

# 1 Analisi dei Requisiti

## 1.1 Descrizione Generale

E' richiesto di sviluppare un'applicazione che permetta di leggere e sfruttare un formato RSS 2.0 opportunamente esteso. Tale applicazione deve essere di utilizzo immediato per l'utente, presentando un'interfaccia semplice e facendo quanto più possibile in modo automatico.

Dovrà anche essere progettata la "specifica" del formato RSS 2.0 esteso. Bisognerà tener conto del fatto che la sua ipotetica applicazione nel mondo reale sarà quella di veicolare notizie per l'Associazione degli Industriali della Provincia di Como. A questo proposito il committente ha richiesto che il documento RSS 2 porti con se due informazioni aggiuntive: la prima di *location*, che indichi la località associata al feed; la seconda di *business*, che indichi l'area di business a cui si riferisce il feed. Dovrà essere possibile da parte dell'utente applicare su questi campi dei filtri, in modo da poter ottenere le notizie sensibili per l'area geografica e/ o il settore di business.

Da ultimo dovrà essere fornita un'utility di conversione per generare degli RSS2 estesi a partire da altri sistemi di syndication (RSS diversi dalla versione 2.0 e Atom).

## 1.2 Glossario

*XML (eXtensible Markup Language)*: Linguaggio di markup di utilizzo generale. Tra le altre cose è utilizzato per codificare notizie seguendo la specifica RSS.

*RSS (Really Simple Syndication)*: E' la tecnologia *standard-de-facto* per la rappresentazione di una serie di notizie da veicolare attraverso mezzi informatici. Essendovi diverse specifiche, da qui in avanti quando ci si riferirà alla versione RSS 2.0 dove non altrimenti specificato.

*Item o Entry*: Un'astrazione che rappresenta una singola notizia.

*Channel o Feed*: Una collezione di Entries. Nel nostro caso è codificato con RSS 2.0 (opportunamente esteso). A questo elemento sono associate le eventuali informazioni relative a location e business.

## 1.3 Considerazioni Preliminari

In questo paragrafo sono affrontate le varie problematiche del sistema riscontrate in fase di progettazione e gli aspetti poco chiari nella descrizione del sistema:

- *I vari "dialetti" del linguaggio RSS.* L’RSS presenta varie versioni, spesso incompatibili tra di loro. Per ovviare questo problema si utilizzeranno librerie prodotte da ingegneri Sun. Queste permetteranno di individuare le varie estensioni di RSS e Atom e convertirle in un formato standard che verrà interpretato dall’applicazione, quindi permetteranno una maggiore compatibilità tra i feeds.
- *Personalizzazione del formato.* Essendo l’applicazione progettata per i membri dell’Unione Industriali di Como, e quindi indirizzata ad un’utenza particolare, sarà possibile effettuare un filtraggio per location, che dovrà essere fatto per i vari comuni della provincia, ed un filtraggio per area di business che sarà fatto secondo i settori industriali.
- *Operazioni standard.* L’utente dovrà essere in grado di effettuare le operazioni basilari che si aspetta di trovare in un comune applicativo di lettura di feed: siano queste l’impostazione e l’organizzazione dei “preferiti”, l’aggiornamento automatico o su richiesta dell’utente, oppure anche solo semplicemente la ricerca di stringhe tra tutte le notizie.
- *La serializzazione dei dati.* L’applicazione dovrà tenere in memoria le impostazioni del sistema, per esempio: le preferenze, la marcatura dei messaggi già letti e la lista dei feeds, necessari per mantenere le impostazioni per le successive sessioni di lavoro. Si rende necessaria la serializzazione dei dati, che saranno richiamati nel riavvio dell’applicazione.
- *L’interfaccia “amichevole”.* Pensando all’utenza dell’applicazione, si cercherà di sviluppare l’interfaccia più semplice possibile con comandi chiari e scelte limitate. Essendo l’applicazione è pensata per i membri dell’Unione Industriali di Como, l’interfaccia grafica dovrà rendere possibile, senza fatica da parte dell’utente, la fruizione di notizie di interesse dell’utente nell’area industriale in un ambito geografico determinato.
- *Le impostazioni di default.* L’applicazione dovrà avere delle buone impostazioni di default, e dovrà essere utile per l’utente senza particolari personalizzazioni. Sarà comunque possibile cambiare le impostazioni dopo l’installazione.
- *Gestione di casi eccezionali:* Essendo le notizie con cui lavoreremo dipendenti da fattori esterni al sistema, sarà necessario considerare il caso in cui l’URL di un feed utilizzato dall’applicazione non esista più o venga cambiato, oppure che utilizzi una specifica “illegale”. Si dovranno impedire incosistenze nel modello così originate.

## 1.4 Attori

Gli attori del sistema rappresentano tutte le entità che interagiscono con il sistema.

- *Utente*: E' il protagonista di tutte le interazioni con il sistema: può visionare e filtrare le entry all'interno dei feed; inoltre può modificare la lista dei feeds e chiedere l'aggiornamento della lista delle entry all'interno della lista dei feeds.
- *La nuvola Internet*: la sorgente delle informazioni da cui il sistema acquisisce le informazioni da fornire all'utente. Sarà soltanto un attore passivo.

## 1.5 Scenari e casi d'uso

Segue il diagramma UML dei casi d'uso principali e più generali del sistema. Gli scenari che saranno ritenuti di maggior interesse, verranno analizzati in dettaglio attraverso dei diagrammi UML di sequenza.

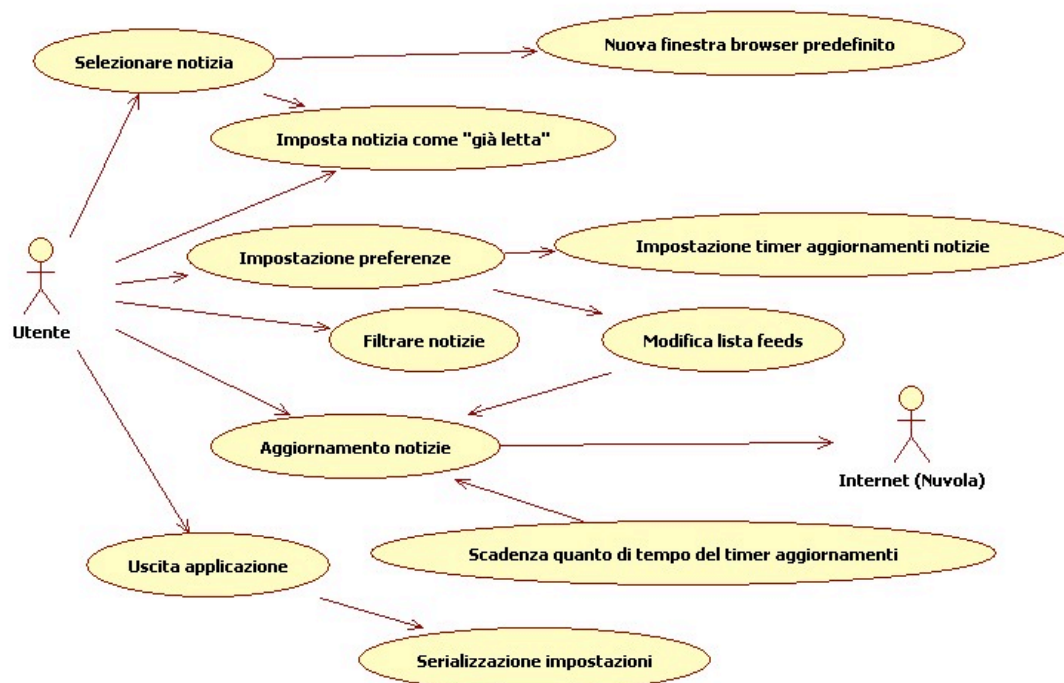


Figura 1: Casi d'uso

Essendo l'Utente l'unico attore "attivo" del sistema ometteremo di specificare gli attori coinvolti nei seguenti scenari.

### 1.5.1 Scenario 1: “selezione e lettura di una notizia”

**Use Case** “Selezionare notizia”

**Pre-condition** Deve esserci almeno una notizia selezionabile.

**Normal flow** L'Utente seleziona la notizia da leggere e chiede al sistema di poter visionare la notizia (ad esempio con un doppio click o con un controllo apposito). Opzionale: Se la notizia non era ancora stata letta, essa sarà marchiata come letta.

**Post-condition** Se la notizia non era ancora stata letta, essa sarà marchiata come letta. Viene aperta una nuova pagina nel browser di default del sistema operativo.

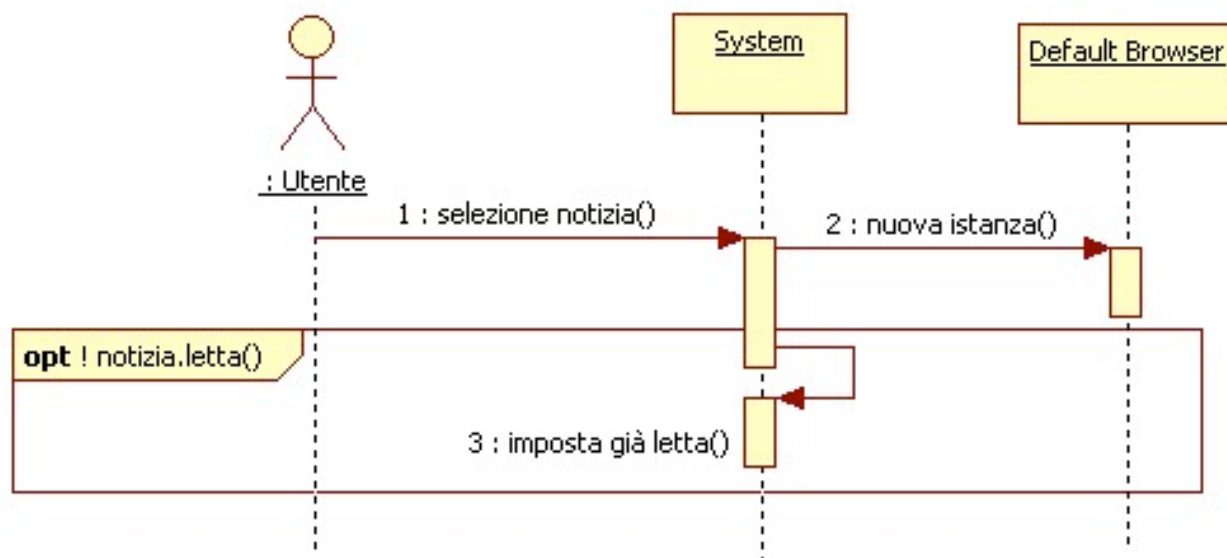


Figura 2: Sequence “selezione e lettura di una notizia”

### 1.5.2 Scenario 2: “filtraggio e ricerca di notizie”

**Use Case** “Filtrare notizie”

**Pre-condition** Nessuna

**Normal flow** L'Utente inserisce, attraverso appositi controlli, dei filtri sotto forma di stringhe.

**Post-condition** La collezione restituita contiene tutte e sole le notizie correlate con il filtro applicato.

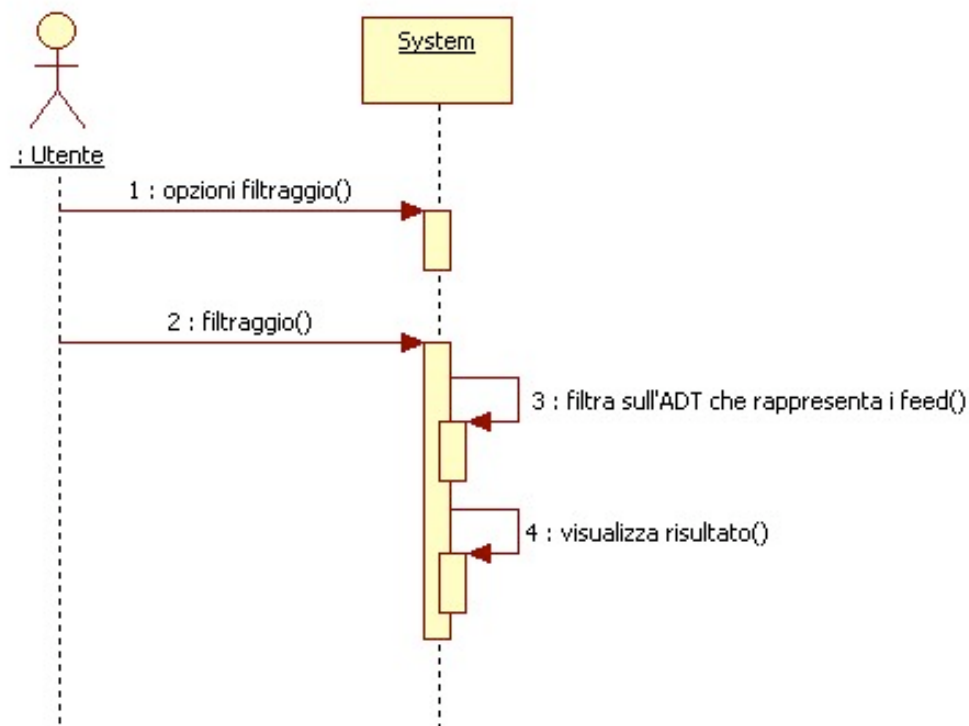


Figura 3: Sequence “ filtraggio e ricerca di notizie”

### 1.5.3 Scenario 3: “modifica preferenze”

**Use Case** “Impostazione preferenze”

**Pre-condition** Nessuna

**Normal flow** L'Utente modifica, attraverso appositi controlli, alcune preferenze del sistema. Questo reagisce opportunamente, dipendentemente dall'impostazione scelta dall'utente.

**Post-condition** Il sistema si porta nello stato corretto.

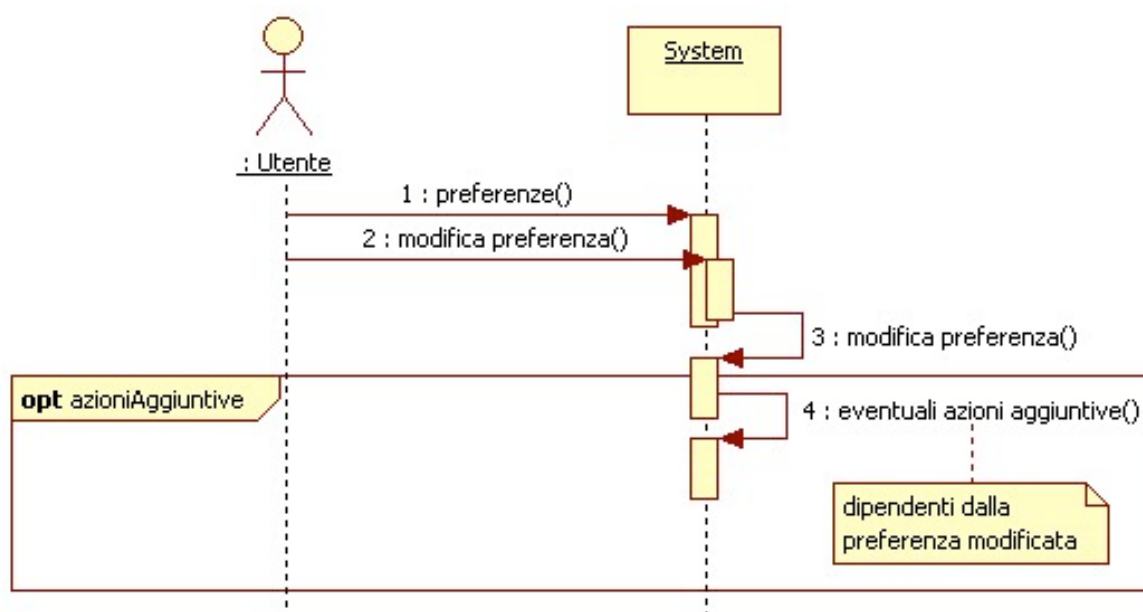


Figura 4: Sequence “ modifica preferenze”

### 1.5.4 Scenario 4: “uscita dall’applicazione”

**Use Case** “Uscita applicazione”

**Pre-condition** Nessuna

**Normal flow** L’Utente esce dall’applicazione. Il sistema reagisce serializzando i dati sensibili in modo trasparente per l’utente, e provvede a terminare.

**Post-condition** Gli oggetti importanti sono serializzati su Filesystem, il processo del sistema non è più esistente.

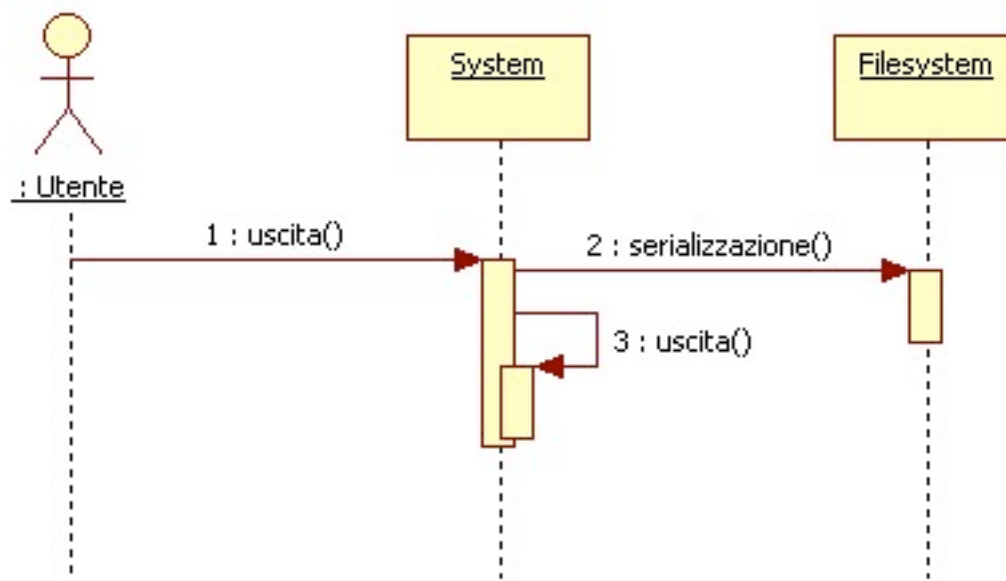


Figura 5: Sequence “uscita applicazione”

## 2 Analisi delle Specifiche

### 2.1 Modello concettuale astratto

L'immagine seguente è una rappresentazione in UML del modello concettuale astratto dell'applicazione.

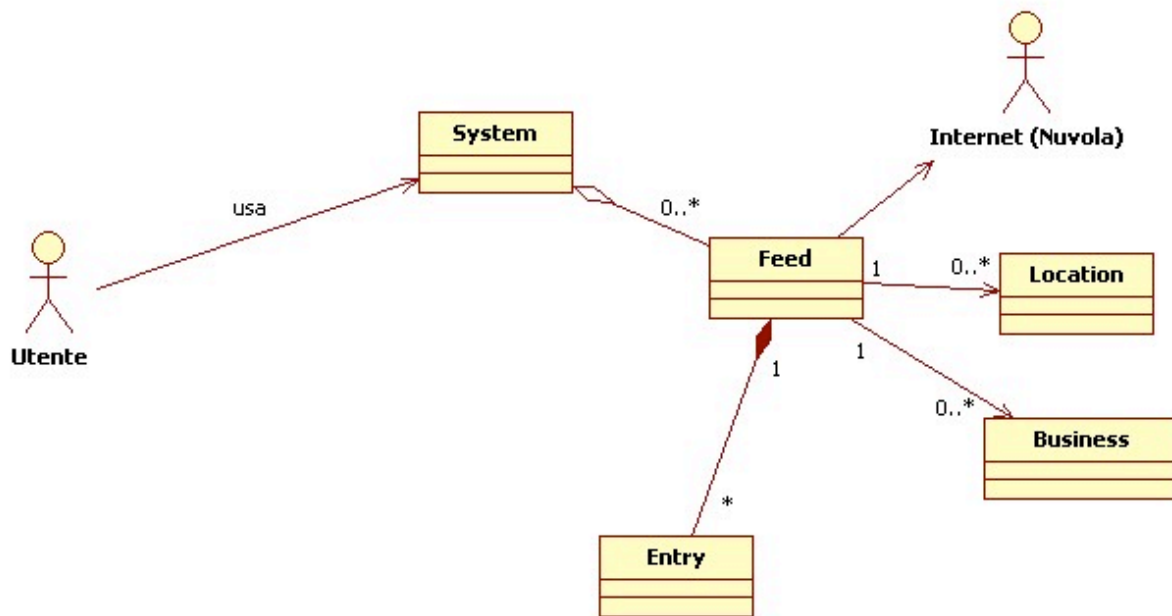


Figura 6: Class "modello concettuale astratto"

L' **Utente** si interfaccia direttamente con il sistema **System**. Questo contiene tra gli altri elementi atti a garantirne il funzionamento (verranno definiti nella fase di Design), la struttura dati che astrae il concetto di Feeds RSS: essa è composta dai due elementi fondamentali **Feed** e **Entry**.

Al **Feed** è possibile associare una o più informazioni di **Location** e **Business**. Inoltre, è previsto il caso in cui il **Feed** sia di carattere generale e non associato alle informazioni appena citate. Si ricordi che il **Feed** trae le sue informazioni da sistemi esterni e remoti (rappresentati qui in generale da **Internet**).