

LABORATORIO DI INGEGNERIA DEI SISTEMI SOFTWARE

Introduction

Si vuole esplorare come sia possibile costruire modelli eseguibili di un sistema software partendo dalla analisi del problema, senza entrare subito nei dettagli tecnologici necessari per la costruzione effettiva del software. Il caso di studio è una applicazione IOT denominata WD (WolfDetection).

Requirements

Una stazione sperimentale per lo studio di lupi allo stato libero, ha dislocato in una zona di interesse un RaspberryPi dotato di una fotocamera, di un sonar HC-SR04 e di un Led (di colore blu).

Nella stazione è disponibile un altro RaspberryPi, dotato di un Led di colore rosso.

Si chiede di costruire un sistema software che:

- permetta l'accensione e lo spegnimento remoto del sonar mediante comandi impariti dalla stazione
- accenda il Led blu quando il sonar rileva un ostacolo a distanza inferiore a un limite prefissato DMIN
- scatti una fotografia dell'ostacolo rilevato a distanza inferiore a DMIN e la invii ad un elaboratore di immagini in grado di capire se l'ostacolo è un lupo
- accenda il Led rosso presso la stazione, se è stato rilevato un lupo
- spenga il Led blu e il Led rosso quando il sonar non rileva più ostacoli a distanza inferiore a DMIN

Requirement analysis

Dai requisiti si deduce l'esistenza di 3 entità modellate come attori con Pojo:

- RemoteRaspberry
- StationRaspberry
- ImageAnalyzer

RemoteRaspberry e StationRaspberry vivono in due contesti differenti. Sempre dai requisiti deduciamo anche l'esistenza di alcuni tipi di messaggio:

- SonarOn e SonarOff: per accendere e spegnere Sonar: scambiati tra StationRaspberry e RemoteRaspberry
 - AnalyzeImage: richiesta di analisi fotografia: scambiata tra RemoteRaspberry e ImageAnalyzer
 - WolfDetected e NothingDetected: messaggi di rilevamento e fine rilevamento lupo
- n.b. da requisiti non abbiamo abbastanza informazioni per determinare la natura di tali messaggi

Problem analysis

Contesto ImageAnalyzer

mentre sappiamo che RemoteRaspberry e StationRaspberry vivono in due contesti differenti, nei requisiti non viene specificato nulla riguardo ad ImageAnalyzer, le possibilità sono sostanzialmente tre:

Contesto	PRO	CONTRO
ImageAnalyzer vive nello stesso contesto di RemoteRaspberry	<ul style="list-style-type: none"> • minore latenza di comunicazione con sonar e camera • possibilità di analizzare e salvare immagini di lupi anche senza connessione 	<ul style="list-style-type: none"> • risorse di calcolo limitate • poco controllo sul modello e difficoltà di modifiche/aggiornamenti
ImageAnalyzer vive nello stesso contesto di StationRaspberry	<ul style="list-style-type: none"> • controllo su come vengono fatte i analisi, facilità di modifiche aggiornamento 	<ul style="list-style-type: none"> • risorse di calcolo limitate • necessità di connessione per analisi immagini
ImageAnalyzer vive in un contesto a sé stante	<ul style="list-style-type: none"> • possibilità di utilizzare HW ad hoc • semplicità di modifiche/aggiornamenti • maggiore indipendenza tra le parti del sistema 	<ul style="list-style-type: none"> • necessità di connessione per analisi immagini

conviene discutere con il cliente per capire le sue necessità

Tipo Messaggi SonarOn e SonarOff

- event: non risulta idoneo poichè questi messaggi vengono scambiati tra due endpoint precisi

- dispatch: potrebbe essere utilizzato tuttavia trattandosi di messaggi che di fatto attivano/disattivano il sistema sembra più appropriato che RemoteRaspberry fornisca un ack sulla riuscita delle operazioni
- request: sembra il tipo più appropriato

Comunicazione AnalyzeImage WolfDetected e NothingDetected

dai requisiti viene detto poco su come debba essere realizzata la comunicazione tra ImageAnalyzer RemoteRaspberry e StationRaspberry in caso di rilevamento lupo. di seguito riportiamo alcune possibilità

1. RemoteRaspberry -> ImageAnalyzer : richiesta analisi immagine tipo Request
ImageAnalyzer -> RemoteRaspberry : risultato analisi immagine tipo Answer
se lupo rilevato: RemoteRaspberry -> StationRaspberry: WolfDetected tipo Dispatch
se lupo non più rilevato: RemoteRaspberry -> StationRaspberry: NothingDetected tipo Dispatch
2. RemoteRaspberry -> ImageAnalyzer : richiesta analisi immagine tipo Dispatch
ImageAnalyzer -> RemoteRaspberry : risultato analisi immagine tipo Dispatch
ImageAnalyzer -> StationRaspberry : risultato analisi immagine tipo Dispatch
3. RemoteRaspberry -> ImageAnalyzer: immagine da analizzare tipo Dispatch
ImageAnalyzer: risultato analisi immagine tipo Event

da quest'analisi il tipo 3. sembrerebbe il più adatto, dato che il rilevamento di un lupo è di fatto un'evento a cui sia RemoteRaspberry che StationRaspberry sono interessati, ed è quindi un tipo di comunicazione 1 a molti

Test plans

Project

Testing

Deployment

Maintenance

By: Alessio Benenati

Email: alessio.benenati2@studio.unibo.it

GIT:

<https://github.com/alessioBenenatiUnibo/IngegneriaSistemiSW.git>

