



UNIVERSITÀ DI PISA

Dipartimento di Informatica

Corso di Laurea Magistrale in Informatica per l'economia e per l'azienda
(Business Informatics)

TESI DI LAUREA

**PROGETTAZIONE E SVILUPPO DI UN SISTEMA
PER LA GESTIONE DI LOG DI EVENTI**

RELATORE

Prof. Roberto BRUNI

Candidato

Francesco SANTERAMO

ANNO ACCADEMICO 2014-15

Riassunto

I processi di business ricoprono nel mondo di oggi una parte fondamentale, guidando i servizi e le funzioni delle aziende e delle organizzazioni di tutto il mondo. Mentre la maggior parte degli attuali sistemi informativi aziendali dispone già di funzioni per l'esecuzione di tali processi e la loro registrazione tramite log di eventi, ancora molto deve essere fatto per quanto riguarda il monitoraggio e l'analisi di queste esecuzioni. In questo ambito si inserisce la disciplina del *Process Mining*, capace per mezzo di innovative tecniche di analisi e monitoraggio dei processi di entrare con successo in questa area di ricerca. Il lavoro svolto si inserisce nel contesto di questa disciplina e nasce dalla necessità di avere un sistema che permetta di creare facilmente, manipolare ed esaminare log di eventi, anche a fini didattici, utilizzando come tecnologia XES, dal 2010 lo standard XML ufficiale per la memorizzazione di log di eventi per quanto riguarda il Process Mining. L'obiettivo di questa tesi è quindi la progettazione e lo sviluppo di una applicazione basata sullo standard XES per la creazione, modifica ed analisi di log di eventi memorizzati utilizzando questo formato. Molte delle funzionalità che sono state pensate per questo nuovo sistema non sono presenti negli altri programmi disponibili per la gestione ed analisi dei log. L'obiettivo del progetto è stato quello di fornire nuove funzionalità a questa area di ricerca, complementando ed integrandosi allo stesso tempo con le altre applicazioni già presenti in questo settore.

INDICE

1	INTRODUZIONE	7
1.1	Ambito del lavoro di tesi	7
1.2	Obiettivo della tesi	12
1.3	Contenuto della tesi	13
2	TECNOLOGIE UTILIZZATE	15
2.1	Standard XES	16
2.1.1	La terminologia utilizzata	18
2.1.2	Il meta-modello di XES	19
2.1.3	Serializzazione XML di XES	36
2.1.4	Estensioni standard	38
2.2	Sviluppo dell'applicazione	49
2.2.1	Il linguaggio di programmazione	50
2.2.2	L'ambiente di sviluppo	52

INDICE

2.2.3	Il pattern Modified Model-View-Controller	53
2.2.4	Le librerie esterne	53
3	PROGETTAZIONE DELL'APPLICAZIONE	65
3.1	Pattern architetturale	65
3.1.1	Il pattern Model-View-Controller tradizionale	66
3.1.2	Il Modified Model-View-Controller	68
3.2	Struttura dei pacchetti Java	71
3.3	Diagrammi UML delle classi	73
3.3.1	Il pacchetto Starter	74
3.3.2	Il pacchetto Model	74
3.3.3	Il pacchetto View	75
3.3.4	Il pacchetto Controller	80
3.3.5	Vista sulle interazioni tra le classi principali del sistema	90
3.3.6	Le directory del progetto	92
4	SVILUPPO: LA GESTIONE DEI DOCUMENTI	95
4.1	L'interfaccia dell'applicazione	96
4.2	Gestione dei documenti	102
4.2.1	La classe EventLogEditor	103
4.2.2	Apertura dei documenti: Open e Recent	108
4.2.3	Salvataggio dei documenti: Save e Save As	114
4.2.4	Chiusura di una finestra: Close	121
4.2.5	Stampa di un documento: Print	128
4.2.6	Funzioni di utilità per l'editing dei contenuti	130
4.2.7	Gestione della dimensione del testo: Zoom	132

5	SVILUPPO: GENERAZIONE E ANALISI DEI LOG	134
5.1	Generazione dei log di eventi	134
5.1.1	Le funzioni Template e Completion dell'editor	136
5.1.2	Definizione di una sintassi per la generazione dei log	142
5.1.3	Generazione di log a partire da espressioni	150
5.2	Analisi dei log di eventi	167
5.2.1	EventLogEditor e il supporto all'analisi: i filtri	167
5.2.2	Validazione di un log XES	171
5.2.3	Analisi di un log XES	174
5.3	Import ed export dei documenti	186
5.3.1	La funzione di import	187
5.3.2	La funzione di export	191
5.4	Errori riscontrati sullo XES XSD 2.0 e sulla libreria OpenXES	
	2.0	200
5.4.1	Errore sullo XES Schema 2.0	200
5.4.2	Errori sulla libreria OpenXES 2.0	202
6	SVILUPPO: GESTIONE DELL'APPLICAZIONE	205
6.1	Disposizione di una nuova finestra all'interno dell'applicazione	205
6.2	Gestione di molteplici finestre: i FooterAreaButton	210
6.3	Interazione con l'interfaccia utente	213
6.3.1	La modifica delle dimensioni della finestra principale	213
6.3.2	La chiusura della finestra principale	219
6.4	Gestione delle configurazioni	222
6.5	Note sul codice: lingua, commenti e JavaDoc	223

INDICE

7 CASI D'USO	226
7.1 La generazione di un log	226
7.2 L'analisi di un log	232
8 CONCLUSIONI	237
Bibliografia	241

CONCLUSIONI

L'obiettivo di questa tesi era la progettazione e lo sviluppo di una applicazione basata sullo standard XES per la creazione, modifica ed analisi di log di eventi memorizzati utilizzando questo formato.

Come abbiamo visto, l'applicazione dà la possibilità di aprire e modificare log di eventi reali utilizzando un editor molto efficace.

L'applicazione permette inoltre di generare nuovi log, utilizzando ad esempio le funzioni di Template e Completion dell'editor, oppure specificando, attraverso il linguaggio di espressioni definito all'interno del sistema, possibili sequenze di attività appartenenti ad una precisa esecuzione del processo.

La generazione di nuovi log può inoltre essere fatta anche attraverso l'utilizzo della semantica in formato CSV per la definizione di sequenze di attività connesse ad una stessa istanza di processo.

Molti sono quindi i modi messi a disposizione per la didattica attraverso i quali un utente può generare un nuovo log e successivamente gestirlo e

CAPITOLO 8. CONCLUSIONI

analizzarlo.

Il sistema sviluppato permette a questo punto la validazione e l'analisi di ogni log aperto e la generazione di un report di analisi completo e dettagliato, che ha il suo massimo significato nella visualizzazione della *footprint matrix* generata a partire dal log stesso: in questa matrice vengono mostrate le dipendenze causali tra le attività del log, dalle quali poi si possono trarre importanti conclusioni sull'esecuzione del processo analizzato.

Grazie ad un intuitivo e completo sistema di filtri, l'applicazione dà modo all'utente di definire i parametri dell'analisi e ottenere un report ed una matrice di maggior interesse per l'analisi del processo.

L'obiettivo dell'analisi, come ricordato spesso in questo lavoro, è la ricerca della conoscenza sul processo in esame, a partire dalla registrazione delle sue esecuzioni, il log di eventi appunto.

Una volta elaborati i log e generati i report di analisi, l'applicazione dà la possibilità di esportare questi contenuti in formati diversi, oltre a quelli testuali che sono l'immediata rappresentazione del contenuto del documento all'interno di un file esterno.

Abbiamo visto infatti come un log possa essere esportato al di fuori del sistema utilizzando la semantica in formato CSV che abbiamo definito nel corso dello sviluppo.

Oppure, come un report di analisi possa essere esportato in formati diversi, a seconda delle esigenze e di quale sia l'utilizzo che intendiamo farne in seguito. I formati disponibili per questo contenuto sono: HTML, PDF, LaTeX e PNG.

Insieme a questa funzionalità di export, ne viene fornita una di import

nel sistema, per poter ricaricare documenti precedentemente esportati.

Durante la progettazione di questo sistema, numerose sono state le funzionalità che si è deciso di sviluppare (molte di queste non presenti in altre applicazioni di questo genere), al fine di rendere questo applicativo uno strumento utile per scopi didattici e interessante all'interno della disciplina del Process Mining e tra gli strumenti già sviluppati per l'analisi dei processi.

L'intento era infatti quello di fornire nuove funzionalità a questa area di ricerca e di integrarsi con le altre applicazioni già presenti in questo ambito.

Quello che è stato ottenuto al completamento del progetto è una applicazione facile da utilizzare, robusta, soprattutto per quanto riguarda la gestione di tutte le possibili situazioni che possono accadere durante il suo utilizzo, adatta alla gestione e all'analisi dei log sia in campo accademico sia in quello aziendale e professionale.

Per quanto riguarda la parte di progettazione e sviluppo inoltre, il progetto presenta codice organizzato in modo chiaro dal punto di vista architetturale, grazie all'utilizzo del pattern Modified Model-View-Controller, largamente commentato e documentato, con tutte le funzionalità implementate finemente interconnesse tra di loro.

Queste caratteristiche facilitano eventuali estensioni future dell'applicazione.

L'obiettivo a lungo termine che si propone questo progetto è quello di continuare a crescere in termini di funzionalità aggiunte e di utenza, con il chiaro intento di diventare un importante strumento all'interno della disciplina dell'analisi dei processi. Ad esempio, delle funzionalità che potrebbero essere facilmente incluse in futuro riguardano il confronto tra footprint matrix, per

CAPITOLO 8. CONCLUSIONI

vedere come piccoli cambiamenti sulle tracce si riflettono sulle dipendenze tra eventi e l'integrazione di altre tecniche di analisi dei log.

L'applicazione è stata sviluppata in Java e verrà rilasciata con licenza Open Source con il nome di *Event Log Manager*.

BIBLIOGRAFIA

- [1] Wil M. P. van der Aalst. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Publishing Company, Incorporated, 1st edition, 2011.

- [2] Wil M. P. van der Aalst, Arya Adriansyah, Ana Karla Alves de Medeiros, Franco Arcieri, Thomas Baier, et al. Process mining manifesto. In Florian Daniel, Kamel Barkaoui, and Schahram Dustdar, editors, *Business Process Management Workshops - BPM 2011 International Workshops, Clermont-Ferrand, France, August 29, 2011, Revised Selected Papers, Part I*, volume 99 of *Lecture Notes in Business Information Processing*, pages 169–194. Springer, 2011.

- [3] Wil M. P. van der Aalst, Ton Weijters, and Laura Maruster. Workflow mining: Discovering process models from event logs. *IEEE Trans. Knowl. Data Eng.*, 16(9):1128–1142, 2004.

BIBLIOGRAFIA

- [4] Apple. Concepts in objective-c programming - model-view-controller, January 2012. <https://developer.apple.com/library/mac/documentation/General/Conceptual/CocoaEncyclopedia/Model-View-Controller/Model-View-Controller.html>.
- [5] Robert Eckstein. Java se application design with mvc, March 2007. <http://www.oracle.com/technetwork/articles/javase/index-142890.html>.
- [6] Eclipse. Eclipse project, April 2015. <https://eclipse.org/eclipse/>.
- [7] Christian W. Günther and Eric Verbeek. Openxes developer guide v 2.0, March 2014. http://www.xes-standard.org/_media/openxes/openxesdeveloperguide-2.0.pdf.
- [8] Christian W. Günther and Eric Verbeek. Xes standard definition v. 2.0, March 2014. http://www.xes-standard.org/_media/xes/xesstandarddefinition-2.0.pdf.
- [9] Java. JavaTM platform, standard edition 8 api specification, June 2015. <https://docs.oracle.com/javase/8/docs/api/>.
- [10] HTML To LaTeX. Html to latex converter in java, April 2015. <http://htmltolatex.sourceforge.net/index.html>.
- [11] OpenCSV. Java library for csv file management, April 2015. <http://opencsv.sourceforge.net/index.html>.
- [12] RSyntaxTextArea. A syntax highlighting text component, April 2015. <http://bobbylight.github.io/RSyntaxTextArea/>.

BIBLIOGRAFIA

- [13] Flying Saucer. Xml/xhtml and css 2.1 renderer in pure java, April 2015.
<https://code.google.com/p/flying-saucer/>.