

LABORATORIO DI INGEGNERIA DEI SISTEMI SOFTWARE

Introduction

Dopo [Appl1-HTTPSpring1](#), abbiamo a disposizione i seguenti prodotti:

- *unibo.http.Appl1HTTPSpring1*, un programma che realizza il core applicativo
- *TestAppl1HTTPSpring1*, che fornisce un test automatizzato per il core applicativo

Requirements

In quest Sprint affrontiamo il problema della gestione dei **comandi stop/resume**

Requirement analysis

Il nuovo requisito impone che la logica applicativa:

- realizzi una fase *reattiva*, in cui il robot deve poter gestire i comandi **stop/resume** mentre sta percorrendo il perimetro della stanza.
- sia predisposta ad operare in seguito a comandi inviati da **CmdConsole**.

Problem analysis

In questo Sprint:

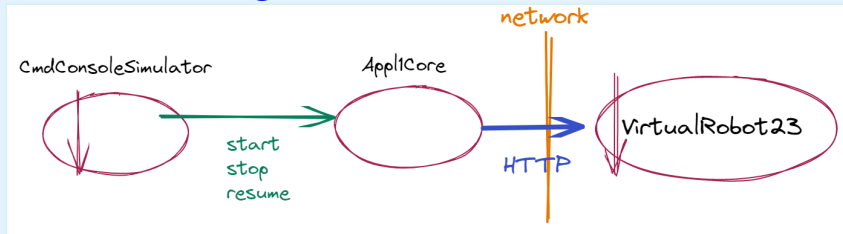
- fissiamo il *requisito non funzionale* di racchiudere la logica applicativa in un singolo POJO, scorporandola dal main program. Ciò allo scopo di poter riusare questo oggetto come una risorsa in future applicazioni.
- evitiamo la complessità della distribuzione, sostituendo la Console remota con un *simulatore* prima e una *console locale* poi.

stop/resume

Il comando **stop** interrompe il movimento del robot (*interrupt*) e riprende solo dopo la ricezione del comando **resume**.

Per gestire le interruzioni, utilizzo il comando *step* definendo [appl1Core](#)

Architettura logica del sistema



Test plans

Un componente *observable* emette degli eventi, che possono essere percepiti da componenti registrati come *observer*. Per **evento** si intende

- un'azione che modifica lo stato del sistema

I componenti registrati come *observer* presso il soggetto *observable* vengono notificati quando si verificano gli eventi.

Project

Testing

Deployment

Maintenance

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

