

Platformă educațională de tip Trivia pentru evaluarea și îmbunătățirea cunoștințelor de specialitate

Mihai-Cezar Viziteu

ACADEMIA DE STUDII ECONOMICE DIN BUCUREȘTI
Facultatea de Contabilitate și Informatică de Gestiune
Coordonator: Prof. univ. dr. Florin Mihai

Rezumat: Într-o lume din ce în ce mai tehnologizată este important ca educația să se adapteze. Prezenta lucrare explorează aspectele teoretice ale gamificării și aplicarea acestora în contextul educațional. A fost implementată o platformă pentru evaluarea și îmbunătățirea cunoștințelor de specialitate, folosind tehnologii moderne precum limbajul Typescript, framework-ul NextJS, biblioteca software React. Cu ajutorul acestora a fost alcătuită o platformă cu rol în gestionarea întrebărilor și gruparea acestora în teste (en. quizzes). Folosind Socket.IO, întrebările sunt transmise către utilizatori. Sunt colectate date analitice pentru analiza ulterioară. Pe baza răspunsurilor date de către utilizatori se pot acorda puncte. Platforma își propune să ofere cadrelor didactice instrumentele necesare pentru realizarea evaluării, având în vedere caracteristicile gamificării.

Cuvinte-cheie: gamificare, platformă educațională, motivație intrinsecă și extrinsecă, socket.io, postgresql, nextjs

Introducere

Într-o lume aflată în continuă schimbare, este necesar ca educația să se adapteze. Aflându-ne într-o eră din ce în ce mai conectată și tehnologizată, oamenii obișnuiesc să își petreacă o perioadă semnificativă din timpul lor pe dispozitive mobile. Nemaiputând preveni prezența dispozitivelor mobile (în special telefoane) în sala de clasă (fiind o componentă indispensabilă a vieților noastre de zi cu zi), acestea pot deveni o resursă extrem de valoroasă în procesul educațional, atât ca modalitate de obținere de informații, cât și pentru realizarea evaluării cunoștințelor.

Deși există deja exemple de succes precum *kahoot* și *quizizz*, modalitatea acestora de funcționare rămâne un mister – codul sursă al acestora nefiind disponibil către utilizatori. Dezvoltarea proprie a unei asemenea platforme aduce după sine ocazia de a implementa și testa noi moduri de joc care nu au fost explorate până acum în mediul educațional. Un astfel de exemplu este jocul *Triviador*, care a reprezentat o mare sursă de inspirație pentru această platformă (a se vedea împărțirea jucătorilor în grupe de câte trei și paginile responsabile cu realizarea și editarea hărților de joc).

Prezenta lucrare încearcă să exploreze utilizarea gamificării ca factor motivator în vederea utilizării unei platforme educaționale online pentru evaluare. O recomandare referitoare la utilizarea platformei care urmează să fie descrisă în această lucrare este folosirea acesteia ca modalitate suplimentară de evaluare, respectiv materializarea recompenselor sub formă de punctaj bonus, utilizarea acesteia nefiind o necesitate pentru obținerea unui rezultat maxim. Platforma poate reprezenta o unealtă pentru motivarea studenților și recompensarea acestora în cadrul activității de predare-învățare-evaluare.

1. Gamificare

Gamificarea este procesul de îmbunătățire a sistemelor, serviciilor, organizațiilor și activităților prin integrarea elementelor și principiilor de proiectare a jocurilor în contexte non-joc (Hamari, 2019). Scopul gamificării este de a crește implicarea, motivația, competiția și participarea utilizatorilor prin utilizarea mecanicilor de joc, cum ar fi punctele, medaliile, clasamentele și alte tipuri de recompense (Robson et al., 2015).

Werbach și Hunter (2012) dezvoltă cadrul MDC (Mecanici, Dinamici, Componente), care reprezintă rezultatele obținute de către jucător în urma acțiunii mecanicilor și dinamicilor, respectiv elemente precum puncte, insigne, bunuri digitale (de exemplu, obiectele dintr-un inventar). Robson et al. (2015) înlocuiește componentele cu emoții și dezvoltă cadrul MDE (Mecanici, Dinamici, Emoții) al principiilor gamificării. În acest context, mecanicile (en. *mechanics*) se referă la „deciziile pe care proiectanții – cei care vor să gamifice un context non-joc – le iau pentru a specifica scopul, regulile, contextul, tipurile de interacționări și limitele situației ce urmează să fie gamificate”. Mecanicile trebuie să fie cunoscute de dinainte și să rămână constante pe parcursul desfășurării activității gamificate (Robson et al., 2015). Dinamicile (en. *dynamics*) sunt „tipurile de comportament ale jucătorilor care apar în urma participării la activitate. În mod contrar față de mecanici, care sunt setate de către proiectant, dinamicile sunt produse de modul în care jucătorii urmează mecanicile stabilite” (Robson et al., 2015). Emoțiile (en. *emotions*) sunt „stări mentale afective și reacții evocate în rândul jucătorilor când participă la o activitate gamificată. Emoțiile sunt produsul modului în care jucătorii urmează mecanicile și generează dinamicile. La fel ca în cazul jocurilor, activitățile gamificate trebuie să fie atractive și distractive” (Robson et al., 2015).

Chou (2015) are o abordare ușor diferită, creând Octalysis - un cadru bazat pe 8 impulsuri principale (en. *core drives*) ale motivației umane, așezate în formă de octogon.



Figura 1. Cele 8 impulsuri de bază ale cadrului Octalysis.
Sursa: Chou (2015)

Acesta împarte cele 8 impulsuri după două clasificări: (1) intrinsec – extrinsec: impulsurile aflate în partea dreaptă a octogonului (empowerment, social influence, unpredictability) stimulează motivația intrinsecă, în timp ce impulsurile aflate în partea stângă

(accomplishment, ownership, scarcity) stimulează motivația extrinsecă și (2) white hat – black hat: impulsurile din partea de jos a octogonului (scarcity, avoidance, unpredictability) sunt „black hat”, utile pentru a crea un sens de urgență pe termen scurt și a face utilizatorii să fie proactivi și concentrați. Impulsurile aflate în partea de sus (accomplishment, meaning, empowerment) au rolul de a oferi sentimentul de bunăstare, scop, însemnătate și fericire.

1.1. Caracteristici ale gamificării

Conform literaturii de specialitate, se identifică următoarele caracteristici ale unei activități gamificate: (1) punctele, (2) insigne, (3) clasamentele, (4) graficele de performanță, (5) poveștile cu semnificație, (6) avatarele și (7) coechipierii.

- Punctele (en. *points*) sunt elementul de bază al gamificării. Acestea sunt acordate în principal ca rezultat al îndeplinirii cu succes a activităților specificate în mediul gamificat și au rolul de a reprezenta numeric progresul jucătorului. Unul dintre cele mai importante scopuri al acestora este de a oferi feedback, permițând măsurarea și recompensarea imediată a comportamentului jucătorului (Werbach & Hunter, 2012);
- Insigne (en. *badges*) sunt reprezentarea vizuală a realizărilor și pot fi obținute și colectate în cadrul mediului gamificat. Acestea confirmă îndeplinirea scopurilor. Pot avea rolul de a stabili un scop, dacă condițiile pentru obținerea lor sunt cunoscute. Pot, de asemenea, să fie un simbol de statut. În același mod ca punctele, insigne oferă feedback (Werbach & Hunter, 2012);
- Clasamentele (en. *leaderboards*) ordonează jucătorii în funcție de un indicator de succes, în acest mod determinând care jucător se descurcă cel mai bine la o activitate. (Werbach & Hunter, 2012). Conform autorilor Werbach și Hunter (2015), clasamentele sunt motivatori dacă se poate avansa relativ ușor, dar demotivatori în momentul în care jucătorii se află în partea de jos a clasamentului. Competiția cauzată de clasamente poate crea presiune socială pentru a crește nivelul de angajament al jucătorului și poate avea un efect constructiv asupra participării și învățării. Efectele pozitive ale competiției sunt mai probabile dacă competitorii sunt la un nivel aproximativ egal de performanță;
- Grafice de performanță (en. *performance graphs*) au rolul de a oferi informații cu privire la evoluția în timp a unui jucător, comparativ cu performanța înregistrată anterior;
- Povești cu semnificație (en. *meaningful stories*) sunt elemente de proiectare a jocurilor care nu afectează direct performanța jucătorilor. Contextul narativ al unei activități gamificate oferă acestuia însemnătate mai departe decât colectarea punctelor și a insinelor;
- Avatarele (en. *avatars*) sunt reprezentarea vizuală a jucătorilor în mediul gamificat. De obicei, acestea sunt alese sau create de către jucător. Principalul rol este să fie distincte vizual, astfel încât să diferențieze un jucător de alt jucător. Avatarele permit jucătorilor să își creeze o identitate și să obțină apartenența la comunitate;
- Coechipierii (en. *teammates*) fie că sunt alți jucători reali sau nu (roboți), pot crea conflict, competiție sau colaborare.

2. Metodologie

În această cercetare, s-a realizat un studiu de tip aplicativ. S-a propus dezvoltarea și validarea unei platforme educaționale de tip Trivia, care să ajute la evaluarea și îmbunătățirea cunoștințelor de specialitate în contabilitate, analiză economică și informatică de gestiune.

Componenta cantitativă s-a axat pe măsurarea performanțelor utilizatorilor, iar componenta calitativă a analizat modul în care aceștia interacționează cu platforma. În cadrul lucrării, s-au propus următoarele obiective:

- Dezvoltarea unei platforme software funcționale pentru testarea cunoștințelor de specialitate;
- Evaluarea eficienței platformei în procesul de învățare;
- Analiza modului în care influențează sesiunile de Trivia consolidarea cunoștințelor utilizatorilor.

Pentru atingerea obiectivelor cercetării, s-au folosit mai multe metode: (1) pentru proiectare și implementare: platforma utilizează o arhitectură clasică, alcătuită din backend, frontend și o bază de date; (2) experiment: s-a testat platforma pe un grup țintă de utilizatori și s-au urmărit rezultatele obținute; (3) observație: s-a monitorizat comportamentul utilizatorilor în timpul sesiunilor de Trivia și (4) analiză statistică: s-au examinat datele colectate pentru a urmări evoluția performanței participanților după utilizarea platformei.

2.1. Tipuri de itemi

Fiind un joc în timp real, sunt necesare tipuri de întrebări (itemi) care pot fi evaluate instant. Deși este promițătoare evoluția modelelor lingvistice de mari dimensiuni (en. *large language models*), acestea nu dețin cunoștințele de specialitate necesare pentru a evalua orice tip de exercițiu. Fiind o platformă de tip Trivia, este important ca răspunsul pentru fiecare tip de item să poată fi determinat într-un timp cât mai scurt. Astfel, platforma implementează o mare parte a itemilor obiectivi și semi-obiectivi, itemii subiectivi neputând fi evaluați în mod automat.

Pe baza itemilor aferenți testului docimologic, au fost stabilite următoarele tipuri de întrebări în cadrul platformei:

Tabel 1. Corespondența itemilor între testul docimologic și platformă

Item din cadrul testului docimologic	Item din cadrul platformei
Alegere duală	True/False
Alegere multiplă; cu un răspuns corect	MultipleChoice
Asociere simplă; asociere cauză-efect	Correspondence
Ordonare	Ordering
Cu răspuns scurt	Nearby
Propoziții lacunare	IncompleteText

2.2. Ce implementează în mod concret platforma din caracteristicile gamificării?

Dintre caracteristicile gamificării enumerate mai sus, platforma implementează în momentul de față următoarele:

- Puncte – jucătorul acumulează puncte de experiență (XP) pe baza răspunsurilor oferite;
- Clasamente – clasamentul de pe pagina Home cu numărul de răspunsuri corecte din ultima săptămână;
- Grafice de performanță – histograma timpilor de răspuns pentru o întrebare;
- Avatare – fiecare utilizator are un avatar generat pe baza numelui de utilizator, care poartă denumirea de „Identicon”;
- Coechipieri – o sesiune de joc necesită mai mulți jucători.

2.3. Impactul pedagogic al gamificării

Platforma urmărește să implementeze concepte motivaționale, care să se alinieze teoriei autodeterminării a lui Ryan și Deci (2000), care are în vedere: (1) autonomia – libertatea de a alege modalitatea de realizare a sarcinii. O modalitate de creștere a autonomiei ar fi posibilitatea de a exersa pe teste/jocuri. Faptul că un joc duce la obținerea unui punctaj suplimentar, dar nu este obligatoriu pentru obținerea notei maxime, poate reprezenta un alt factor care crește autonomia; (2) competență – este necesar ca jucătorii să simtă că progresează odată cu desfășurarea activității de evaluare gamificată. Feedbackul este obținut prin scorul primit în timpul jocului și cimentat prin punctele de experiență acordate; și (3) relaționare – nevoia de a aparține unui grup - studentul face parte dintr-o grupă la facultate, o comunitate online, învață în același timp cu alți studenți etc.

Teoria se bazează pe două aspecte importante ale motivației: (1) intrinsecă – utilizatorii învață de plăcere (practic se joacă), nu sunt obligați, li se stârnește interesul. Conform Ryan și Deci (2000), este imposibil ca o persoană să motiveze intrinsec o altă persoană, dar aceasta îi poate crea un mediu propice pentru satisfacerea celor trei nevoi: autonomie, competență și relaționare; și (2) extrinsecă – utilizatorii învață deoarece vor primi recompense (puncte). Pe baza răspunsurilor date în timpul activității de evaluare, utilizatorii acumulează puncte de experiență. Acestea pot fi cuantificate direct în puncte suplimentare la seminar. Este important ca platforma să ofere un bonus, nu să fie o necesitate pentru obținerea notei maxime, pentru a minimiza influența motivației extrinseci.

Adițional, platforma poate fi utilizată pentru a oferi cadrelor didactice o vedere de ansamblu asupra atingerii obiectivelor educaționale. Aceasta ar trebui să ofere instrumente pentru atingerea primelor 5 niveluri din Taxonomia lui Bloom (reamintire, înțelegere, aplicare, analiză, evaluare).

3. Platforma – Backend

Platforma este alcătuită din 3 componente și anume: componenta NextJS (formată din backend și frontend) și Custom Server. Backend-ul din NextJS și Custom Server sunt ambele servere web și sunt „servite” de la aceeași adresă. Componenta backend din NextJS are rolul de a „servi” componenta frontend din NextJS. De asemenea, include funcționalitatea unui server HTTP, care se bazează pe conceptul de resurse și care sunt gestionate cu verbe precum GET, POST, PUT, DELETE. Resursele sunt entitățile gestionate de către această platformă: Questions, Quizzes, Users. Fiecărei resurse i se aplică operațiile CRUD (Create, Read, Update, Delete). Componenta Custom Server are de asemenea rolul de server HTTP și gestionează crearea de Lobbies. Adițional, gestionează și menține conexiunile Socket.IO, lucru pe care NextJS nu poate să îl facă.

3.1. Tehnologii utilizate

Limbajul TypeScript este un limbaj de programare dezvoltat de Microsoft, care adaugă *static types* și adnotări pentru tipul de date (en. *type annotations*) limbajului JavaScript. Acesta este transformat în JavaScript în urma unui proces numit „compilare”, și poate rula atât pe server, cât și în browser (client).

NextJS este un cadru de dezvoltare (en. *framework*) dezvoltat de către compania Vercel, care oferă componentele necesare pentru crearea de aplicații web full-stack (Vercel, 2025a). Acesta permite utilizarea bibliotecii software React, împreună cu alte biblioteci software instalate de către utilizator. În acest context, NextJS are și rol de bundler (prin Turbopack, care este integrat), facilitând procesul de dezvoltare, respectiv punere în funcțiune a platformei.

Socket.IO este o bibliotecă software pentru comunicare bidirecțională, în timp real, între serverul web și client. Deoarece NextJS este în principal un framework serverless, acesta nu permite menținerea îndelungată a conexiunilor. Astfel, a fost necesară implementarea unui Custom Server (Vercel, 2025b). Acesta are rolul de a gestiona lobby-urile, camerele (en. *rooms*), și de a menține conexiunile Socket.IO. De asemenea, este locul în care este rulată logica jocului.

Conexiunea cu baza de date se realizează folosind Drizzle ORM (Object Relational Mapper), care este o bibliotecă software responsabilă cu definirea și gestionarea schemei bazei de date. Aceasta permite accesarea datelor atât într-un mod similar cu SQL, cât și în mod relațional (Drizzle Team, n. d.). Pe baza schemei declarate în fișierul „*schema.ts*”, se pot deduce tipuri de date corespunzătoare structurii bazei de date. Drizzle oferă posibilitatea utilizării „*drizzle-kit*” pentru a transforma *schema.ts* într-un set de instrucțiuni valide SQL și a-l rula pe baza de date. Pentru a facilita transferul și validarea datelor, fiecărui concept relevant îi corespund mai multe tipuri de date, în funcție de partea din platformă în care este utilizat. Această abordare permite evitarea scurgerilor de date sensibile, precum hash-ul parolei unui utilizator.

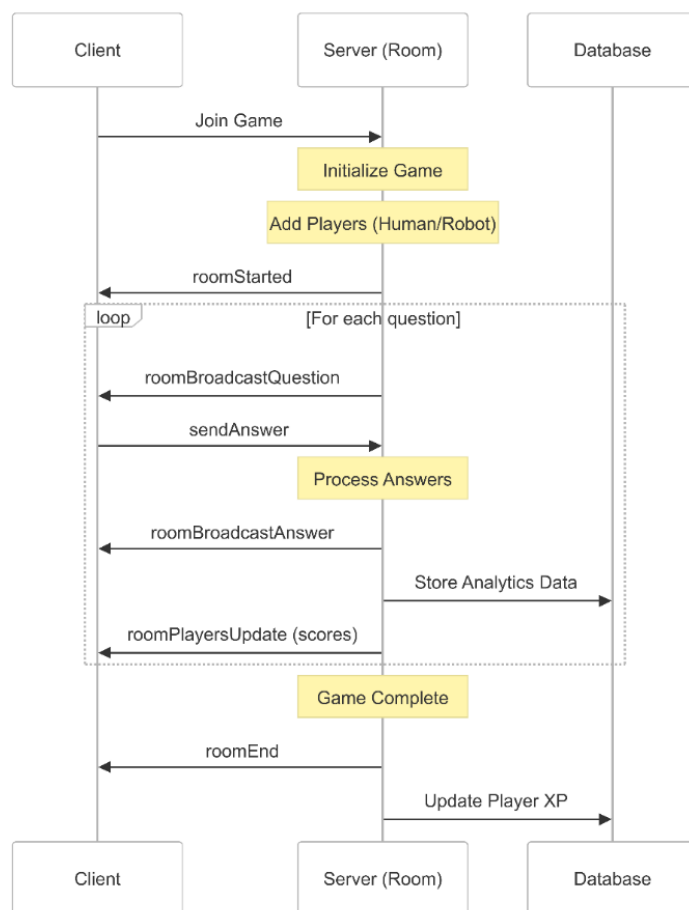


Figura 2. Diagrama sequence pentru comunicarea client – server – baza de date.

Sursa: prelucrare proprie

3.2. Autentificare și autorizare

Sistemul de autentificare și autorizare este realizat cu ajutorul bibliotecii software lucia v3 (Pilcrow, n.d. a). Aceasta are rolul de a abstractiza complexitatea din spatele gestionării sesiunilor, respectiv de a controla ce utilizatori au acces la anumite resurse. Sistemul de autentificare se bazează pe două concepte – session și user.

Un utilizator (en. *user*) este un participant pe platformă – un student, un cadru didactic sau un administrator etc. Platforma stochează despre un utilizator un identificator (ID) unic, un username (nume de utilizator), hash-ul parolei acestuia, rolul, numele întreg, adresa de email, numărul de puncte de experiență deținute, și data și ora la care a fost creat contul. Rolul unui utilizator poate fi administrator, profesor sau student. Administratorul are control deplin al platformei, profesorul are permisiuni asupra întrebărilor și testelor, iar studentul poate doar să își vadă profilul și să participe la joc.

Sesiunile (en. *session*) permit platformei să identifice cererile (en. *requests*) făcute de către utilizatorii autentificați (Pilcrow, n.d. b). O sesiune este formată dintr-un ID unic, ID-ul utilizatorului care deține sesiunea și data expirării sesiunii. Adicional, platforma colectează adresa IP, navigatorul (en. *browser*) și sistemul de operare de pe care a fost generată sesiunea. Unui utilizator îi corespund una sau mai multe sesiuni, pentru fiecare dispozitiv de pe care s-a autentificat.

Identificatorul sesiunii este stocat într-un cookie HTTP-Only denumit „auth_session”, care poate fi accesat de către server și vizualizat în developer tools, dar nu poate fi modificat direct de codul care rulează în browser. Un utilizator cu acest cookie este considerat autentificat, cât timp sesiunea acestuia este validă. O sesiune este considerată validă dacă nu a expirat încă și dacă îi corespunde un utilizator.

Pentru a înregistra un utilizator nou, acesta trebuie să completeze formularul de pe pagina „/auth/signup”. Trimiterea formularului face un request de tip POST către „/api/auth/signup”, unde datele trimise de către utilizator sunt validate, folosind zod. Se verifică dacă exista deja un utilizator cu același username sau cu aceeași adresă de email, apoi parola utilizatorului este „hashed” (procesată folosind un algoritm criptografic ireversibil – Argon2id Internet Engineering Task Force, 2021). Apoi, este generat un userID unic, sunt preluate datele referitoare la adresa IP, browser și sistemul de operare, apoi valorile sunt salvate în baza de date. Dacă oricare dintre acești pași eșuează, procesul de înregistrare este anulat, iar utilizatorului îi este prezentat un mesaj de eroare. În urma înregistrării, este generată o sesiune, lucru care autentifică utilizatorul nou înregistrat.

Pentru autentificarea de pe alte dispozitive, utilizatorul trebuie să completeze formularul de autentificare de pe pagina „/auth/login” cu adresa de email asociată contului și parola. Apăsarea butonului „Login” trimite un POST request către „/api/auth/login”, unde datele din formular sunt verificate. Se caută în baza de date un utilizator cu adresa de email trimisă. Dacă acesta nu este găsit, este afișată o eroare privind faptul că numele de utilizator sau parola sunt invalide. Altfel, parola trimisă prin formular este procesată și comparată folosind funcții specifice Argon2id. Dacă parola este corectă, sunt preluate datele referitoare la adresa IP, browser și sistemul de operare și este generată o nouă sesiune.

Pentru a obține informații despre utilizatorul autentificat, lucia v3 pune la dispoziție funcția „validateRequest”, care primește ca parametri un obiect de tip Request și unul de tip Response. Pe baza acestora, obține valoarea cookie-ului – ID-ul unic al sesiunii, pe care îl folosește pentru a obține obiectul care reprezintă întreaga sesiune și utilizatorul care o deține. Funcția returnează un obiect care conține utilizatorul și sesiunea, sau null pentru amândouă, dacă acesta nu este autentificat.

În sfârșit, pentru deconectare, butonul „Logout” realizează un POST request către „/api/logout”. Funcția validateRequest este utilizată pentru a obține utilizatorul și sesiunea, apoi sesiunea este invalidată și cookie-ul este șters.

3.3. Comunicarea bidirecțională în timp real între server și client bazată pe evenimente

Pentru a implementa o experiență de joc fluidă, este necesară o modalitate de a comunica în timp real. WebSocket este un protocol care permite comunicarea bidirecțională între un client și un server (IETF, 2011). Socket.IO este o bibliotecă software care

implementează comunicarea folosind atât WebSockets, cât și alte două metode (HTTP long-polling, WebTransport). Biblioteca alege cea mai bună metodă în funcție de capabilitățile browserului și configurarea rețelei (Socket.IO, 2025). Deși Socket.IO folosește WebSockets pentru transferul datelor, un client WebSocket nu se poate conecta la un server Socket.IO și viceversa. Acest lucru se datorează faptului că Socket.IO adaugă metadate suplimentare fiecărui pachet (en. *packet*). Avantajele utilizării Socket.IO față de WebSockets simple sunt reconectarea automată, *broadcasting* („difuzarea” către mai mulți clienți), *acknowledgements* (confirmarea primirii unui eveniment).

În fișierul „./lib/socket/types.ts” sunt definite patru tipuri de date necesare implementării Socket.IO:

- *ServerToClientEvents* – evenimentele trimise de către server către client (ex. *error*, *roomBroadcastQuestion*, *roomPlayersUpdate*);
- *ClientToServerEvents* – evenimentele trimise de către client către server (ex. *joinLobby*, *startLobby*, *action*);
- *InterServerEvents* – evenimentele trimise între servere (dacă exista mai multe) (ex. *ping*);
- *SocketData* – datele suplimentare asociate unei conexiuni – profilul utilizatorului, identificator lobby și camerei la care este asociat acesta.

Pe server-side, evenimentele sunt înregistrate în fișierul „./server/socketHandlers.ts”. Este utilizată o funcție de tip middleware (care este executată pentru fiecare conexiune) pentru a obține utilizatorul autentificat și a-l atribui datelor conexiunii. Dacă nu este conectat, utilizatorului îi este atribuit un utilizator de tip Guest (invitat). Este instanțiat un obiect de tip LobbyManager, care gestionează toate Lobby-urile din cadrul platformei. Acesta oferă funcții pentru crearea de Lobbies, ștergerea acestora și inițializarea corectă pe baza Quiz-ului selectat.

3.4. Logica unui joc – Lobby, Room

Un Lobby reprezintă mulțimea tuturor jucătorilor care vor participa la un joc. Acesta este alcătuit pe baza unui Quiz și îi este atribuit un cod care poate fi folosit de jucători pentru a se alătura. Un Room reprezintă o grupă de 3 jucători (umani sau nu) care vor concura. Când funcția *Lobby.start()* este executată, jucătorii sunt amestecați, apoi împărțiți în grupe de câte trei. Dacă nu sunt suficienți jucători, aceștia sunt împărțiți astfel încât să nu existe o grupă cu un singur jucător.

Tabel 2. Împărțirea jucătorilor pe grupe în funcție de numărul acestora

Număr de jucători	Număr de grupe (g)
3k	k grupe
3k+1	k + 1 grupe, din care două cu 2 jucători
3k+2	k + 2 grupe, din care una cu 2 jucători

Se instanțiază g Rooms, iar în cele cu mai puțin de 3 jucători se adaugă „roboți”. Roboții sunt jucători simulați, care aleg un răspuns complet aleator pentru fiecare întrebare și nu generează date analitice. Fiecare Room este pornit de către Lobby. Pentru fiecare întrebare, aceasta este transmisă jucătorilor conectați. Este salvată data și ora la care a fost trimisă întrebarea. Jucătorii transmit înapoi răspunsul ales folosind evenimentul „action”. Răspunsul dat și timpul necesar pentru a răspunde sunt salvate. Când toți jucătorii au răspuns, le este transmis răspunsul corect. După un interval de timp, datele referitoare la răspunsul dat și timpul necesar sunt resetate și se trece la următoarea întrebare. Jucătorii care au răspuns corect selectează (cuceresc) teritorii de pe hartă, în funcție de corectitudinea și timpul de răspuns. Când s-au terminat toate întrebările, jucătorilor le este transmis evenimentul *roomEnd*. Sunt salvate în baza de date punctele de experiență acumulate.

3.5. Baza de date

Platforma folosește pe post de bază de date PostgreSQL, care este o baza de date relațională, open-source. Având peste 35 de ani de dezvoltare activă, aceasta și-a dobândit o reputație puternică cu privire la fiabilitate, robustețe și performanță (The PostgreSQL Global Development Group, 2025). Modelul relațional este prezentat în Figura 3.

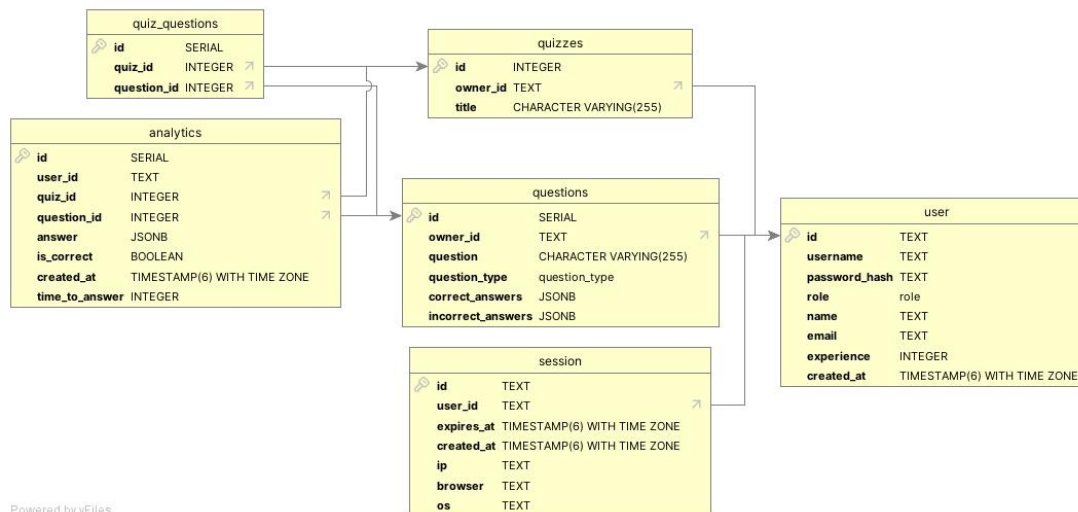


Figura 3. Structura și relațiile dintre tabelele din baza de date.

Sursa: prelucrare proprie

3.5.1. Managementul întrebărilor

Platforma necesită stocarea a mai multor tipuri de întrebări. O implementare inițială a constat în stocarea unei întrebări sub forma unei uniuni etichetate (en. *discriminated union*) (Microsoft, n.d.) de tipuri de date, cu un singur parametru comun „questionType”. Pe baza acestui parametru se putea realiza *type narrowing*. Această metodă presupunea complicarea procedurilor de stocare, elimina siguranța datelor la nivelul bazei de date și făcea dificilă adăugarea pe viitor a unor noi tipuri de întrebări.

Metoda actuală de stocare a întrebărilor în baza de date se bazează pe faptul că, fundamental, o întrebare are răspunsuri corecte și răspunsuri greșite. În cazul anumitor tipuri de întrebări, putem să ne gândim la *incorrectAnswers* ca stocând distractori. Pe lângă acestea, un obiect de tip întrebare mai stochează un ID unic, pe post de cheie primară, identificatorul utilizatorului care a creat întrebarea și cerința întrebării.

Tabel 3. Folosirea câmpurilor din tabelul Questions pentru salvarea tipurilor de întrebări predefinite

Tip întrebare	correctAnswers[]	incorrectAnswers[]
True/False	Valoarea booleană true sau false, în funcție de răspunsul corect	Nimic, răspunsul greșit poate fi calculat în funcție de răspunsul corect
MultipleChoice	Textul variantei corecte	Textul variantelor greșite
Correspondence	Variantele corecte, în ordinea corectă	Distractori (dacă există)
Ordering	Variantele corecte, în ordinea corectă	Nimic
Nearby	Textul variantei corecte	Nimic
IncompleteText	Variantele corecte, în ordinea corectă	Distractori (dacă există)

3.5.2. Managementul datelor analitice

Un alt rol semnificativ îndeplinit de către această platformă educațională este colectarea datelor și analizarea acestora, în vederea îmbunătățirii performanței evaluării. În cadrul unei sesiuni de joc, după ce o întrebare primește răspuns, acesta este înregistrat în baza de date în tabelul Analytics. O înregistrare conține un ID numeric pe post de cheie primară, identificatorul utilizatorului care a dat răspunsul, identificatorul quizului din care face parte întrebarea la care s-a răspuns, identificatorul întrebării la care s-a răspuns, răspunsul dat de către utilizator, o valoare booleană care indică dacă răspunsul oferit este corect, timpul de răspuns (în milisecunde) și data și ora la care a fost dat acest răspuns. Nu sunt colectate date analitice de la jucători care nu sunt umani (ex. roboți). Pe baza datelor analitice obținute, se pot calcula „statistici” pentru fiecare tip de obiect analizat (întrebare, quiz, utilizator). Pentru fiecare întrebare, platforma calculează rata răspunsurilor corecte.

Formula 1. Calculul ratei răspunsurilor corecte

$$\text{Rata răspunsurilor corecte } RRC = \frac{\text{Număr răspunsuri corecte}}{\text{Număr răspunsuri totale}} \in [0,1]$$

Pe baza acestei rate, fiecărei întrebări îi este atribuită o dificultate, reprezentată de culoarea verde – ușor (dacă $RRC \in (0.66, 1]$), galben – mediu (dacă $RRC \in (0.33, 0.66]$), respectiv roșu – dificil (dacă $RRC \in [0, 0.33]$).

De asemenea, pe pagina fiecărei întrebări este afișată o histogramă pentru analiza timpului de răspuns. Pentru aceasta, sunt obținuți timpii de răspuns în milisecunde și ordonați în ordine crescătoare. Este calculată dimensiunea eșantionului ca fiind numărul de răspunsuri totale, respectiv valoarea minimă și maximă. Se stabilește dimensiunea unei grupe de 1000 milisecunde (o secundă). Este creat un vector de obiecte care reprezintă fiecare grupă și sunt numărați timpii de răspuns care se încadrează.

4. Platforma - Frontend

4.1. Tehnologii utilizate

React este o bibliotecă software dezvoltată de Facebook (actual Meta). O interfață grafică construită cu React este formată din componente. Principiul de bază al React este că Interfața Grafică (UI) este o funcție a stării (en. *state*). În React există două fluxuri de date – datele (en. *props*) sunt trimise în jos (de la componenta părinte la componentele copii), iar evenimentele (en. *events*) sunt trimise în sus (de la componentele copii la componenta părinte). Adicional, fiecare componentă are o stare proprie, care poate fi influențată de props și events. Componente avansate pot fi obținute prin compunerea componentelor mai primitive. De exemplu, un tabel este format din rânduri.

Platforma folosește shadcn/ui pe post de bibliotecă de componente. Aceasta oferă componente cu o interfață de programare (en. *API*) comună și compozabilă (Shadcn, n.d.). De asemenea, componentele sunt proiectate să arate consistent. Shadcn/ui integrează Recharts pentru afișarea graficelor.

TanStack Query este o bibliotecă software responsabilă cu gestionarea stării asincrone. Este utilizată pentru obținerea, stocarea în cache, sincronizarea și actualizarea stării de pe server (Tanstack, n.d.).

Platforma este structurată pe mai multe secțiuni dedicate gestionării resurselor: *Home* afișează statistici despre jucători și sesiuni de joc, precum și clasamentul săptămânal; *Play* permite utilizatorilor să se alăture sau să creeze sesiuni de joc pe baza quizurilor disponibile. Secțiunea *Questions* gestionează întrebările platformei, oferind opțiuni de creare, vizualizare și ștergere, în timp ce *Quizzes* centralizează sesiunile de joc existente. *Maps* permite încărcarea de fișiere SVG pentru generarea de hărți interactive, utilizate în platformă. *Profiles* afișează

datele fiecărui utilizator, iar *My Profile* oferă, suplimentar, opțiunea de gestionare a sesiunilor active. *Pagina profilului altui utilizator* afișează întrebările și sesiunile de joc create de acesta.

5. Scenariu de utilizare

Platforma este alcătuită din multiple pagini, fiecare organizate logic pe secțiuni și responsabile cu câte un concept. Înainte de a putea începe o sesiune de joc (Lobby), cadrul didactic trebuie să creeze suficiente întrebări. Ulterior, întrebările vor fi asociate unui Quiz. Pe baza unui Quiz este creat un Lobby. După alăturarea cadrului didactic la Lobby, este afișat un cod QR pe care studenții îl pot scana pentru a se alătura (Figura 4). Jocul poate fi început de către cadrul didactic, respectiv utilizatorul care a creat Lobby-ul.



Figura 4. Panoul din partea stângă a paginii unui joc în etapa de Lobby.
Sursa: prelucrare proprie

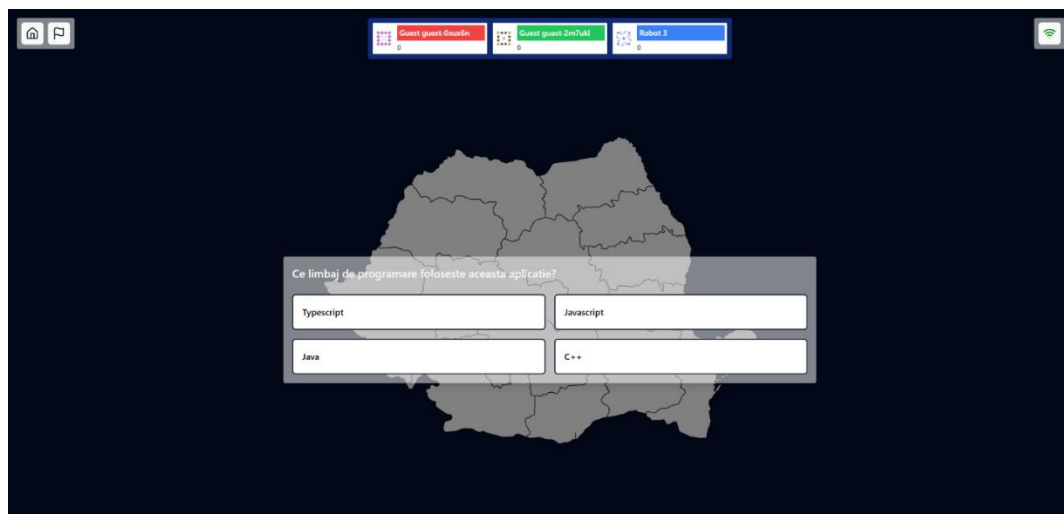


Figura 5. Interfața afișată unui jucător în timpul unei sesiuni de joc.
Sursa: prelucrare proprie

După începerea jocului, participanții sunt împărțiți în grupe de câte trei. Le este afișată întrebarea curentă și interfața necesară pentru a alege un răspuns. De asemenea, în partea de sus a ecranului le este afișat scorul celorlalți jucători (Figura 5). După finalizarea unui joc este afișat clasamentul (Figura 6).

Fig. 3. Clasamentul afișat unui jucător la finalul unui meci

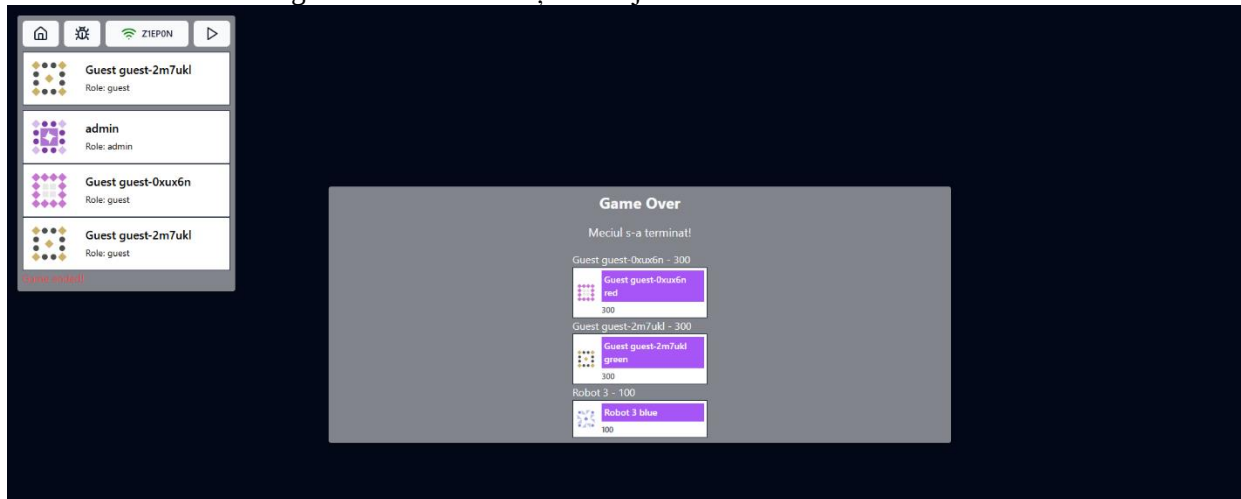


Figura 6. Clasamentul afișat unui jucător la finalul unui meci.

Sursa: prelucrare proprie

6. Implementare și îmbunătățiri posibile

Pentru punerea în funcțiune a platformei a fost creat un container în *Proxmox*, în care au fost descărcate *git* și *nodejs*. Folosind *nginx proxy manager*, a fost setată ca adresă <https://trivia.viziteumihai.ro>, generat un certificat SSL și redirectionat traficul către container.

În urma monitorizării datelor furnizate de *Proxmox*, s-a observat că utilizarea memoriei RAM este constantă, situându-se constant în plaja a 700 MiB. De asemenea, utilizarea maximă a procesorului în perioada observării nu a depășit 1%, concluzia fiind că această platformă poate fi găzduită cu ușurință, fără să necesite o utilizare intensă a resurselor. În stadiul curent, platforma este în proces de dezvoltare. Poate fi utilizată pentru evaluare, dar sunt multiple facilități care ar trebui implementate pentru a îmbunătăți experiența de utilizare. Fiecare Lobby ar putea să aibă asociat un obiect pentru configurare, care să stocheze opțiuni cu privire la timpul de răspuns maxim, multiplicatorul de XP, amestecarea variantelor de răspuns etc.

Deși interfața grafică este suficient de funcțională pe dispozitive mobile (telefoane, tablete), aceasta poate fi optimizată în continuare, pentru a asigura compatibilitatea cu o gamă mai variată de dimensiuni de ecrane.

În cazul întrebărilor, se pot adăuga următoarele:

- Buton de partajare (en. *share*) pentru a trimite mai ușor către alte cadre didactice;
- Etichetarea fiecărei întrebări cu etichete personalizate;
- Etichetarea fiecărei întrebări cu competențele vizate de aceasta, în vederea urmăririi îndeplinirii acestora;
- Posibilitatea de a importa întrebări în formatul AIKEN sau GIFT, pentru a crește compatibilitatea cu alte sisteme precum Moodle;
- Posibilitatea ca jucătorul să poată marca o întrebare ca fiind greșită în timpul desfășurării unui meci – urmând ca aceasta să fie revizuită de către profesor mai târziu;

- Adăugarea unui câmp cu explicații care să fie afișate jucătorilor după evaluarea întrebării.

Pe baza datelor analitice colectate, se mai pot calcula și:

- Rata răspunsurilor corecte pentru Quiz și pentru User;
- Cel mai des ales răspuns pentru o întrebare;
- Frecvențe absolute și relative pentru variantele de răspuns;
- Numărul total de utilizări ale unei întrebări sau ale unui Quiz;
- Dificultatea medie a unui Quiz, în funcție de întrebările adăugate;
- Numărul total de Quizuri jucate de un player.

Platforma ar putea beneficia de pe urma implementării mai multor tipuri de joc. S-ar putea astfel evalua mai multe tipuri de întrebări, precum probleme specifice (de exemplu contabilitate, normalizarea datelor), nemaifiind necesar formatul de tip Trivia.

Pentru a aplica mai îndeaproape caracteristicile gamificării, utilizatorii ar putea să primească insigne, care să fie afișate pe profilurile acestora. Un exemplu de astfel de insignă ar putea fi acordată pentru rezolvarea unui Quiz fără greșeli.

Din punct de vedere tehnic, backend-ul platformei ar putea fi paralelizat, permițând astfel conectarea mai multor jucători în paralel. Numărul de servere utilizat ar putea să se ajusteze în funcție de numărul de utilizatori care folosesc platforma într-un moment, crescând eficiența utilizării resurselor hardware.

Bibliografie

1. Chou, Y.-K. (2015) *Actionable gamification: beyond points, badges, and leaderboards*. United States: Octalysis Group
2. Drizzle Team (n.d.) “Drizzle ORM – Why Drizzle?”. [Online]. Disponibil online la: <https://orm.drizzle.team/docs/overview>
3. Hamari, J. (2019) Gamification. In: G. Ritzer, ed. *The Blackwell encyclopedia of sociology* (pp.1-3). <https://doi.org/10.1002/9781405165518.wbeos1321>.
4. Internet Engineering Task Force (2011) “The WebSocket protocol (RFC 6455)”. [Online]. Disponibil online la: <https://datatracker.ietf.org/doc/html/rfc6455>
5. Internet Engineering Task Force (2021) “Argon2: Memory-hard functions for password hashing and other applications (RFC 9106)”. [Online]. Disponibil online la: <https://datatracker.ietf.org/doc/html/rfc9106>
6. Microsoft (n.d.) “Typescript: Documentation – Narrowing”. [Online]. Disponibil online la: <https://www.typescriptlang.org/docs/handbook/2/narrowing.html#discriminated-unions>
7. Pilcrow (n.d. a) “Lucia v3 documentation”. [Online]. Disponibil online la: <https://v3.lucia-auth.com/>
8. Pilcrow (n.d. b) “Sessions”. [Online]. Disponibil online la: <https://v3.lucia-auth.com/basics/sessions>, accesat ianuarie 2025
9. Robson, K., Plangger, K., Kietzmann, J. H., McCarthy, I. & Pitt, L. (2015) “Is it all a game? Understanding the principles of gamification”, *Business Horizons*, vol. 58, no. 4: 411-420. <https://doi.org/10.1016/j.bushor.2015.03.006>.
10. Ryan, R. M. & Deci, E. L. (2000) “Intrinsic and extrinsic motivations: Classic definitions and new directions”, *Contemporary Educational Psychology*, vol. 25, no. 1: 54-67. <https://doi.org/10.1006/ceps.1999.1020>.
11. Shadcn (n.d.) “Introduction – shad/cn UI”. [Online]. Disponibil online la: <https://ui.shadcn.com/docs>
12. Socket.IO (2025) “Introduction to Socket.IO”. [Online]. Disponibil online la: <https://socket.io/docs/v4/>
13. Tanstack (n.d.), “Overview: Tanstack Query React Docs”. [Online]. Disponibil online la: <https://tanstack.com/query/latest/docs/framework/react/overview>

14. The PostgreSQL Global Development Group (2025) “PostgreSQL: The world's most advanced open source database”. [Online]. Disponibil online la: <https://www.postgresql.org/>
15. Vercel (2025a) “React Foundations: About React and Next.js”. [Online]. Disponibil online la: <https://nextjs.org/learn/react-foundations/what-is-react-and-nextjs>
16. Vercel (2025b) “Configuring: Custom server”. [Online]. Disponibil online la: <https://nextjs.org/docs/pages/building-your-application/configuring/custom-server>
17. Werbach, K. & Hunter, D. (2012) *For the win: How game thinking can revolutionize your business*. EBL-Schweitzer. Philadelphia: Wharton Digital Press.
18. Werbach, K. & Hunter, D. (2015) *The Gamification toolkit: Dynamics, mechanics, and components for the win*. Chicago: Wharton Digital Press.