

Progetto Finale Software Engineering for Autonomous Systems

Iovenitti Riccardo 253093

Eugenio Fantaconi 261978

Fondamentalmente la nostra idea per il progetto di SEAS è quella di implementare lo schema MAPE-K e con l'ausilio di diverse tecnologie (Nashorn JavaScript, MQTT broker, Java OSGi, Machine/Deep Learning) un sistema autonomo per la gestione della "vita" in una SMART HOME. Le funzionalità della SmartHome utilizzeranno dati raccolti dalle abitudini degli inquilini per ottimizzare, ad esempio, il riscaldamento, le luci e in generale la gestione della casa.

Lo schema MAPE-K vorremmo implementarlo così:

MONITORING

La fase di monitoring sarà costituita da dei sensori virtuali che simuleranno variazioni di temperatura, luminosità ambientale adattiva (spegnimento e accensione luce in base all'intensità luminosa dell'ambiente esterno, regolazione areazione in base alla quantità di CO2, riscaldamento in base alla presenza di persone, ecc.) nella SMART HOME.

I dati verranno comunicati al componente di ANALISI tramite un broker MQTT che, inoltre, fungerà da canale comune per la comunicazione tra i vari componenti.

I sensori saranno direttamente collegati allo stato di monitoring rendendolo un blocco unico.

ANALYZING

La fase di analisi è collegata tramite il componente di monitoring da un broker MQTT che vorremmo utilizzare per la trasmissione dei dati "catturati dai sensori".

Poiché in questa fase il sistema deve valutare la situazione dell'environment ha bisogno di comunicare con la *KNOWLEDGE* che sarà costituita da una *rete neurale e/o database*, codice *hardcoded* o *dinamico*.

Per la gestione della rete neurale verrà utilizzato un componente OSGi che fungerà da connettore verso un'implementazione *pure java* realizzata tramite Weka o DeepLearning4j oppure mediante Python (TensorFlow, SK-learn, etc) utilizzando una comunicazione diretta tramite chiamate REST.

A tal proposito il training della rete neurale avverrà tramite un dataset con le abitudini “domestiche” delle persone.

Il risultato dei dati analizzati verrà trasmesso dall'ANALYZER al PLANNER sempre tramite MQTT.

PLANNING

In base alla fase di analisi il planner avrà il compito di elaborare la migliore strategia da effettuare.

Sempre in ottica di migliore integrazione tra le componenti il PLANNER verrà strutturato tramite il framework OSGi.

Ancora non sappiamo se la strategia del PLANNER sarà hardcoded oppure dinamica.

Nel caso sia dinamica, le strategie verranno prelevate da file XML/TXT.

La comunicazione all'ESECUTORE avverrà tramite broker MQTT.

EXECUTING

Si occupa tramite attuatori dell'esecuzione di azioni volte ad un cambiamento nell'environment(se il ciclo MAP ha prodotto e “ritenuto opportuno” una strategia).

La gestione degli attuatori verrà implementata in script JavaScript eseguiti all'interno dell'applicazione stessa.

I vari comandi da eseguire verranno trasmessi agli attuatori direttamente agli attuatori tramite un canale apposito.

La fase di EXECUTING + l'attuazione dei comandi costituisce un blocco unico.

KWNOLEDGE

La knowledge sarà un componente OSGi che memorizzerà i vari stati tramite database/file e rete neurale.

Tutti i controlli con relativi dati numerici verranno visualizzati in una pagina WEB e manipolati tramite funzioni javascript.

Ancora non sappiamo se la pagina web avrà incorporati le funzioni per la manipolazione dei dati e dei controlli, o caricherà dinamicamente i comandi mandati dagli attuatori.

