***Documentul de specificare a cerințelor***

***Software Requirements Specification***

***(SRS) Document***

***<Analiza in tenis folosind AI/ML>***

***<Lescai Valentin-Mihai>***

***<15.04.2025>***

| **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.](#_heading=h.q9tfiickdp7y) Introducere 4

[1.1](#_heading=h.4igw435r02un) Scopul 4

[1.2](#_heading=h.tl2g8uvpt2gh) Convenții ale documentului 4

[1.3](#_heading=h.4wsoh4v4ui5w) Audiență țintă 4

[1.4](#_heading=h.a8ezblimdpjo) Sfera de aplicare 4

[1.5](#_heading=h.cb5eozmzc3gq) Referințe 4

[**2**](#_heading=h.yi4zt29t1g1e) **Descriere generală 4**

[**2.1**](#_heading=h.vqvtkjscz8w) **Perspectiva produsului 4**

[**2.2**](#_heading=h.583wqz9lxelg) **Caracteristici ale produsului 4**

[**2.3**](#_heading=h.7029y8am10v2) **Clase și caracteristici ale utilizatorilor 4**

[**2.4**](#_heading=h.qwnyr9w2byc4) **Mediul de operare 5**

[**2.5**](#_heading=h.okxysvydv0lq) **Constrângeri de proiectare și de implementare 5**

[**2.6**](#_heading=h.gvqtexcs9w00) **Presupuneri și dependențe 5**

[**3**](#_heading=h.ratdjc2fz6ql) **Cerințele sistemului 5**

[**3.1**](#_heading=h.82gzmk9zd8ll) **Funcționalitatea 1 5**

[**3.1.1**](#_heading=h.i6751q5pbwgh) **Descriere generală 5**

[**3.1.2**](#_heading=h.7usytts0dwef) **Flux de interacțiune (scenarii de utilizare) 5**

[**3.1.3**](#_heading=h.ptyo1d9ocary) **Condiții prealabile și constrângeri 6**

[**3.1.4**](#_heading=h.atw2xkyg0s93) **Detaliere cerință 6**

[**3.1.5 Scenarii de eroare și gestionarea excepțiilor 6**](#_heading=h.w0inb9s1i6vv)

[**3.1.5**](#_heading=h.17dq63110ekv) **Dependențe și interacțiuni cu alte funcționalități 6**

[**3.2**](#_heading=h.wxoefau1xli4) **Cerința funcțională 2 6**

[3.3](#_heading=h.rt4875ikheym) .... 6

[4](#_heading=h.5evtz622f8ti) Cerințe pentru interfețe 6

[4.1](#_heading=h.7cx1zq5d5anp) Interfețe cu utilizatorul 6

[4.2](#_heading=h.6gki6rk14e85) Interfețe hardware 6

[4.2.1](#_heading=h.4jbf57upkcec) Configurații Minime Recomandate 7

[4.2.2](#_heading=h.2fxj4tbonunk) Dispozitive Externe Compatibile 7

[4.3](#_heading=h.ocxvgf2138e8) Interfețe de comunicare 7

[4.3.1](#_heading=h.9hkreyigbgw3) Protocoale și Standarde de Comunicare 7

[4.3.2](#_heading=h.4eaw5ipldoml) Cerințe de Securitate în Comunicare 7

[4.4](#_heading=h.9864svnlawxs) Interfețe software 7

[4.4.1](#_heading=h.cerhe7aj3c9h) Tehnologii Utilizate 7

[4.4.2](#_heading=h.2yxhr7dt4ulm) Servicii Externe și API-uri 7

[5](#_heading=h.iij59023jhsb) Cerințe non-funcționale 8

[5.1](#_heading=h.2k0tq3v848jt) Cerințe de performanță 8

[5.2](#_heading=h.8vzzibw4f5ng) Cerințe de siguranță 8

[5.3](#_heading=h.a8ftflw7yi2) Cerințe de securitate 8

[5.4](#_heading=h.na4dy5xn54ae) Atribute de calitate ale software-ului 8

[6](#_heading=h.j85kqkwooce0) Alte cerințe 8

[7](#_heading=h.fhwyrypk9r25) Anexe 8

[7.1](#_heading=h.5m5siitpeu9s) Anexa A: Glosar 8

[7.2](#_heading=h.bj0d58p84nj9) Anexa B: Modele de Analiză 8

[7.3](#_heading=h.7lzlzmf3mt7y) Anexa C: Listă de Probleme 9

# Introducere

## Scopul

Documentul descrie cerințele și specificațiile pentru un sistem de analiză și detecție a statisticilor în tenis. Sistemul utilizează tehnici de AI/ML pentru a analiza și urmări jucătorii și mingea dintr-un video scurt, pentru a calcula date precum viteza loviturii și distanța parcursă de jucători.

## Audiență țintă

Documentul este destinat echipelor de dezvoltare, managerilor de proiect și analiștilor sportivi care vor utiliza acest sistem pentru analiza jocurilor. De asemenea, va fi folosit de antrenori și jucători pentru a îmbunătăți performanțele lor prin analiza detaliată a mișcărilor și vitezelor în timpul jocurilor.

## Sfera de aplicare

Sistemul va fi capabil să proceseze un videoclip de 10 secunde, oferind statistici precise și vizualizări în timp real ale mișcării jucătorilor și mingii. Acesta va analiza toate cadrele din videoclip pentru a calcula viteza loviturilor, distanța parcursă de jucători și pentru a le vizualiza mișcările pe un teren grafic. Sistemul nu va fi dependent de echipamente de înaltă performanță, dar va necesita o unitate GPU pentru a facilita procesarea rapidă a video-urilor.

# Descriere generală

## Perspectiva produsului

Sistemul se bazează pe tehnologia de învățare automată și viziune computerizată, utilizând modelele YOLOv8 pentru detectarea jucătorilor și YOLOv5 pentru mingea de tenis, ambele antrenate pe date relevante din domeniu. Modelul de keypoints ajută la stabilirea coordonatelor terenului pentru a calcula mișcările exacte ale jucătorilor. În final, datele obținute sunt convertite în informații utile pentru utilizator, cum ar fi viteza mingii și distanța parcursă de jucători, care sunt afișate pe un teren grafic.

## Caracteristici ale produsului

Sistemul va îndeplini următoarele funcții esențiale:

* Detectarea jucătorilor și a mingii în video.
* Calcularea vitezei loviturilor pe baza mișcării mingii.
* Calcularea distanței parcurse de jucători pe teren.
* Vizualizarea pe ecran a mișcării jucătorilor și mingii, inclusiv a unui mini teren de tenis, cu simboluri pentru fiecare jucător și minge.
* Oferirea unor statistici detaliate în timp real despre joc.

## Clase și caracteristici ale utilizatorilor

* Analiști sportivi: Vor analiza performanțele jucătorilor prin statistici detaliate (viteza mingii, distanța parcursă, etc.).
* Antrenori: Vor folosi datele pentru a ajuta jucătorii să-și îmbunătățească jocul, observând unde sunt punctele slabe (de exemplu, viteza de deplasare).
* Jucători: Vor folosi informațiile pentru a-și urmări performanțele și pentru a înțelege mai bine cum să-și îmbunătățească jocul.
* Dezvoltatori: Vor contribui la dezvoltarea și întreținerea software-ului.

## Mediul de operare

Sistemul va funcționa pe platformele de operare Windows și Linux, fiind compatibil cu majoritatea configurațiilor hardware de bază. Este recomandat să aibă cel puțin un procesor Intel i5 sau echivalent, 8 GB RAM și un GPU dedicat pentru a asigura procesarea rapidă a videoclipurilor. Aplicația va fi integrată cu OpenCV și TensorFlow pentru procesarea video și implementarea modelelor AI/ML.

# Cerințele sistemului

Toate cerințele din cadrul sistemului sau sub-sistemului pentru a determina rezultatul pe care se așteaptă să-l ofere produsul în raport cu intrarea dată. Acestea constau în cerințe de design, cerințe grafice, cerințe de sistem de operare.

*Acest șablon ilustrează organizarea cerințelor funcționale pentru produs în funcție de caracteristicile sistemului, principalele servicii furnizate de produs. Poți prefera să organizezi această secțiune în funcție de cazul de utilizare, modul de operare, clasă de utilizatori, clasă de obiecte, ierarhie funcțională sau combinații ale acestora, orice ar face cel mai mult sens logic pentru produsul tău.*

## Funcționalitatea 1

*Detectarea jucătorilor și mingii*

### 3.1.0 Descriere generală

* Descriere generala: Sistemul trebuie să detecteze jucătorii și mingea într-un videoclip folosind modelele preantrenate YOLOv8 pentru jucători și YOLOv5 pentru minge. Această funcționalitate va permite sistemului să urmărească pozițiile jucătorilor și mingii în fiecare cadru al videoclipului.
* Flux de interacțiune: Sistemul va procesa fiecare cadru al videoclipului, extrăgând coordonatele pentru fiecare jucător și minge în timp real. Utilizatorii vor vedea pe ecran, într-un mini teren de tenis, pozițiile detectate ale jucătorilor și ale mingii.
* Condiții prealabile: Video-ul introdus în sistem trebuie să fie clar, de înaltă rezoluție și să aibă o durată de aproximativ 10 secunde. Sistemul trebuie să poată detecta rapid mișcarea obiectelor (mingea și jucătorii).

### 3.1.1 Flux de interacțiune (scenarii de utilizare)

* Secvența tipică de acțiuni ale utilizatorului
* Utilizatorul încarcă un videoclip de 10 secunde
* Sistemul detectează jucătorii și mingea din fiecare cadru al videoclipului.
* Coordonatele jucătorilor și ale mingii sunt afișate pe terenul grafic în timp real.
* Răspunsurile așteptate din partea sistemului
* La încărcarea videoclipului, sistemul începe procesarea acestuia.
* După procesare, sunt afișate coordonatele corecte pentru fiecare jucător și minge pe terenul grafic.
* Sistemul afișează un mesaj de succes în cazul în care procesarea s-a finalizat corect sau un mesaj de eroare în caz de probleme.

### 3.1.2 Condiții prealabile și constrângeri

* Videoclipul trebuie să fie în format MP4, AVI sau MOV
* Sistemul trebuie să fie capabil să proceseze videoclipuri cu o frecvență minimă de 30fps.
* Rezoluția videoclipului trebuie să fie de cel puțin 720p pentru a asigura detecția corectă a obiectelor.
* Sistemul necesită o unitate GPU pentru procesarea rapidă a videoclipurilor (recomandat NVIDIA GTX 1060 sau mai bun).
* Sistemul trebuie să detecteze jucătorii și mingea chiar și în condiții de iluminare mai puțin favorabile.

### 3.1.3 Detaliere cerință

| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| --- | --- | --- | --- |
| **REQ-1** | Sistemul trebuie să detecteze jucătorii și mingea din videoclip | Ridicată | Pozițiile jucătorilor și mingii trebuie să fie corecte în cel puțin 95% dintre cadre. |
| **REQ-2** | Coordonatele detectate ale jucătorilor și mingii trebuie să fie afișate pe mini terenul grafic. | Ridicată | Coordonatele trebuie să fie afișate corect și să se alinieze cu pozițiile vizibile în videoclip. |

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

* Ce se întâmplă dacă utilizatorul introduce date invalide?
  + Dacă videoclipul nu respectă cerințele minime de rezoluție sau durată, sistemul va returna un mesaj de eroare „Videoclip invalid”.
* Ce erori sunt prevăzute și cum sunt gestionate?
  + Dacă sistemul nu poate detecta un obiect din cauza unor cadre blurate sau cu mișcare rapidă, se va returna un mesaj de eroare „Imposibil de detectat obiectele”.
  + În cazul în care modelul AI nu poate procesa videoclipul din cauza unui tip de fișier incompatibil, sistemul va solicita un alt format de fișier.

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

**Dependențe**:

* Funcționalitatea de detectare a jucătorilor și mingii depinde de modelele AI antrenate (YOLOv8 și YOLOv5) și de procesarea video eficientă.

**Interacțiuni cu alte funcționalități**:

* **Funcționalitatea 2 - Calcularea vitezei loviturii**: Pozițiile jucătorilor și mingii obținute în această funcționalitate vor fi folosite pentru a calcula viteza mingii.
* **Funcționalitatea 3 - Calcularea distanței parcurse de jucător**: Detectarea corectă a jucătorilor va permite calculul distanței acestora pe teren.

## 

## 3.2 Funcționalitatea 2

*Calcularea vitezei loviturii*

### 3.2.0 Descriere generală

* Descriere generala: După detectarea mingii, sistemul va calcula viteza acesteia pe baza diferenței de poziție între cadrele consecutive și timpul necesar pentru a parcurge acea distanță. Viteza va fi afișată în kilometri pe oră (km/h).
* Flux de interacțiune: Sistemul va analiza mișcarea mingii între cadrele video și va calcula viteza acesteia pe baza diferențelor de poziție. Rezultatele vor fi afișate pe ecran.
* Condiții prealabile:
  + Sistemul trebuie să fi procesat deja videoclipul și să fi detectat coordonatele mingii.
  + Viteza va fi calculată pe baza unui model de distanță fixă (metri) și a timpului dintre cadrele consecutive.

### 3.2.1 Flux de interacțiune (scenarii de utilizare)

* Secvența tipică de acțiuni ale utilizatorului
* Utilizatorul încarcă un videoclip și sistemul detectează mingea.
* După procesare, sistemul calculează viteza mingii între cadrele consecutive.
* Viteza este afișată în km/h pe ecran.
* Răspunsurile așteptate din partea sistemului
* Sistemul va calcula viteza mingii și o va afișa pe ecran în timp real, alături de pozițiile detectate ale mingii.
* Dacă există erori în procesul de calcul al vitezei, sistemul va returna un mesaj „Viteza nu poate fi calculată din cauza erorii de procesare a cadrelor”.

### 3.2.2 Condiții prealabile și constrângeri

* Sistemul trebuie să detecteze cel puțin două poziții ale mingii pentru a calcula viteza.
* Calculul vitezei va fi precis doar pentru cadrele clare, fără distorsiuni sau cadru lipsă.
* Viteza va fi calculată pe baza unui model fix de distanță (în metri), iar rezultatele vor fi exprimate în km/h.

### 3.2.3 Detaliere cerință

| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| --- | --- | --- | --- |
| **REQ-1** | Sistemul va calcula viteza pe baza diferentei de pozitie intre cadrele consecutive | Ridicată | Viteza calculată trebuie să fie corectă în cel puțin 95% dintre cadrele procesate. |
| **REQ-2** | Viteza trebuie să fie calculată și afișată în km/h. | Ridicată | Viteza trebuie să fie afișată pe ecran fără întârzieri, imediat după calcul. |

### 3.2.4 Scenarii de eroare și gestionarea excepțiilor

* Ce se întâmplă dacă utilizatorul introduce date invalide?
  + Dacă nu există suficiente cadre pentru a calcula viteza (de exemplu, din cauza unui cadru pierdut), sistemul va returna un mesaj de eroare „Nu există suficiente date pentru calcularea vitezei”.
* Ce erori sunt prevăzute și cum sunt gestionate?
  + Dacă viteza nu poate fi calculată din cauza lipsei de date corecte (de exemplu, mingea nu a fost detectată corect), va fi afișat un mesaj „Imposibil de calculat viteza”.

### 3.2.5 Dependențe și interacțiuni cu alte funcționalități

**Dependențe**:

* Calculul vitezei depinde de corectitudinea detecției mingii în funcționalitatea anterioară (Funcționalitatea 1).

**Interacțiuni cu alte funcționalități**:

* **Funcționalitatea 3 - Calcularea distanței parcurse de jucător**: Pozițiile mingii și jucătorilor detectate în funcționalitatea anterioară vor fi folosite pentru a calcula mișcarea mingii și a jucătorilor.

## 3.3 Funcționalitatea 3

*Calcularea distanței parcurse de jucător*

### 3.3.0 Descriere generală

* Descriere generala: Această funcționalitate permite calcularea distanței parcurse de fiecare jucător pe teren, pe baza coordonatelor detectate în fiecare cadru al videoclipului. Distanța va fi calculată prin măsurarea diferenței între pozițiile consecutive ale jucătorului, folosind coordonatele acestora. Rezultatele vor fi afișate în unități de măsură reale (metri), pe terenul grafic, pentru a urmări cât de mult s-a deplasat fiecare jucător.
* Flux de interacțiune:
  + Sistemul primește coordonatele jucătorilor, detectate în fiecare cadru din videoclip.
  + Pe baza acestor coordonate, sistemul calculează distanța parcursă de jucător pe teren între cadre consecutive.
  + Distanta este exprimată în metri și este afișată pe terenul grafic, iar utilizatorul poate urmări mișcarea jucătorilor pe teren.
* Condiții prealabile:
  + Sistemul a procesat deja videoclipul și a detectat pozițiile jucătorilor
  + Coordonatele jucătorilor trebuie să fie corect extrase și să fie valabile pentru fiecare cadru al videoclipului.
  + Sistemul trebuie să aibă o scalare corectă a terenului de tenis pentru a converti corect pixelii în unități de măsură reale (metri).

### 3.3.1 Flux de interacțiune (scenarii de utilizare)

* Secvența tipică de acțiuni ale utilizatorului
* Utilizatorul încarcă un videoclip în aplicație.
* Sistemul detectează jucătorii și extrage coordonatele acestora din fiecare cadru al videoclipului.
* Sistemul calculează distanța parcursă de fiecare jucător între cadre consecutive și o afișează pe terenul grafic.
* Utilizatorul poate vizualiza distanța parcursă în timp real pe mini terenul de tenis.
* Răspunsurile așteptate din partea sistemului
* Sistemul va calcula și actualiza continuu distanța parcursă de fiecare jucător.
* Distanța va fi afișată pe terenul grafic în metri, iar mișcarea jucătorilor va fi vizibilă pe teren.
* Dacă există o eroare în calculul distanței (de exemplu, coordonatele sunt incorecte), sistemul va afișa un mesaj de eroare.

### 3.3.2 Condiții prealabile și constrângeri

* Sistemul trebuie să fi procesat corect videoclipul și să fi detectat pozițiile corecte ale jucătorilor în fiecare cadru.
* Conversia coordonatelor din pixeli în metri trebuie să fie realizată corect, având în vedere dimensiunile reale ale terenului de tenis.
* Distanța calculată va depinde de precizia detectării jucătorilor în fiecare cadru, deci erorile de detectare vor afecta rezultatul final.
* Sistemul trebuie să poată calcula distanța într-un timp rezonabil, fără întârzieri semnificative în actualizarea terenului grafic.

### 3.3.3 Detaliere cerință

| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| --- | --- | --- | --- |
| **REQ-1** | Sistemul va calcula distanța parcursă de fiecare jucător pe teren pe baza coordonatelor acestora. | Ridicată | Distanța parcursă de jucători trebuie să fie precis calculată, iar valorile afișate pe teren să fie corecte și consistente. |
| **REQ-2** | Distanta calculată va fi exprimată în metri și afișată pe terenul grafic. | Ridicată | Distanța trebuie să fie actualizată pe terenul grafic la fiecare cadru procesat. |

### 

### 3.3.4 Scenarii de eroare și gestionarea excepțiilor

* Ce se întâmplă dacă utilizatorul introduce date invalide?
  + Dacă există o eroare în detectarea coordonatelor jucătorilor sau a mingii, sistemul va returna un mesaj de eroare „Date invalide pentru calcularea distanței” și va opri procesarea până când datele corecte sunt furnizate.
* Ce erori sunt prevăzute și cum sunt gestionate?
  + Dacă sistemul nu poate calcula distanța din cauza unui număr insuficient de cadre valide (de exemplu, jucătorul nu este detectat în unele cadre), va apărea un mesaj „Imposibil de calculat distanța din cauza erorilor de detectare”.
  + Dacă distanța calculată depășește limitele fizice ale terenului, sistemul va semnala această eroare și va ajusta calculul.

### 3.3.5 Dependențe și interacțiuni cu alte funcționalități

**Dependențe**:

* Funcționalitatea 1: Detectarea corectă a jucătorilor este esențială pentru calculul distanței. Dacă pozițiile jucătorilor nu sunt detectate corect, distanța parcursă nu va fi calculată corect.
* Funcționalitatea 2: Calculul vitezei mingii poate influența analiza mișcării jucătorilor, în special în scenariile în care jucătorii reacționează rapid la mișcarea mingii.

**Interacțiuni cu alte funcționalități**:

* Funcționalitatea 4: Calculul distanței parcurse de jucători este vizualizat pe terenul grafic, iar acest lucru depinde de funcționalitatea de desenare a terenului.
* Funcționalitatea 1 și 2: Coordonatele jucătorilor și ale mingii sunt esențiale pentru calcularea corectă a distanței și vitezei.

## 3.4 Funcționalitatea 4

*Calcularea distanței parcurse de jucător*

### 3.3.0 Descriere generală

* Descriere generala: Această funcționalitate permite desenarea unui teren de tenis grafic pe care sunt reprezentate mișcările jucătorilor și ale mingii. După ce pozițiile jucătorilor și ale mingii sunt detectate și calculate, coordonatele sunt convertite din pixeli în unități reale (metri), folosind o scalare bazată pe dimensiunile reale ale terenului. Sistemul va actualiza terenul grafic în timp real, arătând pozițiile jucătorilor și mingii pe un mini teren de tenis.
* Flux de interacțiune:
  + După detectarea jucătorilor și mingii în videoclip, coordonatele lor sunt extrase.
  + Aceste coordonate sunt convertite din pixeli în unități reale (metri) pe baza unei scalări a terenului.
  + Terenul grafic este actualizat continuu, arătând mișcarea jucătorilor și mingii, cu simboluri corespunzătoare.
* Condiții prealabile:
  + Sistemul trebuie să fi detectat deja coordonatele jucătorilor și ale mingii.
  + Terenul grafic trebuie să aibă dimensiuni prestabilite, iar sistemul trebuie să poată scală corect coordonatele din pixeli în unități de măsură reale (metri).
  + Sistemul trebuie să aibă o componentă grafică (de exemplu, utilizarea unui framework grafic precum Matplotlib sau Pygame pentru desenare).

### 3.3.1 Flux de interacțiune (scenarii de utilizare)

* Secvența tipică de acțiuni ale utilizatorului
* Utilizatorul încarcă videoclipul.
* Sistemul procesează videoclipul și detectează jucătorii și mingea.
* Sistemul convertește pozițiile jucătorilor și ale mingii din pixeli în unități reale, folosind scalarea terenului.
* Pozițiile sunt reprezentate grafic pe terenul virtual, actualizându-se în timp real pe măsură ce videoclipul progresează.
* Utilizatorul poate urmări mișcările jucătorilor și mingii pe terenul grafic.
* Răspunsurile așteptate din partea sistemului
* Sistemul va desena terenul grafic pe baza pozițiilor jucătorilor și mingii.
* Terenul va fi actualizat continuu pe măsură ce se procesează fiecare cadru din videoclip.
* Sistemul va afișa simboluri corespunzătoare pentru fiecare jucător și pentru minge pe teren.
* Pozițiile vor fi corect scalate și afișate în unități de măsură reale.

### 3.3.2 Condiții prealabile și constrângeri

* Videoclipul trebuie să fi fost procesat corect, iar pozițiile jucătorilor și ale mingii să fi fost detectate.
* Dimensiunile terenului de tenis trebuie să fie standardizate și definite
* Terenul grafic trebuie să fie capabil să gestioneze actualizări rapide pentru a reflecta mișcarea în timp real a jucătorilor și mingii.
* Conversia pixelilor în metri presupune că terenul grafic are o scalare corectă, iar dimensiunile terenului sunt constante pe întreaga durată a videoclipului.
* Sistemul trebuie să gestioneze eficient resursele pentru a asigura o actualizare rapidă a terenului grafic în timp real.

### 3.3.3 Detaliere cerință

| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| --- | --- | --- | --- |
| **REQ-1** | Sistemul trebuie să convertească coordonatele jucătorilor și ale mingii din pixeli în metri pe baza scalării terenului. | Ridicată | Conversia trebuie să fie exactă pentru toate cadrele, iar pozițiile trebuie să fie precise pe terenul grafic. |
| **REQ-2** | Terenul grafic trebuie să fie actualizat continuu pe măsură ce videoclipul progresează, arătând mișcarea jucătorilor și mingii. | Ridicată | Terenul trebuie să fie actualizat la fiecare cadru, fără întârzieri semnificative (sub 1 secundă per cadru). |

### 

### 3.3.4 Scenarii de eroare și gestionarea excepțiilor

* Ce se întâmplă dacă utilizatorul introduce date invalide?
  + Dacă coordonatele detectate pentru jucători sau minge sunt greșite sau invalide, sistemul va afișa un mesaj de eroare „Pozițiile detectate sunt invalide pentru desenarea terenului”.
* Ce erori sunt prevăzute și cum sunt gestionate?
  + Dacă terenul grafic nu poate fi actualizat din cauza unor erori de scalare a coordonatelor, sistemul va încerca să reproceze conversia coordonatelor și va alerta utilizatorul cu mesajul „Eroare la scalarea terenului grafic”.

### 3.3.5 Dependențe și interacțiuni cu alte funcționalități

**Dependențe**:

* Funcționalitatea de desenare a terenului depinde de detecția corectă a jucătorilor și a mingii din videoclip (Funcționalitatea 1).
* Calculul corect al distanței parcurse de jucători (Funcționalitatea 3) va influența plasarea acestora pe terenul grafic.

**Interacțiuni cu alte funcționalități**:

* Funcționalitatea 1 - Detectarea jucătorilor și mingii: Pozițiile jucătorilor și ale mingii sunt utilizate pentru a le reprezenta grafic pe terenul virtual.
* Funcționalitatea 3 - Calcularea distanței parcurse de jucători: Distanța parcursă de jucători este reflectată pe terenul grafic prin mișcarea acestora, în funcție de datele obținute.
* Funcționalitatea 2 - Calcularea vitezei mingii: Viteza mingii poate fi afișată pe terenul grafic, ca o informație adițională alături de mișcarea mingii.

# Cerințe pentru interfețe

Această secțiune descrie modul în care sistemul interacționează cu utilizatorii, hardware-ul, alte sisteme software și rețele de comunicație.

## Interfețe cu utilizatorul

Interfața cu utilizatorul (UI) va fi grafică, având rolul de a facilita interacțiunea cu utilizatorii finali, inclusiv analiști, antrenori și jucători. UI-ul va include ecrane și elemente vizuale care vor ajuta utilizatorii să înțeleagă rezultatele analizei video-ului de tenis, cum ar fi viteza mingii și distanța parcursă de jucători, și să vizualizeze mișcările acestora pe terenul grafic.

## Interfețe hardware

### **Configurații Minime Recomandate**

* **PC/Laptop:**
  + **Procesor: minim Intel I5 / AMD Ryzen 5**
  + **RAM: minim 8GB RAM**
  + **GPU: AMD Radeon 610M Graphics**
  + **Spatiu stocare: minim 10GB**
  + **OS: Win 10/11**