**Documentul de proiectare**  
**Robot AI bazat pe RL pentru Urmărire și Manipulare în Fabrici**  
**Versiunea 0.5, 29.04.2025**

**1. Introducere**

**1.1 Scopul documentului**

Acest document descrie arhitectura și designul sistemului robotic bazat pe Reinforcement Learning (RL), care urmărește un robot lider și manipulează cutii/produse în fabrici. Scopul este de a ghida echipa de dezvoltare în implementarea soluției tehnice, asigurând alinierea cu cerințele funcționale și non-funcționale din Documentul de Cerințe (SRS).

**Public țintă:**

* Manageri de proiect
* Ingineri software (AI/RL, ROS)
* Ingineri hardware (senzori, actuatori)
* Testeri

**2. Prezentare generală și abordări de proiectare**

**2.1 Prezentare generală**

**Sistemul constă din:**

* **Robotul urmăritor:** Folosește RL pentru navigare autonomă. Echipat cu senzori: LiDAR, camere RGB-D, encodoare pentru poziționare.
* **Robotul lider:** Furnizează trasee și comenzi.
* **Sistemul de control central:** Gestionează fluxul de date și interacțiunea între componente.

**Abordare:**

* Orientată pe obiecte (Python/C++ pe ROS 2).
* Modulară: Separă componentele de percepție, decizie, acțiune.

**2.2 Presupuneri / Constrângeri / Riscuri**

**Presupuneri:**

* Robotul lider emite semnale de poziție într-un format standardizat.
* Mediul fabricii are marcaje minimale pentru navigare.

**Constrângeri:**

*Hardware:*

* Latency maximă de 100 ms pentru decizii în timp real.
* Autonomie a bateriei ≥ 8 ore.

*Software:*

* Compatibilitate cu ROS 2 Galactic.
* Criptare AES-256 pentru comunicații.

**Riscuri:**

| **Risc** | **Mitigare** |
| --- | --- |
| Pierderea semnalului lider | Oprire automată + notificare către operator |
| Coliziuni cu obstacole | Senzori redundanți (LiDAR + camere) |

**3. Considerații de proiectare**

**3.1 Obiective și linii directoare**

* **Siguranță:** Oprire instantanee la detectare obstacol.
* **Precizie:** Eroare de urmărire ≤ 5 cm.
* **Adaptabilitate:** Învățare online în medii dinamice.

**3.2 Metode de dezvoltare**

* **RL:** Algoritmi PPO (Proximal Policy Optimization).
* **Simulare:** Gazebo + ROS pentru testare.
* **CI/CD:** Docker + Jenkins pentru integrare continuă.

**3.3 Strategii de arhitectură**

**Flux de control:**

* *Percepție:* Senzori → Preprocesare date → Detectare obstacole
* *Decizie:* Model RL → Planificare traseu
* *Acțiune:* Control motoare + braț robotic

**4. Arhitectura sistemului**

**4.1 Vedere logică**

*Descriere:* Fluxul de date între module: /perception → /rl\_navigation → /gripper\_control.

**4.2 Arhitectură hardware**

**Componente:**

* Calculator onboard: NVIDIA Jetson AGX Xavier
* Senzori: RPLIDAR A3, Intel RealSense D435
* Actuatori: Motoare DC cu encodoare, gripper pneumatic

**4.3 Arhitectură software**

**ROS 2 Noduri:**

* /perception: procesare date senzori
* /rl\_navigation: RL pentru urmărire
* /gripper\_control: manipulare cutii

**4.4 Arhitectura informațiilor**

**Date stocate:**

* Jurnale de navigare (SQLite)
* Modele RL antrenate (PyTorch)

**4.5 Comunicări interne**

**Protocoale utilizate:**

* TCP/IP pentru comenzi rapide
* MQTT pentru telemetrie

**5. Proiectarea sistemului**

**5.1 Baza de date**

**Tabele principale:**

* robot\_logs: timestamp, poziție, stare baterie
* training\_data: date pentru reantrenare

**5.3 Interfețe utilizator**

**Operator:**

* Intrare: Butoane fizice Start/Stop
* Ieșire: Interfață web (vizualizare traseu în timp real)

**6. Scenarii de utilizare**

**6.1 Urmărire lider**

1. Operatorul selectează modul „Urmărire”
2. Robotul detectează liderul
3. Navighează și evită obstacolele automat

**6.2 Manipulare cutie**

1. Robotul primește comanda de ridicare
2. Gripper-ul ajustează forța
3. Depozitează cutia la locația indicată

**7. Proiectare detaliată**

**7.1 Hardware**

**Specificații LiDAR:**

* Rază: 12 m
* Frecvență: 15 Hz

**7.2 Software**

**RL Navigation:**

* Input: Poziția liderului + obstacole
* Output: Comenzi de viteză motoare

**7.3 Securitate**

* Autentificare two-factor pentru actualizări firmware
* Verificare integritate la pornire sistem

**Anexe**

**Anexa A: Gestiune modificări**

| **Versiune** | **Data** | **Autor** | **Modificări** |
| --- | --- | --- | --- |
| 0.5 | 29.04.2025 | Soare Robert | Document inițial |

**Anexa B: Acronime**

| **Acronim** | **Descriere** |
| --- | --- |
| RL | Reinforcement Learning |
| ROS | Robot Operating System |