

## *Prototipo framework SAPI, testato e rilasciato in ambiente di sperimentazione*

### *Survey*

Nel presente documento si riassumono i contenuti del deliverable "**D5.2\_1 Prototipo framework SAPI, testato e rilasciato in ambiente di sperimentazione**" relativo all'attività *SP 5.2 Realizzazione prototipale di un framework per l'erogazione personalizzata di contenuti e servizi*, compiuta nell'ambito del quinto Obiettivo Realizzativo (OR 5) "Sperimentazione e valutazione dell'impatto della ricerca".

L'attività SP 5.2 prevede il rilascio del prototipo e nel documento a corredo è descritta l'architettura del framework, viene esposto come creare contenuti e servizi intelligenti per SAPI e viene indicato come l'uso della piattaforma IWT e di altri strumenti creati from-scratch, sono adoperati per la gestione di servizi e contenuti intelligenti sviluppati per SAPI. Il documento inizia con esplicitare i concetti fondamentali su cui si basa la logica del framework SAPI e segue descrivendo brevemente le sue caratteristiche peculiari. I concetti fondamentali sono di seguito riassunti.

**Activity.** È il mattone fondamentale nella definizione di una logica di business associata ad un servizio web erogato da SAPI e rappresenta un processo.

**Entity.** Consente di modellare la modalità di fruizione di un servizio della piattaforma SAPI adattato rispetto ad una particolare situazione (utente e contesto). Le Entity sono state suddivise in atomiche e composte. Una entity atomica può essere interpretata come un oggetto digitale (widget), o risorsa, dotato di una rappresentazione semantica e di metadati associati. Una entity composta (EC) è costituita da entity atomiche (EA) e/o da EC.

**Abstract Interaction Unit.** Una Abstract Interaction Unit (AIU) può essere definita come un insieme di più elementi funzionali attraverso i quali l'utente interagisce con il sistema.

**Errore. L'origine riferimento non è stata trovata.** La fruizione di una entity, da parte dell'utente, avviene attraverso una AIU. Una AIU dipende dal dispositivo utilizzato (caratteristiche hardware e software) e dalla tecnologia impiegata lato presentation.

**Sensori.** In SAPI sono stati analizzati ed identificati tre tipologie di sensori: *Sensori Fisici (SF)*; *Sensori Virtuali (SV)*; e *Sensori Cognitivi (SC)*. I *SF* e *SV* percepiscono le informazioni dal contesto in cui è eseguita l'applicazione (inteso come ambiente, dispositivo e canale di comunicazione). I *SC* permettono l'acquisizione implicita di informazioni inerenti al comportamento utente (preferenze, interessi, difficoltà, etc.)

I dati raccolti tramite i sensori cognitivi sono presenti nella Tabella 1.

**Tabella 1: Dati prelevati tramite i sensori cognitivi.**

Servizi richiesti (Click su Hyperlink, bottoni, etc)
Feedback esplicito (pop up)
Tempo permanenza intera sessione (ora log-in, ora log-out)
Tempo permanenza singola pagina (secondi)
Tempo permanenza singolo servizio (secondi)
Completamento del servizio
Click errati per ogni componente grafico della pagina
Numero accessi alla piattaforma
Frequenza utilizzo singolo servizio

**Profilo Utente.** Le informazioni relative all'utente SAPI e rappresentate nel profilo sono suddivise in 4 categorie: 1) *Informazioni esplicite a lungo termine*; 2) *Informazioni esplicite a breve termine*; 3) *Informazioni implicite a lungo termine*; 4) *Informazioni implicite a breve termine*.

**Modello di Contesto.** Il modello del contesto SAPI estende lo standard UAProf<sup>1</sup>, ma è ben strutturato in modo da far risaltare solo le informazioni utili ai fini della descrizione dei contesti trattati in SAPI. Tutto ciò si realizza mediante un opportuno set di regole che, a partire dal gran numero di dati strutturati secondo l'UAProf, permette di derivare informazioni di più alto livello associate a profili di contesto predefiniti.

**Modello delle abilità utente.** Esso è composto da 1) Una tassonomia delle disabilità visive target che descrive le disabilità visive ritenute di maggiore rilevanza in SAPI ; 2) Un'ontologia di alto livello che descrive le relazioni tra disabilità, i difetti visivi e capacità ad interagire; 3) Un insieme di relazioni esplicite che mettono in relazione le disabilità visive con i difetti e le capacità di interazione.

**Stereotipo.** In SAPI, in particolare, si fa riferimento ad una gerarchia di stereotipi che deve essere associata all'intera situazione user-contestuale. Gli stereotipi dichiarati al sistema sono organizzati in un grafo diretto aciclico (Directed Acyclic Graph) mediante la relazione di ordinamento parziale "generalization of".

<sup>1</sup> User Agent Profile (UAProf) è uno standard definito dall'Open Mobile Alliance (OMA) ed è basato sullo schema definito dal Composite Capabilities/Preference Profiles (CC/PP) per la descrizione dei dispositivi mobili. L'UAProf, infatti, rappresenta proprio una concreta implementazione del CC/PP destinata alla descrizione delle capacità dei dispositivi mobili che supportano il Wireless Application Protocol (WAP).

**Situazione.** Una Situazione in SAPI è individuata dai valori assunti in un dato momento dallo User Context Environment.

**Rule (Regola di Adattamento).** Una Regola di Adattamento  $Ar$  di una Entità  $En$  ad una data Situazione  $S$ , attivabile dall'occorrenza di un evento  $Ev$ , è una particolare Regola Event Condition Action (ECA), il cui significato è il seguente: all'occorrenza dell'evento  $Ev$  in una data situazione  $S$  se la condizione  $Cd$  è verificata allora viene eseguito l'adattamento dell'Entità  $En$  alla situazione  $S$ .

Una regola di adattamento agisce a run-time e tiene conto dei parametri dello User Context forniti dai Sensori Fisici e Virtuali. In SAPI l'evento che attiva una regola è il cambiamento di una Situazione.

**Contenuti Intelligenti.** In ambito SAPI un contenuto intelligente è suddiviso in due diverse categorie *Contenuti Informativi* e *Contenuti Formativi*. Il modello di descrizione dei Contenuti Intelligenti è presentato nella seguente figura:

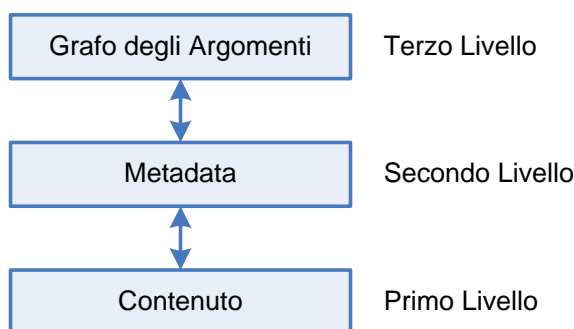


Figura 1: Modello di descrizione dei contenuti intelligenti.

Per introdurre l'organizzazione logica dei componenti di SAPI e per meglio comprendere il suo funzionamento, nel D5.2\_1 sono stati descritti alcuni semplici casi d'uso:

- Normovedente che utilizza il chiosco
- Ipovedente affetto da cataratta che utilizza il chiosco
- Normovedente che utilizza uno Smartphone/PDA

In ogni scenario d'uso sono evidenziate le potenzialità di adattamento ed ottimizzazione offerte da SAPI.

In particolare è possibile distinguere tre ambiti di adattamento:

- in base allo stereotipo utente;
- in base alle condizioni ambientali;
- in base al dispositivo di accesso.

SAPI, inoltre, prevede due tipologie di adattamento:

- 1 Adattamento a *design-time* o *in batch* (definita nel documento col termine Ottimizzazione)
- 2 Adattamento *on the fly*

La prima forma di adattamento è eseguita ottimizzando il modello delle entity agendo direttamente sui metadata delle entity che andranno a comporre il servizio. Tale operazione viene effettuata "in differita" una tantum per ogni stereotipo di utente definito e ad intervalli di tempo precedentemente stabiliti.

La seconda forma di adattamento avviene a run-time. A fronte dell'esecuzione della logica di business, le entity da presentare vengono ulteriormente adattate tenendo conto della situazione in cui si svolge l'interazione.

### Adattamento in base allo stereotipo utente:

In generale, lo stereotipo viene associato all'utente a partire dai dati che vengono richiesti ai clienti all'atto della registrazione o che vengono raccolti dal sistema nel corso delle interazioni.

L'utente, infatti, è caratterizzato da un profilo costituito da una serie di informazioni (facets) che sono contenute in un'ontologia.

Ogni facet viene pesata e convertita in una coordinata di uno spazio multidimensionale: l'utente è rappresentato da un punto in questo spazio.

Gli stereotipi sono anch'essi punti disposti nello spazio multidimensionale: attraverso un criterio di minima distanza viene associato uno stereotipo all'utente (Figura 2).

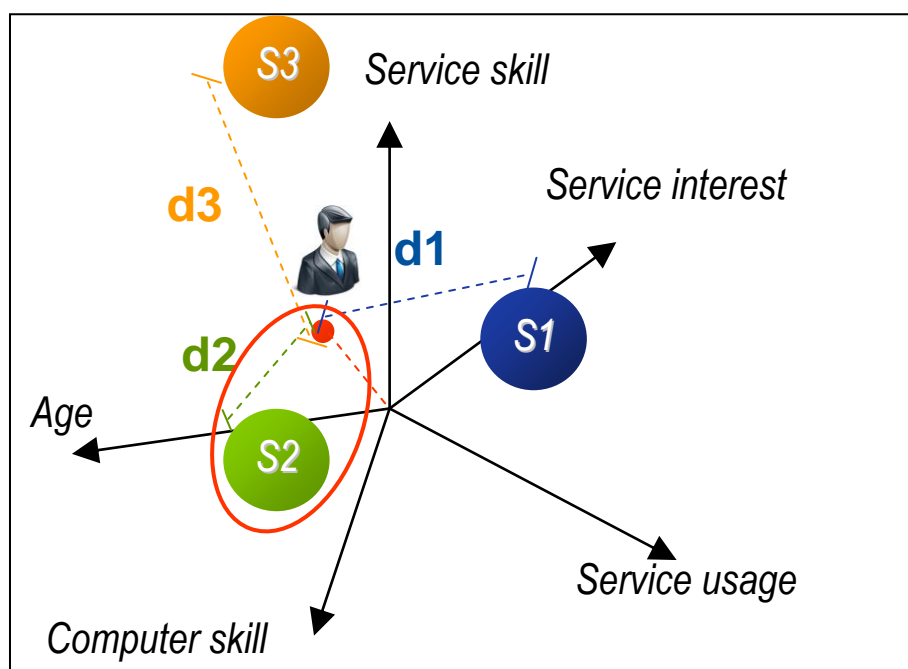


Figura 2: Associazione stereotipo utente.

Uno stereotipo utente è costituito dalla combinazione dei valori assunti dalle seguenti 5 macro-caratteristiche, che identificano diversi aspetti di un utente:

- Informazioni di base
- Interesse
- Knowledge
- Competenza
- Disabilità

La combinazione degli stereotipi relativi ad ogni caratteristica viene poi affiancata allo stereotipo della situazione di contesto ricavata dai dati dei sensori. In questo modo, la coppia stereotipo utente e situazione (device e ambiente) in cui esso opera permette a SAPI di applicare le opportune direttive di adattamento e ottimizzazione.

L'associazione di un utente ad uno stereotipo non è statica, ma, a seguito delle interazioni, può cambiare dinamicamente grazie all'evoluzione del profilo e al conseguente avvicinamento dell'utente ad un altro stereotipo.

In funzione dello stereotipo, SAPI interviene attuando diverse soluzioni:

1. Sulla base delle necessità dell'utente è prevista (a tempo di progetto) un versione ottimizzata delle diverse entity che andranno a comporre dinamicamente le pagine del servizio. Ciascuna entity, infatti, è descritta semanticamente da una serie di attributi che consentono di descriverne la posizione, la grandezza, la versione audio, etc
2. La competenza o skill dell'utente rappresenta una caratteristica dinamica dello stereotipo per effetto della quale un utente può migrare da una classe ad un'altra. La competenza di un utente in un servizio rileva la capacità di fruizione di un servizio tenendo conto di parametri quali il tempo impiegato per completare il servizio, la frequenza di accesso alla piattaforma oppure se l'utente ha completato o meno il servizio. Nel caso l'utente appartenga ad uno stereotipo con un valore di competenza nel servizio "Alto", il sistema effettua a *runtime* una sintesi dei contenuti a video che si riferiscono esclusivamente a quel servizio. Nel caso l'utente appartenga ad uno stereotipo con un valore di competenza nel servizio "Basso" oppure "Medio", il sistema non esegue alcun adattamento.
3. L'interesse, così come la competenza, è una caratteristica dinamica del profilo e misura la propensione di un utente ad utilizzare un determinato servizio o una specifica modalità di pagamento. Il sistema, in base alle scelte effettuate dall'utente nel corso delle interazioni passate, è in grado di effettuare un "salto" nella logica di business, presentando direttamente la pagina che l'utente più probabilmente ha voglia di visitare. Tale soluzione permette di anticipare la scelta dell'utente e di snellire il processo di richiesta di un servizio.

### **Adattamento in base alle condizioni ambientali:**

La piattaforma SAPI, come già detto, è in grado anche di adattarsi rispetto al contesto per far fronte ad esempio a delle condizioni ambientali difficili che possono condizionare l'interazione dell'utente nella fruizione dei servizi richiesti. Il sistema, in condizioni di luminosità elevata per esempio, esegue un adattamento a run-time per effetto del quale

l'interazione avviene in maniera audio abilitando le funzioni TTS/ASR implementate nel framework.

**Adattamento in base al dispositivo d'accesso:**

La piattaforma, sfruttando le informazioni relative al device ricavate dai sensori virtuali, è in grado di adattarsi alla risoluzione del dispositivo di accesso sia esso il chiosco, il PC oppure il palmare. L'ottimizzazione alle particolari esigenze del dispositivo è un'altra delle operazioni che SAPI esegue in batch. In seguito a tali operazioni, vengono generate indicazioni utili al modulo di renderizzazione per la riorganizzazione del layout nello spazio a disposizione.

A ciò si aggiunge il fatto che le pagine realizzate in SAPI sono caratterizzate da un layout "liquido", ovvero occupano sempre tutto lo spazio disponibile della finestra del browser, qualunque sia la risoluzione del monitor.

Dopo aver definito le funzionalità del framework, il documento ne descrive la struttura logica. Dal punto di vista logico si può immaginare che l'architettura SAPI presenti i componenti fondamentali mostrati in Figura 3 e distribuiti su 4 layer.

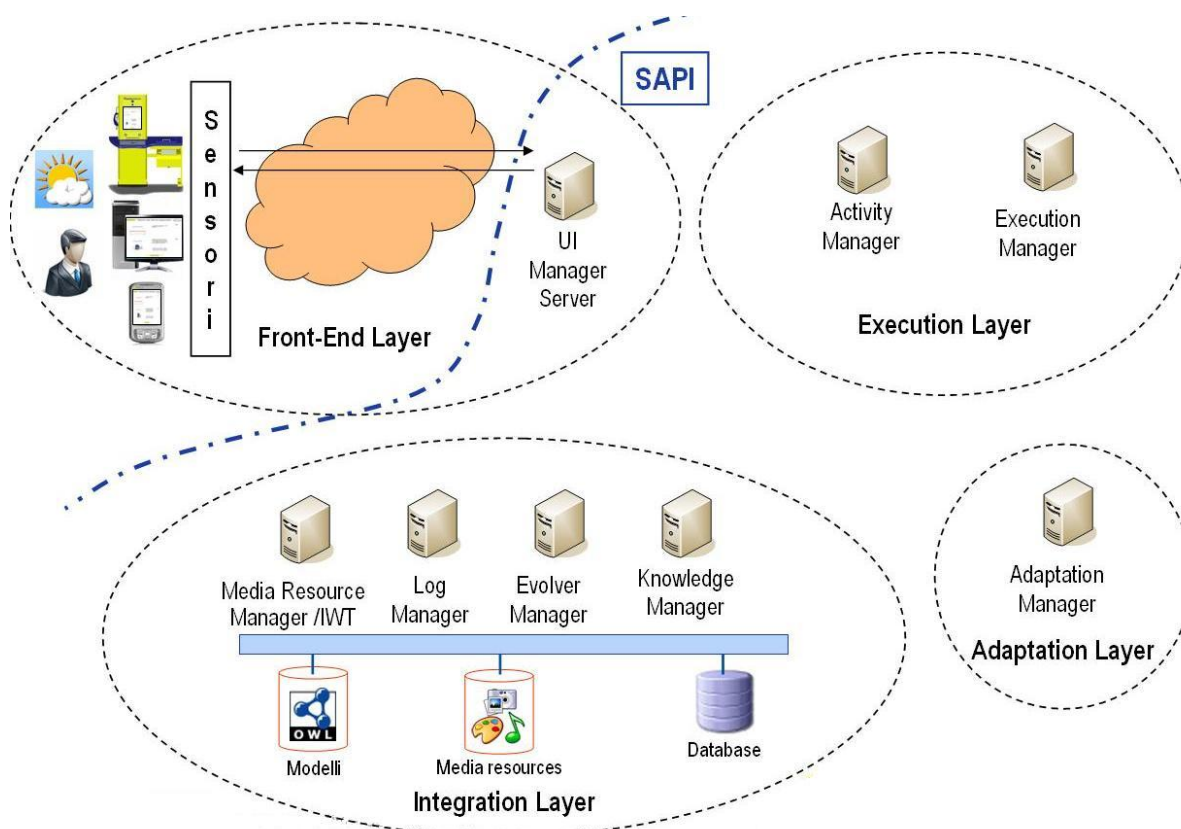


Figura 3: Schema di funzionamento del framework SAPI.

Il *Front-End Layer* si occupa della renderizzazione e del deploy delle interfacce utente per la fruizione del servizio; l'*Execution Layer* si occupa dell'orchestrazione dei servizi offerti dai vari moduli e coordina la composizione del servizio intelligente,

interagendo, se necessario, con il back-end di Poste Italiane; l'*Adaptation Layer* si occupa dell'adattamento a run-time dei contenuti da presentare all'utente in base alle variazioni delle condizioni di contesto in cui opera l'utente stesso; l'*Integration Layer* gestisce tutta la base di conoscenza del sistema, dal profilo utente ai contenuti semantici, ai dati di utilizzo del sistema.

Il flusso di composizione di un servizio intelligente è descritto di seguito.

Al momento della richiesta di un servizio, vengono attivati tutti i sensori disponibili sul client per il monitoraggio dei dati di contesto (ambiente e device) e dei dati di utilizzo del sistema (sensori cognitivi). Le informazioni raccolte vengono inviate con la richiesta all'Activity Manager.

L'Activity Manager interroga il Knowledge Manager per richiedere l'interpretazione dei dati prelevati dai sensori e ricevere lo stereotipo di utente e contesto rispetto al quale adattare il servizio. L'istanza dello stereotipo viene ottenuta per mezzo di inferenze su di una base di conoscenza ontologica.

Allo stesso tempo, i dati di utilizzo del sistema prelevati dai sensori cognitivi vengono storicizzati dal Log Manager in modo che possano essere analizzati in fasi successive. Tali analisi consentono di attivare le logiche del Knowledge Manager per evoluzione del profilo utente e le logiche di ottimizzazione e predizione fornite dall'Evolver.

Identificato lo stereotipo, l'Activity Manager interroga l'Evolver Manager per verificare la possibilità di anticipare la prossima mossa dell'utente a partire dalla sua richiesta e dalla sua storia pregressa di interazioni passate e storicizzate dal Log Manager. In caso di risposta affermativa, l'Activity Manager modifica a run-time il flusso della logica di business.

A questo punto l'Activity Manager conosce l'operazione prevista dalla logica di business e deve recuperare i contenuti intelligenti necessari per la composizione dell'interfaccia. A tal fine viene interrogato il Knowledge Manager, il quale restituisce l'oggetto semantico entity associato alla richiesta dell'utente ed ottimizzato per il particolare stereotipo utente precedentemente identificato. L'operazione di ottimizzazione della entity è eseguita in batch dall'Evolver Manager.

Il contenuto recuperato può necessitare di completamento con dati dinamici o di controlli di validità da richiedere al back-end di Poste Italiane. Pertanto viene interrogato l'Execution Manager che ha in sé la logica esecutiva associata alla particolare fase del flusso di business.

Il contenuto riempito con le elaborazioni della logica di business subisce le operazioni di adattamento a run-time da parte dell'Adaptation Manager, in modo da tenere in conto le variazioni del contesto ambientale e delle variabili dinamiche del dispositivo.

Prima che il contenuto intelligente ottimizzato ed adattato possa essere mandato all'UIManager per la renderizzazione, devono essere recuperate le indicazioni generate in batch dall'Evolver Manager per la riorganizzazione del layout. L'Activity Manager interroga quindi l'Evolver Manager per conoscere quali modifiche al layout deve far applicare al motore di render per tenere in conto delle eventuali limitazioni grafiche o prestazionali del dispositivo.

A questo punto l'Activity Manager dispone di tutte le informazioni e di tutti gli oggetti semantici da inviare all'UIManager per la creazione della pagina e la sua presentazione all'utente finale.

A corredo della descrizione del flusso di composizione di un servizio, il documento D5.2\_1 riporta sequence diagram dettagliati relativi ai diversi casi d'uso analizzati e che descrivono puntualmente le chiamate ai servizi esposti da ciascun componente del framework. In Figura 4 è riportato, a titolo d'esempio, il sequence diagram relativo ad una generica richiesta pervenuta all'execution layer nel caso d'uso di normovedente che utilizza il chiosco.

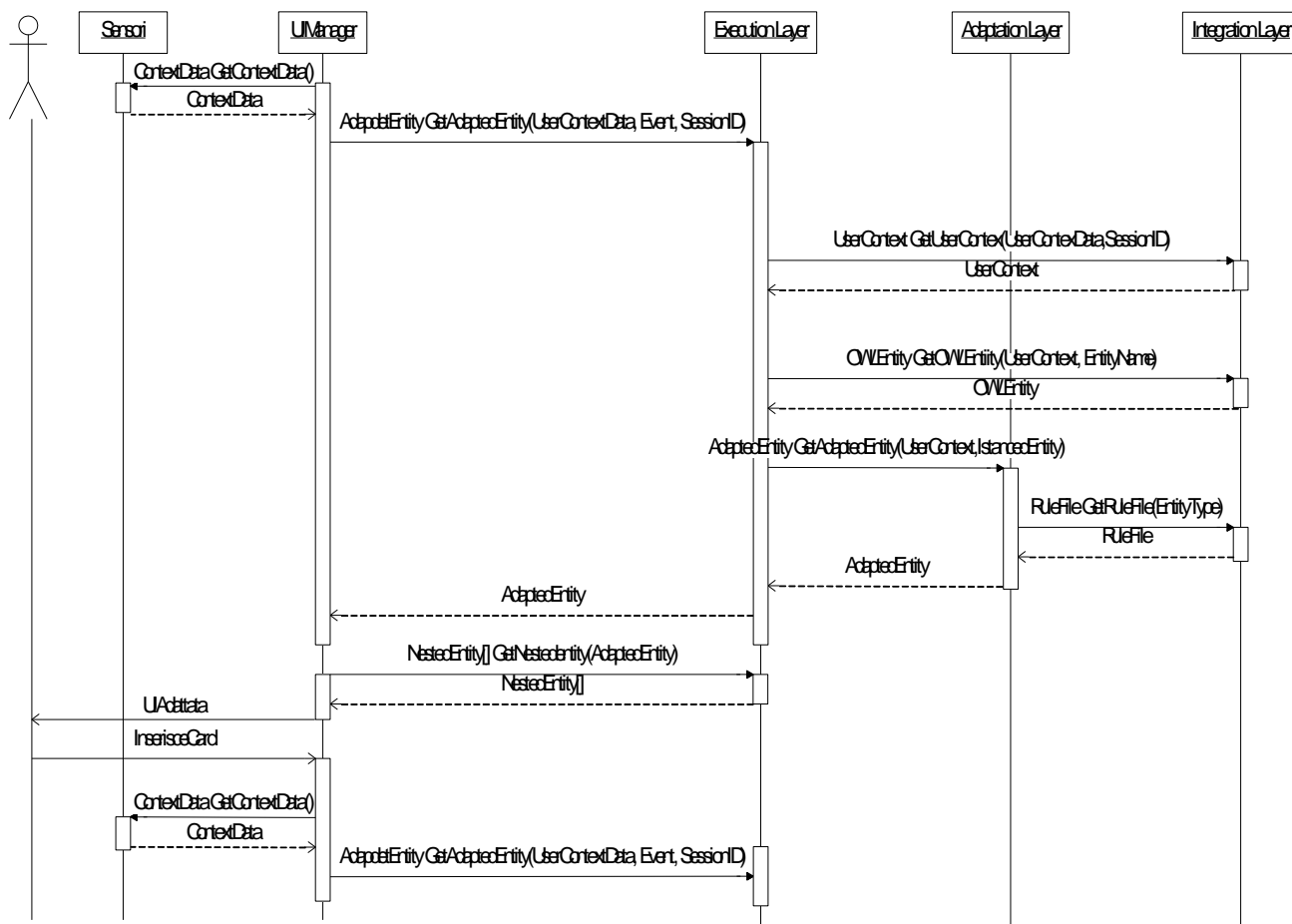


Figura 4: Chiosco dettagli Front-end Layer.

Come è stato detto, le funzionalità di SAPI si basano su inferenze effettuate su modelli ontologici e su analisi di una serie di dati ed informazioni raccolte durante l'interazione dell'utente. Pertanto è stato necessario definire opportunamente una base di conoscenza a supporto del framework. E' stata definita, cioè, una ontologia per ogni modello utilizzato dal sistema e si è scelto OWL come linguaggio di rappresentazione. In particolare sono stati definiti i seguenti modelli:

- Erogazione e Fruizione dei Servizi
- Profilo Utente
- Abilità Dell'Utente
- Contesto
- Entity

Per la gestione dei modelli ontologici si è scelto di utilizzare JENA, un framework basato su Java per la costruzione di Semantic Web Application. Le API del framework consentono di lavorare direttamente sulla rappresentazione OWL dei modelli e, inoltre, sono disponibili dei meccanismi di persistenza che consentono in modo automatico di creare/salvare/aggiornare le istanze dei modelli all'interno di un database relazionale (nel nostro caso MySQL).

E' stato definito, inoltre, un database di supporto e di raccordo con l'Ontology Suite e il Content Repository che contiene le seguenti tabelle:

- Entity Ottima – Stereotipo: una Entity definita a tempo di progetto, collegata ad una data Activity e individuata univocamente attraverso un EntityName, è associata ad uno Stereotipo. La tabella conserva questo il legame entity ottima stereotipo.
- Entity – Activity: permette di associare un Evento Servizio, generato dall'utente, all'activity che gestisce quell'evento.
- Stereotipo – RFID: associa ad uno Stereotipo il tag RFID dell'utente. Questa associazione permette di ricavare dall'RFID l'ultimo stereotipo associato all'utente nella sessione precedente e di caricare, di conseguenza, lo stereotipo di dialogo opportuno per iniziare l'interazione.
- Dati di log: conserva i dati di utilizzo dell'interfaccia utente che servono per le evoluzioni del profilo e le inferenze sulla logica di business.

Il framework SAPI, oltre alle funzionalità di composizione ed erogazione di servizi intelligenti, permette ovviamente di poterli anche progettare e definire. Più precisamente, sfruttando il framework SAPI, è possibile creare:

1. Servizi intelligenti adattivi
2. Contenuti Intelligenti che possono essere:
  - a. Contenuti Informativi: Descrivono un servizio offerto da Poste Italiane. In altre parole i contenuti informativi legati ad un determinato servizio sono pensati per utenti del sistema SAPI che non sono ancora utenti dello specifico servizio e che necessitano solo di una descrizione di massima dello stesso.
  - b. Contenuti Formativi: Sono le istruzioni per l'utilizzo di un determinato servizio o di particolari operazioni afferenti ad un servizio. Una funzionalità interessante che offrono i contenuti formativi, a differenza di quelli informativi, è la possibilità di essere sequenzializzati dato un obiettivo formativo che si vuole raggiungere. Una sequenza di contenuti formativi è detta **percorso d'istruzione**.

Per la creazione di un nuovo servizio intelligente in SAPI si devono svolgere le seguenti operazioni:

1. Definire la logica di business associata al servizio che si traduce in:
  - a. Scrittura del codice delle Activity che di fatto implementano le funzionalità del servizio che si vuole creare e dialogano con il back-end di Poste Italiane;

- b. Aggiunta nel database di una nuova entry che mette in relazione il nuovo Evento Servizio con la rispettiva Activity e con l'Entity Composta utilizzata per la presentazione;
  - c. Definizione dello schema dei dati che si scambiano Entity Composta ed Activity.
2. Definizione delle Entity Atomiche attraverso l'individuazione di un nome univoco che indica l'istanza di una classe AtomicEntity nell'ontologia di rappresentazione;
  3. Scelta del tipo a cui appartiene l'Entity Atomica. Attualmente sono supportati i seguenti tipi (testo, immagine, hyperlink, audio video , bottone e casella di testo);
  4. Definizione delle Entity Composte attraverso l'individuazione di un nome univoco che indica l'istanza di una classe CompositEntity nell'ontologia di rappresentazione;
  5. Creazione delle associazioni che legano le Entity Atomiche alle rispettive Entity Composte;
  6. Definizione del Presentation Model che prevede l'individuazione di un nome univoco (e di conseguenza l'istanza di una classe nel modello di rappresentazione); La definizione del Presentation Model comporta la scrittura del PM Component model in cui viene specificato come una Entity Atomica viene utilizzata a livello di presentazione (definizione del layout e disposizione all'interno della pagina) all'interno dell'Entity Composita alla quale appartiene. La definizione del PM Component model comporta la scrittura dei metadata che caratterizzano le Entity Atomiche.

Per la creazione di contenuti intelligenti (Informativi e Formativi) SAPI sfrutta le caratteristiche della piattaforma di e-learning Intelligent Web Teacher ed in particolare di un suo servizio denominato LIA richiamato attraverso il modulo Intelligent Help System.

IWT consente di:

- Creare ontologie composte da concetti collegati tra loro mediante relazioni;
- Associare ai concetti dei contenuti reali opportunamente descritti mediante dei metadata;
- Fare inferenza sull'ontologia di concetti mediante LIA in modo da generare dei percorsi di istruzione personalizzati in funzione di un obiettivo formativo.

Prima di poter creare dei percorsi di istruzione è necessario che in IWT sia definita una ontologia di dominio nella quale sono contenuti anche i concetti relativi al servizio per il quale stiamo scrivendo i contenuti. LIA è in grado di restituire un percorso di istruzione ad hoc in funzione del servizio (*obiettivo formativo*) e del profilo utente.

Sia i contenuti informativi che formativi sono legati ad una Activity che rappresenta un servizio adattivo di help a cui è associato nel database SAPI un evento di help per il dato servizio.

Le operazioni da compiere per la creazione di un Contenuto Informativo sono le seguenti:

1. Preparazione dei contenuti reali.
2. Definizioni delle Entity Atomiche e/o Composte che descrivono i contenuti adattivi.

Le operazioni da compiere per creare un Contenuto Formativo sono le seguenti:

1. Preparazione dei contenuti reali.
2. Definizioni delle Entity Atomiche e/o Composte che descrivono i contenuti adattivi.
3. Aggiunta dei metadati per associare il contenuto formativo ad un concetto dell'ontologia di IWT

IWT integrato con SAPI consente anche:

- Una gestione centralizzata degli utenti
- La gestione di parte delle informazioni relative alle entity
- La gestione dei contenuti reali e del content repository

Il framework SAPI, dal canto suo, offre tool di supporto al progettista di servizi intelligenti per:

- La progettazione delle entity
- L'ottimizzazione dei colori da utilizzare nell'interfaccia utente
- L'ottimizzazione del layout dell'interfaccia utente

Il documento si conclude con la definizione dettagliata dei test da condurre sul prototipo per valutare le componenti base del framework ed il corretto uso dei moduli software che pilotano le periferiche hardware del chiosco.