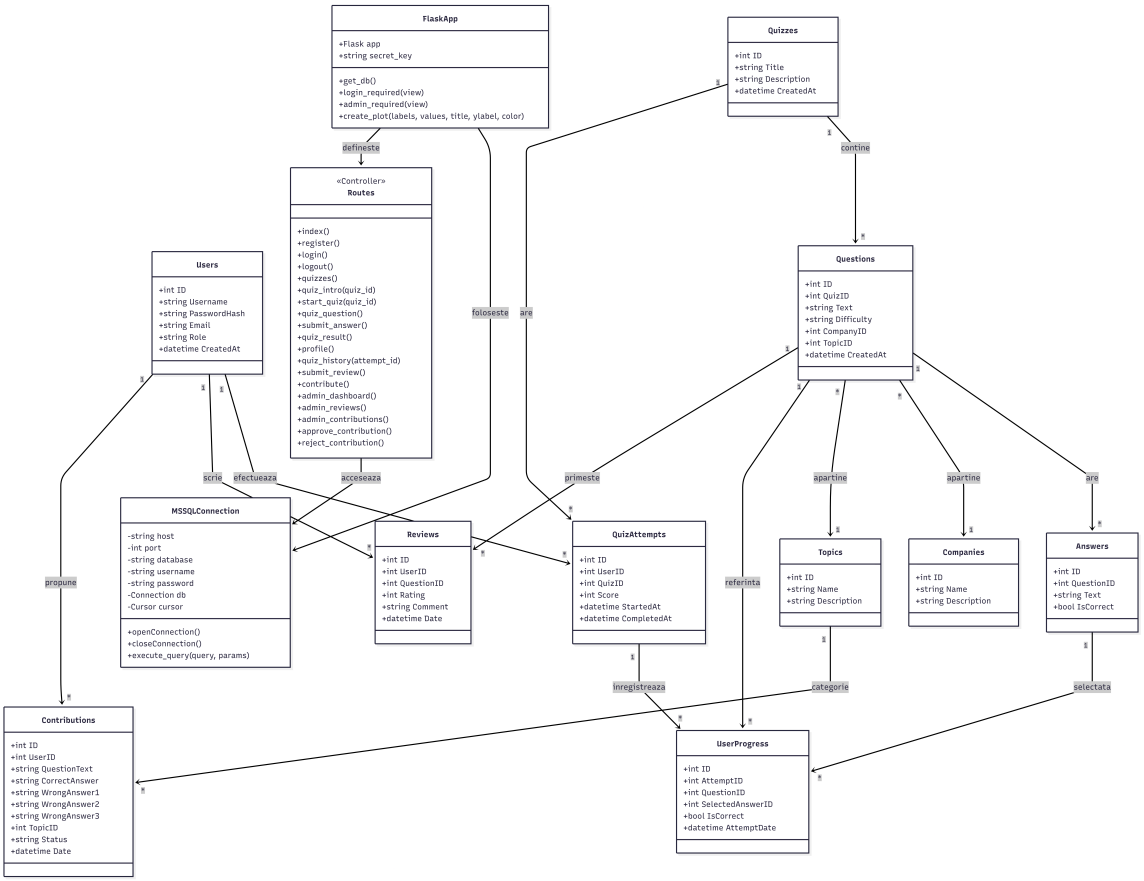


Figure 2: Diagrama de Clase

3.3 Diagrama de stari si fluxul de lucru

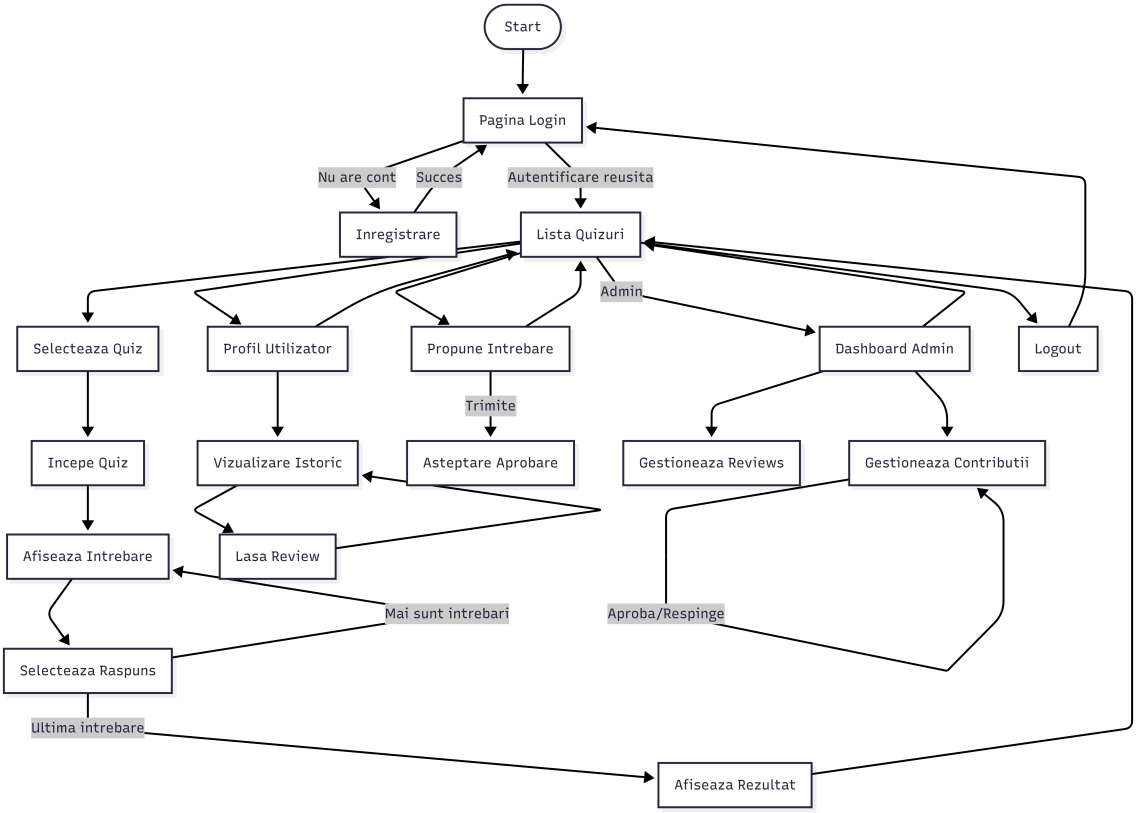
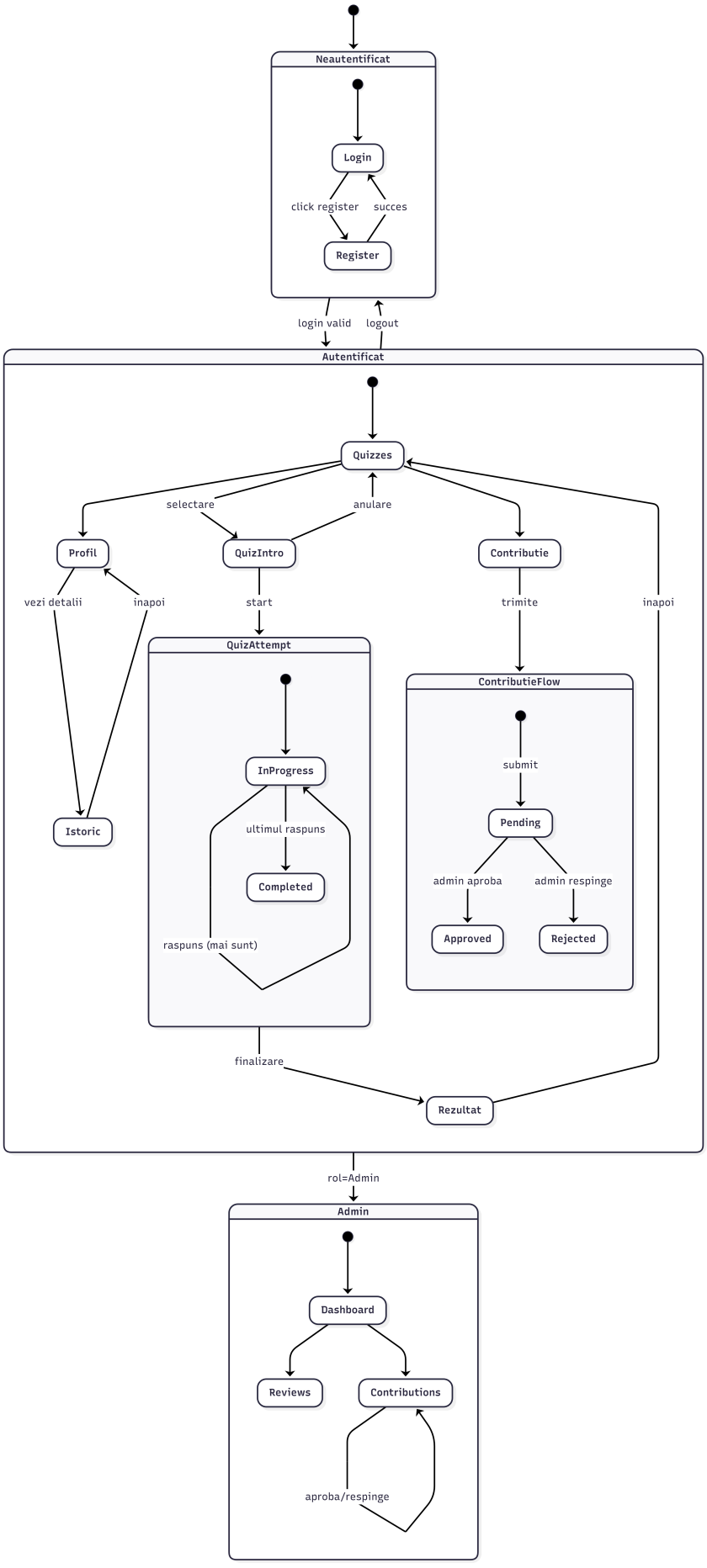


Figure 3: Fluxul de Lucru General


Neautentificat
Login
click register
succes
Register
login valid
logout
Autentificat
Quizzes
selectare
anulare
Profil
QuizIntro
Contributie
vezi detalii
inapoi
start
trimite
inapoi
QuizAttempt
InProgress
ultimul raspuns
Completed
raspuns (mai sunt)
ContributieFlow
submit
Pending
admin aproba
admin respinge
Approved
Rejected
Istoric
finalizare
Rezultat
rol=Admin
Admin
Dashboard
Reviews
Contributions
aproba/respinge

3.4 Prezentarea modului in care se face conexiunea cu baza de date

Conexiunea se realizeaza folosind driver-ul **ODBC pentru SQL Server** prin biblioteca pyodbc.
String de conexiune:

```
conn_str = (
    "DRIVER={SQL Server};"
    "SERVER=localhost,1433;"
    "DATABASE=master;"
    "UID=SA;"
    "PWD=password;"
    "TrustServerCertificate=yes;"
)
```

Modelul de utilizare (Open-Execute-Close):

```
db = get_db()
db.execute_query("EXEC NumeProcedura ?", (param1,))
result = db.cursor.fetchall()
db.closeConnection()
```

3.5 Capturi de ecran pentru interfete si rapoarte

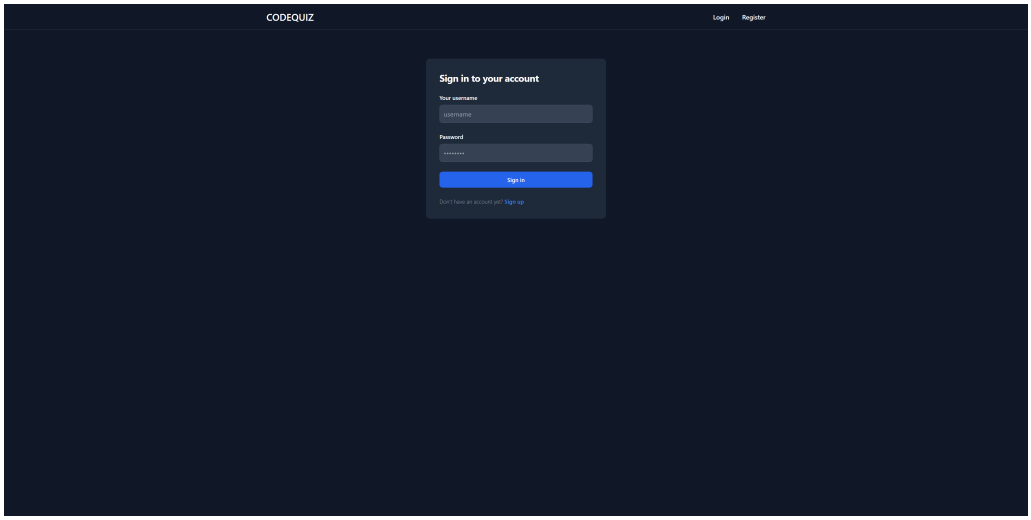


Figure 4: Login Screen

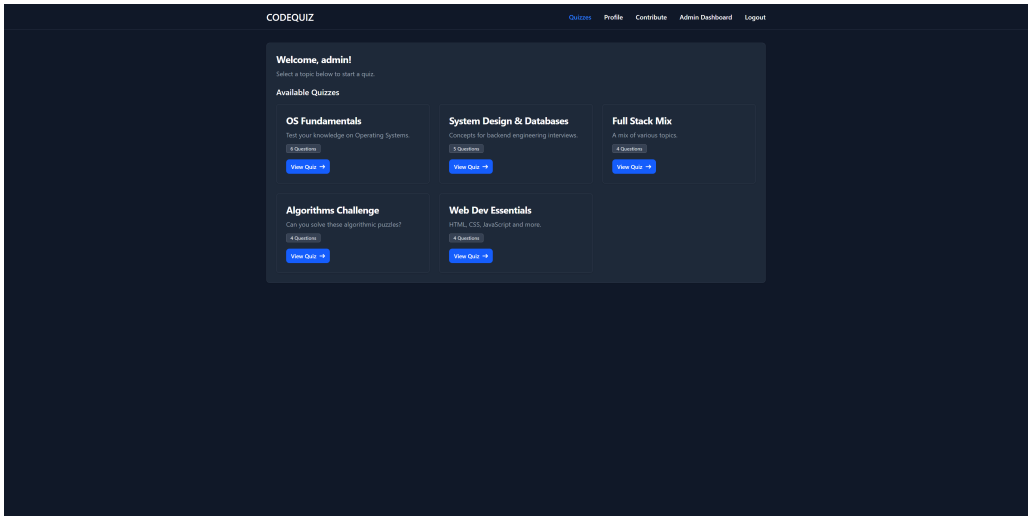


Figure 5: Dashboard Quizzes

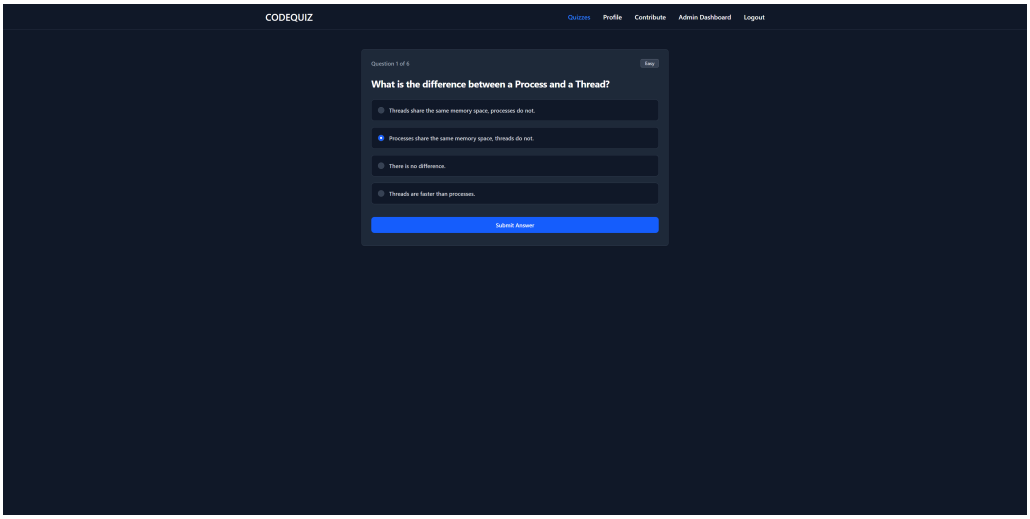


Figure 6: Quiz Question

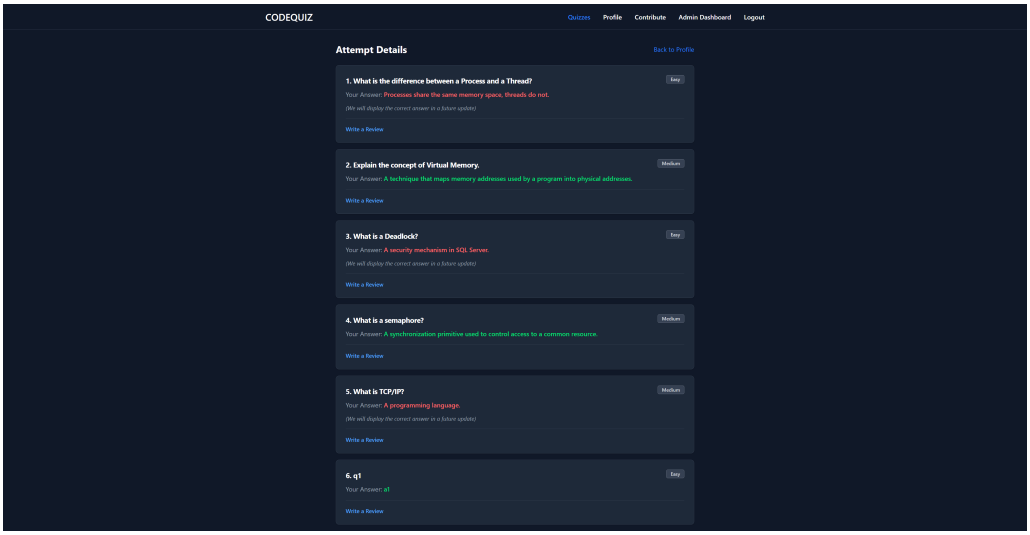


Figure 7: Quiz Result

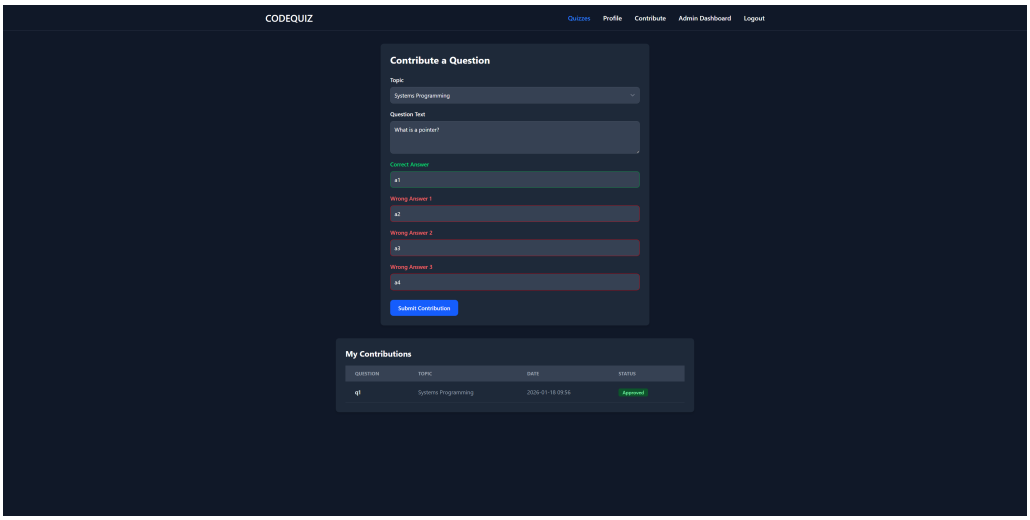


Figure 8: Contribute Page

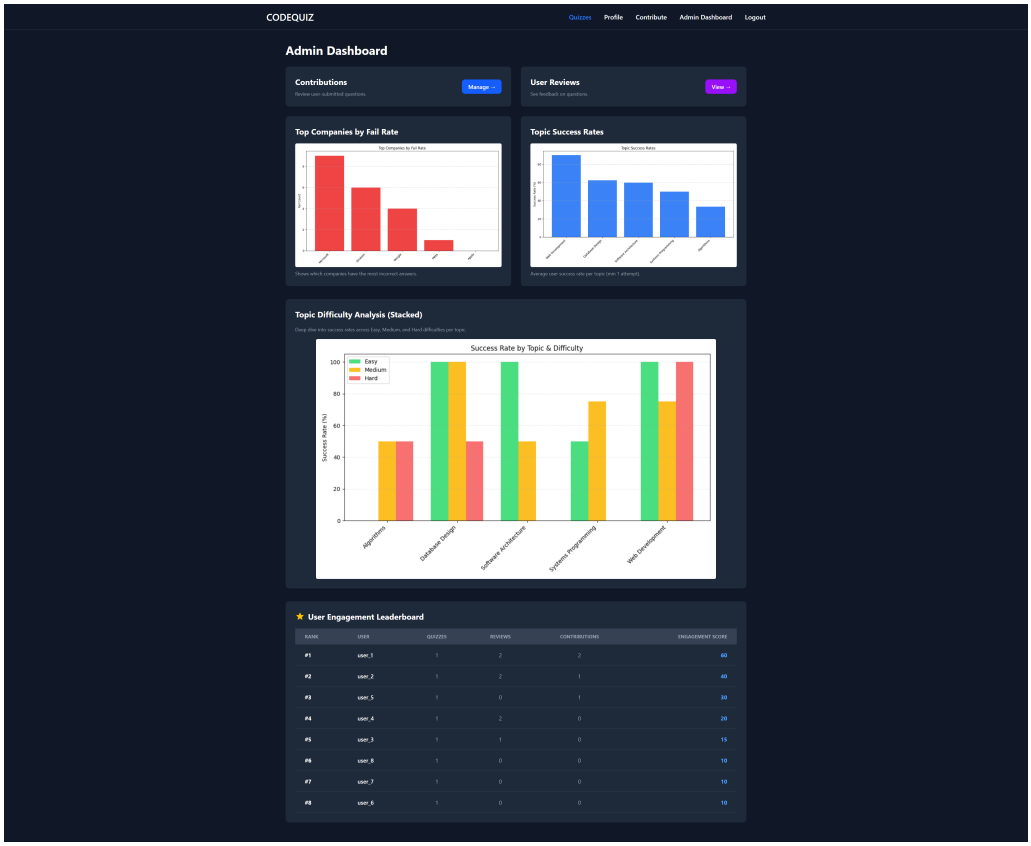


Figure 9: Admin Dashboard

4 Concluzii

Proiectul CodeQuiz a permis consolidarea cunostintelor de baze de date relationale si dezvoltare web. Principalele realizari:

- Implementarea unei baze de date normalizate cu 10 tabele si relatii complexe
- Utilizarea exclusiva a procedurilor stocate pentru acces la date
- Generarea de grafice server-side folosind matplotlib
- Implementarea unui sistem de contributii cu flux de aprobare
- Securizarea aplicatiei prin hash-uri de parole si decoratori de autorizare

5 Bibliografie

1. Documentatie Flask: `https://flask.palletsprojects.com/en/stable/`
2. Laboratorul 07 - SQL Server : Interogari, Variabile si XML: `Laboratorul07-SQLServer:Interogari,VariabilesiXML`
3. Laboratorul 08 - SQLServer: DML, Cursori si Programabilitate: `https://ocw.cs.pub.ro/courses/bdd/laboratoare/08`
4. Laboratorul 09 - Conectare la o Baza de Date: `https://ocw.cs.pub.ro/courses/bdd/laboratoare/09`
5. Matplotlib User Guide: `https://matplotlib.org/stable/users/index.html`
6. Tailwind CSS Documentation: `https://tailwindcss.com/docs`
7. Flowbite - Tailwind CSS component library: `https://flowbite.com/docs/getting-started/introduction/`