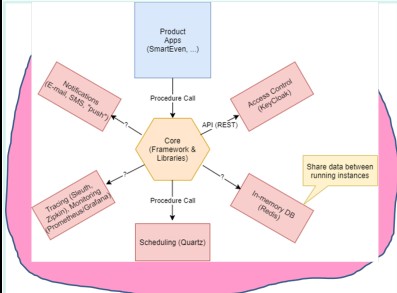
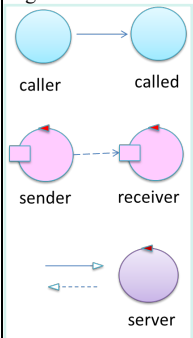
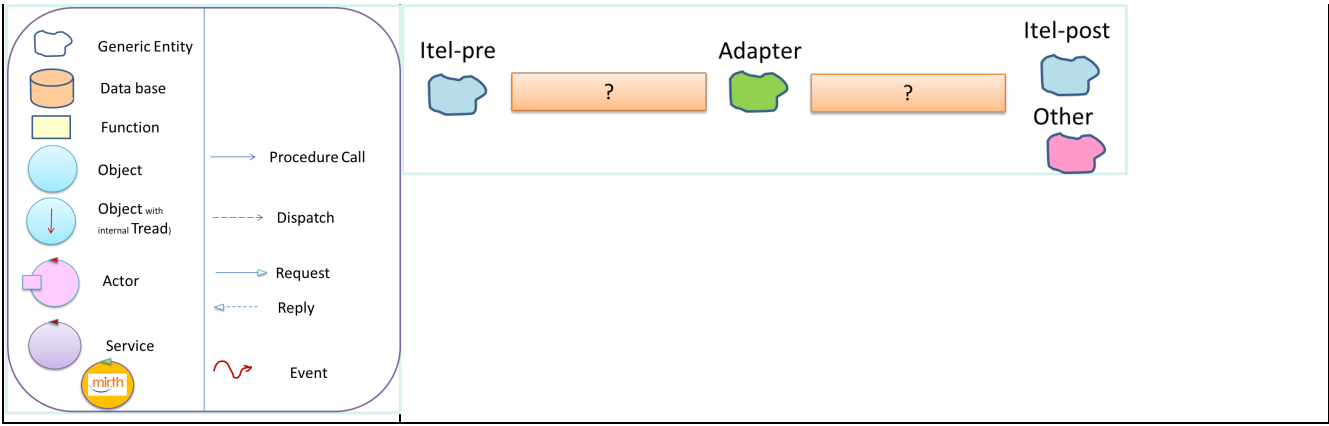


Project Tel | Considerazioni DISI

	<p>ITel-core (cosa e')</p> <ul style="list-style-type: none">• costituisce un supporto formato da componenti implementati come oggetti (POJO) e componenti REST (usati localmente, non esposti come risorse)• che vive (anche replicato) all'interno di una applicazione deployed su una qualche infrastruttura intesa (dalle UserStories) come un elemento non-ITel, ad esempio Docker, Swarm, etc.• che puo' essere condiviso in forma di 'libreria' da piu' applicazioni• essendo i componenti costruiti in modo multi-tenant <p>ITel-core (come viene usato)</p> <ul style="list-style-type: none">• viene usato come un (insieme di) POJO (forse dietro una facade);• i componenti REST <i>non sono visibili all'esterno</i> di ITel-core e sono usati considerando REST come una sorta di 'remote procedure call' alla stregua di GRPC;• interazioni 'non procedure-call' (ad es. publish-subscribe) sono usate (sempre solo <i>internamente</i> a ITel-core) quando rese necessarie da quale utility, ma non sembrano legate a requisiti di livello applicativo; <p>In sintesi:</p> <div>ITel-core fornisce di fatto una 'libreria' e non promuove l'idea di una 'infrastruttura-ITel' di supporto ad applicazioni-ITel</div>
<p>Il ruolo di SpringBoot e SpringCloud</p>	<p>Spring(Boot)</p> <p>Spring(Boot) viene usato, ma senza alcun riferimento all'idea di microservizio.</p> <p>SpringCloud</p> <p>Le problematiche legate alla distribuzione (che hanno spinto allo sviluppo di SpringCloud) sembrano escluse dal modello operativo attuale di ITel.</p>
<p>Il ruolo degli Adapter.</p>	<p>Dall'analisi delle UserStories e dalla discussione svolta emergono caratteristiche/funzionalità/aspetti che vanno oltre a quelli che tipicamente si attribuiscono al concetto di "Adapter Pattern" nei Design Pattern GOF. In particolare:</p> <ol style="list-style-type: none">1. Autonomia - indipendenza/disaccoppiamento dei flussi di controllo:<ul style="list-style-type: none">- in tutte le user story (a parte forse la 4 e la 6) il componente deve mettere in campo una propria architettura di controllo con cui interagire con il sistema esterno (Infrastructure CLI, Health Inst. Service, Domotic System), che permetta di espletare la funzionalità in modo autonomo/indipendente rispetto al controllo del chiamante.2. Strategia di "adattamento" :<ul style="list-style-type: none">- non è limitata ad una semplice traduzione/mapping su standard/tecnologie eterogenee (per avere indipendenza da esse a monte). In alcuni casi il componente appare dover essere in grado di conoscere e applicare aspetti relativi alla <i>logica del dominio</i> e strategie relative. <p>Adapter</p> <p>Nell'attuale scenario sembrano emergere due diverse tipologie di adapter:</p> <ul style="list-style-type: none">• Adapter relativi alla infrastruttura• Adapter relativi alle applicazioni <div>Il ruolo di questi due tipi appare a prima vista alquanto diverso e va meglio precisato, con particolare riferimento alle interazioni e al rapporto logico/funzionale tra cio' che precede e cio' che segue un adapter.</div>
<p>Componenti, interazioni, rappresentazioni</p>	<p>I simboli qui a fianco verranno usati per rappresentare con minor ambiguita' possibile il tipo di componenti in gioco e le forme della loro interazione. Ad esempio:</p>  <p>Schema generico di riferimento</p>



User stories per la infrastructure (Infrastructure-related adapters)

Queste userstories sono tutte collegate allo sviuppo di un Infrastructure Adapter articolato su diversi aspetti:

Infrastructure Adapter

	Occorre chiarire: <ul style="list-style-type: none">il rapporto (on solo in termini di interfacce ma anche di "competenze") tra License Synchronizer e Infrastructure Adapterla interazione
	Occorre chiarire: <ul style="list-style-type: none">il rapporto (on solo in termini di interfacce ma anche di "competenze") tra Metrics Adapter e Infrastructure Adapterla interazione
	Occorre chiarire: <ul style="list-style-type: none">il rapporto (on solo in termini di interfacce ma anche di "competenze") tra Operator e Infrastructure Adapter

User stories per le applicazioni (Application-related adapter)

Gli schemi relativi a queste user stories sembrano al momento solo indicativi di qualcosa che deve ancora essere opportunamente analizzato e definito, anche in termini di requisiti e di funzionalità attese. Rimangono aperte le questioni poste al livello precedente.

	Health Data Adapter
	Sensor Data Adapter
	WebRTC Adapter

Considerazioni

- Con riferimento all'accordo ITEL-DISI relativo a "Progetti di innovazione e diversificazione di prodotto o servizio per le PMI", diamo per scontati i punti:
- f) formare gli sviluppatori già alle dipendenze di I-Tel al fine di acquisire strumenti, modalità e mentalità necessarie per migliorare il processo interno di sviluppo del software;
 - g) impegno a seguire e a mantenere con la I-Tel contatti ed incontri per avanzamento lavori, verifica obiettivi di progetto e fornire supporto tecnologico agli sviluppatori durante le fasi di implementazione del progetto;
 - h) collaborare alla individuazione di uno o più candidati che possano essere integrati nell'organigramma aziendale per ricoprire il ruolo di software architect, che avrà il compito di coadiuvare gli sviluppatori nella manutenzione della piattaforma e nello sviluppo di nuove funzionalità della stessa in maniera autonoma durante e al termine della collaborazione fra le parti; (done con JEAN CLAUDE)
 - i) al termine dell'affiancamento, I-Tel dovrà essere in possesso di tutto il know-how necessario alla modellazione, creazione di interazioni, micro servizi per la gestione della nuova piattaforma.

Per quanto riguarda le parti da **co-progettare e co-prototipare**, il DISI ritiene di particolare interesse i seguenti punti:

- c) interazione con altri software necessari al funzionamento dei prodotti venduti dalla I-Tel;

Questo punto ha condotto al concetto di adapter ed è al centro dell'attuale scenario, con riferimento ai 'progetti' [project HDA](#), [project IOTA](#), [project WA](#).

- e) analizzare, descrivere, specificare e documentare - tramite gli strumenti ritenuti più opportuni - un possibile sistema software performante per le richieste della I-Tel;

Il progetto **project MA** sembra avere caratteristiche **più 'sistemiche'** degli altri e potrebbe prefigurare un campo di indagine per una 'graceful transition' verso il supporto e la erogazione di veri e propri micorservizi (che sembra attualmente 'in background').

Oggetto del Contratto

La bozza attuale prevede di inserire in Art.2:

Il Committente affida all'Università, che accetta alle condizioni di seguito specificate, l'esecuzione di attività di consulenza sul seguente tema:

DA MODIFICARE:

Prestazioni di consulenza tecnica nello sviluppo della nuova piattaforma software multicanale denominata "Sm@rtSuite", basata su micro-servizi quale evoluzione tecnologica della attuale piattaforma operativa denominata "Nestagora" . In particolare saranno forniti 4 prototipi:
 1) Infrastructure Adapter
 2) Health Data Adapter
 3) Sensor Data Adapter
 4) WebRTC Adapter
 Così come meglio descritti nelle 4 pagine della presentazione condivisa tra le parti e riportate come allegato A.

La proposta (per evitare problemi di proprietà del software) è di omettere la frase **saranno forniti 4 prototipi** attraverso una dicitura del tipo che segue:

Le attività di consulenza tecnica consisteranno nella **progettazione e nello sviluppo collaborativo** ITel-DISI di componenti software relativi alla nuova piattaforma qui genericamente denominati con il termine:

1) Infrastructure Layer

e con la sperimentazione della nuova piattaforma "Sm@rtSuite" nello sviluppo di applicazioni basate su microservizi attraverso la co-progettazione e co-costruzione di prototipi denominati:

- 2) Health Data Adapter: per applicazioni nel campo ... CON QUALCHE SPECIFICA DI REQUISITO
- 3) Sensor Data Adapter: : per applicazioni nel campo IOT ... CON QUALCHE SPECIFICA DI REQUISITO
- 4) WebRTC Adapter: : per applicazioni nel campo ... CON QUALCHE SPECIFICA DI REQUISITO

Le attività si svolgeranno in tre fasi:

Fase di Analisi : Luglio-Agosto 2020

Fase di Sperimentazione I : Settembre-Ottobre 2020. Relativa alla progettazione e sviluppo dell'Infrastructure Layer e di un adapter scelto come caso di studio di riferimento per il livello applicativo.

Fase di Sperimentazione II : Novembre-Dicembre 2020. Relativa alla progettazione e sviluppo di prototipi relativi ai rimanenti adapter.

€ 5.000,00 (cinquemila/00) + IVA alla sottoscrizione del presente Contratto; (Luglio)

€ 12.000,00 (euro/00) + IVA al completamento della Fase di Analisi nel par. 2.1;

(Inizio Settembre, per poter pagare borsa/contratto)

€ 8.000,00 (euro/00) + IVA al completamento della Fase Sperimentazione I nel par. 2.1;

(Inizio Novembre, per una eventuale II borsa)

€ 14.960,00 (quattordiciemilanovecentosessantaeuro/00) + IVA al completamento delle prestazioni riportate nel par. 2.1 del presente Contratto

Possibili attività

FASE DI ANALISI - (Luglio - 15 Agosto 2020)

1. Definire e co-realizzare un ambiente di lavoro (**mockitel**) per la **'sperimentazione non interferente di prototipi di adapter e applicazioni'** avvalendosi di componenti Itel-core reali o 'mock' utilizzabili secondo le modalità previste a regime nel sistema reale (inclusa la loro distribuzione in container Docker).
2. Definire in modo più preciso quanto delineato nelle diverse user-stories, per passare da schemi UML-Like a prototipi 'mock' che meglio definiscano le interazioni attualmente previste nei diagrammi di sequenza.
3. Individuare le diverse (se esistono) **tipologie di adapter** e definirne la struttura, interazione e comportamento attraverso opportuni (prototipi) Java su **mockitel**.
4. Individuare due applicazioni-riferimento: **SmartEven** (in corso di sviluppo) e **???** al fine di precisarne i **requisiti** (funzionali e non funzionali) e al fine di definire un modello logico che ponga in evidenza i tipi dei componenti in gioco, la loro interazione e il ruolo degli adapter.
5. Definire meglio l'idea ITel dell'approccio **MonolithFirst** allo sviluppo al fine di delineare un insieme opportuno di ITel-**BoundedContext** e studiare la possibilità di 'graceful transition' di Itel-core ad un insieme di **N(>=1)** microservizi.
6. Concordare un adapter applicativo da approfondire/prototipare nella FASE SPERIMENTAZIONE I

FASE SPERIMENTAZIONE I - (16 Agosto- Ottobre 2020)

FASE SPERIMENTAZIONE II - (Novembre-Dicembre 2020)

By AN-AR Unibo-DISI