

# PROGETTAZIONE

NICOLA MORETTO (MATR. 578258)

19 ottobre 2012

Il documento riporta le informazioni di progettazione riguardanti l'interfaccia grafica per la visualizzazione e la navigazione dei contenuti.

VERSIONE	DATA	MODIFICHE
0.1	15-10-2012	Stesura iniziale del documento.
0.2	17-10-2012	Redatto il capitolo 2.
0.3	18-10-2012	Redatte le sezioni 3.3, 4.3 e 4.4.
0.4	19-10-2012	Redatti i capitoli 3 e 4.

Tabella 1: Registro delle modifiche

## INDICE

---

1	INTRODUZIONE	6
1.1	Obiettivi	6
1.2	To do	6
1.3	Riferimenti informativi	6
2	ARCHITETTURA	7
2.1	Architettura generale	7
2.2	Componenti architetturali	7
2.2.1	Componente Model	7
2.2.2	Componente View	7
2.2.3	Componente Controller	8
3	COMPONENTE MODEL	9
3.1	Package model	9
3.1.1	Content	9
3.1.2	Entity	9
3.1.3	Label	9
3.1.4	Meaning	9
3.1.5	User	9
3.2	Package model.criteria	9
3.2.1	Criterion	10
3.2.2	Author	10
3.2.3	Emotion	10
3.2.4	Intention	10
3.2.5	PublicationDate	10
3.2.6	Rating	10
3.2.7	Topic	10
3.2.8	Type	10
3.3	Package model.filter	10
3.3.1	Filter	10
3.3.2	FilterManager	10
3.3.3	ListFilter	10
3.3.4	RangeFilter	11
3.3.5	SwitchFilter	11
3.3.6	ValueFilter	11
3.4	Package model.search	11
3.4.1	SearchEntityList	11
3.4.2	SearchManager	11
3.4.3	SearchQuery	11
3.4.4	SearchScope	12
3.4.5	SearchTerm	12
3.5	Package model.search.provider	12
4	COMPONENTE VIEW	13
4.1	Package view	13
4.1.1	MainWindow	13
4.2	Package view.content	13
4.2.1	ContentView	13
4.3	Package view.filter	14

4.3.1	FilterContainer	14
4.3.2	FilterView	14
4.3.3	ListFilterView	15
4.3.4	RangeFilterView	15
4.3.5	SwitchFilterView	15
4.3.6	ValueFilterView	15
4.4	Package view.search	15
4.4.1	EntityList	15
4.4.2	EntityView	15
4.4.3	SearchBar	15
4.4.4	SearchScopeSelector	16
4.4.5	LabelEntityList	16
4.5	Package view.timeline	16
5	COMPONENTE CONTROLLER	17
5.1	Package controller	17
5.2	Package controller.filter	17
5.3	Package controller.search	17

## INTRODUZIONE

---

### 1.1 OBIETTIVI

- aggiunta, rimozione o aggiornamento (motore di ricerca) di nuovi ambiti di ricerca;

### 1.2 TO DO

- Aggiungere la lista di informazioni essenziali e aggiuntive di un contenuto nei requisiti.

### 1.3 RIFERIMENTI INFORMATIVI

1. Analisi dei requisiti (*analisi\_dei\_requisiti\_1.0* allegata alla presente documentazione);
2. Sistema di classificazione (*sistema\_di\_classificazione\_2.0* allegato alla presente documentazione).

## ARCHITETTURA

---

### 2.1 ARCHITETTURA GENERALE

L'architettura del sistema software rispecchia il design pattern architetturale MVC, che prevede e garantisce la separazione delle tre componenti fondamentali del sistema:

**MODEL** Racchiude i dati e le informazioni dell'applicazione e definisce le modalità di accesso e fruizione degli stessi da parte delle altre componenti (*Controller* e *View*).

**VIEW** Rappresenta l'interfaccia grafica mediante la quale vengono visualizzate le informazioni e i dati conservati nel *Model* e l'utente può interagire con il sistema. La rilevazione dell'avvenuta interazione dell'utente è responsabilità di tale componente, mentre la gestione della reazione è demandata al *Controller*.

**CONTROLLER** Incorpora la logica di controllo dell'applicazione, inizializzando il sistema e traducendo l'interazione dell'utente con l'interfaccia grafica (*View*) in operazioni sui dati (*Model*).

### 2.2 COMPONENTI ARCHITETTURALI

#### 2.2.1 Componente Model

La componente *model* conserva tutte i tipi di informazioni connessi alla ricerca:

- gli ambiti (etichette, frasi);
- i criteri (termini di ricerca, entità);
- i filtri (argomento, data di pubblicazione, emozioni, ...);
- i risultati (contenuti informativi).

#### 2.2.2 Componente View

La componente *view* rappresenta l'interfaccia grafica mediante la quale l'utente interagisce con il sistema per effettuare una ricerca, raffinarne i criteri o consultarne i risultati.

**LIVELLI** Essa è organizzata in quattro livelli (o strati) distinti, che includono rispettivamente:

- gli strumenti e le informazioni connesse alla ricerca (barra di ricerca, etichette, entità, ...);
- i filtri di ricerca;
- i risultati della ricerca (proprietà e relazioni dei contenuti informativi);
- la cronologia (organizzazione temporale dei contenuti).

### 2.2.3 Componente Controller

La componente *controller* gestisce l'interazione dell'utente con l'interfaccia grafica e le operazioni connesse alla ricerca e alla visualizzazione dei contenuti informativi, tra cui:

- la configurazione dei criteri di ricerca (selezione di un'entità di un'etichetta, gestione delle entità);
- il reperimento dei contenuti informativi corrispondenti ai criteri di ricerca;
- la creazione e la gestione dei filtri di ricerca;
- l'aggiornamento dei risultati di ricerca visualizzati a fronte di modifiche alle entità e ai filtri di ricerca;
- la consultazione dei contenuti informativi da parte dell'utente;
- ...



## COMPONENTE MODEL

---

Questo capitolo illustra i componenti del *Model*, per ciascuno dei quali è indicato il nome della classe accompagnata da un'identificazione sintetica, separati dal carattere ' | ' (separatore verticale).

### 3.1 PACKAGE MODEL

#### 3.1.1 *Content* | *Contenuto informativo*

La classe rappresenta un generico contenuto informativo pubblicato dagli utenti nella piattaforma.

##### PROPRIETÀ

- Argomento
- Autore
- Data di pubblicazione
- Giudizio
- Tipo
- Titolo

#### 3.1.2 *Entity* | *Entità del dominio*

La classe modella le entità del dominio della piattaforma.

#### 3.1.3 *Label* | *Etichetta del dizionario*

La classe modella le etichette del dizionario della piattaforma, ciascuna delle quali può possedere molteplici accezioni, cui sono associate altrettante entità distinte.

#### 3.1.4 *Meaning* | *Accezione di un'etichetta*

La classe modella le entità del dominio della piattaforma.

#### 3.1.5 *User* | *Utente*

La classe rappresenta un generico utente della piattaforma.

### 3.2 PACKAGE MODEL.CRITERIA

Ciascun contenuto può essere classificato in accordo a differenti criteri, ciascuno dei quali prende in esame una differente proprietà del contenuto medesimo, come l'autore, la data di pubblicazione, il tipo, le emozioni associate, ....

Ciascuna proprietà può essere propria del contenuto o funzionale al contesto di ricerca (attinenza, ...).

#### 3.2.1 *Criterion* | *Criterio di classificazione*

Interfaccia di tutti i criteri di classificazione.

#### 3.2.2 *Author* | *Autore*

#### 3.2.3 *Emotion* | *Emozione*

#### 3.2.4 *Intention* | *Intenzione*

#### 3.2.5 *PublicationDate* | *Data di pubblicazione*

#### 3.2.6 *Rating* | *Giudizio*

#### 3.2.7 *Topic* | *Argomento*

#### 3.2.8 *Type* | *Tipo di contenuto*

### 3.3 PACKAGE MODEL.FILTER

Ciascun filtro è associato ad una proprietà dei risultati di ricerca (*Criterion*) e partiziona automaticamente l'insieme dei possibili valori in due sottoinsiemi: ammessi o bloccati. Nella configurazione iniziale, tutti i valori possibili di una proprietà sono ammessi, mentre l'utente può intervenire secondo modalità differenti per alterare tale partizionamento (vedi sezione 4.3).

Il valore che ciascun contenuto (da intendersi come risultato di una ricerca) ha rispetto ad una proprietà associata ad un filtro può quindi appartenere ad uno dei due sottoinsiemi: il contenuto viene mostrato solo se tutti i valori delle proprietà in questione risultano ammessi.

#### 3.3.1 *Filter* | *Filtro di ricerca*

Tale componente rappresenta l'interfaccia dei filtri per il raffinamento dei risultati di una ricerca e viene implementata dalle classi che modellano le tipologie standard di filtri di ricerca (*ListFilter*, *RangeFilter*, *SwitchFilter* e *ValueFilter*).

#### 3.3.2 *FilterManager* | *Gestione dei filtri*

Tale componente rappresenta l'interfaccia del package *model.filter*, utile a esporre le funzionalità per l'istanziamento e la gestione dei filtri.

DESIGN PATTERN: Facade, Singleton

#### 3.3.3 *ListFilter* | *Filtro con lista di valori*

Il componente modella un filtro basato su una lista di possibili valori, ciascuno dei quali può essere autorizzato o bloccato dall'utente.

### 3.3.4 *RangeFilter* | Filtro con intervallo di valori

Il componente modella un filtro basato su un intervallo di valori ordinati (numeri, date, ...). Siano *inf* e *sup* rispettivamente l'estremo inferiore e superiore dell'intervallo: risultano dunque ammessi tutti e soli i valori *v* tali che  $v \in [inf, sup]$ .

All'utente è consentito scegliere i valori desiderati di *inf* e *sup* alle seguenti condizioni:

- *sup* sia un valore valido o  $+\infty$ ;
- *inf* sia un valore valido o  $-\infty$ ;
- $min \leq max$ ;
- se è definito un valore attuale *current*, allora deve valere  $min \leq current \leq max$ .

Se l'insieme dei valori ordinati prevede un minimo *min* e/o un massimo *max* si aggiungono le seguenti condizioni:

- $sup \leq max$ ;
- $inf \geq min$ .

### 3.3.5 *SwitchFilter* | Filtro a doppio stato

Il componente modella un filtro il cui partizionamento è predefinito e invariabile e che l'utente può solamente decidere di abilitare o disattivare.

### 3.3.6 *ValueFilter* | Filtro con soglia di valore

Il componente modella un filtro basato su una soglia di valore. Sia *value* il valore scelto: in tal caso risultano ammessi tutti e soli i valori *x* validi tali che  $x \geq value$ ; se l'insieme prevede un minimo *min* e/o un massimo *max* dev'essere soddisfatta anche la condizione  $min \leq x \leq max$ .

## 3.4 PACKAGE MODEL.SEARCH

### 3.4.1 *SearchEntityList* | Gestione della ricerca

### 3.4.2 *SearchManager* | Gestione della ricerca

## DESIGN PATTERN Facade

### 3.4.3 *SearchQuery* | Query di ricerca

Tale componente modella la query di ricerca, ossia la stringa inserita dall'utente nella barra di ricerca (*SearchBar*), contenente una lista di termini o espressioni separati da virgola.

La classe fornisce i metodi per effettuare l'analisi sintattica della stringa al fine di estrapolare i termini (*SearchTerm*) da cercare e stabilire quali di essi siano etichette o frasi.

#### 3.4.4 *SearchScope* | Ambito di ricerca

La classe rappresenta il generico ambito di ricerca, che l'utente può selezionare mediante il corrispondente selettore (*SearchScopeSelector*).

#### 3.4.5 *SearchTerm* | Chiave di ricerca

La componente rappresenta un'interfaccia comune a tutti i tipi di termini di ricerca, ossia frasi e etichette.

Ciascun termine di ricerca viene memorizzato all'interno di un'istanza di *SearchQuery* e viene passato al *SearchManager* per consentire l'esecuzione della ricerca.

### 3.5 PACKAGE MODEL.SEARCH.PROVIDER

COMPONENTE VIEW

---

Questo capitolo illustra i componenti del *View*, per ciascuno dei quali è indicato il nome della classe accompagnata da un'identificazione sintetica, separati dal carattere ' | ' (separatore verticale).

## 4.1 PACKAGE VIEW

4.1.1 *MainWindow* | *Finestra principale*

La finestra principale rappresenta il contenitore all'interno del quale vengono inseriti e opportunamente collocati tutti gli elementi grafici dell'interfaccia, tra cui gli strumenti di ricerca, i filtri e i contenuti.

## 4.2 PACKAGE VIEW.CONTENT

4.2.1 *ContentView* | *Contenuto informativo*

Si tratta del componente grafico deputato a rappresentare graficamente un contenuto informativo, risultato di una ricerca effettuata dall'utente, e le relative informazioni, in formato grafico o testuale e distinte in essenziali o aggiuntive.

**INFORMAZIONI ESSENZIALI** Le informazioni essenziali sono utili per una immediato e preciso inquadramento del contenuto informativo da parte dell'utente al fine di stabilirne la rilevanza soggettiva.

**AUTORE** L'autore del contenuto è l'utente (*User*) che lo ha pubblicato all'interno della piattaforma e viene rappresentato testualmente mediante il suo *nome utente o proprio*.

**ATTINENZA** Il grado di attinenza di un contenuto rispetto ai criteri di ricerca corrisponde - in percentuale - al rapporto tra le entità assegnate e quelle cercate. Tale informazione viene rappresentata graficamente variando proporzionalmente la *dimensione* dell'elemento grafico.

**DATA DI PUBBLICAZIONE** La data di pubblicazione del contenuto viene indicata testualmente.

**TIPO** Il tipo di contenuto viene rappresentato graficamente mediante la forma<sup>1</sup> specifica dell'elemento grafico e può essere:

- Comunicazione
- Discorso
- Domanda
- Evento
- Pensiero

---

<sup>1</sup> Le forme utilizzate sono elementari per garantire l'immediata riconoscibilità da parte di qualsiasi tipo di utente, evitando l'impiego di forme potenzialmente ambigue o ignote a seconda del suo profilo sociale, culturale, geografico, ...

- Recensione
- Risposta

**TITOLO** Il titolo assegnato al contenuto rappresenta un'informazione chiave e viene rappresentata in formato testuale.

**INFORMAZIONI AGGIUNTIVE** Le informazioni aggiuntive forniscono all'utente dettagli utili per approfondire l'esame di un contenuto.

**ARGOMENTO** Ciascun contenuto può riferirsi al più ad un argomento, rappresentabile in maniera grafica (colore o simbolo) o testuale.

**EMOZIONI** A ciascun contenuto possono essere associate diverse emozioni, che esprimono lo stato d'animo dell'autore al momento della sua redazione o pubblicazione. La lista delle emozioni associate ad un contenuto viene visualizzata in formato testuale.

**ENTITÀ** A ciascun contenuto possono essere assegnate delle etichette, che riferiscono altrettante entità del dominio da visualizzare in formato testuale (mediante il rispettivo nome).

**INTENZIONI** A ciascun contenuto possono essere associate delle intenzioni, che possano fornire delle linee guida interpretative ai lettori. La lista delle intenzioni associate ad un contenuto viene visualizzata in formato testuale.

#### 4.3 PACKAGE VIEW.FILTER

Ciascun filtro è associato ad una proprietà dei risultati di ricerca, ossia i contenuti informativi, e partiziona automaticamente l'insieme dei possibili valori in due sottoinsiemi: ammessi o bloccati. Nella configurazione iniziale, tutti i valori possibili di una proprietà sono ammessi, mentre l'utente può intervenire secondo modalità differenti per alterare tale partizionamento.

Il valore che ciascun contenuto (da intendersi come risultato di una ricerca) ha rispetto ad una proprietà associata ad un filtro può appartenere ad uno dei due sottoinsiemi: un contenuto viene mostrato solo se tutti i valori delle proprietà in questione risultano ammessi.

##### 4.3.1 *FilterContainer* | Livello filtri

Il *FilterContainer* raccoglie e mantiene separati in un livello distinto gli elementi grafici di tipo *FilterView*, che forniscono all'utente gli strumenti per regolare i filtri di ricerca.

**DESIGN PATTERN:** Facade, Singleton.

##### 4.3.2 *FilterView* | Filtro di ricerca

Tale componente rappresenta l'interfaccia delle componenti grafiche che rappresentano le tipologie standard dei filtri per il raffinamento dei risultati di una ricerca, ossia (*ListFilterView*, *RangeFilterView*, *SwitchFilterView* e *ValueFilterView*).

#### 4.3.3 *ListFilterView* | Filtro con lista di valori

Il componente grafico - associato alla classe *ListFilter* - visualizza le liste dei valori ammissibili e bloccati e consente all'utente di:

- spostare un valore da una lista all'altra;
- azzerare il filtro, spostando automaticamente tutti i valori nella lista degli ammissibili.

#### 4.3.4 *RangeFilterView* | Filtro con intervallo di valori

Il componente grafico - associato alla classe *RangeFilter* - permette all'utente di specificare l'estremo inferiore e superiore dell'intervallo dei valori ammissibili secondo le regole definite nella suddetta classe.

#### 4.3.5 *SwitchFilterView* | Filtro a doppio stato

Il componente grafico consente di abilitare o disattivare un filtro di tipo *SwitchFilter* e ne visualizza lo stato corrente.

#### 4.3.6 *ValueFilterView* | Filtro con soglia di valore

Il componente grafico - associato alla classe *ValueFilter* - consente all'utente di modificare il valore della soglia secondo le condizioni previste dalla suddetta classe.

### 4.4 PACKAGE VIEW.SEARCH

#### 4.4.1 *EntityList* | Elenco delle entità cercate

Al termine della ricerca, tale componente visualizza le entità (*EntityView*) riferite dalle etichette e cercate tra i contenuti.

#### 4.4.2 *EntityView* | Entità

Tale componente grafico visualizza le informazioni essenziali associate ad un'entità e consente all'utente di:

- visualizzare la lista delle entità padre;
- visualizzare la lista delle entità figlie;
- sostituire l'entità corrente con un padre o un figlio.
- eliminare l'entità corrente.

#### 4.4.3 *SearchBar* | Barra di ricerca

La barra di ricerca rappresenta il componente grafico (campo di testo) mediante il quale l'utente può inserire i termini di ricerca.

#### 4.4.4 *SearchScopeSelector* | *Selettore dell'ambito di ricerca*

Il selettore dell'ambito di ricerca è un componente grafico (menu a tendina) che permette all'utente di circoscrivere la ricerca ad informazioni specifiche:

##### TUTTO

Estende la ricerca a tutti i tipi di informazioni indicizzate o ricercabili all'interno della piattaforma.

##### ETICHETTE

Limita la ricerca alle sole etichette assegnate ai contenuti.

##### FRASI

Limita la ricerca alle informazioni presenti in un contenuto (titolo, corpo, ...).

#### 4.4.5 *LabelEntityList* | *Accezioni di un'etichetta*

Ove il sistema individui tra i termini cercati delle etichette aventi accezioni multiple, tale componente grafico provvedere a mostrare - per ciascuna etichetta - la lista delle entità (*EntityView*) riferite e consente all'utente di selezionare quella rispetto cui intenda procedere con la ricerca.

Al termine della ricerca, l'entità selezionata viene mostrata nel componente *EntityList*.

### 4.5 PACKAGE VIEW.TIMELINE



## COMPONENTE CONTROLLER

---

Questo capitolo illustra i componenti del *Controller*, per ciascuno dei quali è indicato il nome della classe accompagnata da un'identificazione sintetica, separati dal carattere ' | ' (separatore verticale).

5.1 PACKAGE CONTROLLER

5.2 PACKAGE CONTROLLER.FILTER

5.3 PACKAGE CONTROLLER.SEARCH