**Documentul de specificare a cerințelor**

**Software Requirements Specification**

**(SRS) Document**

**Imbunatatirea si optimizarea unui brat robotic contolat prin ai pentru manipularea pieselor de sah**

**15.04.2025**

**Vilcu Elena Roxana**

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

| Versiune | Autor(i) principali | Descriere versiune | Dată |
| --- | --- | --- | --- |
| 1.0 | Vilcu  Elena Roxana | Versiune initiala a documentului completat pentru laborator | 13.04.2025 |

|  |
| --- |
| **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

## Scopul

Acest document descrie specificațiile detaliate pentru proiectul Îmbunătățirea și optimizarea unui braț robotic controlat prin AI pentru manipularea pieselor de șah. Scopul principal este dezvoltarea unui sistem robotic inteligent care să automatizeze jocul de șah prin detectarea mișcărilor jucătorului și răspuns automat al brațului robotic, în funcție de deciziile AI-ului de joc.

## Convenții ale documentului

Documentul urmează convențiile standard de redactare a cerințelor software, folosind identificatori de tip REQ-x pentru cerințele funcționale și specificând termenii tehnici în glosar.

## Audiență țintă

Documentul este destinat evaluatorilor academici.

## Sfera de aplicare

Proiectul se concentrează pe dezvoltarea unui braț robotic inteligent, controlat de un AI de șah, capabil să interacționeze fizic cu piesele de pe tabla de joc. Sistemul citește poziția pieselor printr-o matrice de senzori reed, ia decizii prin intermediul modelului AI și execută mutările fizic cu ajutorul brațului robotic.

## Referințe

* Documentatie Arduino Mega
* Documentatie driver PCA9685
* Documentatie multiplexor CD74HC4067
* Cod Python pentru integrarea cu Arduino
* Model AI de sah pregatit anterior
* Video referinta brat robotic: DIY 5DOF Robotic Arm (Arduino)

# Descriere generală

## Perspectiva produsului

Produsul face parte dintr-un sistem educațional și demonstrativ destinat învățării și aplicării conceptelor de automatizare, inteligență artificială și robotică. Sistemul integrează un braț robotic controlat de un AI.

## Caracteristici ale produsului

Produsul este un sistem integrat care combină un braț robotic, senzori reed și un model de inteligență artificială pentru a automatiza jocul de șah. Sistemul detectează mișcările jucătorului uman printr-o matrice de senzori reed montați sub tabla de șah și trimite aceste informații către un model AI antrenat pentru decizii de joc. AI-ul analizează poziția pieselor și decide următoarea mutare, iar instrucțiunile sunt transmise brațului robotic, care execută mutarea pe tablă. La finalul fiecărui ciclu de joc, sistemul oferă un semnal sonor pentru a informa utilizatorul că este rândul său. Sistemul poate funcționa și în regim de testare, fără utilizarea senzorilor reed, pentru verificarea comunicării dintre AI și braț.

## Clase și caracteristici ale utilizatorilor

Utilizatorii vizați ai acestui sistem sunt studenții și profesorii interesați de aplicații de robotică și inteligență artificială, dezvoltatorii hardware și software care doresc să extindă sistemul și evaluatorii tehnici care analizează funcționalitatea și performanța proiectului.

## Mediul de operare

Hardware-ul include Arduino Mega, multiplexoare CD74HC4067, driver servo PCA9685, șase servomotoare, senzori reed, buton de comandă și buzzer. Software-ul rulează pe un calculator cu sistem de operare Windows sau Linux, cu Arduino IDE și Python instalate.

## Constrângeri de proiectare și de implementare

Principalele constrângeri ale proiectului sunt legate de precizia mecanică a brațului robotic, care influențează direct acuratețea executării mutărilor pe tabla de șah. De asemenea, acuratețea citirii senzorilor reed este esențială pentru a evita interpretarea eronată a poziției pieselor. Comunicația dintre Arduino și calculator se realizează prin interfață serială, ceea ce poate introduce o anumită latență în transmiterea datelor. Funcționarea corectă a întregului sistem depinde și de o alimentare stabilă a componentelor hardware, precum și de configurarea corectă a software-ului pe calculatorul utilizat.

## Presupuneri și dependențe

Se presupune că utilizatorul cunoaște regulile de bază ale șahului. Funcționarea corectă a sistemului depinde de integritatea conexiunilor electrice și de configurarea corespunzătoare a software-ului. Toate componentele hardware trebuie să fie alimentate corect pentru a asigura funcționarea continuă a sistemului.

# Cerințele sistemului

## Detectarea pozitiei pieselor

### Descriere generală

Sistemul detectează poziția curentă a pieselor pe tabla de șah utilizând o matrice de 64 de senzori reed conectați prin multiplexoare la Arduino Mega.

### Flux de interacțiune (scenarii de utilizare)

Utilizatorul mută o piesă și apasă butonul de comandă. Sistemul citește valorile de la senzori și trimite datele către AI.

### Condiții prealabile și constrângeri

Sistemul trebuie să fie alimentat și conectat la calculator. Senzorii reed trebuie să fie montați corect pentru a asigura citirea precisă a datelor

### Detaliere cerință

|  |  |  |  |
| --- | --- | --- | --- |
| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| **REQ-1** | Sistemul trebuie să citească toate cele 64 de poziții de pe tabla de șah. | ridicată | datele sunt recepționate integral fără erori |
| **REQ-2** | Sistemul trebuie sa trimita matricea de senzori catre AI | ridicată | AI-ul primește matricea corectă |

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

În cazul în care utilizatorul nu apasă butonul de comandă, sistemul rămâne în așteptare și nu inițiază procesul de citire a senzorilor. Dacă o piesă nu este detectată corect din cauza unei poziții necorespunzătoare pe senzor, sistemul continuă fluxul de procesare cu datele disponibile fără a întrerupe funcționarea. Sistemul nu include o funcționalitate de semnalare a erorilor prin mesaje de alertă, ci se bazează pe buna funcționare a componentelor hardware și pe interacțiunea directă a utilizatorului pentru identificarea eventualelor probleme.

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

Funcționalitatea de citire a senzorilor depinde direct de apasarea butonului de comandă de către utilizator, deoarece această acțiune inițiază procesul de citire a poziției pieselor. Sistemul funcționează în mod secvențial, așteptând interacțiunea utilizatorului pentru fiecare etapă a jocului. De asemenea, citirea corectă a senzorilor este esențială pentru transmiterea datelor către modelul AI, care are nevoie de o matrice completă sau parțială pentru a calcula mutarea următoare. În lipsa unei detecții perfecte, sistemul continuă fluxul de procesare cu informațiile disponibile, fără a depinde de alte mecanisme de verificare sau corecție automată.

## Decizia AI și execuția mutării

### Descriere generală

Sistemul AI calculează următoarea mutare și transmite instrucțiunile către brațul robotic pentru a executa mutarea fizic pe tabla de șah.

### Flux de interacțiune

După recepționarea matricei de la senzorii reed, AI-ul calculează mutarea și trimite comenzile de execuție către brațul robotic

### Condiții prealabile și constrângeri

Calculatorul trebuie să ruleze modelul AI. Driver-ul PCA9685 trebuie să fie conectat și funcțional pentru controlul servomotoarelor.

### Detaliere cerinta

|  |  |  |  |
| --- | --- | --- | --- |
| **Cerință** | **Descriere** | **Prioritate** | **Criterii de acceptanță** |
| **REQ-3** | AI-ul trebuie să primească datele și să calculeze mutarea corectă. | ridicată | AI-ul propune o mutare validă |
| **REQ-4** | Brațul robotic trebuie să execute mutarea transmisă de AI. | ridicată | brațul deplasează piesa corect pe tabla de șah |

### Scenarii de eroare si gestionarea exceptiilor

Dacă AI-ul nu primește datele de la senzori, sistemul rămâne în așteptare până când sunt disponibile informațiile necesare. Dacă brațul robotic întâmpină dificultăți mecanice, cum ar fi o aliniere greșită a piesei, acesta își finalizează mișcarea cu poziționarea actuală și se retrage în poziția de repaus fără a întrerupe ciclul de joc.

### Depedente si interactiuni

Această funcționalitate depinde de corecta recepționare a datelor de la senzorii reed și de transmiterea acestora către AI. Execuția mutării este dependentă de buna funcționare a driver-ului PCA9685 și de conectarea corectă a servomotoarelor brațului robotic.

# Cerințe pentru interfețe

## Interfețe cu utilizatorul

Interfața cu utilizatorul este reprezentată de consola aplicației Python, care afișează informații despre starea sistemului și pașii parcurși în procesul de joc. De asemenea, utilizatorul primește un semnal sonor prin intermediul buzzer-ului pentru a indica momentul în care este rândul său să mute piesa. Sistemul este gândit pentru a funcționa fără interfață grafică dedicată, utilizatorul interacționând direct cu placa Arduino prin butonul de comandă și prin observarea mișcărilor brațului robotic

## Interfețe hardware

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

Sistemul necesită un calculator echipat cu procesor minim Intel i3, memorie RAM de minimum 4 GB și sistem de operare Windows 10 sau Linux. Pentru funcționarea optimă a sistemului este necesar ca Arduino IDE și Python să fie instalate corespunzător.

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

* Arduino Mega
* Multiplexoare CD74HC4067
* Driver servo PCA9685
* Senzori reed
* Brat robotic cu sase servomotoare

## Interfețe de comunicare

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

Comunicarea dintre Arduino și calculator se realizează prin interfață serială USB. Această conexiune permite transmiterea datelor de la senzori către aplicația Python, precum și transmiterea comenzilor de control către brațul robotic.

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

Comunicarea dintre componente se face local, fără să fie nevoie de criptare sau autentificare, deoarece sistemul nu este conectat la internet și nu este expus la riscuri externe.

## Interfețe software

### Tehnologii Utilizate

Sistemul utilizează Arduino IDE pentru programarea plăcii Arduino Mega și Python pentru implementarea logicii de procesare a datelor și integrarea modelului AI.

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

Sistemul funcționează complet local și nu necesită integrarea cu servicii externe sau API-uri

# Cerințe non-funcționale

## Cerințe de performanță

Sistemul trebuie să detecteze modificările de pe tabla de șah în maximum cinci secunde de la apăsarea butonului de comandă. După calcularea mutării de către AI, brațul robotic trebuie să finalizeze executarea mutării în maximum zece secunde pentru a menține fluența jocului.

## Cerințe de siguranță

Pentru prevenirea deteriorării componentelor, brațul robotic trebuie să se oprească automat în caz de blocaj mecanic. Sistemul trebuie să fie alimentat corect, respectând specificațiile fiecărui component hardware, pentru a preveni suprasolicitarea și supraîncălzirea circuitelor.

## Cerințe de securitate

Sistemul este destinat utilizării într-un mediu de laborator, unde accesul este restricționat. Nu sunt necesare măsuri suplimentare de securitate a datelor sau comunicației.

## Atribute de calitate ale software-ului

Software-ul trebuie să fie clar documentat, modularizat și ușor de întreținut pentru a permite extinderea viitoare a funcționalităților și integrarea unor componente suplimentare. Codul trebuie să fie lizibil și comentat corespunzător pentru a facilita depanarea și actualizările ulterioare.

# Alte cerințe

Se recomandă păstrarea unui istoric al versiunilor pentru codul sursă și documentație, pentru a urmări evoluția proiectului. În vederea îmbunătățirilor viitoare, se va avea în vedere optimizarea algoritmului de control al brațului robotic și îmbunătățirea preciziei de detectare a senzorilor reed.

# Anexe

## Anexa A: Glosar

* Arduino Mega reprezintă microcontrolerul utilizat pentru coordonarea componentelor hardware ale sistemului.
* PCA9685 este driverul utilizat pentru controlul servomotoarelor brațului robotic.
* Senzor reed este un senzor magnetic utilizat pentru a detecta prezența pieselor pe tabla de șah.
* Multiplexor CD74HC4067 este un circuit utilizat pentru extinderea numărului de intrări digitale disponibile pe Arduino.
* Python este limbajul de programare utilizat pentru logica sistemului și integrarea modelului AI.

## Anexa B: Modele de Analiză

Modelul AI utilizat pentru deciziile de joc este antrenat anterior și preia matricea de senzori pentru a genera mutarea optimă.

## Anexa C: Listă de Probleme

Se identifică necesitatea îmbunătățirii preciziei brațului robotic pentru o manipulare mai sigură a pieselor de șah. Există posibilitatea apariției interferențelor electromagnetice care pot afecta citirea corectă a senzorilor reed. De asemenea, optimizarea fluxului de date între Arduino și aplicația Python poate reduce latența generală a sistemului.