**Documentul de specificare a cerințelor**

**Software Requirements Specification**

**(SRS) Document**

**Sistem informatic de recomandare pentru o platformă de cursuri bazat pe inteligența artificială**

**02.04.2025**

|  |
| --- |
| **Istoricul versiunilor** |

| Versiune | Autor(i) principali | Descriere versiune | Dată |
| --- | --- | --- | --- |
|  |  |  |  |

|  |
| --- |
| **Revizuiri și aprobări** |

Istoric aprobări

| Aprobă | Versiune | Semnătură | Dată |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

Istoric revizuiri

| Revizor | Versiune | Semnătură | Dată |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |
| --- |
| **Cuprins** |

[1. Introducere 4](#_Toc190689617)

[1.1 Scopul 4](#_Toc190689618)

[1.2 Convenții ale documentului 4](#_Toc190689619)

[1.3 Audiență țintă 4](#_Toc190689620)

[1.4 Sfera de aplicare 4](#_Toc190689621)

[1.5 Referințe 4](#_Toc190689622)

[**2 Descriere generală 4**](#_Toc190689623)

[**2.1 Perspectiva produsului 4**](#_Toc190689624)

[**2.2 Caracteristici ale produsului 4**](#_Toc190689625)

[**2.3 Clase și caracteristici ale utilizatorilor 4**](#_Toc190689626)

[**2.4 Mediul de operare 5**](#_Toc190689627)

[**2.5 Constrângeri de proiectare și de implementare 5**](#_Toc190689628)

[**2.6 Presupuneri și dependențe 5**](#_Toc190689629)

[**3 Cerințele sistemului 5**](#_Toc190689630)

[**3.1 Funcționalitatea 1 5**](#_Toc190689631)

[**3.1.1 Descriere generală 5**](#_Toc190689632)

[**3.1.2 Flux de interacțiune (scenarii de utilizare) 5**](#_Toc190689633)

[**3.1.3 Condiții prealabile și constrângeri 6**](#_Toc190689634)

[**3.1.4 Detaliere cerință 6**](#_Toc190689635)

[**3.1.5 Scenarii de eroare și gestionarea excepțiilor 6**](#_Toc190689636)

[**3.1.5 Dependențe și interacțiuni cu alte funcționalități 6**](#_Toc190689637)

[**3.2 Cerința funcțională 2 6**](#_Toc190689638)

[3.3 .... 6](#_Toc190689639)

[4 Cerințe pentru interfețe 6](#_Toc190689640)

[4.1 Interfețe cu utilizatorul 6](#_Toc190689641)

[4.2 Interfețe hardware 6](#_Toc190689642)

[4.2.1 Configurații Minime Recomandate 7](#_Toc190689643)

[4.2.2 Dispozitive Externe Compatibile 7](#_Toc190689644)

[4.3 Interfețe de comunicare 7](#_Toc190689645)

[4.3.1 Protocoale și Standarde de Comunicare 7](#_Toc190689646)

[4.3.2 Cerințe de Securitate în Comunicare 7](#_Toc190689647)

[4.4 Interfețe software 7](#_Toc190689648)

[4.4.1 Tehnologii Utilizate 7](#_Toc190689649)

[4.4.2 Servicii Externe și API-uri 7](#_Toc190689650)

[5 Cerințe non-funcționale 8](#_Toc190689651)

[5.1 Cerințe de performanță 8](#_Toc190689652)

[5.2 Cerințe de siguranță 8](#_Toc190689653)

[5.3 Cerințe de securitate 8](#_Toc190689654)

[5.4 Atribute de calitate ale software-ului 8](#_Toc190689655)

[6 Alte cerințe 8](#_Toc190689656)

[7 Anexe 8](#_Toc190689657)

[7.1 Anexa A: Glosar 8](#_Toc190689658)

[7.2 Anexa B: Modele de Analiză 8](#_Toc190689659)

[7.3 Anexa C: Listă de Probleme 9](#_Toc190689660)

# Introducere

### 1.1 Scopul

Scopul acestui document este de a defini cerințele funcționale și non-funcționale ale sistemului LearnJy – o platformă de e-learning bazată pe inteligență artificială. Documentul va ghida echipa de dezvoltare, testare și management de proiect în proiectarea, implementarea și verificarea produsului software.

### 1.2 Convenții ale documentului

* Abrevierile importante sunt listate în Anexa B.
* Numele claselor de utilizatori sunt scrise cu majuscule (ex: **STUDENT**, **INSTRUCTOR**, **ADMIN**).
* Termenii tehnici sunt notați în format italic la prima apariție.

### 1.3 Audiență țintă

| **Rol** | **Interes în document** |
| --- | --- |
| Manager de proiect | Asigurarea alinierii cu obiectivele afacerii |
| Dezvoltatori | Implementarea cerințelor funcționale |
| Testeri | Crearea de scenarii de test și validare |
| Designeri UI/UX | Crearea interfețelor centrate pe utilizator |

### 1.4 Sfera de aplicare

LearnJy este o platformă educațională modernă care utilizează algoritmi AI pentru a personaliza traseul de învățare al utilizatorilor. Obiectivul principal este de a îmbunătăți eficiența procesului de învățare prin recomandări dinamice, evaluări automate și certificare digitală. Platforma oferă beneficii directe pentru instituții educaționale, companii de training și utilizatori individuali.

### 1.5 Referințe

* Document de proiectare LearnJy (versiunea 1.0)
* OWASP Top 10 ([https://owasp.org](https://owasp.org/))
* Documentație FastAPI ([https://fastapi.tiangolo.com](https://fastapi.tiangolo.com/))
* Documentație React ([https://reactjs.org](https://reactjs.org/))
* Documentație Neo4j (<https://neo4j.com/docs/>)

# 2. Descriere generala

### 2.1 Perspectiva produsului

LearnJy este un sistem LMS complet nou, proiectat să funcționeze independent, dar extensibil pentru integrare cu sisteme existente (ex: Zoom, Google Meet, Moodle). Platforma este oferită ca serviciu web și nu necesită instalare locală.

### 2.2 Caracteristici ale produsului

* Autentificare și înregistrare utilizatori
* Recomandări AI de cursuri
* Înrolare și urmărire progres
* Teste automate și feedback
* Emiterea de certificate
* Management cursuri și lecții
* Mesagerie internă între utilizatori
* Interfață administrator pentru controlul platformei

### 2.3 Clase și caracteristici ale utilizatorilor

| **Clasă utilizatori** | **Roluri și caracteristici** |
| --- | --- |
| STUDENT | Accesează cursuri, primește recomandări, susține teste |
| INSTRUCTOR | Creează și administrează cursuri și lecții |
| ADMIN | Gestionează platforma, utilizatorii și rapoartele globale |

Utilizatorii pot fi ocazionali sau activi frecvent, cu diferite niveluri de experiență tehnică. Platforma este concepută pentru a fi ușor accesibilă indiferent de acest nivel.

### 2.4 Mediul de operare

* **Frontend:** Browsere moderne (Chrome, Firefox, Safari, Edge)
* **Backend:** Python 3.11, FastAPI
* **Bază de date:** Neo4j AuraDB Cloud
* **Stocare fișiere:** AWS S3
* **Dispozitive suportate:** Desktop, laptop, smartphone (iOS 14+/Android 9+)

### 2.5 Constrângeri de proiectare și implementare

* Platforma trebuie să respecte GDPR
* Comunicarea între componente trebuie să fie securizată (HTTPS, JWT)
* Integrare cu Stripe pentru plăți (API REST)
* Interfața trebuie să fie responsivă (mobile-friendly)
* Baza de date trebuie să suporte interogări graf eficiente

### 2.6 Presupuneri și dependențe

* Utilizatorii au acces la internet stabil
* Accesul se face de pe dispozitive compatibile
* Serviciile externe (Stripe, AWS, Neo4j) sunt disponibile
* Modelele AI au suficientă date pentru a livra rezultate relevante
* Proiectul va evolua incremental, cu feedback continuu de la utilizatori

# 3. Cerințele sistemului

### 3.1 Funcționalitatea 1 – Autentificare utilizator

#### 3.1.1 Descriere generală

Autentificarea permite utilizatorilor existenți să se conecteze în platforma LearnJy prin furnizarea credențialelor (email și parolă). Funcționalitatea este necesară pentru a asigura accesul securizat la funcțiile personalizate (ex: înrolare la cursuri, mesagerie, progres).  
Este folosită de toți utilizatorii: **STUDENT**, **INSTRUCTOR**, **ADMIN**, imediat după accesarea aplicației.

#### 3.1.2 Flux de interacțiune (scenariu de utilizare)

1. Utilizatorul accesează pagina de login.
2. Introduce emailul și parola.
3. Apasă pe butonul „Autentificare”.
4. Sistemul validează datele și, dacă sunt corecte:
   * Generează un token JWT.
   * Redirecționeză utilizatorul către dashboard-ul personal.
5. Dacă datele sunt incorecte:
   * Se afișează un mesaj de eroare („Email sau parolă incorectă”).

#### 3.1.3 Condiții prealabile și constrângeri

* Utilizatorul trebuie să fie deja înregistrat.
* Sistemul trebuie să aibă conectivitate cu baza de date.
* Răspunsul API trebuie să fie sub 500 ms.

#### 3.1.4 Detaliere cerință

| **Cod** | **Cerință** | **Prioritate** | **Criterii de acceptanță** |
| --- | --- | --- | --- |
| REQ-1 | Utilizatorul trebuie să poată introduce email și parolă | Ridicată | Formularele acceptă input valid |
| REQ-2 | Sistemul trebuie să valideze automat formatul emailului | Ridicată | Se afișează eroare pentru format invalid |
| REQ-3 | Sistemul trebuie să cripteze parolele folosind hashing (bcrypt) | Ridicată | Nicio parolă stocată în clar în DB |
| REQ-4 | După autentificare, sistemul trebuie să returneze un token JWT | Ridicată | Tokenul este valabil timp de 60 minute |
| REQ-5 | Sistemul trebuie să răspundă cu mesaj de eroare dacă datele nu corespund | Medie | „Email sau parolă incorectă” este afișat |

#### 3.1.5 Scenarii de eroare și gestionarea excepțiilor

* Email nevalid: mesaj tip “Format email invalid”
* Parolă incorectă: mesaj tip “Email sau parolă incorectă”
* Server indisponibil: mesaj generic “Autentificarea nu este disponibilă momentan. Încearcă mai târziu.”
* Cont inexistent: „Contul nu a fost găsit. Creează unul nou.”

#### 3.1.6 Dependențe și interacțiuni cu alte funcționalități

* Depinde de funcționalitatea de **înregistrare** (crearea contului).
* Generează token JWT utilizat în toate solicitările autentificate către API.
* Interacționează cu modulul de **gestionare roluri** pentru a direcționa utilizatorul în interfața corespunzătoare (Student, Instructor, Admin).

### 3.2 Funcționalitatea 2 – Înrolare la curs

#### 3.2.1 Descriere generală

Înrolarea permite utilizatorilor înregistrați să se înscrie la cursurile disponibile în platforma LearnJy. Această funcționalitate este esențială pentru progresul educațional personalizat și pentru activarea traseului de învățare AI. Utilizatorii **STUDENT** utilizează această funcție din pagina unui curs listat.

#### 3.2.2 Flux de interacțiune (scenariu de utilizare)

1. Utilizatorul accesează catalogul de cursuri.
2. Selectează un curs disponibil.
3. Apasă butonul „Înrolează-te”.
4. Sistemul verifică eligibilitatea (ex: cont activ, curs neînrolat anterior).
5. Sistemul adaugă relația INROLAT între utilizator și curs în Neo4j.
6. Utilizatorul este redirecționat către pagina cursului.

#### 3.2.3 Condiții prealabile și constrângeri

* Utilizatorul trebuie să fie autentificat.
* Cursul selectat trebuie să fie activ (status = activ).
* Dacă înrolarea implică plată, plata trebuie să fie procesată cu succes anterior.

#### 3.2.4 Detaliere cerință

| **Cod** | **Cerință** | **Prioritate** | **Criterii de acceptanță** |
| --- | --- | --- | --- |
| REQ-6 | Utilizatorul trebuie să poată solicita înrolarea într-un curs | Ridicată | Buton „Înrolează-te” funcțional pentru cursuri active |
| REQ-7 | Sistemul trebuie să valideze dacă utilizatorul nu este deja înscris | Ridicată | Dacă deja înrolat, sistemul nu mai permite înscrierea |
| REQ-8 | Sistemul trebuie să adauge relația INROLAT în baza de date | Ridicată | Relația este creată în Neo4j |
| REQ-9 | După înrolare, sistemul trebuie să confirme înscrierea | Medie | Afișarea mesajului „Te-ai înscris cu succes la curs!” |
| REQ-10 | Dacă plata este necesară, trebuie procesată înainte de înrolare | Ridicată | Confirmare tranzacție Stripe înainte de crearea relației |

#### 3.2.5 Scenarii de eroare și gestionarea excepțiilor

* Curs inexistent: mesaj „Cursul selectat nu mai este disponibil.”
* Eroare server la creare relație: mesaj „Înscrierea a eșuat. Încearcă din nou.”
* Utilizator deja înrolat: mesaj „Ești deja înscris la acest curs.”
* Eșec la plată: mesaj „Plata nu a putut fi procesată. Încearcă altă metodă.”

#### 3.2.6 Dependențe și interacțiuni cu alte funcționalități

* Depinde de funcționalitatea de **autentificare** (user login).
* Interacționează cu **modulul de plăți** (Stripe) pentru cursurile premium.
* Actualizează datele utilizatorului pentru a permite parcurgerea lecțiilor și evaluărilor din curs.

# 4 Cerințe pentru interfețe

### 4.1 Interfețe cu utilizatorul

Interfața platformei LearnJy este web-based și mobile responsive, fiind dezvoltată cu React (Next.js).  
Designul este orientat spre ușurința utilizării și accesibilitate (respectând standardele WCAG 2.1).

**Ecrane principale:**

* **Pagina de autentificare:** Câmpuri email, parolă, buton „Autentificare” + link „Am uitat parola”.
* **Dashboard utilizator:** Recomandări de cursuri, progres în cursuri, notificări recente.
* **Pagina cursului:** Informații generale, progres lecții, buton „Continuă” sau „Finalizare curs”.
* **Test online:** Întrebări, variante multiple de răspuns, buton „Trimite testul”.
* **Mesagerie internă:** Inbox cu conversații, răspunsuri rapide.
* **Profil utilizator:** Editare date personale, vizualizare certificări.

**Ghiduri de stil:**

* Font principal: Roboto, 14px–18px
* Butoane principale: alb pe fundal albastru (#0056D2)
* Mesaje de eroare: roșu (#E53935)
* Layout adaptiv: Flexbox și Grid CSS pentru optimizare mobilă

### 4.2 Interfețe hardware

LearnJy este un sistem cloud și necesită doar dispozitive standard pentru accesarea platformei.

#### 4.2.1 Configurații Minime Recomandate

* **PC/Laptop:**
  + Procesor: Intel Core i5 (sau echivalent AMD)
  + RAM: 8GB
  + Spațiu de stocare: minim 500MB pentru cache/browser
  + Browser: Chrome 95+, Firefox 90+, Edge 95+
* **Dispozitive mobile:**
  + Android 9.0+ sau iOS 14+
  + RAM: 4GB+
  + Browsere mobile actualizate

#### 4.2.2 Dispozitive Externe Compatibile

* Cameră web pentru participarea la cursuri live (opțional).
* Microfon pentru feedback audio (opțional).

### 4.3 Interfețe de comunicare

#### 4.3.1 Protocoale și Standarde de Comunicare

* **HTTP/HTTPS:** Pentru toate interacțiunile client-server.
* **WebSockets:** Pentru notificări în timp real (ex. mesaje interne, actualizări progres).
* **TLS 1.2+:** Asigură criptarea traficului.

#### 4.3.2 Cerințe de Securitate în Comunicare

* Toate datele transmise trebuie criptate (TLS 1.2 sau superior).
* Autentificarea și autorizarea se fac folosind protocolul OAuth 2.0, cu sesiuni protejate prin JWT (JSON Web Token).

### 4.4 Interfețe software

#### 4.4.1 Tehnologii Utilizate

* **Backend:** Python 3.11 – FastAPI
* **Frontend:** React 18 – Next.js
* **Bază de date:** Neo4j AuraDB Cloud

#### 4.4.2 Servicii Externe și API-uri

| **Serviciu/API** | **Utilizare** |
| --- | --- |
| Stripe API | Procesare plăți pentru cursurile plătite |
| Firebase Authentication | Resetare parole și validare email (opțional) |
| AWS S3 | Stocare fișiere multimedia (imagini, PDF, video) |
| Neo4j Bolt API | Interogare bazei de date graf bazată pe relații |
| Google reCAPTCHA | Protecție împotriva bot-urilor la autentificare și înregistrare |

# 5 Cerințe non-funcționale

### 5.1 Cerințe de performanță

* Timp de răspuns API pentru operațiuni uzuale (login, înrolare, accesare cursuri) trebuie să fie sub **500 ms** în condiții normale de utilizare.
* Sistemul trebuie să fie capabil să suporte minim **10.000 de utilizatori simultan** fără degradare semnificativă a performanței.
* Recomandările AI trebuie generate într-un timp mediu de **sub 5 secunde** per utilizator.
* Platforma trebuie să asigure **99,5% disponibilitate lunar** (SLA garantat).

### 5.2 Cerințe de siguranță

* Sistemul trebuie să asigure backup automat al datelor esențiale (utilizatori, cursuri, progres) cel puțin **o dată pe zi**.
* Recuperarea din backup trebuie să fie posibilă în maximum **2 ore** în caz de incident major.
* Mecanisme anti-abuz (ex: limitare rate de login pentru a preveni atacurile brute-force) trebuie să fie implementate.
* Proceduri de restaurare a datelor și continuare a activității în caz de eșec server (Disaster Recovery Plan).

### 5.3 Cerințe de securitate

* Toate datele transmise între client și server trebuie să fie criptate folosind **TLS 1.2** sau superior.
* Datele sensibile (ex: parole) trebuie criptate folosind algoritmi de hashing sigur (ex: **bcrypt**).
* Implementarea autentificării OAuth 2.0 pentru toate serviciile externe și autentificare JWT internă.
* Conformitate cu **Regulamentul General privind Protecția Datelor (GDPR)**:
  + Dreptul utilizatorilor de a cere ștergerea datelor personale.
  + Dreptul de export a datelor proprii într-un format standardizat (JSON/CSV).

### 5.4 Atribute de calitate ale software-ului

* **Ușurință în întreținere:** Cod modular, documentat, separare clară frontend/backend.
* **Adaptabilitate:** Posibilitatea adăugării rapide de noi funcționalități sau integrări externe.
* **Flexibilitate:** Arhitectură bazată pe microservicii pentru scalare independentă.
* **Utilizabilitate:** Interfață intuitivă, respectând principiile de UX/UI și accesibilitate WCAG 2.1.
* **Fiabilitate:** Sisteme de logging și monitorizare automate pentru erori și evenimente critice.
* **Portabilitate:** Posibilitatea rulării în medii cloud multiple (AWS, Azure, GCP).

# 6 Alte cerințe

* **Cerințe de internaționalizare:** Platforma trebuie să permită adăugarea facilă de noi limbi printr-un sistem de traduceri (ex: JSON key-value) fără modificări de cod.
* **Cerințe legale:** Respectarea GDPR și a altor reglementări europene privind datele personale.
* **Cerințe privind bazele de date:** Sistemul trebuie să suporte Neo4j Aura pentru gestionarea grafurilor de relații complexe între utilizatori, cursuri și interacțiuni.

# 7 Anexe

### 7.1 Anexa A: Glosar

| **Termen/Acronim** | **Definiție** |
| --- | --- |
| LMS | Learning Management System (Sistem de management al învățării) |
| JWT | JSON Web Token, metodă de transmitere a informației securizate între părți |
| OAuth 2.0 | Protocol de autorizare standardizat pentru acces la resurse API |
| Neo4j | Bază de date orientată pe grafuri, folosită pentru relații complexe |
| API | Application Programming Interface, interfață de comunicație între software-uri |
| GDPR | Regulamentul General privind Protecția Datelor, reglementare UE pentru protejarea datelor personale |
| TLS | Transport Layer Security, protocol criptografic pentru comunicații sigure |

## Anexa B: Modele de Analiză