

1. Quale è la finalità?

La finalità della prima fase è quella di comprendere la differenza tra un sistema software monolite e uno distribuito basato microservizi, attraverso un percorso pratico che parte dalla progettazione e realizzazione del primo porta alla realizzazione del secondo grazie a un processo step-by-step di refactoring. Il percorso in questione permette anche di fare esperienza nell'affrontare casi in cui bisogna apportare modifiche, anche significative, al sistema software. Nelle prossime fasi del corso lo scopo potrebbe essere quello di capire come migliorare e velocizzare la produzione di software e capire quale architettura adottare per un sistema o sotto-sistema.

2. Sistemi realizzati e sperimentati?

Sistema monolite di partenza in Java, aggiungendo le classi Grid e Cell, e il sistema M2M basato su WebSocket.

3. Abilità conseguite?

Per quanto riguarda le abilità di stampo pratico, sono state conseguite: utilizzo base di Gradle, utilizzo base del framework SpringBoot, migliore capacità di utilizzo dell'IDE Eclipse; in più, durante la prima fase del corso, si ha la possibilità/opportunità di poter approfondire e sviluppare un capacità di utilizzo base di Docker. Per le conoscenze teoriche, vi sono: cosa sia un DSL, cosa avviene nella pratica quando vi è un'interazione asincrona, l'importanza dell'indipendenza tecnologica del sistema software, il funzionamento del protocollo MQTT.

4. Perché ConwayLife come esempio?

La scelta di ConwayLife come caso di studio potrebbe essere dovuta alla logica applicativa non estremamente elevate e, forse, alla sua prestazione ad essere progettato come un sistema software basato su microservizi, in quanto si possono individuare facilmente i suoi componenti e questi sono in numero ridotto.

5. Perché la scelta di Java come linguaggio?

La scelta di Java come linguaggio di programmazione per lo sviluppo del sistema potrebbero ricadere sulla sua portabilità, data dalla JVM che permette di eseguire uno stesso programma su qualsiasi piattaforma senza modificarlo, e sulla sua espansione nell'ambiente dello sviluppo software, che fa sì vi siano molti framework a disposizione.

6. Aspetti dell'IdS richiamati o introdotti durante la Fase 1?

Durante le varie fasi evolutive dei prototipi sviluppati del sistema, sono stati richiamati vari aspetti e concetti dell'ingegneria del software, come l'analisi dei requisiti, l'analisi del problema, il concetto di core business, il principio di singola responsabilità, la dependency inversion e i design pattern Singleton e Observer, e il refactoring.

7. Il sistema ottenuto è basato su microservizi?

Il sistema finale ottenuto è basato su microservizi: i suoi componenti (GUI, servizio Conway, e, nel caso di comunicazione M2M, CallerMQTT), infatti, sono moduli

software del tutto indipendenti e interagenti fra loro attraverso dei protocolli di comunicazione.

8. Perché lo sviluppo e l'utilizzo di librerie custom?

Lo sviluppo di librerie custom durante la realizzazione di un sistema software ha lo scopo di conferire delle caratteristiche a quest'ultimo che non si possono ottenere con altre tecnologie. Nel caso di studio di Conway, la libreria custom `unibo.basicomm23-1.0.jar` contiene classi che permettono al sistema software di essere indipendente dalla tecnologia utilizzata per le interazioni tra i vari componenti; di conseguenza, dà la possibilità di poter cambiare il/i protocollo/i di comunicazione senza modificare tutto il codice.

Repository Git: https://github.com/brihval/ISS_Valerio_Brighi