



UNIVERSITÀ DI PISA

Corso di Laurea in Informatica Umanistica

LAUREA TRIENNALE

**Dalla codifica del manoscritto all'edizione  
elettronica: il caso del Codice Pelavicino**

**Candidato:** *Alessio Miaschi*

**Relatore:** *Enrica Salvatori*

**Correlatore:** *Roberto Rosselli Del Turco*

Anno Accademico 2014-2015

*A Chiara*

# Indice

1	Introduzione .....	5
2	Il Codice Pelavicino .....	8
2.1	La storia e i contenuti .....	8
2.2	Le caratteristiche .....	10
2.3	Le edizioni.....	11
3	La codifica del manoscritto.....	12
3.1	Introduzione all'edizione digitale.....	12
3.1.1	L'edizione digitale e l'Image-based digital edition.....	12
3.1.2	Le fasi e il team di lavoro .....	14
3.2	La codifica di testi .....	15
3.2.1	XML e gli schemi di codifica.....	17
3.2.2	La TEI.....	19
3.3	La codifica del Codice Pelavicino.....	20
3.3.1	Gli strumenti.....	20
3.3.2	Lo schema di codifica.....	22
3.3.3	La codifica .....	23
4	Dalla codifica dei testi a EVT .....	33
4.1	Il sistema di correzione .....	33
4.1.1	Il linguaggio XSL .....	33
4.1.2	L'interfaccia grafica e le fasi di realizzazione.....	35
4.2	L'edizione digitale con EVT .....	40
4.2.1	Fasi preliminari.....	40
4.2.2	L'interfaccia grafica .....	41
5	<i>I signa tabellionis</i> .....	44
5.1	Cenni storici .....	44
5.2	<i>I signa</i> nelle altre edizioni .....	46
5.3	<i>I signa</i> nel Codice Pelavicino digitale.....	49

5.3.1 Il <facsimile> XML.....	49
5.3.2 La pagina HTML.....	51
6 Conclusioni .....	55
Bibliografia.....	57

# 1 Introduzione

La seguente relazione intende focalizzarsi sulla lunga attività di tirocinio e di successivo approfondimento svoltasi durante il piano di digitalizzazione del Codice Pelavicino, un importante manoscritto medievale che contiene le copie di oltre 500 documenti notarili riguardanti la diocesi di Luni.<sup>1</sup> Il progetto, realizzato sotto la supervisione della professoressa Enrica Salvatori e il professor Roberto Rosselli Del Turco, è stato commissionato dall'Accademia Lunigianese di Scienze "Giovanni Cappellini" della Spezia<sup>2</sup>, con l'intenzione di realizzare un'edizione digitale in grado di restituire importanza al volume e, al tempo stesso, di fornire agli studiosi e agli utenti interessati un innovativo strumento per la consultazione dei testi antichi.

La realizzazione del progetto ha messo in luce le proprietà peculiari della nuova editoria digitale, evidenziando anzitutto il cambio di prospettiva e di approccio ai contenuti nel passaggio dalla pubblicazione cartacea a quella elettronica. A differenza della prima, infatti, il processo di digitalizzazione richiede diverse fasi di lavoro e il coinvolgimento di più figure professionali con specifiche competenze in ambito letterario e informatico.<sup>3</sup> Inoltre, è proprio a partire da queste tipologie di progetti che, nel corso degli anni, si è sviluppata la figura dell'informatico umanista, il quale, grazie alla sua formazione, è in grado di unire fra di loro i diversi strumenti informatici e di utilizzarli per la rielaborazione del sapere umanistico.<sup>4</sup> L'insieme di questi aspetti ha dunque portato alla ridefinizione dei principi base dell'editoria tradizionale, in favore di un approccio più complesso, ma in grado di fornire all'utente una serie di strumenti per un'analisi approfondita dei testi antichi.

---

<sup>1</sup> Ennio Callegari, *Appunti per una storia di Sarzana*, Sarzana, Coop Liguria, 1995.

<sup>2</sup> Sito ufficiale, <http://www.accademiacapellini.it/index.html> (visitato il 25/01/2015).

<sup>3</sup> J.Rydberg-Cox, *Digital libraries and the challenges of digital humanities*, Chandos Publishing, Oxford, 2006.

<sup>4</sup> Anne Burdick, Johanna Drucker, Peter Lunenfeld, Todd Presner, Jeffrey Schnapp, *Digital Humanities*, Cambridge, The Mit Press Cambridge, 2012, pp. 38-39.

L'edizione pensata per il Codice Pelavicino permette di mettere a confronto il testo e le immagini che raffigurano le pagine del manoscritto, fornendo inoltre una serie di strumenti in grado di facilitare e di affinare la ricerca e lo studio dei contenuti. Questa tipologia di pubblicazione digitale viene definita *image-based digital edition* e, data la sua ricchezza, rappresenta una delle forme più complesse di edizioni elettroniche. L'esempio al quale si è fatto riferimento per la realizzazione del Codice Pelavicino Digitale è il progetto del *Digital Vercelli Book* che, per il confronto tra testo e immagine, ha utilizzato un particolare strumento di visualizzazione sviluppato presso l'Università di Pisa: il software EVT (*Edition Visualization Technology*).

Il progetto di digitalizzazione si è svolto in diverse fasi, ognuna delle quali è stata curata da un diverso gruppo di lavoro. La seguente tesi intende focalizzarsi principalmente sulle fasi che, dalla codifica del manoscritto, hanno portato all'edizione digitale vera e propria. Per predisporre i documenti del volume alla visualizzazione in EVT, infatti, è stato necessario, per prima cosa, convertire i testi in un formato processabile dai computer e in grado di arricchire il libro con tutte le informazioni relative alla sua struttura e ai suoi contenuti. L'operazione di codifica per mezzo di un linguaggio di markup rappresenta una delle fasi più significative dell'intero progetto, poiché costituisce il ponte fra il manoscritto nella sua forma originaria e l'edizione digitale vera e propria.

Tutti gli strumenti e le soluzioni grafiche in grado di migliorare la consultazione e lo studio del manoscritto sono stati sviluppati, invece, nelle fasi successive del progetto. Nello specifico, le immagini delle carte sono state ridimensionate e rinominate per permettere al software EVT di applicare i suoi specifici tool, mentre un sistema di correzione, elaborato durante la codifica dei documenti, ha permesso di impostare buona parte delle scelte visive per il trattamento degli elementi più importanti dei testi.

L'ultimo capitolo della tesi si focalizza sui *signa tabellionis*, i simboli attraverso i quali i notai certificavano l'autenticità dei documenti, e sul loro utilizzo all'interno dell'edizione digitale. Data la loro importanza, infatti, è stato deciso di predisporre una pagina HTML, esterna alla visualizzazione in EVT, con la lista

dei notai e dei corrispettivi *signa* presenti nel manoscritto, in modo da fornire all'utente un ulteriore strumento di consultazione e di visualizzazione per i numerosi contenuti rintracciabili tra le pagine dell'antico volume.

## 2 Il Codice Pelavicino

Il Codice Pelavicino è un manoscritto in cartapeccora risalente alla seconda metà del XIII secolo e conservato presso l'Archivio del Capitolo della Cattedrale di Sarzana. Composto da 426 carte e da oltre cinquecento documenti, il volume rappresenta una delle fonti principali riguardo la diocesi di Luni per un periodo di circa trecento anni e, al tempo stesso, può essere considerato un “*esemplificativo delle tracce documentarie lasciate in Lunigiana dal transito degli imperatori.*”<sup>5</sup> Il codice venne confezionato per volontà del vescovo Enrico da Fucecchio (1273-1297), al fine di trascrivere e conservare atti importanti per la chiesa di Luni.<sup>6</sup>

### 2.1 La storia e i contenuti

In seguito all'imprigionamento e all'esilio di Guglielmo vescovo di Luni, che fu catturato nel 1241 mentre si stava recando a Roma per il Concilio Lateranense, il vicario dell'Imperatore, Oberto Pelavicino, ordinò la raccolta dei documenti che avevano riguardato i rapporti tra la diocesi e l'Impero<sup>7</sup> fino alla nomina del suo incarico. Il lavoro venne portato avanti fino al 1247, anno della morte del marchese, e il lungo elenco ricognitivo di beni, diritti, censi in denaro o in natura spettanti al vescovo confluì in quella parte del manoscritto a noi nota con il nome di *Liber Magister*.<sup>8</sup> Non si è purtroppo conservato il testo originario, ma l'idea di un codice che conservasse la copia dell'archivio diocesano spinse il vescovo Enrico da Fucecchio a ordinare la trascrizione non solo del *Liber Magister* (che abbiamo appunto in copia nel Codice Pelavicino), ma anche di tutti i documenti che il prelado riteneva indispensabili a dimostrare i diritti della curia sui territori di cui si erano impossessate varie realtà signorili. Fece inoltre ricopiare e riunire da

---

<sup>5</sup> Enrica Salvatori, *Imperatore e signori nella Lunigiana della prima metà del XIII secolo*, p. 3.

<sup>6</sup> Lupo Gentile, *Il regesto del Codice Pelavicino*, in *Società ligure di storia patria*, XLIV, Genova, 1912, p. IX.

<sup>7</sup> Ennio Callegari, *Appunti per una storia di Sarzana*, Sarzana, Coop Liguria, 1995.

<sup>8</sup> G.Pistarino, *Per una nuova edizione del Codice Pelavicino* cit.



uno scriba anonimo e da maestro Egidio (attivo dal 1287 al 1289) tutti gli antichi privilegi del vescovato lunense in un unico volume.<sup>9</sup>

Il Codice in realtà è composto da quattro unità distinte che sono state rilegate assieme solo in età moderna. La prima parte, composta da 8 carte, è caratterizzata da un lungo *Indice* e da una *Memoria*, un inventario dell'archivio vescovile così come si presentava all'epoca di Enrico da Fucecchio. La seconda parte consiste nel già citato *Liber Magister* di Oberto Pelavicino (in 23 carte) che, come fa notare Luigi Podestà, non è altro che una copia dell'originale.<sup>10</sup> La terza sezione è sicuramente la più consistente ed è costituita dal *Liber Iurium* vero e proprio. Al suo interno sono trascritti più di cinquecento documenti di natura pubblica o privata, distinguibili in varie tipologie: atti giudiziari, carte di procura, di vendita e di locazione, obbligazioni, alienazioni d'immobili, privilegi, concessioni, ecc. La quarta ed ultima parte del manoscritto, originariamente estranea al Codice, è composta da una seconda redazione del *Liber Magister* che, grazie alla scrittura, è databile alla fine del XIV secolo.

Tra i documenti più interessanti del Codice è giusto ricordare i numerosi privilegi e concessioni del Papa e dell'Impero alla curia di Luni. Si possono citare, a questo proposito, il privilegio del 963 di Ottone I dato alla Rocca di S. Leo, quello del 1183 con il quale Federico I concede al vescovo Pietro il contado lunense, la ripa e il pedaggio dei porti di Luni e di Avenza e quello del 1297 (l'ultimo) che ha per oggetto una sentenza di Guidotto, cappellano del papa, provocata da violenze e danni recati dai Sarzanesi al vescovo Antonio da Camilla.<sup>11</sup> Il primo documento del manoscritto, invece, è databile al 900 e si tratta di un privilegio del re d'Italia Berengario che conferma al vescovo Odelberto i privilegi concessi alla Chiesa lunense dai suoi predecessori.

---

<sup>9</sup> Lupo Gentile, *Il regesto del Codice Pelavicino*, p. IX.

<sup>10</sup> Luigi Podestà, *I vescovi di Luni*, Modena, Tip. Vincenti, 1895.

<sup>11</sup> Lupo Gentile, *Il regesto del Codice Pelavicino*, n. 17, n. 18, n. 369, n. 21, n. 19.

## 2.2 Le caratteristiche

Il Codice Pelavicino è un volume pergameneo rilegato con pannello di legno<sup>12</sup> ed è stato compilato da quattro mani diverse. La lingua utilizzata è il latino, anche se si tratta di una forma tarda, ricca di abbreviazioni e di errori dovuti alle trascrizioni degli atti notarili da parte degli scribi.

Nella scrittura del volume si nota l'intervento di diversi amanuensi. Il primo, di cui si ignora il nome, ha redatto la parte iniziale del *Liber Iurium*, dopodiché il lavoro fu ripreso, per volere di Enrico da Fucecchio, da uno scriba di origini francesi, il maestro Egidio di Bligny sur Ouche, che grazie allo stile tipico delle *littere scholasticae* elaborate in ambiente universitario seppe riprodurre efficacemente i monogrammi e i timbri notarili presenti nei testi originali. Diversa, invece, la scrittura che caratterizza i primi ventidue documenti del *Liber Iurium*: lo stile rotondeggiante, riconducibile al tipo della rotunda italiana, si discosta dal tratto spezzato caratteristico della forma gotica. Si possono inoltre osservare, a margine di alcuni documenti, una certa quantità di didascalie ascrivibili al sec. XVI e XVII.<sup>13</sup>

L'accuratezza nella scrittura e nella riproduzione dei *signa tabellionis*, nella seconda metà del manoscritto, viene sempre meno, così come l'utilizzo dell'inchiostro rosso a scopo ornamentale (usato, nella prima parte, per le lettere iniziali, per il corpo delle maiuscole e per la legatura). In seguito all'azione deleteria dell'umidità e all'uso frequente del Codice, alcuni caratteri risultano ad oggi intellegibili.

I *signa tabellionis*, ovvero i disegni che i notai apponevano prima della loro sottoscrizione (a garanzia di autenticità)<sup>14</sup>, sono stati riprodotti con grande cura nelle prime trascrizioni del maestro Egidio e, in alcuni casi, sono stati ricopiati

---

<sup>12</sup> Lupo Gentile, *Il regesto del Codice Pelavicino*, p. VII.

<sup>13</sup> Lupo Gentile, *Il regesto del Codice Pelavicino*, p. X.

<sup>14</sup> Enciclopedia Treccani, voce *Tabellone*, [http://www.treccani.it/enciclopedia/tabellone\\_\(Dizionario\\_di\\_Storia\)/](http://www.treccani.it/enciclopedia/tabellone_(Dizionario_di_Storia)/) (visitato il 29/12/2014).

anche nella parte iniziale dei documenti, non sappiamo se riprendendo l'originale o a scopo puramente decorativo.

## 2.3 Le edizioni

Nonostante il Codice Pelavicino abbia fornito abbondante materiale per gli studi sulla Lunigiana tra i secoli X e XIII, sono pochi gli studiosi che, nel corso degli anni, si sono dedicati all'indagine del volume e a un'attenta descrizione sotto l'aspetto paleografico e librario.<sup>15</sup>

Nel 1914, lo storico Michele Lupo Gentile (1880 - 1959) pubblicò *Il Regesto del Codice Pelavicino* che, tutt'oggi, rappresenta la fonte più ricca e approfondita dell'intero manoscritto. Il volume contiene la trascrizione di buona parte dei documenti, nonostante si tratti spesso e volentieri di testi non integrali, ed è stato arricchito grazie alla presenza dei numerosi regesti: riassunti, più o meno ampi, del contenuto di ogni singolo documento.<sup>16</sup>

In seguito alla pubblicazione di Lupo Gentile, altri studiosi si sono interessati al manoscritto, scrivendo diversi articoli. Tra questi, le *Correzioni critiche di alcune date nel Regesto del Codice Pelavicino per il socio U. Mazzini*<sup>17</sup>, pubblicate dallo stesso Gentile, sono state integrate come appendice all'opera del 1914.

---

<sup>15</sup> Pistarino, Geo, 1942. *Un crittogramma nel Codice Pelavicino dell'Archivio Capitolare di Sarzana*. In "Giornale Storico e Letterario della Liguria", XVIII-3/4, pp. 186–187.

<sup>16</sup> Enciclopedia Treccani, voce *Regesto*, <http://www.treccani.it/enciclopedia/regesto/> (visitato il 30/12/2014).

<sup>17</sup> U. Mazzini, *Correzioni critiche di alcune date del Regesto del Codice Pelavicino*, Società ligure di storia patria, XLIV, Genova, 1914.

## 3 La codifica del manoscritto

### 3.1 Introduzione all'edizione digitale

Con la speranza di restituire importanza a una delle principali fonti di studio sulla Lunigiana medievale e ottenere un maggiore riscontro rispetto a una pubblicazione cartacea, è stato deciso di realizzare un'edizione digitale del Codice Pelavicino, sfruttando i nuovi metodi informatici per la digitalizzazione dei testi antichi. Tale progetto verrà distribuito sul web per la libera consultazione e, grazie ad una serie di strumenti per l'indagine e l'approfondimento dei contenuti, darà la possibilità a qualsiasi tipologia di utente di approcciarsi al manoscritto in maniera dinamica e innovativa.

Prima di passare alle diverse fasi di lavoro e, nello specifico, alle questioni portate avanti da noi studenti, è però necessario soffermarsi sul concetto di edizione digitale, in modo da poterne inquadrare le caratteristiche e le possibili realizzazioni.

#### 3.1.1 L'edizione digitale e l'Image-based digital edition

L'idea di una digitalizzazione massiccia dei manoscritti antichi, con il duplice scopo di tutelare i supporti originali e fornire l'accesso a documenti non altrimenti consultabili, costituisce, a oggi, una priorità ineludibile<sup>18</sup>. La necessità di un tale progetto ha portato alla realizzazione delle cosiddette edizioni digitali: pubblicazioni in formato elettronico in grado di fornire una serie di funzioni esclusive per l'analisi e la ricerca all'interno del testo. Questi prodotti sono il frutto di una particolare interpretazione dell'informazione testuale<sup>19</sup> e, per questo

---

<sup>18</sup> Degl'Innocenti, Emiliano. 2007. *Il Progetto di digitalizzazione dei Plutei della Biblioteca Medicea Laurenziana di Firenze*. In *Digitalia. Rivista digitale dei beni culturali*, volume 1, pp. 103-114;

<sup>19</sup> Buzzetti, Dino, 2009. *Digital Edition and Text Processing*. In *Text Editing, Print and Digital World*, Farnham, Ashgate, p. 45.

motivo, non possono essere considerati come una semplice conversione di formato.

La realizzazione di un'edizione digitale richiede diverse fasi di lavorazione e, perciò, si serve di numerose figure professionali.<sup>20</sup> Nonostante un progetto del genere possa sembrare il frutto di un'operazione complessa e particolarmente costosa, in realtà rappresenta una valida alternativa ad una pubblicazione tradizionale e, come vedremo, offre una serie di soluzioni vantaggiose per la consultazione dei testi antichi.

Nel caso specifico del Codice Pelavicino, si è deciso di basarsi su un modello particolare di edizione digitale: l'*image-based digital edition* (Kevin S. Kiernan). In questo caso, ogni pagina del testo viene collegata alla sua rispettiva immagine, in modo da fornire anche la consultazione del volume nella sua forma originaria. Naturalmente, tutte le potenzialità e gli strumenti delle edizioni digitali tradizionali possono essere a loro volta integrati per garantire l'accesso a un prodotto che possa superare i limiti intrinseci di una pubblicazione cartacea<sup>21</sup>.

Le *image-based digital edition* hanno iniziato ad affermarsi come importanti risorse per gli studi umanistici solamente negli ultimi anni<sup>22</sup> e, per questo motivo, sono ancora poco diffuse. A questo proposito, si possono citare il progetto dell'*Electronic Beowulf*<sup>23</sup>, composto da una vasta selezione di immagini del manoscritto originale<sup>24</sup>, e il *Digital Vercelli Book*, sviluppato con il software EVT su iniziativa di Roberto Rosselli del Turco, ricercatore di filologia Germanica

---

<sup>20</sup> Peter L. Shillingsburg, *From Gutenberg to Google*, Cambridge, Cambridge University Press, pp. 94-95.

<sup>21</sup> Kevin Kiernan, *Electronic Textual Editing: Digital Facsimiles in Editing*, [http://www.tei-c.org/About/Archive\\_new/ETE/Preview/kiernan.xml](http://www.tei-c.org/About/Archive_new/ETE/Preview/kiernan.xml) (visitato il 09/01/2015).

<sup>22</sup> Dekhtyar, Alex, Ionut E. Iacob, Jerzy W. Jaromczyk, Kevin Kiernan, Neil Moore, Dorothy Carr Porter, 2006. *Support for XML markup of image-based electronic editions*. In *International Journal on Digital Libraries*, Volume 6, pp. 55-69.

<sup>23</sup> <http://ebeowulf.uky.edu> (visitato il 09/01/2015).

<sup>24</sup> Per ulteriori informazioni: Prescott, Andrew, 1997. *The Electronic Beowulf and Digital Restoration*. In *"Literary and Linguistic Computing"*, n.12, 185-95.

presso l'Università di Torino e docente di Codifica di testi presso l'Università di Pisa.

L'edizione digitale del Codice Pelavicino è stata sviluppata proprio sulla base del software EVT che, per l'occasione, è stato ampliato e modificato per potersi adattare alla particolare composizione del manoscritto. Le funzionalità e le caratteristiche dell'edizione elettronica verranno discusse in maniera approfondita nel quarto capitolo.

### **3.1.2 Le fasi e il team di lavoro**

Come già specificato, la digitalizzazione dell'informazione e, in particolare, quella dei manoscritti antichi, richiede l'intervento di più figure professionali. Affidare i diversi compiti a una sola persona, infatti, comporterebbe allo squilibrio in una delle fasi di lavoro e, quindi, alla perdita di tutta la complessità e la particolarità di tali progetti. Anche per questo motivo, negli ultimi anni sono nate una serie di figure interdisciplinari, come l'informatico umanista, in grado di gestire in maniera assai più efficace la varietà dei contenuti e degli strumenti necessari per la realizzazione di queste edizioni.<sup>25</sup>

Il progetto di digitalizzazione del Codice Pelavicino è iniziato con l'acquisizione delle immagini del manoscritto e con la trascrizione dei singoli documenti, di cui si sono occupati Enrica Salvatori, Laura Balletto e Edilio Riccardini. Parallelamente alla trascrizione è stata realizzata l'edizione critica, grazie alla quale sono state inserite una serie di informazioni fondamentali (sotto forma di note o di metadati in fronte ai documenti) per la consultazione e l'analisi dei singoli testi.

Nella seconda fase del progetto, il team di studenti si è occupato della codifica del manoscritto, sotto la supervisione di Roberto Rosselli del Turco ed Enrica Salvatori. Si tratta della fase intermedia, fondamentale per estrarre le informazioni dal testo e predisporre i documenti del volume alla visualizzazione e alla

---

<sup>25</sup> Blanke, Tobias, Elena Pierazzo, Peter A. Stokes, 2014. *Digital Publishing Seen from the Digital Humanities*. in *Logos*, vol 25, no. 2, pp. 16–27.

consultazione. La codifica si articola in più fasi e presuppone l'utilizzo di diversi strumenti informatici, nonché una serie di conoscenze sulle caratteristiche e sui contenuti dei documenti. Di questa fase se ne discuterà in maniera più approfondita nelle pagine successive.

Della trasformazione del materiale codificato nel formato per la visualizzazione se n'è occupato il team di sviluppo di EVT, composto da Roberto Rosselli del Turco, Chiara Di Pietro, Julia Kenny e Raffaele Masotti.

L'ultima fase del progetto, ovvero la realizzazione del sito web, fondamentale per introdurre l'edizione digitale e per informare su eventuali aggiornamenti o modifiche, è stata assegnata a Chiara Mannari.

L'intero progetto è stato supervisionato da Enrica Salvatori, professore associato di Storia medievale presso l'Università di Pisa.

### **3.2 La codifica di testi**

Un testo è una struttura complessa in grado di veicolare informazione e di articolarla su più livelli: sequenze di caratteri che si combinando per formare le parole, frasi, strutture linguistiche astrarre raggruppate in unità con funzioni specifiche (titoli, capitoli, paragrafi, ecc.). La maggior parte di questi contenuti sono forniti in maniera implicita e soltanto l'essere umano è in grado di estrapolarli in maniera rapida e con apparente assenza di sforzo.<sup>26</sup> Il computer, invece, memorizzando ed elaborando i dati sotto forma di codice binario, avrà bisogno di ricevere un testo 'trasformato', in un'estensione *Machine Readable Form* (MRF). La trasformazione del 'messaggio' in un formato processabile dal calcolatore viene definita *codifica*.

La rappresentazione di ciascun carattere alfanumerico nella forma del codice binario, ovvero in una sequenza di due soli simboli 0 e 1 associati in sequenze da

---

<sup>26</sup> A. Lenci, S. Montemagni, V. Pirrelli, *Testo e computer. Elementi di linguistica computazionale*, Roma, Carocci, 2005, p. 53.

8 cifre (*byte*), prende il nome di codifica di livello zero. La codifica di alto livello, invece, si occupa di arricchire il testo con informazioni riguardanti la struttura linguistica e testuale, con l'intenzione di rendere esplicito tutto ciò che è congetturale o implicito.<sup>27</sup> L'operazione di marcatura del testo è resa possibile dai linguaggi formali (*markup languages*) che, attraverso una serie di regole, sono in grado di descrivere i meccanismi di rappresentazione dei diversi contenuti e, al tempo stesso, di renderli decifrabili per gli elaboratori elettronici.

La codifica (con riferimento particolare a quella di alto livello) non può essere ridotta, dunque, alla semplice scansione del documento (es. in formato PDF) o all'utilizzo di software OCR per ricavarne una versione in formato ASCII (la forma più diffusa per la rappresentazione dei caratteri alfanumerici mediante codice binario<sup>28</sup>). L'operazione di marcatura, infatti, estrapolando una quantità di informazioni implicite non ricavabili altrimenti, si pone ad un livello di trattamento testuale assai più profondo.

I linguaggi di marcatura sono molteplici e possono essere utilizzati per ottenere risultati differenti, secondo le specifiche esigenze. Nonostante ciò, Michael Sperberg-McQueen ha identificato una serie di 'assiomi' che ogni *markup* dovrebbe essere in grado di rispettare per essere definito tale. Innanzitutto, il linguaggio deve riflettere un'interpretazione del testo ed essere in grado di esprimerla. In secondo luogo, deve tener conto dell'estensibilità, in riferimento alle diverse caratteristiche testuali, e della capacità di descrivere l'organizzazione linguistica. È necessario, infine, che i testi siano rispettati nella loro struttura, lineare e gerarchica, tenendo conto anche della loro valenza storica e culturale.<sup>29</sup>

---

<sup>27</sup> A. Lenci, S. Montemagni, V. Pirrelli, *Testo e computer. Elementi di linguistica computazionale*, cit.

<sup>28</sup> Enciclopedia Treccani, voce *ASCII*, <http://www.treccani.it/vocabolario/ascii/> (visitato il 30/12/2014).

<sup>29</sup> C. M. Sperberg-McQueen, *Text in the Electronic Age: Textual Study and Text Encoding, with Examples from Medieval Text*, 1991.



### 3.2.1 XML e gli schemi di codifica

XML (*eXtensible Markup Language*) è uno dei più importanti metalinguaggi di markup utilizzati per la codifica dei testi. È flessibile, indipendente da ogni sistema operativo e permette di creare, archiviare e diffondere diverse tipologie di documenti digitali. Inoltre, può essere utilizzato per creare linguaggi aggiuntivi.<sup>30</sup>

XML, la cui prima versione fu rilasciata nel febbraio del 1998<sup>31</sup>, ambisce a sostituire la complessità di SGML (*Standard Generalized Markup Language*), uno dei primi linguaggi di markup mai sviluppati, provando a superare anche i limiti intrinseci di HTML. Come qualsiasi altro linguaggio di codifica, si serve di una serie di elementi per delimitare le diverse porzioni di testo e per estrapolare l'informazione. È definito *estensibile*, in linea con uno dei principi fondamentali di Sperberg-McQueen, poiché qualunque utente ha la possibilità di creare elementi personalizzati in base alle proprie esigenze.<sup>32</sup>

Un documento XML può essere aperto con un qualsiasi editor di testo e, per essere definito 'ben formato', deve rispettare tutte le caratteristiche della sintassi interna. Inoltre, il file può essere validato in base ad uno *schema di codifica* di riferimento. L'operazione di validazione è ritenuta indispensabile nel caso si abbia a che fare con documenti complessi, quali ad esempio un testo letterario o un manoscritto.

#### Schema di codifica

Prima di passare alla marcatura vera e propria, può essere utile definire un insieme di regole restrittive in modo da poter delimitare la codifica solo ed esclusivamente ai contenuti e alle informazioni ricercate.<sup>33</sup> La dichiarazione degli elementi

---

<sup>30</sup> Steven Holzner, *Real World XML*, Peachpit Press, 2003.

<sup>31</sup> W3C Press Release, *The World Wide Web Consortium Issues XML 1.0 as a W3C Recommendation*, <http://www.w3.org/Press/1998/XML10-REC> (visitato il 31/12/2014).

<sup>32</sup> Cunningham, Lawrence A., 2006. *Language, Deals and Standards: The Future of XML Contracts*. In *Research Paper*, N. 93, p. 6.

<sup>33</sup> Text Encoding Initiative, *A Gentle Introduction to XML*, <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SG.html> (visitato il 31/12/2014).

permessi, dei dati a loro associati e delle relazioni gerarchiche, espressi sotto forma di un linguaggio informatico, prende il nome di schema di codifica. Un documento di markup, nello specifico un file XML, può essere considerato valido solo ed esclusivamente se è ben formato e se rispetta tutte le norme che sono state impostate nello schema, dalla corrispondenza tra i nomi degli elementi e degli attributi alla verifica delle regole relative alla sintassi.

Tra i principali vantaggi dei linguaggi schema vi è l'estensibilità, fondamentale per apportare modifiche allo schema durante la codifica, e la possibilità di determinare la validità di un documento tramite uno specifico programma, spesso integrato nell'editor di testo.<sup>34</sup>

I linguaggi scritti con la stessa sintassi di XML, nonostante la verbosità, sono preferibili, poiché maggiormente espressivi e utilizzabili con diverse applicazioni.<sup>35</sup> In precedenza, per la validazione, si faceva riferimento alla DTD (Document Type Definition), un linguaggio molto compatto, ma difficilmente intellegibile e scritto con una diversa sintassi.

L'XML Schema, come ad esempio Relax NG (Regular Language for XML Next Generation), si serve di una collezione astratta di metadati, consistenti in un set composto, principalmente, da dichiarazioni di attributo e di elemento. Può essere utilizzato anche per generare documenti di consultazione.

Lo schema è così dichiarato all'interno dell'intestazione di un file XML:<sup>36</sup>

```
<?xml-model href="SchemaRelaxNG.rng" type="application/xml"
schematypens="http://relaxng.org/ns/structure/1.0"?>
```

---

<sup>34</sup> Murata, Makoto, Dongwon Lee, Murali Mani, Kohsuke Kawaguchi, 2004. *Taxonomy of XML Schema Languages using Formal Language Theory*. In *ACM Journal Name*, Vol. V, No. N, pp. 1-3.

<sup>35</sup> Lee, Dongwon, Wesley W. Chu, 2000. *Comparative Analysis of Six XML Schema Languages*. In *ACM SIGMOND Record*, Los Angeles.

<sup>36</sup> Eric van der Vlist, *Relax NG*, O'Reilly Media, 2003.

### 3.2.2 La TEI<sup>37</sup>

La TEI (*Text Encoding Initiative*) è un consorzio che ha sviluppato, in ambito linguistico e letterario, uno standard per la rappresentazione dei testi in formato digitale. La missione della TEI è quella di sviluppare una serie di linee guida per la codifica dei testi e per sostenere e condividere il loro uso da parte di diverse categorie, più o meno ampie, di utenti. Attraverso un testo di riferimento, le *Guidelines for Electronic Text Encoding and Intherchange*<sup>38</sup>, il consorzio definisce un linguaggio di markup (in XML) che comprende un numero elevato di elementi, di tipo strutturale e semantico, definiti per mezzo di un certo numero di schemi di codifica, organizzati in una struttura modulabile e personalizzabile. Grazie a queste caratteristiche, ognuno studioso ha la possibilità di sviluppare il proprio progetto partendo da una serie di linee guida, scegliendo i moduli necessari e modificando, in qualsiasi momento, le definizioni degli elementi.

La necessità di realizzare uno standard che permettesse la creazione e l'interscambio di documenti digitali venne discussa, per la prima volta, durante un convegno tenutosi al Vassar College di Poughkeepsie a New York, nel 1987. In seguito a questo incontro furono redatti i cosiddetti *Poughkeepsie Principles*, ossia i principi base ai quali doversi attenere per la realizzazione di uno schema di codifica dei testi per la ricerca umanistica.<sup>39</sup> Il TEI Consortium, l'associazione non profit per lo sviluppo dello standard TEI, fu fondato nel 2000.

L'attuale versione delle linee guida (TEI P5) è stata pubblicata alla fine del 2007 ed è basata su XML e su schemi di codifica Relax NG e DTD tradizionale. Tra le numerose e importanti novità è stato inserito un modulo per la descrizione dei manoscritti, che è stato ampiamente sfruttato per la codifica del Codice Pelavicino.

---

<sup>37</sup> Sito ufficiale, <http://www.tei-c.org/index.xml> (visitato il 02/01/2015).

<sup>38</sup> Disponibili in diversi formati a questo indirizzo: <http://www.tei-c.org/Guidelines/P5/> (visitato il 02/01/2015).

<sup>39</sup> Per ulteriori informazioni: <http://www.tei-c.org/Vault/ED/edp01.htm#b2b1b3b3b3> (visitato il 02/01/2015).

La struttura modulare delle *Guidelines* permette di scegliere soltanto i moduli corrispondenti alle proprie esigenze, in modo da realizzare rapidamente uno schema appropriato. Ogni modulo contiene un certo numero di elementi, organizzati in classi strutturali o semantiche. Gli attributi, invece, sono così distinti: globali, se disponibili per tutti gli elementi, o specifici, se circoscritti ad un numero limitato di elementi. Lo schema di codifica, disponibile nei formati DTD, W3C e Relax NG, può essere creato utilizzando TEI Roma<sup>40</sup>, uno script disponibile sul web o come shell script su Linux.

### **3.3 La codifica del Codice Pelavicino**

Per l'edizione digitale del Codice Pelavicino si è scelto di far riferimento allo standard TEI, utilizzando i moduli e gli elementi predisposti dalle *Guidelines*. Data la ricchezza e la varietà del manoscritto, per la prima fase della codifica si è stabilito di partire da una selezione di 145 documenti tratti dal *Liber Iurium*. In seguito alla messa online del progetto, si potrà passare ad una seconda fase di codifica.

Prima della marcatura, è stato necessario stabilire i diversi strumenti di lavoro (editor, piattaforma di condivisione, ecc.) e predisporre uno schema di codifica, in modo da poter restringere il campo alle informazioni necessarie e, dunque, a un numero limitato di moduli e di elementi.

#### **3.3.1 Gli strumenti**

I testi del manoscritto sono stati studiati, numerati e trascritti al computer dagli storici che, per ogni documento, hanno creato un file Word e l'hanno arricchito con tutte le informazioni necessarie per la marcatura dei contenuti.

Per facilitare la codifica e per rendere esplicite le informazioni da estrarre, sono stati evidenziati con diversi colori tutti gli elementi interessati (v. fig. 1) e, per

---

<sup>40</sup> <http://www.tei-c.org/Roma/> (visitato il 02/01/2015).

evitare casi di ambiguità e inutili ripetizioni, sono stati aggiunti dei commenti al bordo del testo.

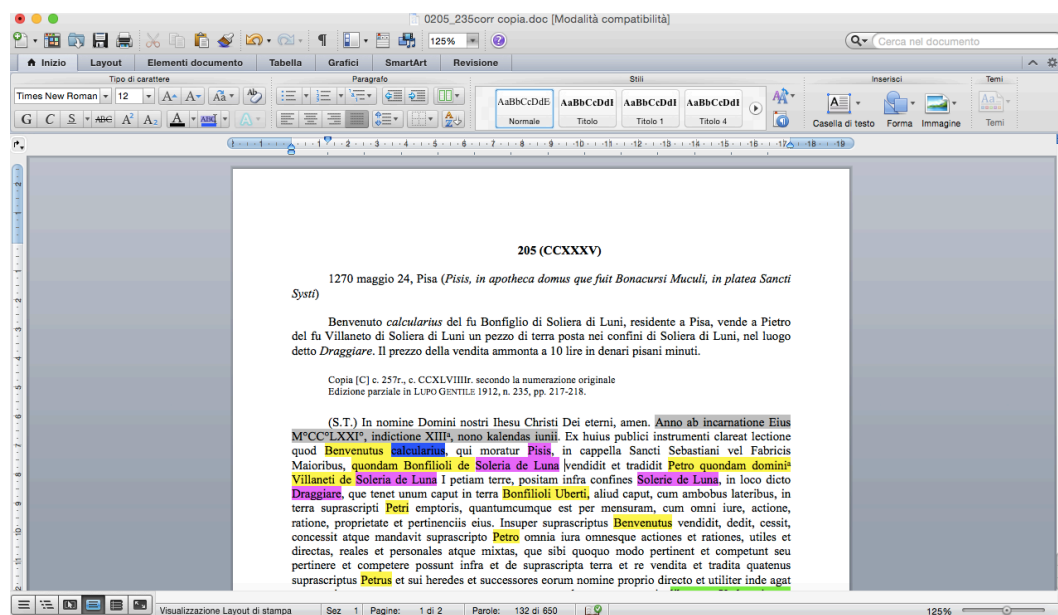


Figura 1. Esempio di documento nel formato Word

In secondo luogo, è stato fondamentale stabilire una piattaforma collaborativa dove poter inserire tutti i file e i documenti necessari alla codifica. La condivisione dei contenuti, infatti, rappresenta uno dei momenti più importanti dell'intera realizzazione del progetto, poiché dà la possibilità a ogni membro del team di lavorare in remoto mantenendo il materiale sempre a disposizione degli altri<sup>41</sup>, permettendo così un controllo costante della codifica. Per l'edizione digitale del Codice Pelavicino, sono state utilizzate due piattaforme: Google Drive, per la condivisione degli appunti, del modello di codifica e del manuale, e Dropbox, per i documenti del manoscritto e per i diversi file XML.

Nelle ultime fasi del progetto è stato utilizzato anche un repository GIT<sup>42</sup>, al quale alcuni dei ricercatori accedevano tramite il client SourceTree (v. fig. 2), che ha permesso non solo la condivisione dei documenti XML, ma anche il confronto e la correzione immediata dei conflitti dei diversi file.

<sup>41</sup> Pitti, Daniel V., 2004. *Designing Sustainable Projects and Publications*. In *A Companion to Digital Humanities*, Oxford, Blackwell.

<sup>42</sup> Per ulteriori informazioni: <http://git-scm.com/about> (visitato il 05/01/2015).

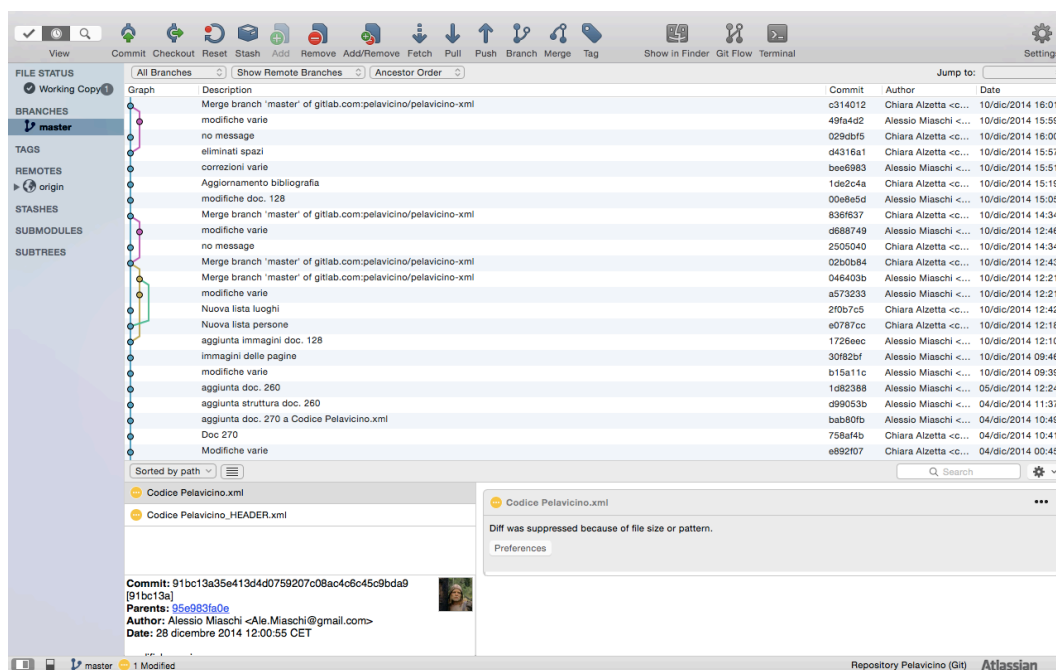


Figura 2. Esempio di schermata in SourceTree

Nonostante i numerosi editor di testo freeware, per la codifica del Codice Pelavicino si è preferito far affidamento a un'applicazione più avanzata. Per questo motivo è stato scelto <oxygen/><sup>43</sup>, un software proprietario che predispone un sistema per le correzioni automatiche e per le segnalazioni d'errore, in modo da facilitare l'annotazione del testo.

### 3.3.2 Lo schema di codifica

Una delle fasi preliminari più importanti consiste, naturalmente, nell'impostazione dello schema di codifica. Per prima cosa, gli studiosi che hanno lavorato alla trascrizione dei documenti hanno stabilito tutti gli elementi e le informazioni necessarie da estrarre dal manoscritto e si sono occupati di illustrare la struttura e le caratteristiche dei testi al team di codifica. Solo dopo una serie di verifiche e di indicazioni preliminari è stato possibile passare alla predisposizione dello schema vero e proprio, in modo da permettere una validazione costante durante il markup del testo.

<sup>43</sup> Sito ufficiale, <http://www.oxygenxml.com> (visitato il 3/01/2015).

La realizzazione di uno schema XML personalizzato, in casi come quello della codifica di manoscritti antichi, è assai preferibile, data la peculiarità dei contenuti non facilmente adattabili a degli standard. Nonostante ciò, per l'edizione del Codice Pelavicino si è deciso di far riferimento ai moduli e agli elementi predisposti dalle *Guidelines* TEI, ridefinendo l'uso di alcuni tag per poterli adeguare alle esigenze.

Tramite lo script TEI Roma è stato possibile realizzare uno schema di codifica completo di tutti i moduli principali (*tei*, *header*, *textstructure* e *core*), fondamentali per la descrizione di un qualsiasi documento TEI, e in più sono stati aggiunti *namesdates*, *figure*, *transcr* e *msdescription*<sup>44</sup>, che rispondono alle specifiche caratteristiche del progetto.

### 3.3.3 La codifica

I 145 documenti del manoscritto sono stati codificati all'interno di un unico file XML con nodo radice <TEI> e suddiviso, come da standard<sup>45</sup>, in due macro elementi: <teiHeader> e <text>. Solo in un secondo momento è stato aggiunto, allo stesso livello gerarchico, l'elemento <facsimile>, necessario per la realizzazione dei collegamenti fra contenuti specifici del testo e immagine.

Il <teiHeader> corrisponde all'intestazione dell'intero manoscritto, all'interno della quale sono stati inseriti tutti i metadati relativi alla provenienza, alla pubblicazione e alla descrizione sintetica del volume. Il <text> è il corpo del file XML e contiene i testi, così come sono stati predisposti nei documenti Word, e la bibliografia di riferimento. L'unico elemento a non contenere testo è il <facsimile>. Al suo interno, infatti, si trovano solo ed esclusivamente le pagine del Codice e i riferimenti ai numerosi *signa tabellionis*.

La struttura del file XML si presenta dunque così:

---

<sup>44</sup> Per ulteriori informazioni: TEI: P5 Guidelines, *Dates, People and Places* (Cap. 13), *Tables, Formulae, Graphics and Notated Music* (Cap. 14), *Representation of Primary Sources* (Cap. 11), *Manuscript Description* (Cap. 10), <http://www.tei-c.org/Guidelines/P5/> (visitato il 03/02/2015).

<sup>45</sup> Lou Burnard, *What is the Text Encoding Initiative?*, Marsiglia, OpenEdition Press, 2014.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE TEI SYSTEM "schema/tei_cp.dtd">
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>
    ...
  </teiHeader>
  <facsimile>
    ...
  </facsimile>
  <text>
    ...
  </text>
</TEI>

```

Le prime due righe del file XML sono occupate dalla *processing instruction*, l'istruzione che attesta il linguaggio utilizzato e la codifica dei caratteri, e dalla dichiarazione dello schema di codifica.

## Il <teiHeader>

Il <teiHeader>, come è già stato espresso in precedenza, racchiude una serie di informazioni rappresentative del manoscritto che devono essere collocate allo stesso livello gerarchico di <text>, ossia del contenuto dei testi. Tutti i metadati necessari per fornire una descrizione accurata del volume sono raggruppati all'interno di un altro elemento, il <fileDesc> che, riprendendo la definizione sul sito della TEI, “*contiene una descrizione bibliografica completa di un file elettronico*”.<sup>46</sup>

I tre elementi principali contenuti nel <fileDesc> sono il <titleStmt>, il <publicationStmt> e il <sourceDesc>, e sono così rappresentati nel documento:

```

<teiHeader>
  <fileDesc>
    <titleStmt>
      ...
    </titleStmt>

```

---

<sup>46</sup> Sito ufficiale, <http://www.tei-c.org/release/doc/tei-p5-doc/it/html/ref-fileDesc.html> (visitato il 04/01/2015), cit.



```

        <publicationStmt>
            ...
        </publicationStmt>
        <sourceDesc>
            ...
        </sourceDesc>
    </fileDesc>
</teiHeader>

```

Il **<titleStmt>** racchiude, tramite gli elementi **<title>** e **<respStmt>**, tutte le informazioni relative al titolo dell'opera e ai responsabili dell'edizione critica, della codifica e della revisione, mentre il **<publicationStmt>** raggruppa i dati riguardo la pubblicazione e la distribuzione del documento elettronico.

Per le informazioni concernenti le caratteristiche specifiche del manoscritto, è stato utilizzato l'**<msDesc>**, uno degli elementi principali del modulo *msdescription*, inserito all'interno del **<sourceDesc>**. L'**<msDesc>** si compone, a sua volta, di quattro unità descrittive: **<msIdentifier>**, **<msContents>**, **<physDesc>** e **<history>**. Il primo, oltre a specificare il nome con il quale è identificato il Codice, raccoglie una serie di dati sulla sua collocazione fisica; il secondo specifica la lingua del testo, in questo caso il latino, e predispone un campo con un breve riassunto informativo sugli autori; **<physDesc>** e **<history>**, invece, attraverso una serie di elementi interni, forniscono tutte le informazioni sulla composizione fisica (dal materiale alla descrizione delle decorazioni) e sulla storia del volume.

Infine, sempre all'interno del **<sourceDesc>**, sono state inserite anche delle liste, fondamentali per la realizzazione degli indici di ricerca e di consultazione dell'edizione digitale.

## Le liste

Di alcuni elementi marcati nel testo, in particolare i nomi di persona e di luogo, si è deciso di creare delle liste per poterli normalizzare e per creare degli indici, essenziali per la consultazione dinamica e per la ricerca dei contenuti all'interno del manoscritto. Attraverso la predisposizione degli elenchi, infatti, è stato possibile associare una serie di metadati relativi a ogni persona o luogo presente

nel testo, creando così una corrispondenza in grado di arricchire l'edizione digitale e, al tempo stesso, di permettere una navigazione completa e approfondita.

La lista delle persone, racchiusa all'interno di un **<listPerson>**, è composta da tanti elementi **<person>** quante sono le persone menzionate nel Codice. A sua volta, ognuno di questi tag comprende una serie di campi descrittivi per la codifica dei metadati. Per distinguere i nomi e per creare la corrispondenza tra le liste e le numerose occorrenze nel testo, ogni **<person>** è dotato di un attributo identificativo, l'*@xml:id*.

Esempio:

```
<person xml:id="Aiutuspres">
  <persName>
    <forename>Adiutus / Aiutus</forename>
    <surname>de valle Ylioli</surname>
  </persName>
  <sex>M</sex>
  <occupation n="1">presbiter</occupation>
</person>
```

La lista dei luoghi è stata impostata in maniera analoga. All'interno di un unico **<listPlace>** sono stati inseriti numerosi **<place>**, distinguibili dai diversi *@xml:id*, che contengono tutti i metadati necessari per l'identificazione e la descrizione delle singole località.

Esempio:

```
<place xml:id="Carrara">
  <settlement type="corte">Carraria</settlement>
  <placeName type="new">Carrara</placeName>
  <district type="comune">Aulla</district>
</place>
```

Data la ricchezza delle informazioni e degli elementi presenti nel testo, ogni lista, prima di essere inserita all'interno del documento XML, è stata compilata e aggiornata in un file Microsoft Excel (v. fig. 3), in modo da facilitare la codifica e l'organizzazione dei metadati.

	A	B	C	D	E
450	CalderonusCastronovo	Calderonus	de Castronovo	M	
451	Cancellerius	Cancellerius		M	
452	CancellinusFalcinello	Cancellinus	de Falcinello	M	
453	Cancius	Cancius		M	
454	CanciusPaganni	Cancius / Cançus	quondam Paganni / de Castronovo quondam Pagani	M	
455	CantillinaFal	Cantillina		F	domina de Falcinello
456	Çapironellus	Zapironellus	qui prenomatur Pirolus, quondam Actocanini de Masio	M	
457	Caracosa	Caracosa		F	
458	Carandus	Carandus		M	
459	Carbo	Carbo		M	
460	CarboBottacciFal	Carbo	de Falcinello /quondam Bottaccii	M	dominus de Falcinello
461	Carbomass	Carbo		M	massarius
462	Carbuncellus	Carbuncellus		M	
463	CarusAmicusnot	Carus Amicus		M	notarius sacri palacii
464	Casolinus	Casolinus		M	
465	Castagnola	Castagnola		F	
466	CastagnolusTalamelli	Castagnolus	de Summo Tranci quondam Talamelli	M	
467	CastagnolaCastronovo	Castagnola	de Castronovo	F	
468	CastellanaFancelli	Castellana	filia quondam Fancelli	F	

Figura 3. Una sezione della lista dei nomi come appare su Microsoft Excel

Attraverso il Mapping XML<sup>47</sup>, una specifica funzione disponibile nella scheda Sviluppo di Excel, è stato possibile esportare i contenuti della tabella direttamente in formato XML, rispettando così la struttura originaria delle liste. Per prima cosa, è necessario generare un file XML con un modello vuoto della lista di riferimento e associarlo, tramite la funzione *mapping*, alla tabella Excel. A questo punto basterà trascinare la struttura XML sulle celle per completare l'operazione e, quindi, generare il documento. Naturalmente, perché la mappatura possa funzionare, è indispensabile che i due file abbiano la stessa struttura.

All'interno del `<sourceDesc>` è stata predisposta anche la lista dei *signa tabellionis*, grazie alla quale è stato possibile collegare i singoli elementi dei timbri nel testo ad una struttura più approfondita, con link alle immagini e riferimenti ai rispettivi notai.

## Il `<text>`

I documenti del Codice sono stati codificati all'interno del `<text>`, il tag che racchiude tutto il corpo del file XML. Trattandosi di un manoscritto composto da più testi, è stato però necessario distinguere ogni singolo documento con

<sup>47</sup> Per ulteriori informazioni: <http://office.microsoft.com/it-it/excel-help/eseguire-e-annullare-il-mapping-di-elementi-xml-HP010206397.aspx> (visitato il 5/01/2015).

altrettanti **<text>** e raggrupparli in un unico **<group>**, l'elemento che, secondo la definizione, contiene l'oggetto di un testo composito.<sup>48</sup>

Di seguito è riportata la struttura del corpo del file XML:

```
<text xml:id="text">
  <group xml:id="group">
    <text> ... </text>
    <text> ... </text>
    ...
  </group>
</text>
```

Per distinguere i numerosi **<text>** è stato predisposto un *@xml:id* diverso per ogni documento. I valori dell'attributo sono stati ripresi dalle due numerazioni utilizzate per classificare e ordinare i diversi testi, secondo questo schema: "Numerazione originale\_Numerazione moderna". Ad esempio, il documento con numerazione originale CCXL e con numerazione moderna 210, presenterà nel **<text>** il seguente attributo: `xml:id="CCXL_210"`.

## I documenti

Prima della trascrizione del testo vero e proprio, ogni **<text>** è arricchito da una serie di informazioni che forniscono dettagli di vario tipo sul documento. Queste indicazioni sono raggruppate nell'elemento **<front>**, identificato da un *@xml:id*. Al suo interno, oltre ai due **<titlePart>** e al **<docDate>**, utilizzati rispettivamente per trascrivere le due numerazioni (originale e moderna) e la data topica e cronica del testo di riferimento, vi sono numerosi **<div>**.

La presenza di diversi attributi permette di determinare al meglio tutti gli elementi e, al tempo stesso, favorisce la navigazione all'interno dei contenuti per le fasi successive del progetto. Nello specifico, il *@type*, come si può intuire dal nome, precisa ulteriormente la tipologia del tag; l'*@xml:id*, invece, identifica in maniera univoca l'elemento, associandolo al documento tramite uno schema simile a quello utilizzato per i **<text>**.

---

<sup>48</sup> Sito ufficiale, <http://www.tei-c.org/release/doc/tei-p5-doc/it/html/ref-group.html> (visitato il 05/01/2015), cit.

Di seguito è riportata la struttura del **<front>**:

```
<front xml:id="NumerazioneOriginale_front">
  <titlePart type="NumerazioneNuova"> ... </titlePart>
  <titlePart type="NumerazioneOrig"> ... </titlePart>

  <docDate xml:id="NumerazioneOriginale_docDate">
    <date when="anno-mese-giorno"> ... </date>
    <placeName> ... </placeName>
  </docDate>

  <div type="regesto" xml:id="NumerazioneOriginale_regesto">
    <p> ... </p>
  </div>

  <div type="footer" xml:id="NumerazioneOriginale_footer">
    <div type="orig_doc"> ... </div>
    <div type="biblio"> ... </div>
    <div type="crit_notes"> ... </div>
    <div type="appr"> ... </div>
  </div>
</front>
```

Il primo **<div>** contiene il regesto, in altre parole un breve riassunto del documento dotato di tutte le informazioni necessarie per una rapida comprensione dei contenuti<sup>49</sup>. Nel testo possono essere presenti alcuni termini in latino: questi sono inseriti all'interno dell'elemento **<term>**, al quale è associato un attributo *@xml:lang* con valore "latin". Il secondo, invece, presenta una struttura più complessa. Al suo interno, infatti, sono presenti quattro ulteriori **<div>**, ognuno dei quali è identificabile per mezzo dell'attributo *@type*. Il primo include le informazioni riguardanti la collocazione del documento all'interno del manoscritto; il secondo contiene la bibliografia di riferimento, con rimando alla lista bibliografica vera e propria (in coda al file XML); il terzo racchiude eventuali note critiche e il quarto, infine, predispone uno spazio per altri testi di approfondimento.

---

<sup>49</sup> Come è già stato specificato nel capitolo relativo alla storia del Codice (v. cap. 2).

Allo stesso livello gerarchico del **<front>** c'è il **<body>**, l'elemento che contiene l'intero corpo del documento. Al suo interno, le diverse informazioni sono state codificate per mezzo di numerosi elementi. Prima di passare alla loro descrizione, però, bisogna fare una distinzione tra elementi strutturali e semantici. I primi, non trasmettendo alcun tipo di informazione, si occupano di riprodurre la struttura originaria del testo (in paragrafi, in versi, ecc.) nel documento elettronico; i secondi, invece, sono utilizzati per marcare semanticamente i contenuti ricercati. I tag strutturali adoperati per la codifica del Codice Pelavicino sono due: **<p>** e **<pb/>**. Il primo permette di distinguere i paragrafi, mentre il secondo segnala il cambio di pagina. Entrambi gli elementi presentano due attributi, *@xml:id* e *@n*, che sono usati per distinguerli dagli altri. Ad esempio, il documento che al suo interno presenta il cambio dalla pagina 214 verso alla 215 recto, conterrà il seguente **<pb/>**: `<pb n="215r" xml:id="fol_215r"/>`. Dopo aver impostato il testo dal punto di vista strutturale, si può passare alla codifica delle diverse informazioni.

Quasi tutti i documenti, nonostante la molteplicità di contenuti, presentano una o più date di redazione, espresse in numeri romani. Si tratta di un'informazione fondamentale, poiché permette di collocare temporalmente un determinato documento. Per questo elemento è utilizzato **<date>** che, oltre al resto, consente la memorizzazione della data in cifre arabe (nel formato anno-mese-giorno), grazie all'uso dell'attributo *@when*.

Anche per i mestieri e per le diverse strutture citate nel Codice è previsto l'utilizzo di due elementi. I primi sono codificati per mezzo di un **<roleName>** privo di attributi, non essendo prerogativa degli studiosi tener conto del collegamento fra l'impiego e l'eventuale persona associata; i secondi, invece, vengono racchiusi all'interno di un **<orgName>**. Data la varietà degli insediamenti, ogni **<orgName>** è contraddistinto da un *@type* che ne specifica la tipologia. Ad esempio, la chiesa di Ponzanello sarà resa così nel file XML: `<orgName type="chiesa">ecclesia de Ponçanello</orgName>`.

Per la codifica delle monete è stato necessario adattare uno degli elementi predisposti dalle TEI Guidelines: il tag **<measure>**. Questo elemento, infatti, è

genericamente usato per qualsiasi unità di misura ma, data la sua generalità, può essere impiegato anche per fini più specifici. Per questo elemento sono previsti tre eventuali attributi: il *@type*, che viene utilizzato per descrivere la tipologia della moneta (*imperiales*, *ianuenses*, ecc.), *@quantity*, che contiene la quantità espressa in cifre arabe e *@unit*, che informa circa l'unità di riferimento nelle tre possibili casistiche: *denari*, *soldi* e *lire*.

Perché le liste del **<teiHeader>** possano trovare una corrispondenza con le numerose occorrenze nel testo, è necessario che ogni nome venga annotato con un preciso elemento. Nello specifico, è importante che ogni persona sia codificata all'interno di un **<persName>** e ogni luogo all'interno di un **<placeName>**. Il collegamento tra le singole voci della lista e i nomi presenti nel testo è reso possibile grazie all'utilizzo di un attributo *@ref*, al quale dovrà essere dato come valore l'*@xml:id* corrispondente preceduto dal simbolo cancelletto: #.

Qui di seguito un esempio di **<persName>** tratto dal file XML:

```
<persName ref="#GerardusVeronus">Gerardinus</persName>
```

I *signa tabellions*, invece, sono marcati nel testo tramite l'elemento vuoto **<ptr/>**: un puntatore utilizzato per fare riferimento a un altro tag del file XML. Il **<ptr/>** è dotato di due attributi: un *@ref*, utilizzato per creare la corrispondenza tra le voci nella lista dei *signa* (inclusa nel **<teiHeader>**), e un *@facs*, fondamentale per la realizzazione del collegamento testo immagine fra i timbri notarili (v. cap. 5).

Ciascun documento è caratterizzato, inoltre, da una serie di interventi editoriali, nello specifico note e correzioni. Anche queste tipologie di informazioni necessitano di una loro codifica e, per questo motivo, sono stati predisposti due elementi: **<note>** e **<add>**. Il primo è utilizzato per le note ed è provvisto di tre attributi: *@place*, per la collocazione dell'annotazione nel testo di riferimento (con valore standard "foot"), *@n*, per la numerazione, e *@xml:id*, per l'identificazione univoca del tag all'interno del file XML. L'elemento **<add>**, invece, è destinato alle correzioni e, nello specifico, alle aggiunte per motivi di corruzione o di scarsa comprensibilità del testo. Il tag, dotato di un attributo

*@place* per la disposizione testuale dell'aggiunta, contiene la parte di lemma mancante e, perciò, deve essere così inserito all'interno del termine codificato:

```
ips<add place="inline">i</add>us
```

## La bibliografia

La bibliografia completa dell'edizione digitale è inserita all'interno di un **<back>**, allo stesso livello gerarchico del `<text xml:id="text">`. Le diverse voci sono racchiuse in diversi **<bibl>** e, per mezzo dell'attributo *@xml:id*, si ricollegano direttamente ai **<ref>** bibliografici contenuti nel `<div type="biblio">`.<sup>50</sup> Ad esempio, il **<ref>** con *@target="#LUPO\_GENTILE\_1912"* troverà nel **<back>** una corrispondenza di questo tipo: `<bibl xml:id="LUPO_GENTILE_1912">M. Lupo Gentile, <emph>Il regesto del codice Pelavicino</emph>, in «Atti della Società Ligure di Storia Patria», XLIV (1912).</bibl>`.

Tutti i **<bibl>** sono inseriti, a loro volta, all'interno di un'unica **<listBibl>**, l'elemento che può contenere una lista di citazioni bibliografiche di qualsiasi natura.<sup>51</sup>

Di seguito è riportata la struttura completa del **<back>**:

```
<back>
  <div type="biblCompleta">
    <listBibl>
      <bibl xml:id="..."> ... </bibl>
      <bibl xml:id="..."> ... </bibl>
      <bibl xml:id="..."> ... </bibl>
      ...
    </listBibl>
  </div>
</back>
```

---

<sup>50</sup> Che si trova nel **<front>** di ogni documento.

<sup>51</sup> Sito ufficiale, <http://www.tei-c.org/release/doc/tei-p5-doc/it/html/ref-listBibl.html> (visitato il 07/01/2015), cit.



## 4 Dalla codifica dei testi a EVT

### 4.1 Il sistema di correzione

Durante la codifica del manoscritto, è stato sviluppato un secondo sistema di visualizzazione di natura differente rispetto a quella di EVT. Quest'interfaccia web, infatti, è stata realizzata con l'intenzione di fornire ai correttori una visuale comprensibile e consultabile dei documenti codificati, in modo da poter segnalare eventuali errori o dimenticanze. Al tempo stesso, buona parte delle soluzioni grafiche predisposte nel sistema di correzione sono state poi integrate nello scenario EVT per l'edizione digitale vera e propria.

Il sistema di correzione è stato realizzato tramite XSLT, uno dei tre linguaggi di programmazione dei fogli di stile per i documenti in formato XML.

#### 4.1.1 Il linguaggio XSL<sup>52</sup>

Un file XML, perché possa essere rappresentato correttamente dai browser o da altri sistemi per la resa grafica dei contenuti, deve essere associato a un foglio di stile, cioè a un linguaggio in grado di descrivere il modo in cui un documento elettronico deve essere visualizzato.

I fogli di stile XSL (*eXtensible Stylesheet Language*) sono stati creati appositamente per XML e possono essere utilizzati in combinazione con altri linguaggi, ad esempio con i CSS (*Cascading Style Sheet*). Com'è noto, lo standard XML prevede che i contenuti di un documento siano separati dalla formattazione della pagina in cui saranno pubblicati e, per questo motivo, ogni file XSL dovrà essere associato in maniera esplicita all'XML corrispondente.

In realtà, XSL è solamente un'etichetta che comprende tre linguaggi molto diversi fra loro: XSLT (*XSL Transformation*), per trasformare un file XML in un

---

<sup>52</sup> Extensible Stylesheet Language (XSL) Version 1.1, <http://www.w3.org/TR/xsl/> (visitato l'11/01/2015).

documento di un altro formato (ad es. HTML); XSL-FO (*XSL Formatting Objects*), per l'applicazione degli stili e del modo di apparizione di un documento XML; XPath (*XML Path*), usato nei fogli di stile XSLT per descrivere come accedere alle parti di un file XML.

L'XSLT<sup>53</sup> è linguaggio di trasformazione che permette di convertire un documento XML in diversi formati, in modo da renderlo correttamente visualizzabile dai diversi browser o da altri software. È diverso dagli altri linguaggi di programmazione perché è basato su una serie di regole di trasformazione ricorsive che dichiarano che cosa deve essere generato quando si incontra un certo elemento nel documento.<sup>54</sup> Le varie *template rules* possono essere disposte in qualsiasi ordine all'interno del file.

Di seguito è riportata la struttura di una generica regola XSLT che estrae il valore di un elemento `<title>` racchiuso in un `<bibl>`, e lo inserisce all'interno di un tag `<h1>`:

```
<xsl:template match="bibl">
  <h1>
    <xsl:value-of select="title"/>
  </h1>
</xsl:template>
```

La trasformazione avviene mediante un processore XSLT che, dopo aver letto il documento XML in input, crea un albero corrispondente. In seguito, il processore inizia a percorrere l'albero confrontando i singoli nodi con le regole presenti nel foglio di stile. Quando viene trovata una *template rule* corrispondente, viene prodotto l'output secondo le istruzioni della regola. Il processore passa così al nodo successivo, fino alla fine del documento XML. Il risultato finale sarà un albero di output.

---

<sup>53</sup> XSL Transformation (XSLT) Version 1.0, <http://www.w3.org/TR/xslt> (visitato l'11/01/2015).

<sup>54</sup> Doug Tidwell, *XSLT: 2nd Edition*, Sebastopol, O'Reilly Media, 2008, pp. 1-3.

## 4.1.2 L'interfaccia grafica e le fasi di realizzazione

Il sistema di correzione realizzato in XSLT permette di navigare all'interno dei 145 documenti codificati e di esplorarne i diversi contenuti. Ogni documento è racchiuso all'interno di una scheda che presenta, in fronte, tutti i metadati inseriti nel <front> XML e, nel corpo, il testo vero e proprio (v. fig. 4 e 5).

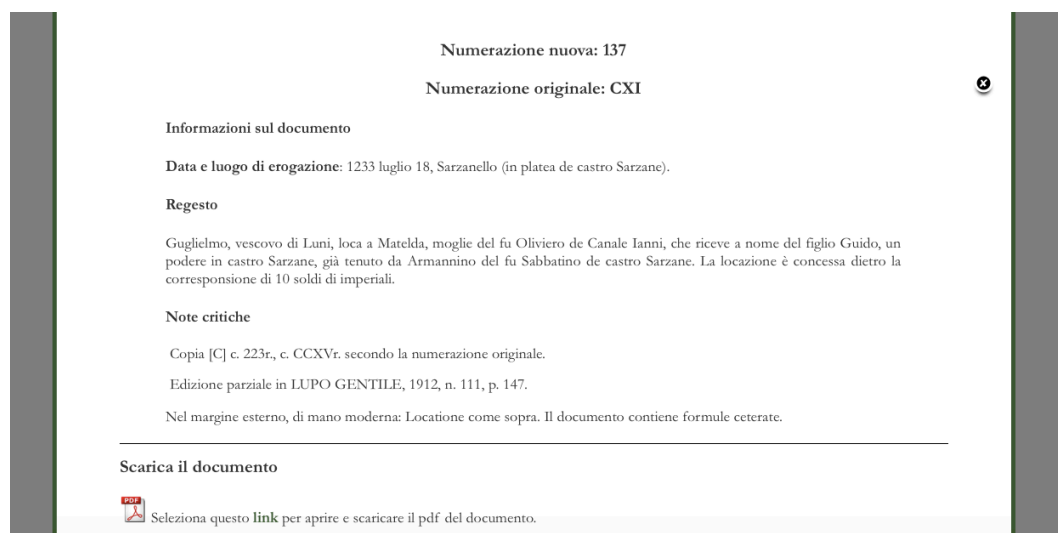


Figura 4. Esempio di documento all'interno dello strumento di correzione (fronte)

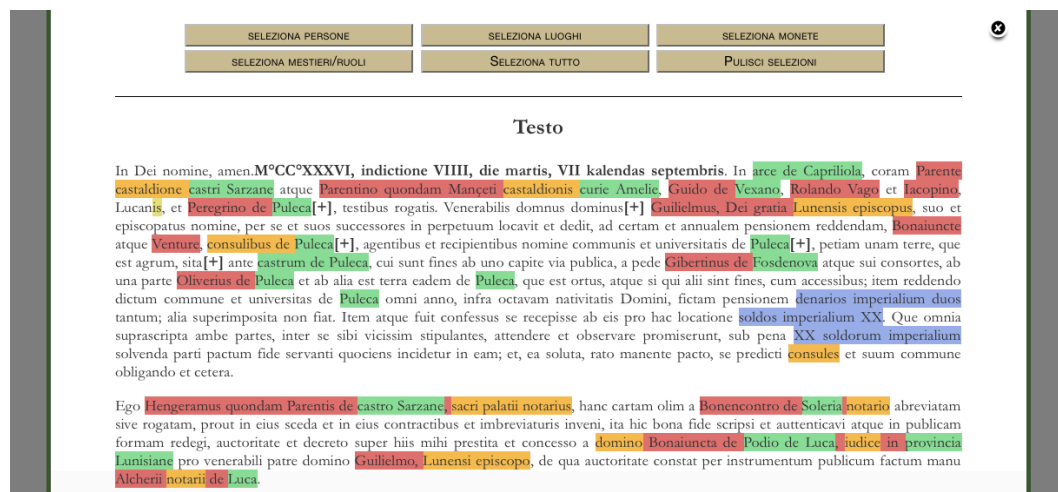


Figura 5. Esempio di documento all'interno dello strumento di correzione (corpo)

Com'è possibile osservare dalla fig. 5, ogni elemento della codifica è reso in maniera differente, in modo da esplicitare l'informazione e da favorire la correzione. In particolare, le note [+], le immagini dei *signa tabellionis* e le date normalizzate sono visualizzabili, al click, per mezzo di finestre pop-up, mentre i nomi di persona e di luogo, i ruoli e le monete possono essere evidenziati con

diversi colori grazie a degli appositi bottoni. In seguito alla generazione delle liste e al loro inserimento nel **<teiHeader>** del file XML, sono stati realizzati dei pop-up anche per le persone e le località presenti nel testo, affinché siano rese consultabili tutte le informazioni presenti nelle singole voci degli elenchi.

Il documento XSLT è composto da una serie di *template rule*, ognuna delle quali si occupa di generare un diverso output per il file HTML in uscita. La prima regola, collocandosi direttamente sulla radice del file XML (il nodo **<TEI>**), origina lo scheletro **<html>** dell'interfaccia: al suo interno sono inseriti l'**<head>** con i collegamenti esterni e il **<body>** contenente il `<div id="contenitore">` e il `<div id="bloccoChiusura">`, i blocchi che racchiudono rispettivamente i 145 documenti codificati e il footer della pagina HTML. All'interno del `<div id="contenitore">`, oltre ai documenti, sono presenti un'intestazione dell'interfaccia e un paragrafo che illustra le principali funzionalità del sistema di correzione.

Di seguito è riportata la struttura della prima *template rule*:

```
<xsl:template match="tei:TEI">
  <html>
    <head>
      ...
    </head>
    <body>
      <div id="contenitore">
        ...
      </div>
      <div id="bloccoChiusura">
        ...
      </div>
    </body>
  </html>
</xsl:template>
```

Ciascun testo viene racchiuso all'interno di un **<div>** invisibile e viene associato ad un altro **<div>** cliccabile tramite i due attributi `@id` e `@data-id`, compilati con lo stesso valore (la numerazione moderna del documento). Attraverso il seguente

script jQuery (inserito nell'<head> del file HTML), è possibile attivare la finestra modale<sup>55</sup> in grado di mostrare il contenuto del testo nascosto:

```
jQuery(function ($) {
    $('.cliccabile .basic').click(function (e) {
        var id=$(this).closest('div').attr('data-id');
        $('#'+id).modal();
        return false;
    } )
} )
```

L'elemento che contiene solo la numerazione moderna del documento (<h2 class="basic">) viene associato ad un evento *click()*. Successivamente, la funzione jQuery *modal()* viene richiamata su una variabile *id*, precedentemente compilata con il valore equivalente al *@data-id* del <div id="cliccabile">. In questo modo, grazie alla corrispondenza degli attributi prima citati (*@id* e *@data-id*), è possibile visualizzare il documento racchiuso nel <div> invisibile mediante il click sull'<h2> ad esso collegato.

Le altre *template rule* presenti nel file XSLT si riferiscono principalmente alla resa grafica dei vari elementi codificati all'interno dei documenti. Le informazioni racchiuse all'interno dei diversi <front> sono state trasformate utilizzando una serie di elementi standard, in modo da disporre ordinatamente i numerosi metadati.

Prima del testo, ogni documento presenta una <form> con una serie di <input> cliccabili. Ognuno di questi bottoni rimanda a un'ulteriore funzione jQuery, inserita nel file esterno *script.js*, che permette di evidenziare le persone, le località, i ruoli e le monete disponibili nel documento. Perché la funzione possa riconoscere le quattro tipologie di elementi, è necessario che altre *template rules* trasformino i tag XML in una serie di <span> distinguibili per categoria. Ad esempio, l'elemento <persName> sarà reso così nel file HTML: <span class="persName">.

---

<sup>55</sup> Una finestra figlia che richiede all'utente di interagire con essa prima di tornare alla pagina iniziale.

Di seguito è riportata la funzione jQuery per l'evidenziazione dei nomi di persona:

```
$(document).on('click', 'personeb', function(event) {
    event.preventDefault();
    var id=$(this).closest('.blocco').attr('.id');
    $('#'+id+' .persName').css('background-color' , '#dd7171');
    $('#'+id+' .placeName').css('background-color' ,
    'transparent');
    $('#'+id+' .measure').css('background-color' ,
    'transparent');
    $('#'+id+' .role').css('background-color' , 'transparent');
} );
```

Lo script, dopo aver associato all'<input> con class="personeb" l'evento *click()*, attribuisce ad una variabile *id* il valore dell'@*id* del <div id="blocco"> più vicino al pulsante, che corrisponde alla numerazione moderna del documento. Così, attraverso una regola CSS per la colorazione dei tag, sarà possibile evidenziare solamente gli elementi appartenenti a quel determinato documento. Naturalmente, trattandosi dello script relativo ai nomi di persona, le restanti righe di codice dovranno nascondere l'eventuale colore di sfondo degli altri tag, attribuendo al 'background-color' il valore 'transparent'. L'evento *preventDefault()* permette di rimanere nella stessa posizione, evitando di essere riportati in cima alla pagina.

Le finestre pop-up per gli elementi che possono mostrare del materiale aggiuntivo (persone, località, note, date e *signa tabellionis*) sono inserite direttamente all'interno del file HTML e, mediante un ultimo script, possono essere nascoste o mostrate al click. Prendendo in analisi i nomi di luogo, la *template rule* corrispondente avrà la seguente struttura:

```
<xsl:template match="tei:placeName[starts-with(@ref, '#')]">
    <span class="placeName">
        <span class="click">
            <xsl:apply-templates/>
        </span>
    </span>
    <span>
        <xsl:attribute name="class">place</xsl:attribute>
```

```

        <xsl:for-each select="//tei:place[@xml:id=substring-
after(current()/@ref, '#)']">
            ...
        </xsl:for-each>
    </span>
</span>
</xsl:template>

```

La regola XSLT affianca al nome di località, racchiuso all'interno di due **<span>**, un ulteriore **<span>** con attributo `class="place"` e, mediante il ciclo `xsl:for-each`, lo compila con tutti i dati presenti nella specifica lista del **<teiHeader>** relativi a quel determinato luogo. La scheda generata viene poi nascosta tramite i CSS, per essere mostrata solo al click sull'elemento. Lo script jQuery in grado di eseguire l'operazione è il seguente:

```

$(document).ready(function() {
    $('.click').click(function() {
        $(this).next().toggle();
        return false;
    });
});

```

L'elemento con `class="click"`, che corrisponde allo **<span>** contenente il nome della località, viene associato ad un evento `click()` che, dopo essere stato attivato, richiama la funzione `toggle()` sul tag successivo, ovvero la scheda di metadati generata dalla *template rule* precedentemente illustrata. La funzione `toggle()` permette di mostrare e nascondere l'elemento selezionato.

Una volta completato, il file XSLT può essere associato al documento XML tramite la funzione “*Configure Transformation Scenario(s)*”<sup>56</sup> dell'editor **<oxyen>**. Il risultato dell'associazione, come già specificato, sarà un file HTML di output, al quale dovranno essere associati tutti gli script e i CSS necessari per una corretta visualizzazione dell'interfaccia.

---

<sup>56</sup> Per ulteriori informazioni: <http://www.oxygenxml.com/doc/ug-author/index.html#topics/xml-transformation-with-xslt.html> (visitato il 13/01/2015).

## 4.2 L'edizione digitale con EVT

Nonostante il sistema di correzione sia in grado di offrire una visualizzazione completa dei documenti e una consultazione delle diverse informazioni, per l'edizione digitale del Codice Pelavicino si è deciso di far riferimento a un software in grado di produrre un'*image-based digital edition*.

Il programma di consultazione EVT (*Edition Visualization Technology*), sviluppato come progetto di ricerca presso l'Università di Pisa, è un'applicazione web open source scritta in Javascript, che richiede soltanto l'uso di un comune browser per essere utilizzata. Il software permette di sfogliare virtualmente il manoscritto in diversi modi e si serve di alcuni interessanti strumenti per la consultazione: lente d'ingrandimento, filtri per l'elaborazione grafica dell'immagine, collegamento fra testo e pagina del volume, ecc.<sup>57</sup> EVT è stato sviluppato ai fini del progetto di digitalizzazione del Libro di Vercelli (*Vercelli Book*)<sup>58</sup>, un antico manoscritto redatto alla fine del X secolo, ed è tuttora in fase di sviluppo.

### 4.2.1 Fasi preliminari

Prima di passare all'interfaccia di consultazione completa e alle sue principali caratteristiche, è importante soffermarsi su alcune delle convenzioni che sono state impostate durante la codifica e nella fase immediatamente successiva.

Anzitutto, affinché il software EVT possa riconoscere e disporre in maniera corretta i vari elementi della codifica, è necessario che i contenuti del file XML siano inseriti all'interno di diversi `<div>`, com'è già stato illustrato nel capitolo precedente, e che ognuno di essi sia identificato con un diverso `@xml:id`, per evitare casi di ambiguità. Anche il testo di ogni documento, prima di essere suddiviso nei vari paragrafi, dovrà essere inserito in un `<div>`, in modo da permettere al software EVT di individuarlo correttamente.

---

<sup>57</sup> Il Software di visualizzazione: EVT, <http://vbd.humnet.unipi.it/?p=54> (visitato il 14/01/2015).

<sup>58</sup> Vercelli Book Digitale, <http://vbd.humnet.unipi.it> (visitato il 14/01/2015).



Sempre a livello di codifica, è importante che l'elemento vuoto `<pb/>` faccia riferimento in maniera esplicita alle pagine del manoscritto tramite i suoi attributi. In questo modo, il software EVT sarà in grado di associare al cambio di pagina l'immagine corrispondente, fornendo così all'utente la possibilità di sfogliare contemporaneamente il testo codificato e la carta a esso collegato. I due attributi utilizzati (`@n` e `@xml:id`), com'è già stato esposto nelle pagine precedenti, riprendono il nome che è stato assegnato alle pagine del volume e, a tal proposito, è fondamentale che le due possibili terminazioni, "r" per "recto" e "v" per "verso", siano sempre specificate.

L'edizione digitale in EVT si serve poi di tre dimensioni diverse per le immagini del manoscritto: standard, per la consultazione tradizionale; thumbnail, per la visualizzazione completa dell'elenco delle carte; alta risoluzione, per l'attivazione della lente d'ingrandimento. Nella fase successiva alla codifica, sarà dunque necessario ridimensionare le fotografie e rinominarle secondo uno schema ben preciso, in modo da permettere anche il collegamento con i relativi `<pb/>` del file XML. Nello specifico, le immagini corrispondenti a una determinata pagina dovranno essere rinominate in questa maniera: `fol_numerazionePagina.jpg` per la dimensione standard, `fol_numerazionePagina_small.jpg` per la miniatura e `fol_numerazionePagina_big.jpg` per la versione ad alta risoluzione.

Una volta completate queste operazioni e inseriti tutti i file nelle apposite cartelle del pacchetto EVT Builder, sarà possibile generare l'edizione digitale applicando il foglio di stile `evt_builder.xsl` al documento XML.

#### **4.2.2 L'interfaccia grafica**

L'edizione digitale del Codice Pelavicino permette una navigazione approfondita del manoscritto ed è in grado di fornire all'utente una serie di strumenti e funzionalità innovative per consultare, al tempo stesso, le immagini del volume e il testo dei documenti codificati (v. fig 6 e 7). Nonostante buona parte delle soluzioni grafiche siano state riprese dall'edizione del Vercelli Book, il software EVT è stato modificato per potersi adeguare alle caratteristiche specifiche del Codice.



Figura 6. Interfaccia EVT: pagina del manoscritto e regesto

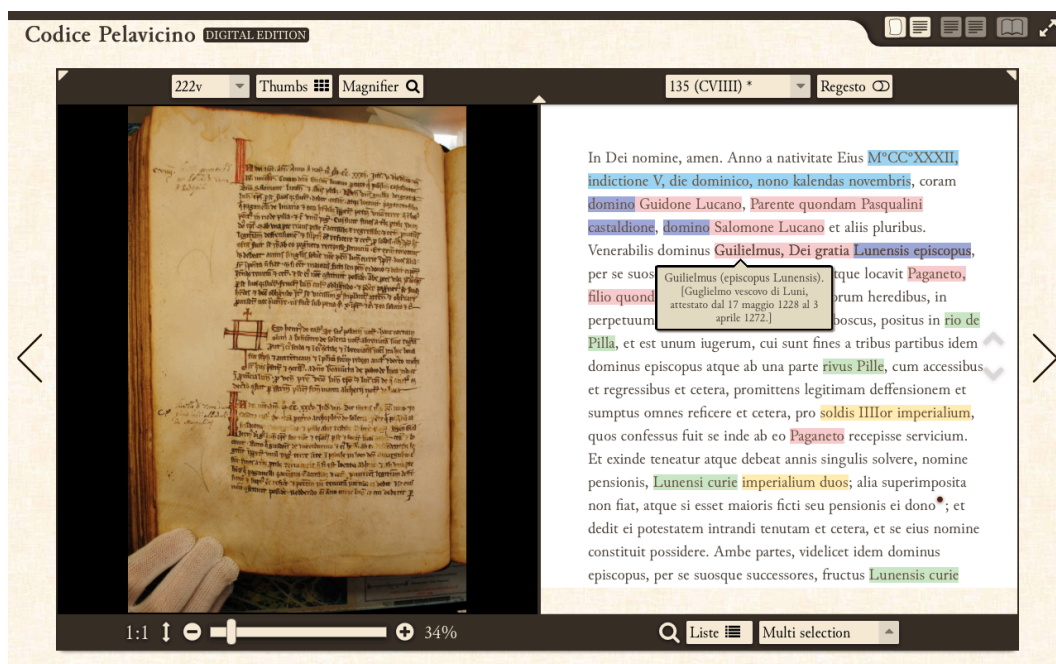


Figura 7. Interfaccia EVT: pagina del manoscritto e testo codificato

Attraverso un'interfaccia grafica pulita e di facile comprensione, l'utente può navigare contemporaneamente tra le pagine codificate e le immagini del manoscritto e, grazie a un bottone collocato sulla barra degli strumenti, può decidere se nascondere o mostrare il regesto di un determinato documento. Il testo è esplorabile in maniera integrale e, com'è possibile osservare dalla figura 7,

buona parte delle soluzioni grafiche utilizzate nel sistema di correzione sono state inserite nel programma di consultazione EVT. In particolare, l'evidenziazione dei contenuti e i pop-up supplementari per i nomi di persona e di luogo, per le date e per i *signa tabellionis* sono stati ripresi e reintegrati nella veste grafica dell'edizione digitale.

La sezione "Liste", espandibile al click, permette di esplorare gli elenchi di persona e di località che sono stati inseriti nel **<teiHeader>** del file XML e consente, inoltre, di ricercare tutte le occorrenze di un determinato nome direttamente nel testo dei documenti.

Per quanto riguarda il supporto alle immagini, l'edizione digitale dispone di uno zoom e di una lente d'ingrandimento per la visualizzazione dei dettagli, come, ad esempio, le aggiunte di mano moderna nei margini esterni dei documenti. Inoltre, la sezione "Thumbs" permette di visualizzare una schermata comprensiva di tutte le carte del manoscritto, sfruttando le miniature generate nelle fasi precedenti alla generazione dell'edizione digitale.

## 5 I *signa tabellionis*

Tra le caratteristiche che contraddistinguono il Codice Pelavicino, vi è senz'altro la numerosa quantità di simboli utilizzati dai rogatari dei documenti per autenticare e convalidare le varie sottoscrizioni: i *signa tabellionis*. L'insieme di questi "segni" notarili rappresenta, infatti, un'importante fonte di studio per la diplomatica e per la paleografia medievale e può offrire, al tempo stesso, uno spaccato storico sul prestigio di questa categoria professionale in un periodo compreso fra il X e il XIII secolo.

Per questo motivo, durante la lavorazione dell'edizione digitale è stato deciso di approfondire lo studio sui *signa tabellionis*, in modo da sviluppare degli strumenti in grado di favorirne la visibilità e una consultazione maggiormente approfondita.

### 5.1 Cenni storici

Nel corso dei secoli, la figura del notaio assunse sempre più importanza e prestigio, fino a raggiungere, nel XII secolo, il definitivo assetto di *persona publica* impegnata nelle strutture della realtà sociale e istituzionale. Questa posizione privilegiata si manifestò, in maniera esplicita, nelle capacità di attribuire a una scrittura una vera e propria valenza giuridica, grazie anche all'autenticazione del documento notarile mediante l'adozione di un segno identificativo personale, il *signum tabellionis*.

Nonostante non si possa parlare di segni del tabellionato fino a quando il notaio non riuscì a raggiungere la *publica fides*, va notato che già nel II secolo d. C. anche lo *scriptor* poneva in fondo al documento il suo sigillo.<sup>59</sup> La presenza del *signum tabellionis* (o *signum notarii*) nei documenti medievali è, dunque, il frutto di una lunga evoluzione che, per certi aspetti, può essere messa a confronto con il mutamento dello stesso testo notarile. È evidente, infatti, che nel passaggio dalla

---

<sup>59</sup> Giorgio Costamagna, *Il notaio a Genova tra prestigio e potere*, Roma, 1970, pp. 138-139.

*carta* e dal *breve* all'*instrumentum*, verso la metà del XII secolo, anche il simbolo identificativo del notaio abbia subito una sua mutazione, deformandosi fino ad assumere le forme più disparate.<sup>60</sup> La *carta*, una delle tipologie documentarie più diffuse tra il XI e il XIII secolo, era destinata a transazioni di carattere definitivo e ad atti di liberalità quali la compravendita, la permuta, il testamento e la donazione. Il *breve*, invece, era destinato ad investiture, refute e accordi e, di norma, era chiuso semplicemente dal nome del notaio che indicava di avere scritto il testo. Le *carte* continuarono ad essere utilizzate ancora fin verso la metà del sec. XII, ma subito dopo, anche per effetto del risorto studio del diritto romano, apparirono i primi esempi di documenti di un nuovo tipo, che davano notizia in forma oggettiva dell'avvenuto negozio giuridico, omettendo qualunque sottoscrizione dell'autore, dell'azione, e indicando i nomi dei testimoni e la chiusura del notaio: gli *instrumenta*.<sup>61</sup>

Grazie allo studio dei primi *signa tabellionis*, è stato possibile dimostrare come tali caratteri siano stati originati da influenze di tachigrafia sillabica e, specialmente, dall'uso delle note rappresentanti le parole *notarius* e *subscripti*, già rintracciabili nei papiri ravennati del VI secolo.<sup>62</sup> La tachigrafia era un metodo di scrittura che impiegava segni, abbreviazioni o simboli (es. note tironiane<sup>63</sup>) per rappresentare le lettere, le parole o le frasi di un testo. Attraverso un'elaborazione secolare, alcune di queste note tachigrafiche finirono per trasformarsi in un simbolo distintivo da utilizzare per particolari forme d'invocazione e, naturalmente, per rappresentare l'autorità notarile. La riprova del fatto che nei

---

<sup>60</sup> Giorgio Costamagna, *Il notaio a Genova tra prestigio e potere*, Roma, pp. 140-142.

<sup>61</sup> Cesare Manaresi, *Diplomatica* in *Enciclopedia Italiana Treccani*, Milano 1931, [http://www.treccani.it/enciclopedia/diplomatica\\_%28Enciclopedia\\_Italiana%29/](http://www.treccani.it/enciclopedia/diplomatica_%28Enciclopedia_Italiana%29/) (visitato il 25/01/2015).

<sup>62</sup> Giorgio Costamagna, *Studi di paleografia e di diplomatica*, Roma, 1972, pp. 7-11.

<sup>63</sup> Le note tironiane sono i segni tachigrafici che devono il loro nome a Marco Tullio Tiro, lo scriba di Cicerone che, per trascrivere i discorsi di quest'ultimo, aveva inventato un sistema di simboli per abbreviare le parole d'uso frequente. Per ulteriori informazioni: Wikipedia, Voce Note tironiane, [http://it.wikipedia.org/wiki/Note\\_tironiane](http://it.wikipedia.org/wiki/Note_tironiane) (visitato il 18/01/2015).

primi *signa* si può scorgere anche una forma d'invocazione, sta nella possibilità di accertarne la presenza anche all'inizio del documento.<sup>64</sup>

L'origine vera e propria del *signum notarii* è stata ampiamente discussa e, a oggi, sono due le correnti più accreditate. La prima, portata avanti dallo storico e archivista francese Marie-Claude Guigue, sostiene che il segno notarile sarebbe una derivazione del *signum manuale*, il simbolo della personalità giuridica nell'età giustiniana; la seconda, sorretta dagli studi di Enrico Donato Petrella, ritiene, invece, che il *signum* rappresenterebbe la logica conseguenza dall'abitudine presso i notai d'invocare il nome di Dio prima di accingersi a compiere una qualsiasi azione di valenza giuridica.<sup>65</sup>

Dopo il 1000, il simbolo notarile andò poco a poco deformandosi e, con la scomparsa della tachigrafia sillabica, si ridusse a uno strano intreccio di elementi difficilmente scomponibili. Nonostante ciò, non dovrà essere confuso con il *signum communis*, il simbolo utilizzato nei documenti comunali e composto da un asse verticale tagliato a croce da due trasversali racchiudenti un nodo.<sup>66</sup>

## 5.2 I signa nelle altre edizioni

Prima di poter impostare gli strumenti per una miglior resa dei *signa tabellionis*, è stato necessario effettuare una serie di ricerche per verificarne la presenza e la specifica trattazione in documenti simili a quelli che compongono il Codice Pelavicino. Nello specifico, sono state prese in considerazione due fonti cartacee e alcune pubblicazioni digitali disponibili gratuitamente sul web. Solo in seguito a queste indagini è stato possibile sviluppare una serie di soluzioni grafiche che potessero aggiungere qualcosa a quelle già esistenti.

---

<sup>64</sup> Giorgio Costamagna, *Il notaio a Genova tra prestigio e potere*, p. 142.

<sup>65</sup> Giorgio Costamagna, *Studi di paleografia e di diplomatica*, pp. 34-35.

<sup>66</sup> Costamagna, Giorgio, 1962. *Note di diplomatica comunale. Il "Signum Communis" e il "Signum Populi" a Genova nei secoli XII e XIII*. In *Miscellanea di Storia Ligure in onore di Giorgio Falco*, Milano, Feltrinelli Editore, pp. 105-115.

La maggior parte delle pubblicazioni cartacee, com'è possibile osservare, ad esempio, nei testi che compongono il *Registrum Magnum del Comune di Piacenza*<sup>67</sup>, si limitano ad applicare le norme comunemente rispettate nelle edizioni documentarie, segnalando la presenza dei *signa* notarili tramite la dicitura S.T. racchiusa tra due parentesi. Nonostante ciò, nell'edizione de *I Libri Iurium della Repubblica di Genova*<sup>68</sup>, è possibile osservare anche uno schema riassuntivo di tutti i documenti, all'interno del quale è stato predisposto un campo per le sottoscrizioni e/o le autentiche. Nella colonna relativa alla fonte sono stati inseriti il nome dei rogatari o, in caso di documenti pubblici, la forma di convalidazione (sigillo, bulla, *signum communis*, ecc.), dando così forma ad una sorta di database cartaceo per le autenticazioni notarili: una scelta che è stata messa in pratica anche per l'edizione digitale del Codice Pelavicino.

Nonostante la digitalizzazione dei testi antichi sia in grado di fornire una serie di strumenti per estrapolare e rielaborare i diversi contenuti, per quanto riguarda il trattamento dei *signa tabellionis* non sono stati trovati esempi che proponessero soluzioni efficaci in aggiunta alle norme documentarie indicate in precedenza. Il progetto *Telma*<sup>69</sup>, ad esempio, benché presenti una notevole quantità di trascrizioni di documenti antichi (nello specifico *carte* originali), si limita a segnalare i simboli notarili tramite la dicitura [*signum*], senza predisporre, dunque, uno strumento per il collegamento fra il testo e l'immagine o un metodo d'indicizzazione dei contenuti. D'altra parte, le numerose edizioni digitali che mettono a disposizione anche la navigazione attraverso gli indici (es. *Le Petit Thalamus de Montpellier*<sup>70</sup>) sono solite circoscrivere gli elenchi per la ricerca solamente ai nomi di persona, di luogo e delle cose notevoli.

---

<sup>67</sup> Ettore Falconi, Roberta Peveri (a cura di), *Il Registrum Magnum del Comune di Piacenza*, Milano, Giuffrè, 1984-1997.

<sup>68</sup> Antonella Rovere, Dino Puncuh (a cura di), *I Libri Iurium della Repubblica di Genova: Introduzione*, Roma, Ministero per i beni culturali e ambientali, 1992.

<sup>69</sup> Sito ufficiale, <http://www.cn-telma.fr> (visitato il 20/01/2015).

<sup>70</sup> Sito ufficiale, <http://thalamus.huma-num.fr> (visitato il 20/01/2015).

Tuttavia, l'archivio collaborativo di *Monasterium.net*<sup>71</sup> ha messo a disposizione un interessante strumento per l'annotazione e la catalogazione degli elementi tipici del testo (v. fig. 8), come appunto i *signa tabellionis*. L'*image tool* permette, infatti, di intervenire direttamente sull'immagine del documento e, tramite una barra espandibile, dà la possibilità all'utente di delimitare e descrivere le zone interessate. In questo modo, oltre alla trascrizione del testo, si potrà consultare il facsimile annotato con tutte le informazioni ritenute necessarie per una miglior comprensione dei diversi contenuti. Si tratta comunque di uno strumento ancora poco utilizzato dagli studiosi e, data la sua genericità, non è particolarmente indicato per l'esclusivo approfondimento dei simboli notarili.

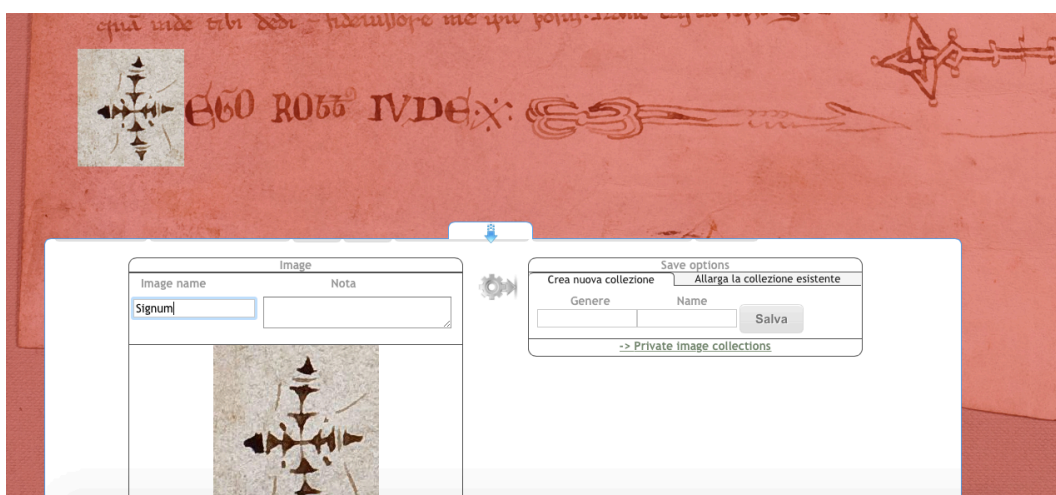


Figura 8. Esempio di annotazione nel tool di Monasterium

Terminata l'operazione di ricerca e raccolte tutte le informazioni necessarie, è stato deciso di sviluppare, oltre al <facsimile> XML per il collegamento fra il testo e l'immagine, una pagina HTML con l'indice di tutti i notai e i rispettivi *signa tabellionis*, in modo da fornire agli utenti un ulteriore strumento per la consultazione degli elementi principali della codifica.

---

<sup>71</sup> Sito ufficiale, <http://monasterium.net/mom/home> (visitato il 20/01/2015).



## 5.3 I *signa* nel Codice Pelavicino digitale

### 5.3.1 Il <facsimile> XML

La corrispondenza fra gli elementi del testo codificato e la loro raffigurazione rappresenta uno degli aspetti peculiari delle *image-based digital edition* e, per questo motivo, è stato deciso di predisporla anche per l'edizione digitale del Codice Pelavicino. Tramite l'elemento <facsimile>, inserito allo stesso livello gerarchico del <teiHeader> e del <text>, è stato possibile creare il collegamento mirato, unendo fra di loro i tag che identificano i *signa tabellionis* (<ptr/>) e le riproduzioni dei simboli notarili presenti nelle carte del manoscritto. Nonostante si sia trattato di un'operazione meccanica e non particolarmente complessa, la realizzazione del <facsimile> ha richiesto diverse fasi di lavorazione e, per la delimitazione delle aree da evidenziare, si è dovuto far riferimento ad uno specifico software open source: IMT (*Image Markup Tool*).<sup>72</sup>

Per prima cosa, l'elemento <facsimile> è stato suddiviso in una serie di <surface><sup>73</sup>, ognuno dei quali è stato collegato alla rispettiva pagina (<pb/>) nel testo codificato, mediante la corrispondenza fra i due attributi @corresp e @xml:id. In seguito, all'interno di ogni <surface> è stato inserito un <graphic/> per indicare la posizione fisica dell'immagine, e una serie di <zone> per delimitare le aree dei *signa* attraverso le quali creare il collegamento. La corrispondenza tra le zone localizzate e gli elementi identificativi dei simboli notarili è stata possibile grazie alla predisposizione di due attributi con lo stesso valore: @xml:id per gli <zone> e @facs per i <ptr/>.

Di seguito è riportata la struttura di un generico <surface>:

```
<surface xml:id="surf_numeroPagina" corresp="#fol_numeroPagina">
  <graphic url="images/single/numeroPagina.jpg" width="1200px"
  height="1793px"/>
</surface>
```

---

<sup>72</sup> Sito ufficiale, [http://tapor.uvic.ca/~mholmes/image\\_markup/](http://tapor.uvic.ca/~mholmes/image_markup/) (visitato il 21/01/2015).

<sup>73</sup> L'elemento <surface> definisce una superficie scritta in termini di uno spazio rettangolare di coordinate. Per ulteriori informazioni: <http://www.tei-c.org/release/doc/tei-p5-doc/it/html/ref-surface.html> (visitato il 21/01/2015).

```

<zone xml:id="st_numeroPagina_001" rendition="signum"
ulx="xxx" uly="xxx" lrx="xxx" lry="xxx" rend="visible"/>
<zone xml:id="st_numeroPagina_002" rendition="signum"
ulx="xxx" uly="xxx" lrx="xxx" lry="xxx" rend="visible"/>
</surface>

```

Com'è possibile osservare dalla struttura XML, l'elemento **<zone>**, per delimitare le aree sensibili dell'immagine, si serve di quattro coordinate spaziali sotto forma di attributi. Per inserire questi valori all'interno del file di codifica è stato utilizzato IMT, uno specifico software open source, sviluppato da Martin Holmes, che permette all'utente di definire aree rettangolari di annotazione (v. fig. 8) e di salvarle direttamente in un documento XML conforme allo standard TEI P5.

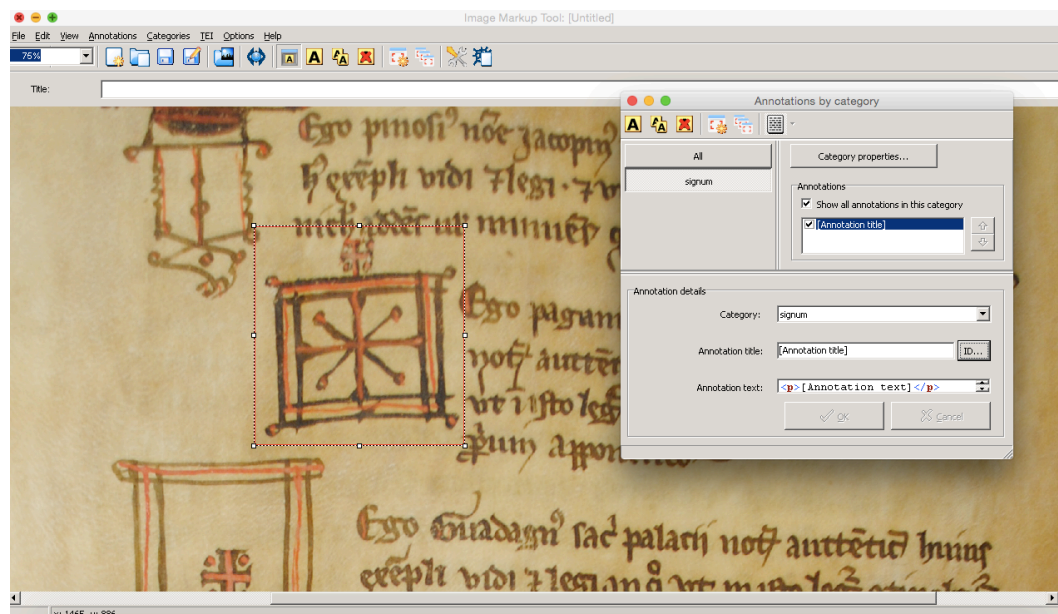


Figura 9. Schermata per l'annotazione d'immagine in IMT

Per ogni immagine sono state delimitate le aree interessate e, grazie al campo "Annotation Title", è stato possibile attribuire a ognuna di esse un diverso *@xml:id* direttamente nell'interfaccia del software. In questo modo, una volta prodotto il file XML, è bastato copiare le varie **<zone>** generate in IMT all'interno del **<surface>** corrispondente.

Quest'operazione è stata ripetuta per tutte le 138 carte dei testi codificati, fino al completamento del **<facsimile>**.

Una volta predisposto il collegamento tra testo e immagine, è stato deciso di realizzare, a livello grafico, una corrispondenza di tipo “*hot spot*”, in modo da mostrare i *signa tabellionis* ed eventuali informazioni aggiuntive (non ancora stabilite) anche nella sezione di EVT contenente il testo codificato.

### 5.3.2 La pagina HTML

La pagina HTML con l’indice dei notai è stata realizzata con l’intenzione di fornire all’utente uno strumento di consultazione esterno alla visualizzazione in EVT in grado di catalogare una quantità maggiore d’informazioni senza però appesantire la resa grafica dell’edizione digitale. Attraverso le regole di trasformazione XSLT, applicate principalmente alla lista XML delle persone e dei *signa tabellionis*, è stato possibile creare un unico **<div>** da poter integrare nel tema WordPress del sito web. Nella sezione *Materiali* è stata predisposta, infatti, una pagina HTML vuota, dal titolo *I notai del Codice*, all’interno della quale poter inserire il **<div>** precedentemente creato e anche una breve introduzione all’argomento a cura di Enrica Salvatori.

Come già annunciato, oltre alla lista delle persone è stata sfruttata anche quella dei *signa tabellionis*, all’interno della quale sono stati raccolti tutti i simboli notarili presenti nei documenti codificati. La lista è composta da una serie di **<item>** collegati ai rispettivi **<ptr/>** del testo tramite l’attributo *@xml:id* e contenenti dei tag **<figure>** per catalogare le singole immagini. Ogni **<figure>** presenta tre sotto-elementi: un **<head>**, per identificare la tipologia del simbolo, un **<graphic/>**, per la collocazione fisica dell’immagine, e un **<figDesc/>**, per una breve descrizione dell’aspetto o del contenuto della figura (ancora da compilare).<sup>74</sup> Inoltre, alcuni **<figure>** dispongono di un attributo *@corresp*, al quale è stato assegnato un valore equivalente all’*@xml:id* del notaio legato a quel determinato simbolo notarile, in modo da creare una corrispondenza anche con gli elementi della **<listPerson>**.

Di seguito è riportata la struttura di un **<item>**:

---

<sup>74</sup> Text Encoding Initiative, <http://www.tei-c.org/release/doc/tei-p5-doc/it/html/ref-figDesc.html> (visitato il 23/01/2015), cit.

```

<item xml:id="numDocumento_st_numSignum">
  <figure corresp="#IdentificatoreNotaio">
    <head>(S.T.)</head>
    <graphic
      url="images/signum/numDocumento_st_numSignum.jpg"/>
    <figcaption/>
  </figure>
</item>

```

Il **<div>** integrato nel tema WordPress del sito contiene la lista cliccabile di tutti i notai e, per ognuno di essi, è possibile visualizzare il nome completo, la qualifica e tutte le immagini dei *signa* corrispondenti (v. fig. 9). Ogni simbolo notarile può essere visualizzato nelle sue dimensioni originali tramite uno script *lightbox* in jQuery. Inoltre, cliccando sul numero del documento associato ad ogni figura, sarà possibile aprire, in un'altra finestra, il testo corrispondente nella visualizzazione in EVT.

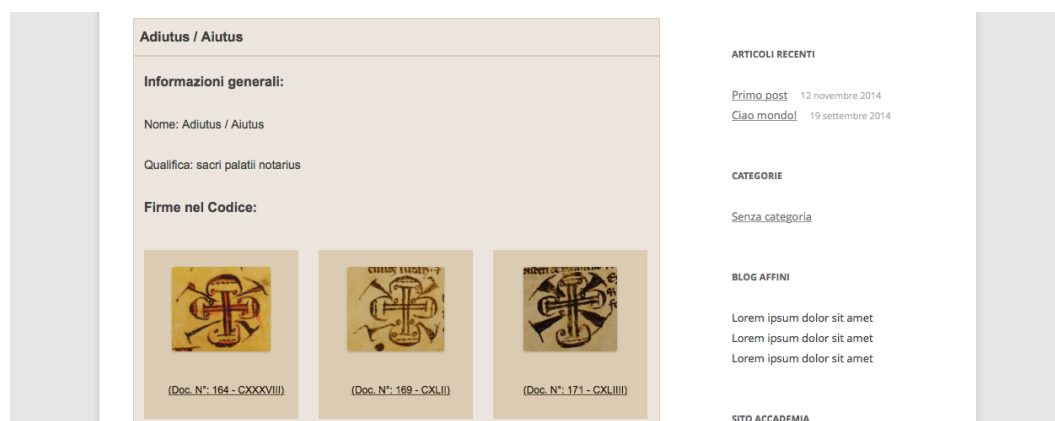


Figura 10. Una schermata della pagina

Per creare la lista dei notai e per collegare a ogni voce le varie informazioni, è stato necessario impostare, nel file XSLT, una serie di cicli *xsl:for-each* concatenati: il primo ha permesso di isolare e di compilare con i dati principali tutte le schede dei notai; il secondo ha associato ad ogni rogatario le immagini dei *signa* corrispondenti; il terzo ha attribuito ad ogni figura il link al documento nel sistema di consultazione EVT.

In coda al **<div>** sono stati inseriti due script jQuery, fondamentali per rendere la pagina dinamica e facilmente consultabile: il primo, servendosi della funzione *accordion()*, è stato utilizzato per rendere visualizzabili le singole schede

solamente al click sul pannello **<h3>** corrispondente; il secondo, invece, è servito per predisporre una riga di testo predefinita per tutti i notai sprovvisti di *signa tabellionis*.

Di seguito è riportata la struttura del primo script:

```
$j=jQuery;
$(document).ready(function() {
    $('#accordion').accordion( {
        collapsible: true;
        active: false;
        heightStyle: "content";
    } );
});
```

La funzione *accordion()*, richiamata sul **<div>** con attributo `id="accordion"`, attiva tutti gli **<h3>** contenenti i nomi dei notai e, al click su ognuno di essi, rende visibile il blocco immediatamente successivo, ovvero il **<div>** corrispondente alla scheda del determinato rogatario. Per mezzo delle tre opzioni dichiarate nelle righe successive del codice, è stato possibile impostare lo script nella maniera più corretta. Nello specifico, `collapsible` e `active` sono servite per chiudere tutte le schede al caricamento della pagina, mentre `heightStyle` ha permesso di ridimensionare in maniera dinamica i blocchi contenenti le informazioni.

Per inserire all'interno delle schede sprovviste di *signa tabellionis* il tag **<p>** contenente la frase «*Nessun timbro presente nei documenti*», è bastato impostare uno script in grado di verificare l'assenza di elementi nel **<div>** predisposto per la raccolta dei simboli. La funzione *trim()*, inserita nella riga di codice relativa al controllo, rimuove tutti gli spazi bianchi all'inizio e alla fine di una qualsiasi stringa di testo.<sup>75</sup>

Di seguito è riportata la struttura dello script:

```
$j=jQuery;
$(document).ready(function() {
    $('.timbro').each(function() {
```

---

<sup>75</sup> jQuery, <http://api.jquery.com/jquery.trim/> (visitato il 24/01/2015).

```
        if (!$j.trim($j(this).html()).length) {  
            $j(this).html('<p>Nessun timbro presente nei  
            documenti</p>')  
        }  
    } );  
} );
```

Lo script *lightbox*, per la visualizzazione dei *signa* a dimensione originale, è stato inserito come link direttamente nell’**<head>** della pagina WordPress, ed è stato richiamato sui vari **<img>** tramite l’attributo *@data-lightbox*, al quale è stato assegnato come valore l’*@xml:id* del simbolo di appartenenza.

Una volta generato il **<div>** contenitore, attraverso la funzione “*Configure Transformation Scenario(s)*” di *<oXygen/>*, è stato possibile integrarlo nella pagina dei notai predisposta dalla responsabile del sito, Chiara Mannari.

## 6 Conclusioni

L'evoluzione delle nuove tecnologie e degli strumenti informatici ha portato, negli ultimi anni, a un notevole mutamento nell'approccio ai contenuti umanistici, ridefinendo i metodi e le pratiche delle varie discipline. Inoltre, con l'avvento delle pubblicazioni digitali e con l'incremento d'accessibilità dei diversi contenuti, anche l'editoria e il ruolo stesso dell'editore hanno subito una serie di rinnovamenti. Nonostante ciò, le numerose barriere culturali<sup>76</sup> e le necessità di profondi cambiamenti hanno frenato, in parte, la creazione e la diffusione di questi progetti.

Com'è stato possibile osservare in queste pagine, il processo di digitalizzazione del Codice Pelavicino ha richiesto un impegno costante da parte delle varie persone impegnate e una serie di conoscenze specifiche difficilmente applicabili da una sola figura professionale, come uno storico o un informatico. Tuttavia, la complessa organizzazione del progetto ha permesso ai diversi gruppi e ai singoli individui di mettere in pratica le proprie competenze e, proprio grazie a questa collaborazione, è stato possibile individuare le varie esigenze e problematiche di percorso.<sup>77</sup> Inoltre, la realizzazione dell'edizione elettronica, richiedendo un "*uso creativo della tecnologia digitale per far avanzare la ricerca umanistica*"<sup>78</sup>, ha dato la possibilità a noi studenti di Informatica Umanistica di intervenire in più stadi della lavorazione del progetto.

A questo proposito, è interessante osservare come una delle fasi principali di tutto il processo di digitalizzazione, la codifica del manoscritto, riesca a rappresentare pienamente l'essenza pratica dell'informatico umanista. L'annotazione del testo, infatti, richiedendo una conoscenza approfondita sia dei contenuti umanistici da

---

<sup>76</sup> Tammaro, Anna Maria, 2012. *L'Informatica Umanistica e l'interdisciplinarietà*. In *Umanisti e risorse digitali*, Università di Parma.

<sup>77</sup> McCarty, Willard, 2012. *Collaborative Research in the Digital Humanities*. In *Collaborative Research in the Digital Humanities*, Farnham, Ashgate Publishing Limited, pp. 1-10.

<sup>78</sup> Gold, Matthew K., 2012. *The Digital Humanities Moment*. In *Debates in the Digital Humanities*, Minneapolis, University of Minnesota Press, Introduction.

digitalizzare sia degli strumenti e dei linguaggi informatici necessari per rielaborare le numerose informazioni, non può essere ricondotta solamente ad una delle due principali discipline di riferimento perché, di fatto, appartiene a entrambe. Per impostare la codifica del Codice Pelavicino, ad esempio, è stato necessario dapprima entrare in contatto con il manoscritto vero e proprio, per comprenderne la particolare conformazione e le caratteristiche distintive e, in un secondo momento, è stato possibile impostare il linguaggio e le eventuali soluzioni di annotazione, con particolare attenzione anche per le esigenze grafiche e strutturali delle fasi successive del progetto.

La codifica dei testi, dunque, può essere intesa anche come un sistema di organizzazione e di mediazione fra i contenuti e i risultati di un progetto più ampio, ed è proprio questo uno dei motivi per i quali può essere considerata una delle concretizzazioni più adatte per il lavoro dell'informatico umanista.

In conclusione, il progetto di digitalizzazione del Codice Pelavicino, oltre alle sue originali caratteristiche, ha permesso di mettere in luce anche un metodo di lavoro innovativo, improntato sulla collaborazione fra più persone e sull'integrazione fra le varie discipline: un approccio completamente diverso dai principi dell'editoria tradizionale. Questi aspetti distintivi, però, hanno in parte frenato la diffusione e la visibilità delle edizioni digitali, considerate ancora troppo complesse e difficili da realizzare. Da questo punto di vista, il software di visualizzazione EVT è stato sviluppato anche con l'intenzione di predisporre uno standard in grado di rendere più agevole la fase di pubblicazione. Pertanto, l'auspicio è che l'importanza delle edizioni digitali venga riconosciuta anche dalle dottrine più conservatrici, affinché queste pubblicazioni possano trovare un maggior riscontro nel panorama del web odierno.



## Bibliografia

- Alessandro Lenci, Simonetta Montemagni, Vito Pirrelli, *Testo e computer. Elementi di linguistica computazionale*, Roma, Carocci, 2005;
- Antonella Rovere, Dino Puncuh (a cura di), *I Libri Iurium della Repubblica di Genova: Introduzione*, Roma, Ministero per i beni culturali e ambientali, 1992.
- Blanke, Tobias, Elena Pierazzo, Peter A. Stokes, 2014. *Digital Publishing Seen from the Digital Humanities*. In *Logos*, vol 25, no. 2, pp. 16–27.
- C.M. Sperberg-McQueen, 1991. *Text in the Electronic Age: Textual Study and Text Encoding, with Examples from Medieval Text*. In “Literary and Linguistic Computing” 6 (1): pp. 34-46;
- Costamagna, Giorgio, 1962. *Note di diplomatica comunale. Il “Signum Comunis” e il “Signum Populi” a Genova nei secoli XII e XIII*. In *Miscellanea di Storia Ligure in onore di Giorgio Falco*, Milano, Feltrinelli Editore, pp. 105-115.
- Cunningham, Lawrence A., 2006. *Language, Deals and Standards: The Future of XML Contracts*. In *Research Paper*, N. 93, p. 6.
- Dekhtyar, Alex, Ionut E. Iacob, Jerzy W. Jaromczyk, Kevin Kiernan, Neil Moore, Dorothy Carr Porter, 2006. *Support for XML markup of image-based electronic editions*. In *International Journal on Digital Libraries*, Volume 6, pp. 55-69.
- Doug Tidwell, *XSLT: 2nd Edition*, Sebastopol, O’Reilly Media, 2008.
- Ettore Falconi, Roberta Peveri (a cura di), *Il Registrum Magnum del Comune di Piacenza*, Milano, Giuffrè, 1984-1997.
- Giorgio Costamagna, *Il notaio a Genova tra prestigio e potere*, Roma, 1970.
- Giorgio Costamagna, *Studi di paleografia e di diplomatica*, Roma, 1972.
- Lee, Dongwon, Wesley W. Chu, 2000. *Comparative Analysis of Six XML Schema Languages*. In *ACM SIGMOND Record*, Los Angeles.
- Lou Burnard, *What is the Text Encoding Initiative?*, Marsiglia, OpenEdition Press, 2014.

- McCarty, Willard, 2012. *Collaborative Research in the Digital Humanities*. In *Collaborative Research in the Digital Humanities*, Farnham, Ashage Publishing Limited, pp. 1-10.
- Michele Lupo Gentile, *Il regesto del Codice Pelavicino*. In *Atti della società ligure di storia patria*, Volume XLIV, Genova, 1912;
- Murata, Makoto, Dongwon Lee, Murali Mani, Kohsuke Kawaguchi, 2004. *Taxonomy of XML Schema Languages using Formal Language Theory*. In *ACM Journal Name*, Vol. V, No. N, pp. 1-3.
- Peter L. Shillingsburg, *From Gutenberg to Google*, Cambridge, Cambridge University Press, pp. 94-95.
- Pitti, Daniel V., 2004. *Designing Sustainable Projects and Publications*. In *A Companion to Digital Humanities*, Oxford, Blackwell.
- Podestà, Luigi, 1894. *I vescovi di Luni dall'anno 895 al 1289. Studi sul Codice Pelavicino nell'Archivio Capitolare di Sarzana*. In *Atti e Memorie della Reale Deputazione di Storia Patria per le Province Modenesi*, s.4, VI, pp. 5–157;
- Susan Schreibman, Ray Siemens, John Unsworth (a cura di), *A Companion to Digital Humanities*, Oxford, Blackwell, 2004.

### **Sitografia**

- Electronic Beowulf, <http://ebeowulf.uky.edu> (visitato il 09/01/2015).
- Git, <http://git-scm.com> (visitato il 05/01/2015).
- Le Petit Thalamus, <http://thalamus.huma-num.fr> (visitato il 20/01/2015).
- Monasterium, <http://monasterium.net/mom/home> (visitato il 20/01/2015).
- Telma, <http://www.cn-telma.fr> (visitato il 20/01/2015).
- Text Encoding Initiative, *A Gentle Introduction to XML*, <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SG.html> (visitato il 31/12/2014).
- Text Encoding Initiative, *Electronic Textual Editing: Digital Facsimiles in Editing*, [http://www.tei-c.org/About/Archive\\_new/ETE/Preview/kiernan.xml](http://www.tei-c.org/About/Archive_new/ETE/Preview/kiernan.xml) (visitato il 09/01/2015).
- Vercelli Book Digitale, <http://vbd.humnet.unipi.it> (visitato il 14/01/2015).

- W3C, *Extensible Stylesheet Language (XSL) Version 1.1*, <http://www.w3.org/TR/xsl/> (visitato l'11/01/2015).
- W3C, *XSL Transformation (XSLT) Version 1.0*, <http://www.w3.org/TR/xslt> (visitato l'11/01/2015).
- W3C Press Release, *The World Wide Web Consortium Issues XML 1.0 as a W3C Recommendation*, <http://www.w3.org/Press/1998/XML10-REC> (visitato il 31/12/2014).