

# Software Requirements Specification (SRS) Document

---

<Recunoașterea semnelor de circulație utilizând machine learning>

<05.05.2025>

# 1. Introducere

## 1.1 Scopul

Acest document specifică cerințele funcționale și non-funcționale pentru o aplicație web ce utilizează algoritmi de machine learning pentru recunoașterea automată a semnelor de circulație din imagini. Aplicația are ca scop creșterea siguranței rutiere și facilitarea procesului de antrenare pentru viitorii conducători auto prin oferirea unui sistem precis și rapid de identificare a semnelor rutiere.

## 1.2 Audiență țintă

Utilizatorii vizați includ elevi școli de șoferi, instructori auto, dezvoltatori software din domeniul auto, autorități de trafic și oricine dorește să exerseze identificarea semnelor de circulație.

## 1.3 Sfera de aplicare

Proiectul se aliniază obiectivelor generale ale sistemelor de educație rutieră și siguranță publică. Recunoașterea automată a semnelor poate fi utilizată în sisteme de asistență a conducătorilor auto, aplicații educaționale sau chiar vehicule autonome.

## 1.4 Referințe

- GTSRB Dataset
- OpenCV
- TensorFlow/Keras
- Documentația ISO pentru semnalizare rutieră

# 2. Descriere generală

## 2.1 Perspectiva produsului

Aplicația este un sistem bazat pe web care permite utilizatorilor să încarce imagini conținând semne rutiere, iar sistemul clasifică semnul utilizând un model ML antrenat. Poate fi folosită independent sau integrată în alte soluții educaționale sau auto.

## 2.2 Caracteristici ale produsului

- Încărcare imagini
- Clasificare automata
- Feedback vizual asupra rezultatului
- Istoric al imaginilor procesate
- Acces API public pentru integrare

## 2.3 Clase și caracteristici ale utilizatorilor

- Utilizatori neautentificați: acces limitat (doar testare)
- Utilizatori autentificați: acces complet la funcționalități
- Admini: gestionare utilizatori și date de antrenament

## 2.4 Mediul de operare

Platforma va rula în containere Docker pe un server Linux, cu Django ca framework backend, PostgreSQL ca bază de date, nGINX ca reverse proxy și Tensorflow pentru procesarea ML.

## 2.5 Constrângeri de proiectare

- Resurse ridicate pentru antrenare ML
- Calitate imagini influențează precizia
- Accesibilitate multiplatformă

## 2.6 Presupuneri și dependențe

- Utilizatorii au acces la Internet
- Set de date etichetat corect
- Dependențe software: Python, Tensorflow, Django, Docker

### 3. Cerințe ale sistemului

#### 3.1 Funcționalitatea 1: Încărcarea imaginii

- Utilizatorul încarcă o imagine cu semn rutier
- Sistemul afișează rezultatul identificării

#### 3.2 Funcționalitatea 2: Istoric identificări

- Utilizatorii autentificați pot accesa istoricul complet al imaginilor și rezultatelor

#### 3.3 Funcționalitatea

#### 3: Contribuții comunitare

- Utilizatorii pot trimite imagini noi pentru antrenarea modelului

#### 3.4 Funcționalitatea 4: API public

- Acces la o parte din funcționalitate prin chei API

#### 3.5 Funcționalitatea 5: Notificări

- Avertizări în caz de erori, rezultate incerte, etc.

#### 3.6 Detaliere cerințe

- REQ-1: Internet obligatoriu

- REQ-2: Imagine validă (JPG/PNG)
- REQ-3: Autentificare pentru funcționalități avansate

## 4. Cerințe pentru interfețe

### 4.1 Interfațe cu utilizatorul

- Pagina principală, upload, rezultate, autentificare, admin

### 4.2 Interfețe hardware

- PC: CPU quad-core, 8GB RAM
- GPU recomandat pentru antrenare

### 4.3 Interfețe de comunicare

- HTTPS, REST API, WebSocket pentru notificări

### 4.4 Interfețe software

- Django, Tensorflow, PostgreSQL, Redis

## 5. Cerințe non-funcționale

### 5.1 Performanță

- Timp de răspuns < 2s
- Uptime > 99.9%

### 5.2 Siguranță

- Criptare date, backup regulat

## 5.3 Securitate

- Autentificare, validare fișiere, protecție CSRF/XSS

## 5.4 Calitate software

- Ușor de folosit, portabil, testabil, scalabil

## 6. Alte cerințe

- Suport multilingv, accesibilitate, integrare CI/CD

## 7. Anexe

### 7.1 Glosar

ML, AI, UI, UX, SSL, REST, GPU, API

### 7.2 Probleme

Diversificarea datelor, balansarea claselor, scalarea infrastructurii