

LABORATORIO DI INGEGNERIA DEI SISTEMI SOFTWARE

Introduction

Requirements

Il *VirtualRobot* deve percorrere (una volta) il bordo perimetrale della stanza rappresentata nella *scena di WEnv*.

Requirement analysis

Per l'analisi completa dei requisiti, si veda: [application1](#)

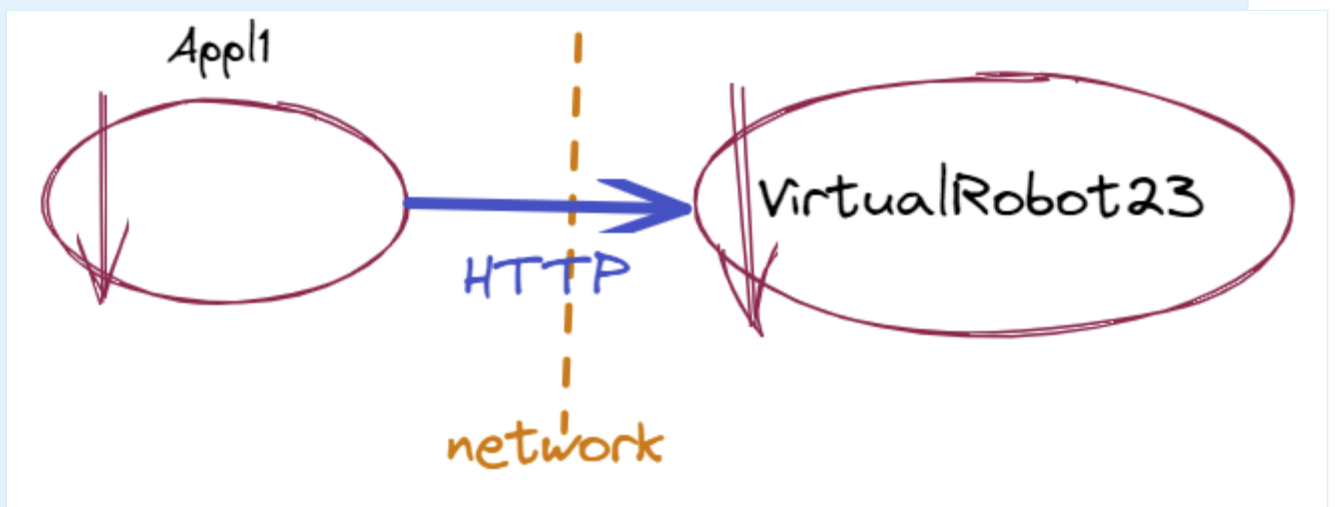
Per quanto riguarda lo sprint1:

- l'applicazione invia il comando **start** al robot tramite **interazione sincrona**

Problem analysis

L'applicazione ha un comportamento proattivo: una volta ricevuto il comando **start**, il robot deve eseguire autonomamente il giro della stanza.

L'architettura logica può essere ridotta a:



Test plans

Dato il requisito, possiamo pianificare come azione di User Acceptance Test:

- **testStartNoStop**: dopo che *Appl1* ha ricevuto (da *CmdConsole*) il comando **start**, occorre verificare che il robot abbia iniziato la sua attività, cioè sia in moto e che, al termine, esso risulti fermo in HOME avendo completato un giro.

Project

Una prima possibile soluzione è:

- inviare un comando di movimento abbastanza lungo per garantire una collisione con il muro di fronte
- dopo la collisione, inviare un comando di rotazione a sinistra per quattro volte, così da percorrere il bordo della stanza.

Single-responsibility principle

Ogni funzione rappresenta una singola azione: walkAtBoundary

- definisco una serie di comandi base per il robot: VrobotMsgs
- definisco una variabile **Nedges** per avere informazioni per il testing

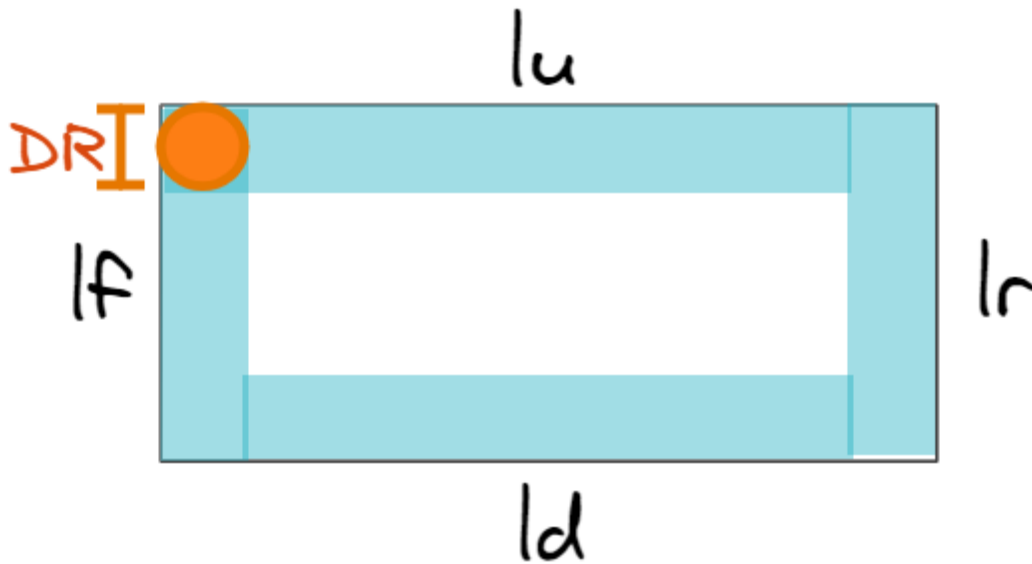
requestSynch

- la funzione **requestSynch** interagisce con il robot via HTTP, restituendo l'esito del movimento.
- rappresenta un supporto alla comunicazione, indipendente dalla logica del problema
- primo possibile **refactoring**: creazione di supporti riusabili per le comunicazioni unibo.basicomm23

Da un punto di vista *logico*, possiamo definire delle operazioni specifiche per interagire con il robot: robot moves

- l'operazione **step** muove il robot in avanti per un tempo dato
- a differenza dell'operazione **forward**, **step** restituisce *true/false* a seconda dell'esito

Modello della stanza



- la stanza è rettangolare: i lati opposti sono uguali
- per percorrere lati uguali, il robot esegue lo stesso numero di passi

Testing

Definito il modello della stanza, una possibile azione di testing per verificare che il robot abbia percorso il perimetro può essere:

- **contare** i passi eseguiti con successo per ogni lato del perimetro
- controllare che il numero di passi su due lati opposti sia uguale

Deployment

Maintenance

- By Letizia Mancini
- email: letizia.mancini3@studio.unibo.it
- GIT repo: <https://github.com/llevtizia/issLab23-ManciniLetizia>
- matricola: 0000926656

