**Documentul de specificare a cerințelor**

**Software Requirements Specification**

**(SRS) Document**

**<Recunoașterea genurilor de știri folosind inteligența artificială>**

**<21.02.2025><0.0.1>**

**<Universitatea Tehnică de Construcții București>**

|  |
| --- |
| **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 3](#_Toc196619836)

[1.1 Scopul 3](#_Toc196619837)

[1.2 Audiență țintă 3](#_Toc196619838)

[1.3 Sfera de aplicare 3](#_Toc196619839)

[1.4 Referințe 3](#_Toc196619840)

[2 Descriere generală 3](#_Toc196619841)

[2.1 Perspectiva produsului 3](#_Toc196619842)

[2.2 Caracteristici ale produsului 4](#_Toc196619843)

[2.3 Clase și caracteristici ale utilizatorilor 4](#_Toc196619844)

[2.4 Mediul de operare 4](#_Toc196619845)

[2.5 Constrângeri de proiectare și de implementare 4](#_Toc196619846)

[2.6 Presupuneri și dependențe 4](#_Toc196619847)

[3 Cerințele sistemului 5](#_Toc196619848)

[3.1 Funcționalitatea 1 5](#_Toc196619849)

[3.1.1 Descriere generală 5](#_Toc196619850)

[3.1.2 Flux de interacțiune (scenarii de utilizare) 5](#_Toc196619851)

[3.2 Funcționalitatea 2 5](#_Toc196619852)

[3.2.1 Descriere generală 6](#_Toc196619853)

[3.2.2 Flux de interacțiune (scenarii de utilizare) 6](#_Toc196619854)

[3.3 Funcționalitatea 3 6](#_Toc196619855)

[3.3.1 Descriere generală 6](#_Toc196619856)

[3.3.2 Flux de interacțiune (scenarii de utilizare) 6](#_Toc196619857)

[3.4 Funcționalitatea 4 6](#_Toc196619858)

[3.4.1 Descriere generală 6](#_Toc196619859)

[3.4.2 Flux de interacțiune (scenarii de utilizare) 6](#_Toc196619860)

[3.5 Funcționalitatea 5 6](#_Toc196619861)

[3.5.1 Descriere generală 6](#_Toc196619862)

[3.5.2 Flux de interacțiune (scenarii de utilizare) 7](#_Toc196619863)

[3.6 Detaliere cerințe 7](#_Toc196619864)

[3.6.1 Scenarii de eroare și gestionarea excepțiilor 7](#_Toc196619865)

[3.6.2 Dependențe și interacțiuni cu alte funcționalități 7](#_Toc196619866)

[4 Cerințe pentru interfețe 8](#_Toc196619867)

[4.1 Interfețe cu utilizatorul 8](#_Toc196619868)

[4.2 Interfețe hardware 8](#_Toc196619869)

[4.2.1 Configurații Minime Recomandate 8](#_Toc196619870)

[4.2.2 Dispozitive Externe Compatibile 8](#_Toc196619871)

[4.3 Interfețe de comunicare 8](#_Toc196619872)

[4.3.1 Protocoale și Standarde de Comunicare 8](#_Toc196619873)

[4.3.2 Cerințe de Securitate în Comunicare 8](#_Toc196619874)

[4.4 Interfețe software 9](#_Toc196619875)

[4.4.1 Tehnologii Utilizate 9](#_Toc196619876)

[4.4.2 Servicii Externe și API-uri 9](#_Toc196619877)

[5 Cerințe non-funcționale 9](#_Toc196619878)

[5.1 Cerințe de performanță 9](#_Toc196619879)

[5.2 Cerințe de siguranță 9](#_Toc196619880)

[5.3 Cerințe de securitate 9](#_Toc196619881)

[5.4 Atribute de calitate ale software-ului 9](#_Toc196619882)

[6 Alte cerințe 10](#_Toc196619883)

[7 Anexe 10](#_Toc196619884)

[7.1 Anexa A: Glosar 10](#_Toc196619885)

[7.2 Anexa C: Listă de Probleme 10](#_Toc196619886)

# Introducere

## Scopul

Inteligența artificială a avut în ultimii ani o creștere exponențială datorită puterii mărite de calcul, a resurselor investite și a algoritmilor dezvoltați. Astfel, informațiile obțiute prin intermediul algoritmilor de I.A au devenit mai bune de-alungul timpului, venind în ajutor și cantitățile din ce în ce mai mari ale datelor de antrenament.

Scopul aplicației web bazate pe rețele neuronale este de a facilita identificarea automată a știrilor din imagini. Prin intermediul unui model de rețea neuronală antrenat pe un set extins de imagini de știri, aplicația permite utilizatorilor să încarce o fotografie a unei știri, iar aceasta va fi clasificată rapid și cât mai corect posibil. Astfel, aplicația poate fi utilizată în diverse domenii, precum educația și cercetarea sau chiar comerțul online, oferind o părere în plus privind specia de știri identificată și datele privind siguranța acesteia.

## Audiență țintă

Atât partea introductivă, cât și elementele de utilizare propriu-zisă a site-ului web sunt accesibile fiecărui cititor. Audiența țintă a aplicației web bazate pe rețele neuronale pentru identificarea știrilor din imagini este diversificată și include mai multe categorii de utilizatori, fiecare având nevoi specifice care pot fi satisfăcute prin utilizarea acestei tehnologii inovative, precum: consumatori și utilizatori ocazionali, agricultori și cultivatori, nutriționiști și specialiști în sănătate, pasionați de gătit, educatori și elevi, cercetători sau persoane cu dizabilități vizuale.

## Sfera de aplicare

Specifică cum se aliniază obiectivele produsului cu obiectivele generale ale sistemului în care se integrează produsul și conturează beneficiile proiectului pentru afacere.

## Referințe

O listă a altor documente la care face referire documentul SRS, inclusiv surse precum site-uri web sau literatură scrisă.

# Descriere generală

## Perspectiva produsului

Datorită folosirii pe scară cât mai largă a inteligenței aritificiale în vaste domenii, în principal de către marile corporații, au început să existe și diferite variante de tip open-source, antrenate cu ajutorul informațiilor disponibile publicului larg, astfel, perspectiva produsului se concentrează pe crearea unui instrument accesibil, ușor de utilizat și extrem de eficient, care răspunde unui număr larg de nevoi ale utilizatorilor din diverse domenii. Această aplicație vizează integrarea tehnologiei de recunoaștere a imaginilor într-o soluție practică, nișată, pentru consumatori, profesioniști, educatori, bucătari sau pasionați.

Viziunea pe termen lung pentru acest produs este să devină un instrument standardizat în recunoaștere atât a știrilor cât și a altor produse alimentare, atât în sectorul de consum, cât și în industrie. Aplicația va continua să evolueze prin integrarea cu noi tehnologii, extinderea bazei de date și îmbunătățirea performanței algoritmilor, astfel încât să răspundă în continuare nevoilor utilizatorilor dintr-o gamă largă de domenii.

## Caracteristici ale produsului

Interfață Utilizator (UI) intuitivă, backend robust, procesarea imaginilor, autentificare, performanță, securitate, documentație și suport.

## Clase și caracteristici ale utilizatorilor

Utilizatorii sunt împărțiți în mai multe clase: utilizatori neautentificați, ce au posibilitatea de a încerca funcționalităție aplicației într-o formă limitată și utilizatorii autentificați, ce nu au nicio limitare privind utilizarea aplicației.

## Mediul de operare

Aplicația va rula într-un mediu containerizat (eng. „containerization”) unde toate dependințele aplicației se află într-un singur loc: aplicația propriu-zisă Django, baza de date Postgresql, și web-serverul nGINX.

## Constrângeri de proiectare și de implementare

Performanța și latența (procesarea imaginilor pot necesita timp, ceea ce poate duce la latențe în răspunsul aplicației), resurse de calcul (modelul de rețea neuronală poate necesita resurse de calcul semnificative, ceea ce poate fi costisitor și poate limita scalabilitatea aplicației), acuratețea modelului (calitatea și cantitatea datelor de antrenare pot afecta acuratețea modelului, astfel încât un set de date insuficient sau de calitate scăzută să producă un răspuns invalid), gestionarea datelor (stocarea imaginilor încărcate și a recunoașterii tipului de știre din acestea poate necesita spațiu de stocare semnificativ), experiența de utilizator (crearea unei interfețe intuitive și ușor de utilizat poate fi o provocare, având în vedere faptul că aplicația este destinată tuturor și trebuie să fie accesibilă oricui, indiferent de dispozitivul folosit și cunoștințele din domeniu).

Aceste constrângeri pot fi gestionate printr-o planificare adecvată, utilizarea unor practici de dezvoltare robuste și testare continuă, având în vedere cele mai bune practici.

## Presupuneri și dependențe

Printre presupuneri se pot regăsi cele tehnice (cunoștințe de programare, infrastructură, utilizarea bibliotecilor și ale framework-urilor), de date (privind calitatea și varietatea datelor), cele privind utilizatorul (acces la internet și dispozitivele compatibile), de securitate (protecția datelor și autentifiarea).

Printre dependențele software se regăsesc dependențele software (python, django, tensorflow, postgres, docker, nginx, daisyui, tailwind), dependențele hardware (capabilitățile de procesare și stocare ale serverului/serverelor), dependențe de servicii externe (precum CDN-urile), dependențe de date (seturi de date etichetate și api-uri externe), dependențe financiare (precum bugetul și costurile recurente privind dezvoltarea, întreținerea și scalabilitatea).

# Cerințele sistemului

În cazul prezentărilor aplicației, utilizarea unui laptop sau calculator personal este complet viabilă, având în vedere faptul că numărul utilizatorilor este scăzut, la fel și complexitatea datelor. Cu toate acestea, se poate lua în considerarea scalabilitatea, iar în acest sens se pot avea în vedere o serie de componente mai performante, precum procesor cu 8 sau mai multe nuclee Intel i7/i9 generație 15 sau AMD Ryzen 7/9, placă video Nvidia din gama RTX, în cazul în care se dorește folosirea CUDA în antrenarea modelului de inteligență artificială cu 8GB sau mai mult VRAM, 16GB RAM sau mai mult, SSD de 512GB sau mai mult, conexiune la Internet de mare viteză (500MB+), sistem de operare Windows/Linux, pentru containerizare se va folosi Docker sau orice alt software pentru containerizarea proiectelor de tip open-source, aplicația fiind formată din mai multe containere, anume: aplicația propriu-zisă ce folosește framework-ul Django versiunea 5, Python 3.11, component library bazat pe tailwind (daisyui), un container pentru nginx, folosit ca reverse proxy, fiind situat în fața WSGI-ului oferit fie de către framework, în cazul rulării într-un mediu de development, fie în fața GUNICORN, în cazul rulării în mediul de producție și un container pentru baza de date PostgreSQL.

## Funcționalitatea 1

Încărcarea fișierelor în formatul specific fotografiilor pentru obținerea informațiilor relevante

### Descriere generală

În urma fotografierii unei știri, poza se va încărca în platformă, aceasta urmând a o prelucra și de a oferi informații relevante în baza informațiilor de antrenament disponibile, precum toxicitatea acesteia sau specia.

### Flux de interacțiune (scenarii de utilizare)

În momentul încărcării fotografiei, aplicația va semnaliza cât timp are loc procesarea, iar la finalul acesteia va fi afișat un răspuns, care poate fi de mai multe tipuri: procesarea a fost finalizată cu succes, oferind informațiile necesare asupra fotografiei încărcate, procesarea a fost finalizată, însă acuratețea informațiilor este scăzută datorită unui cumul de factori, procesarea nu a fost finalizată din cauza unor probleme apărute în fluxul de procesare.

## Funcționalitatea 2

Accesarea informațiilor relevante utilizatorului

### Descriere generală

Platforma oferă posibilitatea accesării tuturor datelor legate de utilizator. De asemenea, există posibilitatea verificării istoricului fotografiilor încărcate, predicția oferită de către platformă, un sistem de rating al predicțiilor și descărcarea acestora.

### Flux de interacțiune (scenarii de utilizare)

Odată accesată platforma, în urma introducerii datelor, utilizatorii pot naviga către paginile destinate contului lor, într-un mod simplu, datorită interfeței intuitive.

## Funcționalitatea 3

Comunitate

### Descriere generală

Utilizatorii înregistrați, a căror cont a fost validat, au posibilitatea de a încărca propriile poze în categoriile existente de știri sau de încărca fotografii a unor specii ce nu au fost introduse până în acel moment în platformă.

### Flux de interacțiune (scenarii de utilizare)

Secțiunea specială privind comunitatea va fi accesibilă numai în cazul utilizatorilor logați, a căror cont a fost validat, prin accesarea unei pagini dedicate comunității. Această secțiune este semnalată ca atare, fiind accesibilă tuturor, aspect ce încurajează colaborarea dintre persoane.

## Funcționalitatea 4

Open API

### Descriere generală

Pentru a permite unui număr cât mai larg de utilizatori să se bucure de resursele existente în platformă, o parte din API-ul platformei va fi disponibil publicului.

### Flux de interacțiune (scenarii de utilizare)

În momentul creării unui cont, un API Key este generat și asociat contului respectiv. Dacă se dorește utilizarea acestuia, trebuie să se semnaleze acest aspect în platformă, prin accesarea secțiunii specifice contului de utilizator. După efectuarea acestui pas, API Key-ul va putea fi folosit pentru trimiterea de request-uri în platformă pe rutele specificate clar, în aceiași secțiune.

## Funcționalitatea 5

Comunicare în timp real

### Descriere generală

Notificări în timp real, în funcție de evenimentul ce a avut loc, precum atingerea limitei de încărcare a fotografiilor, restricționarea accesului sau mesaje de importanță ridicată.

### Flux de interacțiune (scenarii de utilizare)

Cu ajutorul Websocket-urilor, va fi creată o conexiune tip „handshake” între platformă și utilizator, conexiune ce permite comunicarea în timp real. Fiecare notificare va fi disponibilă în colțul de sus al paginii, sugestiv, prin intermediul unui clopoțel.

procesare.

## Detaliere cerințe

|  |  |  |  |
| --- | --- | --- | --- |
| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| **REQ-1** | Conexiune la internet | Ridicată | Fără conexiune la internet, utilizatorul final nu poate accesa platforma |
| **REQ-2** | Existența unui cont validat | Ridicată | Fără un cont, persoanele nu se pot bucura de funcționalitățile platformei |
| **REQ-3** | Existența unei imagini valide | Ridicată | Platforma validează fișierele încărcate și va permite trecerea acestora către procesare doar în cazul în care extensia este validă. |
| **REQ-4** | Respectarea condițiilor și regulilor impuse | Ridicată | Încălcarea condițiilor și a regulilor presupune suspendarea activității contului pe o perioadă determinată sau nedeterminată, în funcție de gravitate. |

### ****Scenarii de eroare și gestionarea excepțiilor****

* În cazul în care utilizatorul nu dispune de Internet, platforma nu poate fi accesată.
* În cazul în care conexiunea la Internet este întreruptă, platforma nu poate întoarce răspunsurile / evenimentele către utilizator.
* Atunci când imaginea nu este validă, platforma fie nu va permite încărcarea acesteia.
* Dacă regulile impuse în platformă sunt încălcate, contul utilizatorului va fi restricționat.
* Platforma devine inaccesibilă atunci când una dintre componente nu este funcțională (baza de date, reverse proxy, modelul ML, mașina ce hostează aplicația).

### ****Dependențe și interacțiuni cu alte funcționalități****

* Mașina „host” este necesar să nu aibă probleme, în caz contrar întreaga platformă va fi afectată, devenind nepracticabilă.
* Reverse proxy (nGINX) este o componentă la fel de critică precum baza de date, iar orice eroare survenite în cazul celor două componente nu va permite accesarea aplicației.
* Modelul de ML este obligatoriu să fie accesibil oricând, în caz contrar utilizatorii nu se vor putea bucura de secțiunea specifică încărcării imaginilor și a prelucrării acestora.

# Cerințe pentru interfețe

## Interfețe cu utilizatorul

Interfețele utilizatorului în platforma curentă se pot împărți în: pagina de înregistrare, pagina de autentificare, dashboard-ul utilizatorului, formularul de încărcare a imaginilor și afișare a răspunsurilor, pagina adresată comunității, panoul de administrare.

## Interfețe hardware

### ****Configurații Minime Recomandate****

* **Backend/hosting: 4 nuclee (CPU), 16 GB RAM, SSD 128 GB, NVIDIA GPU cu suport CUDA, Linux,**
* **PC/Laptop:** 4GB RAM, acces la Internet și browser
* **Dispozitive mobile:** Android 8+ / iOS 12+, 2GB RAM

### ****Dispozitive Externe Compatibile****

* **Dispozitiv 1** – Mouse
* **Dispozitiv 2** – Tastatură
* **Dispozitiv 3** – Cameră web / cameră foto

## Interfețe de comunicare

### ****Protocoale și Standarde de Comunicare****

* **Protocol 1** – HTTPS, protocol de bază pentru interacțiunea dintre browserul utilizatorului și serverul platformei
* **Protocol 2** – REST API, pentru trimiterea request-urilor pe rutele predefinite
* **Protocol 3** – Websockets, pentru comunicarea bidirecțională, în timp real, între client și server
* **Protocol 4** – SMTP, pentru trimiterea notificărilor pe mail

### ****Cerințe de Securitate în Comunicare****

* HTTPS, cu ajutorul SSL, pentru protejarea datelor transmise
* Autentificare sigură, prin intermediul „session cookies”
* Stocarea criptată a datelor utilizatorilor (bcrypt hash)
* Validarea imaginilor încărcate, prin scanarea conținutului malițios, limită de dimensiune și verificarea extensiei fișierului
* Protecție CSRF și XSS
* WSS pentru criptarea și securizarea canalelor de websocket

## Interfețe software

### Tehnologii Utilizate

* **Backend/Frontend:** Django
* **Bază de date:** PostgreSQL, Redis
* **UI:** Tailwind, Daisyui
* **Containerizare:** Docker
* **AI/ML:** Tensorflow, Pandas, numpy

### ****Servicii Externe și API-uri****

| **Serviciu/Interfață** | **Utilizare** |
| --- | --- |
| **OpenAI API** | ChatGPT pentru completarea informațiilor speciei găsite |
| **N8N** | Automatizare de fluxuri și procese |

# Cerințe non-funcționale

## Cerințe de performanță

Uptime 99.9% astfel încât platforma să fie accesibilă tuturor, oricând. Scalabilitate, pentru gesionarea unui număr mai mare de utilizatori fără a afecta performanța. Performanța în timp real, aplicația furnizând răspunsurile în mai puțin de două secunde. Toleranță la erori și recuperare, presupunând recuperarea rapidă după erori și o experiență perturbată a utilizatorilor la un nivel cât se poate de scăzut. Compatibilitatea pentru accesibilitatea multiplatformă și ușurința în utilizare sunt, de asemenea, relevante pentru o bună desfășurare.

## Cerințe de siguranță

Protecția datelor și confidențialitatea acestora (asigurarea faptului că toate datele utilizatorilor și aplicației sunt protejate împotriva accesului neautorizat, a pierderii sau alterării acestora), protecția fizică și acces controlat la infrastructură (serverele și infrastructura fizică sunt protejate împotriva accesului fizic neautorizat sau a altor tipuri de atacuri fizice), disponibilitate (aplicația rămâne disponibilă chiar și în cazul unui incident major.

## Cerințe de securitate

Autentificare și autorizare sigură (doar utilizatorii autorizați au acces la resursele platformei), criptarea comunicațiilor (informațiile sunt protejate împotriva interceptării sau accesului neautorizat), monitorizare și audit (detectarea și prevenirea activităților malițioase).

## Atribute de calitate ale software-ului

Performanță, scalabilitate, disponibilitate, fiabilitate, mentenabilitate, utilizabilitate, securitate, interoperabilitate, flexibilitate, portabilitate, testabilitate, eficiență.

# Alte cerințe

Acuratețea datelor, integrarea cu API-uri externe, mecanism de feedback al utilizatorilor, suport multilingv, accesibilitate (vizuală, mobilă), protecția proprietățiii intelectuale și considerații etice, CI/CD, optimizare (costuri, resurse).

# Anexe

## Anexa A: Glosar

UI (user interface), UX (user experience), DB (database), IA (artificial intelligence, inteligență artificială), GB (gigabyte), MB (megabyte), RAM (random-access memory), WSGI (web server gateway interface), SSD (solid state drive), VRAM (video random-access memory), CDN (content delivery network), ML (machine learning).

## Anexa C: Listă de Probleme

Automatizarea fluxurilor prin integrarea unui API extern (N8N), fie prin intermediul soluției cloud, fie prin intermediul unei instanțe self-hosted; Adăugarea unui rate limiter pentru limitarea request-urilor efectuate de către utilizatori, sporind astfel securitatea platformei; Diversificarea datelor de antrenament și îmbunătățirea calității predicțiilor modelului