

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE · SEDE DI BOLOGNA
DIPARTIMENTO DI INFORMATICA

CORSO DI LAUREA TRIENNALE IN INFORMATICA

APPLICAZIONI INFORMATICHE ALLA FILOLOGIA MODERNA

TESI DI LAUREA TRIENNALE IN
TECNOLOGIE WEB

RELATORE:
PROF.
FABIO VITALI

PRESENTATA DA:
PRIMIANO ARMINIO
CRISTINO

II SESSIONE - II APPELLO

ANNO ACCADEMICO 2018/2019

Questa pagina è lasciata intenzionalmente bianca.

Parole Chiave:

PhiloEditor

Edizione Critica Digitale

Filologia Moderna

Manzoni

Collodi

Indice

1	Introduzione	1
2	La filologia moderna	5
2.1	Filologia	5
2.2	Edizione Critica	8
2.3	Edizione Digitale	9
2.4	Edizione Critica Digitale	9
2.5	Studio delle edizioni critiche	10
2.6	Autori	12
2.7	Manzoni – I Promessi Sposi	12
2.8	Collodi – Pinocchio	13
3	PhiloEditor	15
3.1	Versioni Precedenti	15
3.1.1	PhiloEditor 2.0	15
3.1.2	PhiloEditor 3.0	20
3.2	Versione Attuale	24
3.2.1	PhiloEditor 6.0	24
4	Architettura di PhiloEditor	28
4.1	Tecnologie utilizzate	28
4.1.1	HTML	28
4.1.2	Linguaggi di Markdown	29
4.1.3	PHP	29
4.1.4	Javascript	30

4.1.4.1	Jquery	30
4.1.4.2	WikEd Diff	31
4.1.4.3	Bootstrap	31
4.2	Codice del progetto	32
4.2.1	Back-end	32
4.2.1.1	Registrarsi	32
4.2.1.2	Loggarsi	33
4.2.1.3	Cambiare password	33
4.2.1.4	Richiedere i documenti	33
4.2.1.5	Salvare i documenti	35
4.2.1.6	Condivisione documenti	36
4.2.1.7	Statistiche	36
4.2.2	Front-end	37
4.2.2.1	Caricamento della pagina	37
4.2.2.2	Diffing e Markdown	37
4.2.2.3	Caricamento e salvataggio dei documenti	38
4.2.2.4	Caricamento delle statistiche	38
4.2.2.5	Esportazione TEI	39
4.2.2.6	Barra di ricerca	39
4.2.2.7	Sicurezza e Crittografia	39
5	Valutazione	41
5.1	Confronto con i filologi	41
6	Conclusioni	43
6.1	Conclusioni e sviluppi futuri	43

Capitolo 1

Introduzione

Questa tesi nasce con lo scopo di fornire una piattaforma digitale per lo studio e la creazione di edizioni critiche in ambito filologico.

In particolare questo progetto, che prende il nome di PhiloEditor, è un punto iniziale per la creazione di un legame tra due discipline che sembrano agli antipodi una dall'altra: l'informatica e la filologia.

L'intersezione tra due campi disciplinari diversi, come in questo caso quello della filologia d'autore e quello dell'informatica umanistica, pone diversi problemi di definizione dei concetti utilizzati all'interno dei campi stessi.

Il concetto attorno al quale si sviluppa PhiloEditor è il confronto di varianti presenti in un testo nell'evolversi del tempo.

«Nel campo della filologia d'autore la variante è una modifica di qualcosa di già esistente: una forma, un'espressione o una fase che differisce da un'altra, dotata di senso, all'interno della stessa pagina, o in due testimoni diversi, siano essi manoscritti o a stampa.» [1]

Possedendo le versioni di un testo modificato nel tempo, che parte da una sua versione più primitiva e imperfetta a una versione corretta e riveduta, si ottiene non solo un testo oggettivamente più corretto, ma si ottiene una serie di informazioni necessarie alla comprensione del testo stesso, come

ad esempio i motivi che hanno spinto un autore a modificare il proprio manoscritto.

Questa possibilità di tenere traccia dei testi e delle varianti ci permette infatti di ottenere una rappresentazione stratigrafica di due testi, di due stadi dello stesso testo, superando la visualizzazione sinottica di differenti forme di testo.

«Questa visualizzazione schematica e riassuntiva permea la stragrande maggioranza delle edizioni critiche digitali in circolazione.

All'interno di questo nuovo modello concettuale, le marcature tipografiche e cromatiche, solitamente impiegate per individuare differenti testimoni, sono applicate quindi con profitto a diverse categorie variantistiche. Queste ultime possono essere distinte in categorie correttorie e metodologie correttorie. La combinazione della visualizzazione stratigrafica con la tassonomia variantistica permette quindi di ottenere una descrizione caratterizzante in senso sincronico e diacronico.» [1]

Ed è proprio in questo contesto che PhiloEditor si sviluppa, in un tentativo di avvicinamento di due campi presumibilmente agli antipodi.

PhiloEditor si propone come strumento della filologia moderna per la creazione di edizioni critiche digitali.

PhiloEditor è una piattaforma digitale online che permette nella sua versione attuale di confrontare i testi dei "Promessi Sposi", ventisettanta e quarantana e i testi di "Le avventure di Pinocchio. Storia di un burattino".

«In PhiloEditor le marcature tipografiche e cromatiche, solitamente impiegate per individuare differenti testimoni, sono applicate a diverse categorie variantistiche. Queste ultime possono essere distinte in categorie correttorie e metodologie correttorie.

La combinazione della visualizzazione stratigrafica con la tassonomia variantistica permette quindi di ottenere una descrizione caratterizzante in senso sincronico e diacronico.» [1]

La versione attuale della piattaforma permette di registrarsi, inserendo il proprio nome, cognome e scuola, e di creare la propria versione critica dei testi citati poc'anzi.

I testi vengono salvati direttamente nell'applicazione e ciascun utente può visionare le edizioni critiche di ogni fruitore della piattaforma.

Infine, è possibile scaricare il TEI di ciascun documento in formato xml.

Ogni documento che viene salvato è opportunamente inserito nella sezione "scuola" di cui l'utente fa parte.

Durante tutta la fase di creazione e sviluppo della versione attuale di PhiloEditor è avvenuto un confronto diretto con alcune studentesse del corso in filologia.

Durante i vari incontri si sono riscontrati i limiti della versione 3.0 della piattaforma, i quali hanno portato allo sviluppo di una nuova versione, con nuove funzionalità e caratteristiche.

Ad esempio si era riscontrata la mancata possibilità di cercare delle parole nel testo. I limiti della piattaforma non sono stati l'unico motivo per cui si è cercato di migliorare la versione preesistente.

Uno dei motivi principali per cui si è voluto aggiornare PhiloEditor era ed è il tentativo di rendere PhiloEditor più accessibile in ambito didattico filologico.

Questo ha portato a una revisione completa del codice back-end e ad un sostanziale cambiamento anche del codice front-end.

Uno dei cambiamenti principali apportati è senza ombra di dubbio la divisione netta tra i testi originali degli autori e i testi modificati dagli utenti, indi per cui, si è voluto dividere ogni utente per la propria scuola di appartenenza.

In questo modo, ogni utente è diviso per scuola e ha un quadro molto più chiaro delle modifiche che ciascun utente ha fatto nella propria versione.

Questo è dato dal fatto che le statistiche riguardanti le tipologie di metodologie/categorie correttive sono anch'esse divise per scuola. Così facendo si

possono chiaramente confrontare più scuole in base alle statistiche, ovvero in base alle modifiche che ciascun utente ha apportato al proprio testo.

Sebbene la versione attuale è di fatto già un'ottima piattaforma per la creazione di edizioni critiche, essa però non è perfetta.

Vi sono tante aggiunte e modifiche che potranno essere fatte in futuro, come ad esempio la possibilità di creare utenti normali e utenti speciali, differenziati dal fatto che solo questi ultimi possono modificare i testi e salvarli sulla piattaforma.

Un'altra possibile aggiunta potrebbe essere quella di poter far scegliere all'utente le categorie correttive che gli interessano, o magari di crearne di proprie.

Non voglio soffermarmi troppo sugli esempi di possibili aggiunte ma più sulle potenzialità della piattaforma stessa.

Oltre ad aver ristrutturato completamente il codice back-end del sito, si è cercato di renderlo più dinamico di quanto non fosse, come il fatto che l'inserimento dei testi nell'apposita cartella nel server avviene tutto in automatico senza dover mettere mano al codice. Una volta inseriti i testi di un autore in una apposita cartella, questi, quindi, sono già disponibili sul sito. Con questa facilità di lettura da parte della piattaforma di nuovi testi, trovo molto plausibile che in un possibile futuro si cerchi di rendere dinamica la piattaforma ancora di più.

Capitolo 2

La filologia moderna

Per poter parlare di PhiloEditor in modo corretto e coerente abbiamo bisogno di definire gli aspetti fondamentali su cui esso si basa. L'aspetto primario è ovviamente la filologia, senza la quale non avrebbe nemmeno senso di esistere PhiloEditor stesso. Per poter affrontare e comprendere il problema che PhiloEditor cerca di risolvere è necessario dunque dare delle brevi definizioni dei sopracitati aspetti fondamentali.

2.1 Filologia

«Filologia è quella onorevole arte che esige dal suo cultore soprattutto una cosa, trarsi da parte, lasciarsi tempo, divenire silenzioso, divenire lento, essendo un'arte e una perizia da orafi della parola, che deve compiere un finissimo attento lavoro e non raggiunge nulla se non lo raggiunge lento. Ma proprio per questo fatto è oggi più necessaria che mai; è proprio per questo che essa ci attira e ci incanta quanto mai fortemente, nel cuore di un'epoca del "lavoro": intendo dire della fretta, della precipitazione indecorosa e sudaticcia, che vuol "sbrigare" immediatamente ogni cosa (...). Per una tale arte non è tanto facile sbrigare qualsiasi cosa perché essa ci insegna a leggere bene, cioè a leggere lentamente in profondità, guardandosi avanti

e indietro, non senza secondi fini, lasciando porte aperte, con dita e con occhi delicati.» [2]

Questa è la definizione che è stata proposta dal filosofo, nonché filologo, poeta e saggista Friedrich Nietzsche nel suo scritto *Aurora. Pensieri sui pregiudizi morali* nel 1881.

Ovviamente non è l'unica definizione che si può attribuire alla filologia. Di fatto, non è facile dare una definizione univoca di ciò che è la filologia.

Secondo l'accezione comune attuale, la filologia è dunque un insieme di discipline che studia i testi di varia natura (letterari, storici, politologici, economici, giuridici, ecc.), da quelli antichi a quelli contemporanei, al fine della ricostruzione della loro forma originaria attraverso l'analisi critica e comparativa delle fonti che li testimoniano, e pervenire, mediante varie metodologie di indagine, ad un'interpretazione che sia la più corretta possibile.

Non si può però parlare di filologia senza parlare della figura che più la rappresenta: il filologo. Il filologo studia i testi nella loro evoluzione storica, ricercandone modifiche e cambiamenti. Il passaggio tecnologico tra rotolo, codice, pergamena e carta comporta una necessaria selezione dei testi da riversare nei nuovi supporti, portando a scelte inevitabili dettate dall'interesse e dall'utilità del testo in questione all'epoca; così, molte opere si perdono perché non più riprodotte.

Il procedimento filologico consiste nel tentare di risalire con criteri meccanico-probabilistici e linguistico-formali alla forma originaria di un testo, inevitabilmente corrotta dalla serie di copie che lo hanno tramandato dall'antichità ad oggi.

È poi necessario riscoprire il significato originario del testo e l'intenzione dell'autore. Il filologo deve perciò conoscere in modo approfondito storia e lingua della civiltà nel contesto di studio, e deve avere accesso a fonti e documenti delle epoche in esame, per poter restaurare in modo corretto

l'interpretazione di un passo, una parola o un'opera, argomentando la scelta fatta.

Stabilito che l'oggetto di studio sono i testi, scritti in una lingua data, e considerati, sotto ogni aspetto, come parte integrante della relativa civiltà, gli studi filologici si dividono in varie branche, tra queste cito brevemente la filologia classica e moderna.

«La filologia classica è la scienza che studia i documenti scritti in lingua greca o latina, in vista della retta edizione e della corretta interpretazione di essi nel contesto dell'epoca e della civiltà che li ha prodotti. Purtroppo, di questa produzione e di gran parte di quella seguente, ad esclusione di autori e di opere particolari, rimane in realtà ben poco, spesso pochi frammenti e le citazioni degli studiosi successivi, che, tuttavia, sono più che sufficienti per rendere conto della qualità e dell'importanza degli autori.» [3]

Nel caso della filologia classica, il filologo deve ricercare l'archetipo del testo stesso, nella sua forma originale. Questo è ovviamente un compito molto complesso, in quanto è difficile stabilire se un testo antico è nella sua forma più pura e vicina al testo dell'autore originale stesso. Compito del filologo è dunque quello di evidenziare le linee correttive dell'autore attraverso la catalogazione delle varianti.

Nel caso, invece, della filologia moderna, il filologo invece risulta essere molto più avvantaggiato rispetto al filologo classico, in quanto conosce con certezza sia la versione originale di un testo d'autore sia le sue versioni successive.

Il filologo moderno dunque non ha bisogno di ricercare l'archetipo di un testo e di risalire ad una versione quanto più possibile vicina all'originale; i testi per il filologo moderno sono già a portata di mano.

Il suo scopo è quello di immedesimarsi nell'autore dell'opera, di comprendere i motivi per cui l'autore ha deciso di adoperare un cambiamento nel suo

manoscritto, una variante. Il filologo moderno, quindi, deve entrare nella psiche dell'autore e comprendere e percepire ogni piccolo cambiamento che ha portato l'autore dell'opera a cambiare totalmente o parzialmente una parte della sua opera.

Nel caso di PhiloEditor, la piattaforma risulta utile nella creazione di un'edizione critica, e nel nostro caso specifico, di un'edizione critica digitale.

2.2 Edizione Critica

«L'edizione critica di un testo letterario è il risultato di una serie di operazioni condotte con metodo scientifico, ossia verificabile e dimostrato, che mirano a stabilire, secondo l'ipotesi più economica, la forma del testo in oggetto più vicina possibile alla volontà dell'autore. Il concetto di testo è un concetto dinamico, dal momento che esso subisce variazioni e modifiche, sia da parte dell'autore sia da parte dei copisti durante la trasmissione nel corso del tempo. Le procedure di ricostruzione critica avvengono in due momenti: il primo documentativo, con la raccolta di tutte le testimonianze esistenti del testo stesso, dirette (manoscritti, stampe) e indirette (citazioni presenti in altre opere); il secondo interpretativo, nel quale, attraverso procedimenti differenziati a seconda se si tratta di testi antichi o di testi moderni e contemporanei, si fissa il testo e si procede, ove necessario, al restauro testuale.» [4]

Come si può comprendere da tale esposizione, l'edizione critica, quando è buona, costituisce una garanzia ma non raggiunge la verità assoluta, perché è il risultato di una probabilità che resta aperto a ogni ulteriore verifica o rettifica: è, come ogni atto scientifico, una mera ipotesi.

L'edizione critica si rivolge a tutti i lettori, tanto agli specialisti della disciplina cui appartiene il testo in oggetto quanto al lettore comune. Il lettore comune probabilmente non sa che cosa sia un'edizione critica. Tuttavia

il risultato del lavoro filologico lo riguarda direttamente, perché da quello dipende la qualità del testo che egli legge e che a volte risulta profondamente diverso da altre edizioni.

In conclusione un'edizione critica è un punto d'arrivo del lavoro di disamina e vaglio critico della tradizione ed è il risultato della fase di emendamento degli errori attribuibili all'archetipo. L'edizione critica punta al recupero della lezione originale o di una lezione vicina quanto possibile a quella originale.

2.3 Edizione Digitale

«Si può definire edizione digitale qualsiasi rappresentazione del testo su supporto elettronico, tanto costruita direttamente su tale supporto, mediante linguaggi di codifica (edizione born digital) quanto digitalizzata da fonti cartacee mediante appositi software. Per le sue caratteristiche (facilità di duplicazione, interrogazione, confronto con altre simili), nell'edizione digitale convivono finalità di rappresentazione / interpretazione e di conservazione / archiviazione, spesso con una prevalenza di queste ultime.» [5]

Gli strumenti digitali permettono un allargamento del campo, una visione d'insieme, un'osservazione sintetica dell'ampia rete che lega i diversi testi, ma anche l'avvicinamento estremo all'oggetto di studio, l'analisi ravvicinata se non microscopica, attraverso la riproduzione ad altissima risoluzione dell'originale.

2.4 Edizione Critica Digitale

L'edizione critica digitale unisce sia l'edizione critica che l'edizione digitale cercando di mantenere quanto più possibile i punti cardine da entrambe le parti.

Questo sta a significare che tramite supporti digitali, quali in questo caso

PhiloEditor, si tenta di creare un'edizione critica utilizzando una piattaforma digitale o un supporto/strumento digitale.

2.5 Studio delle edizioni critiche

Date le definizioni sopra citate, siamo in grado di contestualizzare le piattaforme come PhiloEditor e simili.

Infatti, sono sempre più numerosi i progetti di Edizioni Critiche Digitali che permettano di studiare, per mezzo di sistemi open source, il processo di genesi ed evoluzione delle opere attraverso la rappresentazione ed interpretazione delle varianti d'autore. Da più di un ventennio, infatti, il web ha offerto infinite possibilità per la visualizzazione e lo studio dei manoscritti, la loro trascrizione ed edizione critica, ma anche per la marcatura delle stratigrafie correttive, ultima frontiera della filologia d'autore. È tuttavia un paradosso che, nell'era della comunicazione integrata e della costruzione collettiva della conoscenza, i progetti sviluppati, in Italia e all'estero, in contesti di eccellenza e che hanno visto la partecipazione di filologi e informatici in fruttuosa collaborazione, non siano sempre stati partecipati e condivisi, sviluppando realtà spesso isolate tra loro e prodotti non integrati. Il collegamento tra questi progetti permetterebbe invece di mettere a confronto metodi e realizzazioni digitali e procedere nella risoluzione comune di problemi generali e singoli casi di studio.

Che si tratti di Manzoni, Collodi, Nietzsche, Leopardi, e altri ancora, i limiti di una rappresentazione lineare appaiono sempre più che evidenti. L'edizione digitale allora, più che un sistema per abbattere limiti di spazio e costi, più che una riproduzione su schermo della sua controparte cartacea, più che un tentativo di semplificare la fruizione dei testi nella loro complessità genetica e critica, rappresenta un'opportunità unica per ripensare l'idea di testo e la visione tradizionale del processo di scrittura. Questo perché, ciò a cui essa punta, non è dare un'unica versione dell'opera, per quanto "finale", "definitiva" o rappresentativa della *voluntas auctoris*, bensì informare su un processo, dimostrare il

valore del testo nella sua non-linearità e nel suo farsi, illuminare, e non nascondere, quelle “sfilacciate” che i documenti a volte, e non così raramente, lasciano intravedere. [6]

Da un lato, dunque, c'è l'accesso alla dimensione diacronica del testo, al susseguirsi di campagne correttive, ognuna delle quali identifica uno stadio testuale; dall'altro, c'è la coesistenza sincronica di due testi, creata dalla presenza di varianti alternative, che non possono essere gerarchicamente disgiunte, separate, ma rappresentano un prezioso momento di tensione, di approssimazione progressiva, e necessariamente incompleta, verso la perfezione. Ignorando, o riducendo, attraverso un'inadeguata rappresentazione, il valore di questi luoghi, si rinuncerà a conoscere quello “stupendo punto di equilibrio” in cui l'autore “ha lasciato sussistere due varianti nel quale vivono separati elementi che non ha potuto comporre dialetticamente; e benché ognuna da sé riveli un'insufficienza, solo il confronto determina quale”. [7]

L'avvento del digitale ha riaperto la mai sopita discussione sui pro e contro delle edizioni critiche. Infatti, le edizioni critiche propongono al lettore un testo autorevole, frutto di un lavoro critico e di un'attenta valutazione delle fonti documentarie; allo stesso tempo, però, tendono a semplificare troppo certe complessità testuali. Quest'ultimo è un problema maggiormente sentito per testi con folte tradizioni, necessariamente contaminate e caratterizzate da grande varietà linguistica, così anche per testi frammentari e/o incompiuti: l'edizione critica tende a mascherare la complessità della testimonianza documentaria, dando a volte solo sommariamente conto del lavoro editoriale. Oggetto di questo intervento è di proporre una riflessione sui limiti delle edizioni critiche tradizionali, in particolare, per l'analisi linguistica e di mostrare come le edizioni digitali possono ovviare a tale limite.

«L'edizione digitale, infatti, permette da un lato di registrare lo stato del documento fonte a un livello di dettaglio inimmagi-

nabile alla trascrizione condotta su un qualunque programma di video scrittura, in quanto consente di registrare, di correggere e di annotare allo stesso tempo, dall'altro lato consente al lettore di visualizzare il testo in vari formati, per esempio in modo diplomatico o critico, e dall'altro lato ancora si presta a analisi statistiche e quantitative del testo. I testi sono stati estensivamente codificati da un punto di vista linguistico e ortografico, secondo il metodo descritto sopra, il che ha consentito di quantificare il peso dell'intervento di normalizzazione compiuto dall'editore, con risultati se non sorprendenti, per lo meno tali da stimolare la riflessione critica sull'argomento.» [8]

Questi cambiamenti hanno, inoltre, come conseguenza un cambio di percezione di che cosa sia il testo anche da parte del lettore, che può quindi venire confrontato con le problematiche della trasmissione testuale.

2.6 Autori

PhiloEditor attualmente permette di creare edizioni critiche su *I Promessi Sposi* di Manzoni e su *Storie di un burattino* (Pinocchio) di Collodi.

2.7 Manzoni – *I Promessi Sposi*

Il 17 settembre del 1823 Manzoni termina di redigere la prima minuta del romanzo, il cosiddetto *Fermo e Lucia*, iniziata nell'aprile del '21. [9]

Sin da subito si rende conto del parziale fallimento dell'operazione, per ragioni legate sia alla produzione letteraria sia al lessico utilizzato. Alla fine del '23 Manzoni comincia quindi a rivedere e a riscrivere la prima minuta, e a scrivere una seconda introduzione in cui mostra la profonda insoddisfazione per il risultato ottenuto con la prima stesura. La seconda minuta si presenta come un testo singolare, un manoscritto che riutilizza molti fogli della prima e la cui stesura e revisione si intrecciano con il processo di stampa della prima edizione dell'opera, la cosiddetta *Ventisettana* (1825-1827): un

testo in cui Manzoni si impegna in un confronto con la lingua, ricorrendo, da un certo punto in poi, ad un massiccio uso del Vocabolario della Crusca e alla ricerca di testi di autori toscani nei quali confida ancora di trovare una soluzione ad un problema impossibile da risolvere. Il confronto del testo e dell'apparato critico con le postille apposte da Manzoni in margine alla propria copia della Crusca 'veronese' mostra infatti il progressivo instaurarsi di termini ed espressioni certificati linguisticamente dalla Crusca, pur cercando un avvicinamento alla Ventisettana. Questa tendenza si consolida in particolar modo dopo la decisione di accantonare la prima introduzione alla seconda minuta, in cui l'autore giustificava il ricorso necessario ai lombardismi. Dunque Manzoni, dopo l'esperimento del Fermo e Lucia, caratterizzato dall'impiego dei lombardismi, sceglie una lingua ibrida: un italiano comune di area settentrionale, appesantito tuttavia da un lessico letterariamente connotato, prelevato dal dizionario e dagli autori della tradizione toscana. Per arrivare all'edizione definitiva dell'opera, la cosiddetta Quarantana (1840-1842), Manzoni corregge pazientemente e puntualmente singole parole o espressioni fraseologiche, lavorando su una copia tipografica della Ventisettana. Ne risulta quindi un testo pieno di varianti sostitutive, di piccole innovazioni, di correzioni dell'ultimo minuto. Il risultato è quello di un romanzo scritto in una lingua realmente esistente, il fiorentino parlato dalla borghesia colta del suo tempo.

2.8 Collodi – Pinocchio

Nel febbraio del 1883, a poche settimane di distanza dalla fine delle puntate uscite nel «Giornale per i bambini» *Le Avventure di Pinocchio* venivano pubblicate in volume dalla casa editrice Paggi di Firenze. [10]

Il libro fu subito esposto nelle vetrine della Libreria Paggi. Il volumetto faceva parte della sezione dei «Libri per letture» della collana «Biblioteca scolastica» dell'editore Paggi.

Pinocchio aveva esordito sulle pagine del periodico «Giornale per i bambini», rivista per l'infanzia stampata a Roma. Il racconto "Storia di un burattino" uscì a puntate a partire dal primo numero il 7 luglio 1881. La storia si sviluppò in quindici capitoli e Collodi decise di farla finire con Pinocchio

impiccato al ramo della Quercia grande: «E non ebbe fiato per dir altro. Chiuse gli occhi, aprì la bocca, stirò le gambe e, dato un grande scrollone, rimase lì come intirizzito». Il successo del burattino portò la redazione del giornale a chiedere all'autore di continuare il racconto e nel febbraio 1882 nella rivista fu pubblicata questa nota: «Una buona notizia! Vi ricordate del povero burattino che il signor Collodi lasciò attaccato a quell'albero e che pareva morto. Ebbene, ora lo stesso signor Collodi ci scrive per annunciarci che Pinocchio non è morto, anzi è più vivo che mai, e che gli sono accadute delle cose che pare impossibile [...]». La storia riprese con un nuovo titolo, le Avventure di Pinocchio, che si prolungarono per quasi un anno.

Collodi stipulò nel dicembre 1882 il contratto con il libraio fiorentino Felice Paggi per la pubblicazione in volume delle Avventure quando ancora uscivano a puntate sul «Giornale». Ovviamente, fu necessaria una revisione completa del racconto per dargli l'aspetto di una narrazione omogenea, senza i salti o le ripetizioni che si erano generate per gestire il testo a puntate. Questa operazione di editing fu sicuramente opera di Collodi stesso, come è stato dimostrato dagli studiosi: spariscono dunque varie premesse e sunti iniziali mentre vengono aggiunti i sommari al principio di ogni capitolo. La revisione non fu del tutto accurata: alcuni critici hanno notato come l'autore abbia lasciato, secondo Marino Parenti intenzionalmente, molte incongruenze o errori nella trama. Il libro ebbe subito un buon riscontro di pubblico ma la fortuna editoriale di Pinocchio si svilupperà soprattutto nel Novecento tanto da arrivare al secondo milione di copie vendute, come riportato nell'edizione Bemporad (successori di Paggi) del 1921. Il successo di Pinocchio fu dovuto in parte all'indiscutibile valore letterario del testo, di cui però ci si renderà conto successivamente, che, ai pregi della lingua, univa quelli di uno stile in buon equilibrio tra il reale e il fantastico, ma soprattutto all'opportunità che dette Collodi ai suoi piccoli lettori di riconoscere nell'iter faticoso del burattino che diventa ragazzo 'perbene' la loro esperienza di vita.

Capitolo 3

PhiloEditor

3.1 Versioni Precedenti

Prima di passare al prossimo capitolo, dove affronteremo la versione attuale di PhiloEditor, passerò in rassegna due delle versioni che hanno preceduto la versione attuale: PhiloEditor 2.0 e PhiloEditor 3.0

3.1.1 PhiloEditor 2.0

Il prototipo PhiloEditor 2.0 è un'applicazione Web che supporta gli studiosi nell'analisi filologica di testi letterari, permettendo loro di leggere e annotare varianti. Il prototipo permette di confrontare due versioni de *I Promessi Sposi*. [11]

PhiloEditor 2.0 usa internamente uno strumento di diff, ossia un software in grado di processare due varianti di un testo, individuarne le differenze e organizzarle in un elenco di cambiamenti chiamato delta. Uno degli aspetti cruciali nella produzione del delta è la granularità delle operazioni individuate e l'accuratezza del risultato. Nel campo dell'ingegneria del software, ad esempio, il delta è organizzato molto spesso in cambiamenti su linee di codice: una linea di un programma è, infatti, l'unità atomica significativa per gli sviluppatori. In altri casi, invece, può essere utile individuare le differenze a livello di carattere o di strutture principali come capitoli, sezioni, paragrafi, ecc.

Sui testi letterari si ottengono ottimi risultati usando la parola come unità atomica nel calcolo delle differenze. PhiloEditor 2.0 usa questo approccio attraverso la libreria "WikiEd diff:3", il modulo prende in input due varianti in formato testuale e produce il delta come sequenza di operazioni di inserimento e cancellazione di singole parole. Inoltre, attraverso un'analisi successiva del delta, il sistema è in grado di accorpate coppie di cancellazioni e inserimenti giustapposti. Prendendo un esempio da *I Promessi Sposi*: la modifica della parola "lascian" in lasciano combina due sotto-operazioni: la cancellazione di "lascian" e l'inserimento di "lasciano" nella stessa posizione.

Le differenze individuate dal modulo di diff sono mostrate agli utenti in modo chiaro e immediato, attraverso un'interfaccia che permette loro di evidenziare e filtrare i cambiamenti. La peculiarità di questa visualizzazione è l'uso di un approccio stratificato: piuttosto che mostrare le due varianti del testo una di fianco all'altra, PhiloEditor 2.0 mostra il testo della prima variante ed evidenzia le parole modificate. Per ogni parola, l'interfaccia mostra sia il testo originale che quello modificato in sovrapposizione. La vicinanza spaziale delle due versioni di ogni parola permette al lettore di individuare immediatamente le differenze e di confrontarle in modo più veloce.

PhiloEditor 2.0 permette inoltre di classificare i cambiamenti, in modo da creare interpretazioni e analisi personalizzate e condividerle facilmente con altri lettori. Gli utenti abilitati, infatti, possono classificare ogni cambiamento individuato automaticamente dal motore di diff. Se l'individuazione dei cambiamenti è un'operazione oggettiva, infatti, la loro classificazione è soggettiva.

PhiloEditor 2.0 fornisce gli strumenti per annotare i cambiamenti in base al proprio modello di riferimento, ai propri obiettivi e alla propria competenza e sensibilità. Nel prototipo sono state pre-caricate alcune classificazioni specifiche, ma l'applicazione è indipendente e può essere estesa per supportare analisi diverse. Tutte le modifiche classificate nello stesso modo sono visualizzate con lo stesso colore di sfondo, in modo da renderle facilmente individuabili dai lettori. L'interfaccia permette inoltre di filtrare le modifiche in base alla tipologia di appartenenza e di mostrarne solo alcune. Il

sistema, infine, include alcuni pannelli che riassumono dati statistici sulle modifiche mostrando, ad esempio, quante modifiche di ogni tipo ci sono in ogni documento o come sono distribuite su tutti i documenti. Questi strumenti permettono, quindi, agli studiosi di accedere velocemente a dati aggregati e fare ulteriori indagini sui testi.



Figura 3.1: Statistiche PhiloEditor 2.0

In PhiloEditor 2.0 abbiamo la possibilità di marcare ciascuna coppia di varianti presente nel testo associandone una metodologia o una categoria correttoria, o entrambe, utilizzando il ‘pacchetto’ preconstituito di marcature

tipografiche e cromatiche.



Figura 3.2: Versioni e Stili PhiloEditor 2.0

Quel ramo del lago di Como che volge a mezzogiorno tra due catene non interrotte di monti, tutto a seni e a golfi, a seconda dello sporgere e del rientrare di quelli, **vien**<[viene] quasi a un tratto a restringersi e a prender corso e figura di fiume, tra un promontorio a destra, e un'ampia **costiera dall'altra parte**<[riviera di rincontro]; e il ponte, che ivi congiunge le due rive, par che renda ancor più sensibile all'occhio questa trasformazione, e segni il punto in cui il lago cessa, e l'Adda **ricomincia**<[ricomincia], per ripigliar poi nome di lago dove le rive, allontanandosi di nuovo, **lascian**<[lasciano] l'acqua distendersi e **rallentarsi**<[allentarsi] in nuovi golfi e in nuovi seni. La **costiera**<[riviera], formata dal deposito di tre grossi torrenti, scende appoggiata a due monti contigui, l'uno detto di san Martino, l'altro, con voce lombarda, il **Resegone**,<[] dai molti suoi cocuzzoli in fila, che in

Figura 3.3: Testi a Confronto PhiloEditor 2.0

Il giallo indica l'abbassamento stilistico (riviera » costiera; dai bastioni » di su le mura; rispondono verso » guardano a), il blu i casi di toscanizzazione (viene » vien; ricomincia » rincomincia; allentarsi » rallentarsi), il rosso le varianti grafiche (Resegone » Resegone). L'azione interpretante dello studente – o dello studioso – non si esaurisce nella scelta del tasto da schiacciare. Per assegnare la corretta marcatura a ciascuna coppia di varianti, è necessario preliminarmente reperire le informazioni riguardanti i fenomeni linguistici e stilistici tornando agli strumenti tradizionali della ricerca letteraria.

Inoltre, è possibile risegmentare autonomamente le varianti erroneamente generate dal delta, in accordo con il principio di comparabilità delle varianti. Infine, generando istogrammi e diagrammi a torta che mostrano l'incidenza di un singolo fenomeno, l'utente può misurare la prossimità, o viceversa la distanza, tra questa rappresentazione quantitativa dell'iter linguistico del romanzo nelle sue due ultime stazioni testuali e la bibliografia specifica sull'argomento. PhiloEditor 2.0, oltre ad avere – come si è visto – interessanti potenzialità analitiche e didattiche, è applicabile ad altri casi la cui storia genetica presenti puntuali varianti d'autore, manoscritte o a stampa, eventualmente dopo aver modificato le tipologie di marcatura a seconda del testo in oggetto (e. g. categorie argomentative, narratologiche etc.).

3.1.2 PhiloEditor 3.0

PhiloEditor 3.0 è una estensione della versione precedente. La principale novità di questa versione è l'aggiunta di Pinocchio al comparto letterario presente nella piattaforma e la possibilità di esportare i testi in formato TEI. La Text Encoding Initiative (TEI) è un consorzio di istituzioni internazionali, di ambito linguistico e letterario, che ha sviluppato uno standard per la rappresentazione dei testi in formato digitale. Scopo della TEI è quello di sviluppare e mantenere una serie di linee guida di alta qualità per la codifica di testi umanistici e per sostenere il loro uso da parte di istituzioni, comunità di progetti e singoli individui. Il consorzio ha sede presso l'Institute for Advanced Technology in the Humanities dell'Università della Virginia, negli Stati Uniti d'America.

Attraverso le cosiddette Guidelines for Electronic Text Encoding and Interchange, la TEI definisce un linguaggio di markup (in XML) per la digitalizzazione dei testi, utile, in particolar modo, per coloro che intendono costituire archivi e banche dati testuali. L'idea di fondo era quella di creare uno standard per digitalizzare i testi letterari (in particolare testi antichi), per poterli conservare nel tempo in maniera efficiente. Dalla codifica digitale si ottengono anche altri benefici, come la portabilità dei testi, la facilità di archiviazione e la facilità di gestione attraverso gli strumenti informatici. [12] In PhiloEditor 3.0 è possibile esportare un documento modificato in un file di tipo XML. Al suo interno si possono evidenziare le differenze che intercorrono tra le due versioni di testo.

```

<?xml version="1.0"?>
· <TEI xmlns="http://www.tei-c.org/ns/1.0">
  - <teiHeader>
    - <fileDesc>
      - <titleStmt>
        <title>Le avventure di Pinocchio.Storie di un burattino</title>
        <author>Carlo Collodi</author>
      </titleStmt>
      - <editionStmt>
        <edition>Edizione collaborativa</edition>
        - <respStmt xml:id="Teresa">
          <resp>Codifica del capitolo 1</resp>
          <name>Teresa Gargano</name>
        </respStmt>
      </editionStmt>
      - <publicationStmt>
        <publisher>Teresa</publisher>
        <pubPlace>Nessuna</pubPlace>
        <date>2017</date>
        - <availability>
          <licence>Roma</licence>
        </availability>
      </publicationStmt>
      - <sourceDesc>
        - <bibl xml:id="vVolume">
          <author>Ornella Castellani Pollidori</author>
          <title>Le avventure di Pinocchio / Carlo Collodi ;
            edizione critica a cura di Ornella Castellani Pollidori</title>
          <pubPlace>Pescia,PT</pubPlace>
          <publisher>Fondazione nazionale Carlo Collodi</publisher>
          <date>1983</date>
        </bibl>
        - <bibl xml:id="VRivista">
          <author>Carlo Collodi</author>
          <title>Storie di un burattino</title>
          <pubPlace>Roma</pubPlace>
          <publisher>Giornale per bambini</publisher>
          <date>1881-1883</date>
        </bibl>
      </sourceDesc>
    </fileDesc>
  - <encodingDesc>

```

Figura 3.4: TEI PhiloEditor 3.0 parte 1

```

    <p>Marcatura basilare delle varianti interduntive.
    delle precisazioni scolastiche, innalzamento espressivo, varianti grafiche</p>
  </encodingDesc>
  - <revisionDesc>
    <change who="#Teresa" when="2017-04-16"/>
    [...]
    <change who="#Teresa" when="2017-04-16"/>
  </revisionDesc>
</teiHeader>
- <text>
  - <body>
    - <div xml:id="Capitolo 1" type="chapter">
      - <p xml:id="p1">
        - <app type="replace-insert">
          <rdg resp="#void" source="#vRivista"> </rdg>
          <rdg resp="#void" source="#vVolume"> Come andò
          che Maestro Ciliegia, falegname, trovò un pezzo di legno,
          che piangeva e rideva come un bambino </rdg>
        </app>
      </p>
    </div>
    [...]
    - <p xml:id="p18">
      - <app type="replace">
        <rdg resp="#void" source="#vRivista"/>
        <rdg resp="#void" source="#vVolume"/>
      </app>
    </p>
  </div>
</body>
</text>
</TEI>

```

Figura 3.5: TEI PhiloEditor 3.0 parte 2

Una prima parte iniziale mostra le informazioni riguardanti l'autore delle modifiche e le informazioni sul testo modificato tra cui: titolo, data di pubblicazione, publisher, edizione. La seconda parte, invece, mostra le differenze di testo tra una versione e l'altra con le apposite marcature.

La struttura di PhiloEditor 3.0 migliora il tutto in piccoli dettagli, cambiando anche il layout finale del sito, creando tre colonne di cui la prima contenente la lista delle opere e i documenti modificati dagli utenti, la seconda in cui viene caricato il testo e la terza contenente le modifiche che è possibile

apportare al testo.

The screenshot displays the PhiloEditor 3.0 web interface. At the top, a dark navigation bar contains the text "PhiloEditor 3.0" and menu items: "INFO", "AIUTO", "STATISTICHE", and "ESPORTA TEI". On the right side of this bar, it says "BENVENUTO, PRIMIANO CRISTINO".

The main content area is divided into three vertical sections:

- Left Sidebar (OPERE DISPONIBILI):** Lists various works under categories like "I PROMESSI SPOSI" and "LE AVVENTURE DI PINOCCHIO". "CAPITOLO 1" is highlighted in green.
- Center Text Editor:** Displays the text of "Capitolo I" from "Come andò che Maestro Ciliegia, falegname, trovò un pezzo di legno, che piangeva e rideva come un bambino". The text includes several paragraphs with some words highlighted in blue (e.g., "accendere il fuoco e per").
- Right Panel (VERSIONI E STILI):** Contains a "VISTA" and "MODIFICA" section, followed by buttons for "RIVISTA", "VOLUME", "VERT.", and "ORIZZ.". Below this is a "Metodologie correttorie" section with a list of checked options: "Inserimento", "Cancellazione", "Ordine delle parole", "Ridondanza sintattica", "Ridondanza sintattica", "Ridondanza sintattica", "Precisioni scolastiche", "Ridondanza semantica", "Ripetizioni", "Toscanismi", "Varianti grafiche", and "Varianti interpuntive".

At the bottom of the interface, a copyright notice reads: "© 2016 Fabio Vitali, Gioia Donati, Maria Luce Bottazzo, Gianmarco Spinaci Alma Mater Università di Bologna".

Figura 3.6: PhiloEditor 3.0

3.2 Versione Attuale

3.2.1 PhiloEditor 6.0

PhiloEditor 6.0 è la versione attuale della piattaforma.

In questa versione sono state aggiunte e cambiate diverse funzionalità presenti nelle precedenti versioni.

Non appena si apre la pagina risultano immediate le differenze con le versioni precedenti.

Prima tra tutte l'inserimento di una pagina iniziale che mostra e contiene alcune informazioni dei due testi presenti sulla piattaforma.

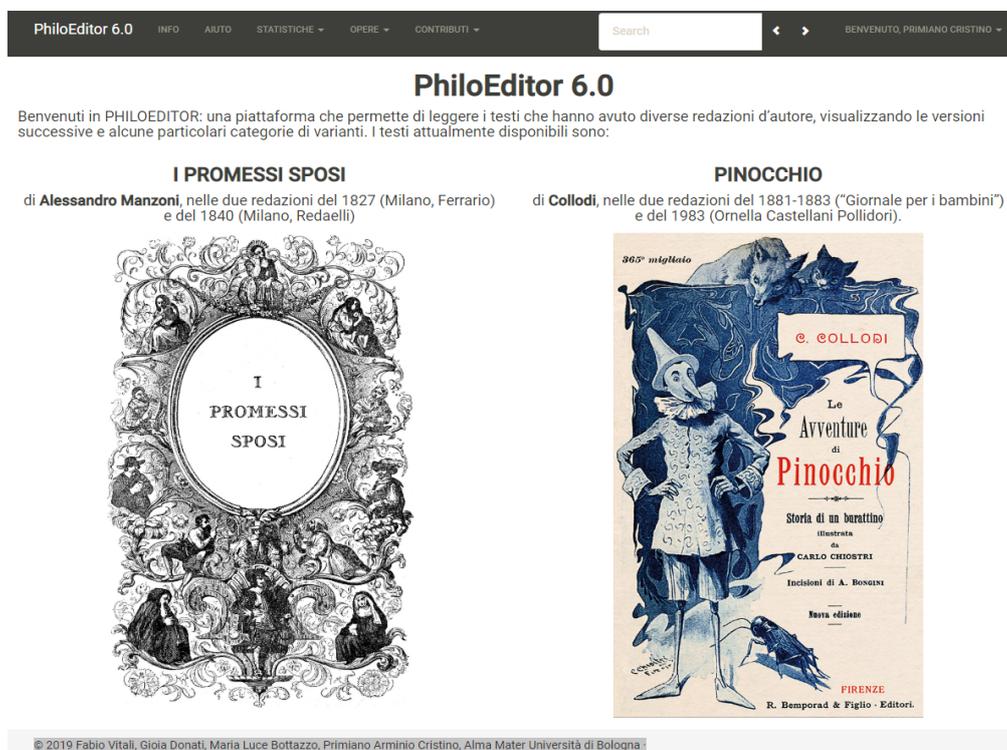


Figura 3.7: PhiloEditor 6.0

Ulteriori differenze riguardanti il layout della pagina sono quelle riguardanti la navbar, come ad esempio lo spostamento dal lato sinistro della pagina della sezione "opere e contributi", o come l'inserimento di una barra di ricerca.

Opere e Contributi sono diventati dei menù a tendina raggiungibili dalla navbar.

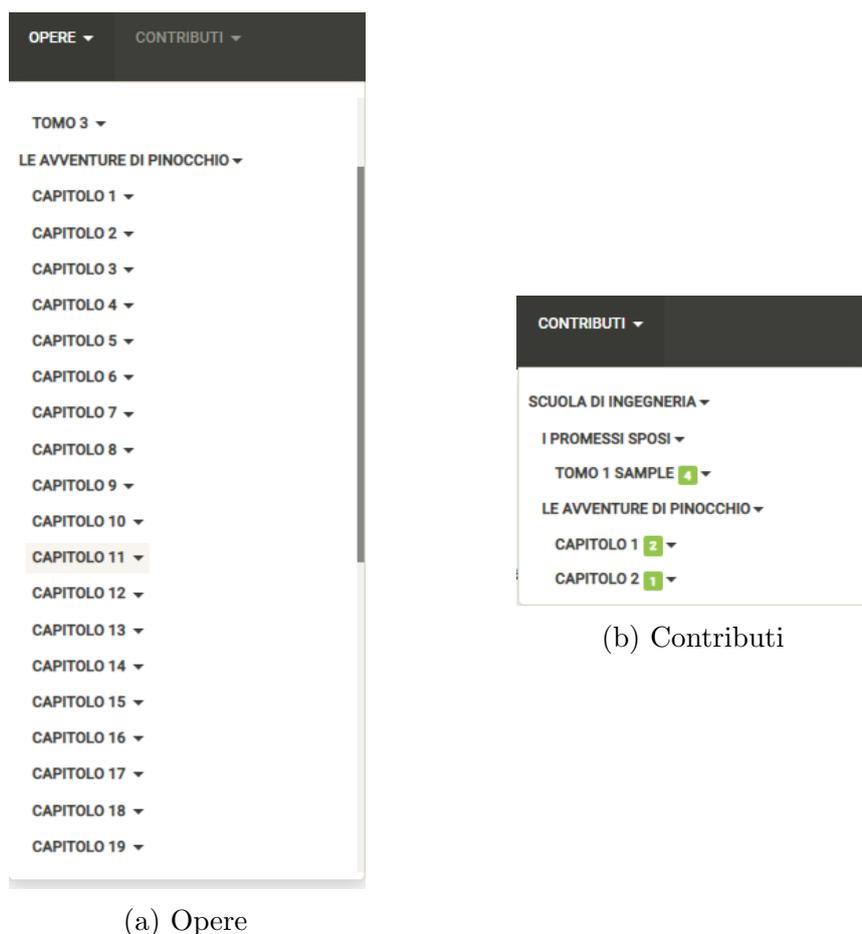


Figura 3.8: Opere e Contributi

Una delle funzionalità largamente richiesta è stata la barra di ricerca

che permette di cercare porzioni di testo all'interno del documento. Vengono colorate di giallo tutte le occorrenze del testo ricercato e in arancione la corrispondenza attuale. Le frecce invece permettono di andare alla successiva e alla precedente occorrenza .

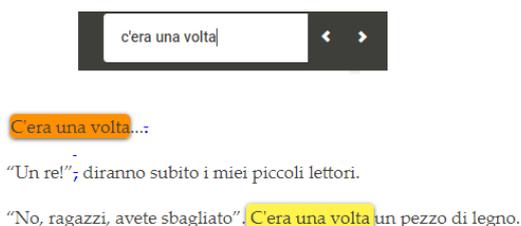


Figura 3.9: Esempio SearchBar

Sulla destra della pagina è possibile visualizzare "Modifica e Vista". Sebbene tutta la parte di aggiunta nel testo di una metodologia sia rimasta invariata, la possibilità di modificare e salvare i testi è stata data unicamente alle persone che sono registrate sulla piattaforma.

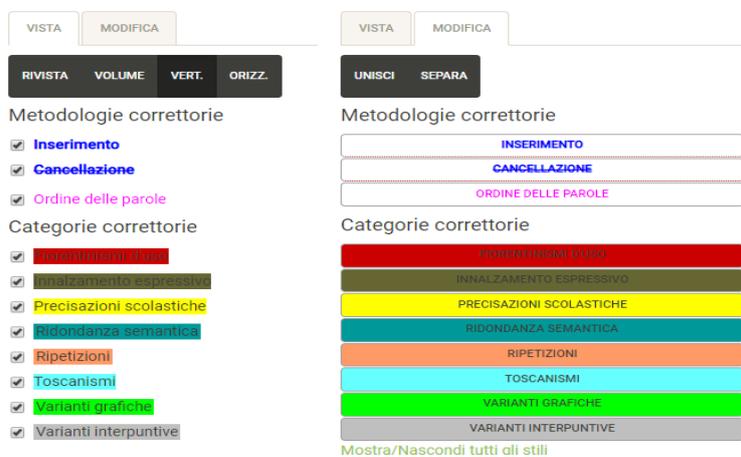


Figura 3.10: Vista e Modifica

Le statistiche sul documento e sull'intera opera sono rimaste pressochè invariate.

Nelle immagini seguenti si possono vedere le diverse tipologie di statistiche presenti sulla piattaforma.

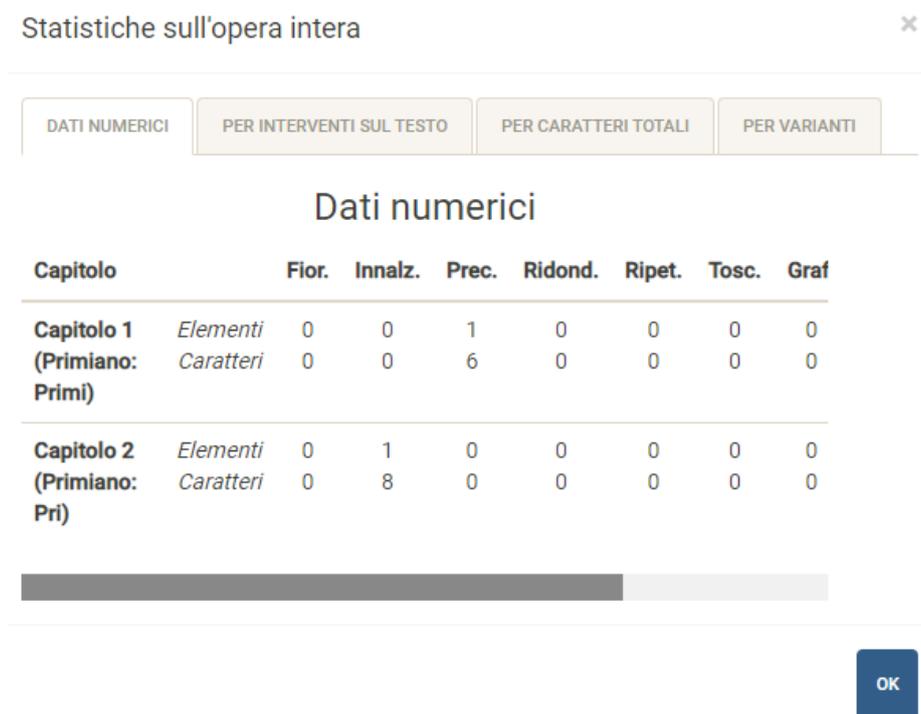


Figura 3.11: Statistiche: Dati Numerici

Capitolo 4

Architettura di PhiloEditor

4.1 Tecnologie utilizzate

4.1.1 HTML

«L'HTML è un linguaggio di formattazione che descrive le modalità di impaginazione o visualizzazione grafica (layout) del contenuto, testuale e non, di una pagina web attraverso tag di formattazione. Sebbene l'HTML supporti l'inserimento di script e oggetti esterni quali immagini o filmati, non è un linguaggio di programmazione: non prevedendo alcuna definizione di variabili, strutture dati, funzioni o strutture di controllo che possano realizzare programmi, il suo codice è in grado soltanto di strutturare e decorare dati testuali.» [13]

Il linguaggio HTML associa al contenuto una struttura grafica (layout) ben definita, modificabile in ogni sua parte, sia grazie a librerie esterne, sia all'interno della pagina web stessa, in modo tale da mostrare al fruitore del documento una pagina Web ben strutturata. Questo si realizza grazie all'utilizzo di tag diversi. Ogni tag specifica un diverso ruolo per i contenuti compresi dall'inizio del tag fino al suo termine. Un esempio è il seguente: <"Nome del tag">"contenuto del tag" </"fine tag">.

«L'HTML è a tutti gli effetti un linguaggio di Markup di tipo descrittivo. Il linguaggio di markup è un insieme di regole che descrivono i meccanismi di rappresentazione (strutturali, semantici, presentazionali) di un testo; facendo uso di convenzioni rese standard, tali regole sono utilizzabili su più supporti. Perciò, la tecnica di formattazione con marcatori (detti espressioni codificate) richiederà una serie di convenzioni, proprie appunto di un 'linguaggio a marcatori di documenti'.» [14]

4.1.2 Linguaggi di Markdown

«Markdown è un linguaggio di markup leggero e versatile, che si differenzia rispetto ad altri per la sua elevata leggibilità. Esso si basa su costrutti sintattici semplici, che non alterano significativamente la struttura del testo annotato. Grazie a questa caratteristica, i documenti markdown possono essere creati, letti e visualizzati con un semplice editor di testo, senza che questo richieda la comprensione di complessi tag come avviene nel caso di altri linguaggi di markup.» [15]

4.1.3 PHP

«PHP, acronimo ricorsivo di: PHP: Hypertext Preprocessor, è un popolare linguaggio di scripting che è particolarmente adatto allo sviluppo web. PHP è un linguaggio di programmazione interpretato, concepito per generare pagine web dinamiche. L'interprete di PHP è un software libero rilasciato sotto la licenza PHP. Una pagina PHP solitamente ha estensione .php. Tutto ciò che sta in una pagina PHP è sempre mostrato in output, soltanto ciò che sta fra i tag di apertura `<?php` e `?>`

è prima interpretato dal motore PHP. Ciò è molto pratico per creare una completa pagina HTML e lanciare comandi PHP solo dove necessario.» [16]

4.1.4 Javascript

«In informatica JavaScript è un linguaggio di scripting orientato agli oggetti e agli eventi, comunemente utilizzato nella programmazione Web lato client (recentemente esteso anche al lato server) per la creazione, in siti web e applicazioni web, di effetti dinamici interattivi tramite funzioni di script invocate da eventi innescati a loro volta in vari modi dall'utente sulla pagina web in uso (mouse, tastiera, caricamento della pagina ecc...).

Tali funzioni di script, utilizzati dunque nella logica di presentazione, possono essere opportunamente inserite in file HTML, in pagine JSP o in appositi file separati con estensione .js poi richiamati nella logica di business. Ultimamente il suo campo di utilizzo è stato esteso alle cosiddette Hybrid App (app ibride), con le quali è possibile creare app per più sistemi operativi utilizzando un unico codice sorgente basato appunto su JavaScript, HTML e CSS.» [17]

4.1.4.1 JQuery

«jQuery è una libreria JavaScript per applicazioni web. Nasce con l'obiettivo di semplificare la selezione, la manipolazione, la gestione degli eventi e l'animazione di elementi DOM in pagine HTML, nonché implementare funzionalità AJAX.

Le sue caratteristiche permettono agli sviluppatori JavaScript di astrarre le interazioni a basso livello tra interazione e animazione dei contenuti delle pagine. L'approccio di tipo modulare di jQuery consente la creazione semplificata di applicazioni web

e versatili contenuti dinamici.

Jquery è un software libero, distribuito sotto i termini della Licenza MIT.» [18]

4.1.4.2 WikEd Diff

«WikEd Diff è una libreria gratuita di JavaScript per confrontare del testo presente sulla stessa riga.

Mentre WikEd Diff è stato sviluppato e ottimizzato per confrontare il testo sorgente di Wikipedia, funziona benissimo per qualsiasi tipo di testo.

La libreria è completamente documentata ed è gratuita.» [19]

4.1.4.3 Bootstrap

«Bootstrap è una raccolta di strumenti liberi per la creazione di siti e applicazioni per il Web. Essa contiene modelli di progettazione basati su HTML e CSS, sia per la tipografia, che per le varie componenti dell'interfaccia, come moduli, pulsanti e navigazione, così come alcune estensioni opzionali di JavaScript. Dalla versione 2.0 supporta anche il responsive web design. Ciò significa che il layout delle pagine web si regola dinamicamente, tenendo conto delle caratteristiche del dispositivo utilizzato, sia esso desktop, tablet o telefono cellulare. A partire dalla versione 3.0, Bootstrap ha adottato il responsive design come impostazione predefinita, sottolineando il suo essere nata come libreria multidispositivo e multiplatforma.

I membri della comunità hanno tradotto la documentazione del software di Bootstrap in varie lingue (tra cui cinese, spagnolo e russo). La qualità della documentazione e degli esempi d'uso pratico è considerato uno dei punti di forza di Bootstrap.» [20]

4.2 Codice del progetto

Il progetto è stato scritto principalmente in due linguaggi differenti, in PHP per la parte back-end e in javascript per la parte front-end.

4.2.1 Back-end

La parte di back-end si occupa di:

- Registrare un nuovo utente sulla piattaforma.
- Loggare e sloggare un utente sulla piattaforma.
- Cambiare password.
- Fare richieste get per i documenti.
- Salvare o modificare un documento.
- Condividere un documento.
- Fare richieste per le statistiche dei documenti.

4.2.1.1 Registrarsi

Il codice riguardante la registrazione è molto semplice: non appena arriva la richiesta da parte del client di registrarsi, i dati del client, quali nome, cognome, password, scuola di appartenenza, e sesso, vengono salvati nelle apposite variabili.

Successivamente viene fatto un controllo iniziale nel caso in cui l'utente sia già registrato, e in questo caso si comunica al client la mancata registrazione. Viceversa se l'utente non viola questo vincolo, i dati vengono inseriti in un json presente nel server.

Per quanto riguarda le password, queste vengono criptate lato client.

4.2.1.2 Loggarsi

Come per la registrazione, inizialmente vengono presi i dati dell'utente e salvati momentaneamente in alcune variabili.

Successivamente viene fatta la ricerca dell'utente tramite nome, e la correttezza della password tramite matching.

Per il logout invece si tratta di azzerare i cookie preesistenti.

4.2.1.3 Cambiare password

Il codice del documento php non è stato particolarmente modificato, se non nella rimozione del controllo md5 in quanto questo pezzo è stato spostato in un altro documento.

La restante parte è rimasta identica.

Si tratta come nei casi precedenti della ricerca dell'utente in un file json e di un controllo di matching tra password da voler cambiare e password vecchia.

4.2.1.4 Richiedere i documenti

Il codice per la getFile cambia drasticamente, si passa dall'averne un unico file, ad avere due file separati.

Il primo si occupa di prendere i documenti dei sorgenti dei testi e di ritornarli al client.

Il secondo, invece, ritorna i documenti modificati dagli utenti.

Il codice presente all'interno dei due è molto simile: si differenziano, nel caso dei documenti Contributors, soltanto per l'aggiunta di alcune informazioni necessarie riguardanti il documento, ovvero le informazioni riguardanti la data di creazione del documento, chi la pubblica, le modifiche apportate e le responsabilità.

Queste aggiunte vengono ricercate nella sottocartella Metadata presente nel file, in cui sono contenuti sia le statistiche, sia le informazioni.

Per la parte in comune il codice è strutturato in modo da ricercare ricorsivamente tutte le cartelle presenti in un path, e se si trovano dei documenti,

aggiungerli al json da ritornare al client.

Nel fare questa operazione, per ogni documento vengono ritornati anche l'identificativo del documento che servirà da matching per le statistiche, seguiti dalla versione del documento, come ad esempio 1827 e 1840 per I Promessi Sposi, dal path del documento, dall'etichettatura e infine dall'ordine che ha lo scopo di distinguere tra file di tipo html e di tipo txt.

Parte del json che riguarda queste informazioni è nel seguente formato:

```
[
  {
    "id": "1",
    "authors": [
      "Primiano"
    ],
    "versions": [
      "RIVISTA",
      "VOLUME"
    ],
    "tei": {
      "encodingDesc": "Precisazione scolastica",
      "revisionDesc": [
        {
          "when": "2019-11-06",
          "who": "#Primiano"
        }
      ],
      "publicationStmt": {
        "publisher": "Primi",
        "pubPlace": "Bologna",
        "date": "2019",
        "license": "nessuna"
      },
      "editionStmts": [
        {
```

```

        "resp": "Capitolo 1",
        "name": "Primiano Cristino",
        "id": "Primiano"
    }
]

```

4.2.1.5 Salvare i documenti

Il codice riguardante il salvataggio dei file sul server cambia dalla versione precedente.

Avendo diviso i sorgenti e i file modificati in due distinte parti, si è deciso di salvarle in altrettante distinte parti.

Ciò ha comportato il dover salvare i file in cartelle separate, una chiamata Sources e una Contributors.

Inoltre avendo aggiunto un campo in più nella registrazione di un utente, ovvero la scuola di appartenenza, si è deciso di creare delle sottocartelle divise proprio in questo modo.

```

-Dir Sources
-Dir   01 - Promessi Sposi
-Dir     00 - Metadata
-Dir     01 - Tomo 1
-File       01.txt
-File       02.txt
-Dir     02 - Tomo 2
-Dir     03 - Tomo 3
-Dir   02 - Pinocchio
-Dir     00 - Metadata
-Dir     01 - Capitolo 1
-Dir     ...
-Dir     36 - Capitolo 36

```

```
-Dir Contributors
-Dir      01 - Scuola di Scienze
-Dir      01 - Promessi Sposi
-Dir      00 - Metadata
-Dir      01 - Tomo 1
-File      01.html
-Dir      02 - Tomo 2
-Dir      03 - Tomo 3
-Dir      02 - Pinocchio
-Dir      00 - Metadata
-Dir      01 - Capitolo 1
-Dir      ...
-Dir      36 - Capitolo 36
```

Il codice inizia proprio con la creazione, se necessaria, delle varie cartelle in Contributors.

Una volta fatto, si modifica il path di destinazione del file, in quanto la cartella può, invero, essere stata appena creata.

Conclusasi questa parte, viene fatto un if per differenziare un documento da salvare come nuovo, rispetto ad un altro da aggiornare.

Conclusasi anche questa parte viene mandato un response di ok al client.

4.2.1.6 Condivisione documenti

Questa parte è rimasta completamente identica alla versione precedente, e si tratta semplicemente di un controllo di matching sull'utente e sul proprietario del documento, seguito dall'aggiunta delle informazioni del documento e del nome utente della persona a cui si vuole condividere.

4.2.1.7 Statistiche

Anche questa parte di codice non è stata modificata.

Il codice semplicemente ritorna le informazioni presenti nel file stats.json

4.2.2 Front-end

La parte di front-end si occupa di tutta quella parte client su cui l'utente lavorerà.

Le principali funzioni che andrò a descrivere sono:

- Controlli iniziali e caricamento della pagina.
- Il diffing tra i due testi.
- Caricamento e salvataggio dei documenti.
- Caricamento delle statistiche.
- La barra di ricerca.
- Sicurezza e Crittografia
- Esportazione TEI.

4.2.2.1 Caricamento della pagina

Inizialmente vengono settate tutte le funzioni da collegare agli oggetti presenti nel DOM, come ad esempio possono essere i bottoni.

In seguito si fixano errori dovuti alle dimensioni dello schermo, come ad esempio il ridimensionamento delle immagini.

Vengono successivamente caricati e inseriti tutti documenti dal server.

Infine, c'è il controllo dei cookie per loggare automaticamente l'utente.

4.2.2.2 Diffing e Markdown

La funzione di diffing è stata completamente ripensata in quanto, passando da una struttura dei sorgenti formata da Capitoli a una struttura in Tomi, la computazione risultava molto lunga.

Quindi si è optato per l'aggiunta di una barra di caricamento da aggiornare in modo asincrono.

Si è quindi deciso di scompartare tutta la funzione di diffing in piccole porzioni, e ad ogni fine computazione aggiornare la barra di caricamento.

Infine, tutte le funzioni spezzettate vengono riunite in un'unica funzione asincrona.

Una volta caricati i testi dal server, a questi viene applicata la funzione di WikEd Diff.

Ovviamente questa non restituisce un documento subito pronto ma una sua approssimazione.

A questo punto non rimane che fixare i vari errori e trasformare il documento ricevuto in un DOM.

Per fare questo, però, c'è bisogno di un parser da un file che è un ibrido tra html e markdown.

Quindi, si trasforma il documento in html puro.

Una volta fatto ciò si rende infine compatibile il documento con l'esportazione TEI.

4.2.2.3 Caricamento e salvataggio dei documenti

Per il caricamento dei documenti si è dovuto dividere la richiesta ajax al server in due distinte richieste, la prima per ottenere le opere, e la seconda per ottenere i documenti modificati dagli utenti.

Una volta terminate le chiamate ajax si resettano i precedenti campi nel menù a tendina presente nella navbar, e si aggiornano con i nuovi file ricevuti.

La funzione che permette di salvare i documenti è divisa in due parti: una prima parte si occupa di ottenere tutte le informazioni per il TEI, e una seconda si occupa di fare una richiesta ajax per postare o aggiornare sia il documento, sia le informazioni dello stesso.

4.2.2.4 Caricamento delle statistiche

Per questa parte di codice viene fatta inizialmente una chiamata ajax per ottenere tutte le statistiche presenti sul server.

Dopo aver fatto questo si fa un confronto tra gli identificativi presenti nelle statistiche e gli identificativi presenti nei documenti caricati.

Si ottengono in questo modo tutte le informazioni riguardanti ogni singolo

documento.

Questi vengono poi caricati per creare la tabella finale in cui visualizzare le statistiche.

4.2.2.5 Esportazione TEI

Per l'esportazione non si è cambiato il codice javascript, bensì il codice del TEI.xsl presente nella cartella metadata.

La funzione semplicemente controlla il testo e le sue informazioni, per poi dividere il body del TEI tra le parti che comprendono la classe "replace" e quelle che non lo comprendono.

4.2.2.6 Barra di ricerca

La barra di ricerca è stata implementata nel seguente modo:

Come prima cosa si ricerca la parola tramite l'apposita searchbar.

Poi, per ogni carattere mostrabile a testo viene chiamata la funzione per evidenziare i matching su testo.

Quindi per ogni stringa trovata gli si aggiunge una particolare classe, highlight, con dei particolari stili in css, come ad esempio il colore, la grandezza, ecc... .

Ogni volta che si aggiunge una lettera l'algoritmo riparte nuovamente.

Vi è inoltre la possibilità, premendo ENTER o cliccando sulle frecce a lato, di passare alla successiva occorrenza della parola nel documento.

Tramite la funzione jump infatti si salta alla successiva/precedente occorrenza.

Infine è presente la funzione clean che rimuove le classi precedentemente create; queste in particolare vengono rimosse prima di un salvataggio e prima di un'esportazione TEI.

4.2.2.7 Sicurezza e Crittografia

Si è voluta aggiungere una funzione che criptasse le password da salvare sul server in modo tale che nessuno possa accedere direttamente alla password in chiaro, ma alla sua versione criptata.

```
function krypton(a){
  for (var i=0;i<100;i++){
    a=CryptoJS.SHA1(a);
  }
  for (var i=0;i<100;i++){
    a=CryptoJS.SHA256(a);
  }
  for (var i=0;i<100;i++){
    a=CryptoJS.SHA3(a);
  }
  return md5(a);
}
```

Capitolo 5

Valutazione

5.1 Confronto con i filologi

Come già anticipato nell'introduzione vi sono stati alcuni incontri, sia via mail, sia via skype con due studentesse di filologia che conoscevano già in precedenza PhiloEditor.

Questi incontri sono serviti come punto d'inizio di lavoro.

Infatti molti dei cambiamenti sia di layout sia di funzionalità di PhiloEditor sono dipesi proprio da questi incontri.

Ad esempio si è discusso dell'inserimento o meno di alcune immagini campione nella pagina iniziale, seguite da una loro breve descrizione, al fine di mostrare brevemente il contesto filologico in cui PhiloEditor si trova.

Queste immagini seppure siano un prototipo, servono a dare un'idea di che cosa sia PhiloEditor.

Riguardo al layout, si è passati da una pagina divisa in tre colonne a una pagina divisa in due, in cui a sinistra è presente il testo e a destra le metodologie/categorie correttorie.

Questo spostamento è dovuto al fatto che lo spazio di lettura del testo risultava troppo piccolo, un terzo della totalità dello schermo.

Si è quindi deciso di spostare la colonna di sinistra nella navbar e di portare a due terzi la larghezza del testo visibile.

Per quanto riguarda le metodologie correttorie si è deciso di lasciarle nella

colonna di destra, in quanto, in questo modo, si possono visualizzare sullo schermo sia le metodologie con i suoi colori, sia il testo stesso modificato con un particolare colore.

Si è inoltre discusso anche di alcune funzionalità nuove come la barra di ricerca.

Questa ha lo scopo di ricercare parte di testo all'interno del testo attualmente visualizzato, con la possibilità in futuro di espanderlo alla totalità dei testi.

Una delle modifiche più importanti di questa versione, quale la ristrutturazione architettonica del codice back-end, è dovuta, infatti, al tentativo di dividere concettualmente gli utenti e le loro modifiche dai testi originali degli autori.

Si è quindi pensato anche a possibili versioni future in cui solo alcuni utenti specifici potranno creare dei testi modificati.

Si è poi discusso di alcune funzionalità che potrebbero essere aggiunte in futuro.

In conclusione, PhiloEditor 6.0 non è la versione definitiva, ma è solo uno dei tanti step di sviluppo.

Si è successivamente discusso proprio sulle versioni future di PhiloEditor.

Capitolo 6

Conclusioni

6.1 Conclusioni e sviluppi futuri

PhiloEditor è quindi una piattaforma web che cerca di creare un connubio tra l'informatica e la filologia.

Questa ricerca ha spinto verso la creazione di diverse versioni della piattaforma, fino al raggiungimento dell'attuale 6.0.

Infine per concludere, voglio parlare brevemente di alcuni aspetti che caratterizzano PhiloEditor.

Come già detto in precedenza, PhiloEditor è molto versatile dal punto di vista del back-end e dell'inserimento di nuovi testi.

Non si può dire la stessa cosa di altri aspetti.

In PhiloEditor ci sono tante piccole migliorie che si potranno sicuramente applicare in futuro e che renderanno PhiloEditor migliore di quel che è già.

In quest'ultima sezione voglio descrivere alcune modifiche che sono state pensate per essere introdotte in futuro.

Prima tra tutte la possibilità di estendere la search non solo al documento presente sul testo in questo momento, ma anche a tutti i documenti presenti sul server.

Una sua possibile soluzione potrebbe essere un motore di ricerca server-side

che filtra per parole.

La seconda aggiunta che è stata pensata è quella di dividere gli utenti in utenti normali e utenti speciali. I primi potranno solo accedere alla piattaforma e visualizzare i testi presenti su di essa, mentre i secondi potranno anche modificare i testi stessi, similmente a quanto accade ora tra utente registrato e visitatore esterno.

La soluzione ideale è quella di aggiungere nella piattaforma un form con lo scopo di passare da un utente normale ad un account speciale.

Una delle soluzioni future sarà sicuramente quella di poter cliccare le immagini in copertina e far partire determinate funzioni.

Bisogna effettivamente pensare a cosa far mostrare una volta cliccate le immagini.

Per come è strutturata l'opera attuale, è possibile vedere soltanto le statistiche di un'opera all'interno di una determinata scuola, in modo tale da poter confrontare varie scuole anche in ambito didattico.

La prossima funzionalità da aggiungere è quella di estendere la visualizzazione delle statistiche a tutte le scuole.

Per quanto riguarda il layout si è pensato di cambiare il dropdown che fa riferimento a "OPERE" e "CONTRIBUTI", specificatamente a "OPERE", in quanto, non avendo più i documenti dei testi modificati nella stessa cartella dei testi degli autori, risulta ripetitivo dover cliccare due volte per aprire un documento, ad esempio prima su "TOMO 1" e successivamente su "CONFRONTA 1827 E 1840".

E per ultimo ma non meno importante, si è considerata l'idea di ingrandire e aggiornare il modale di "AIUTO", ovvero la pagina che mostra e descrive il funzionamento di PhiloEditor all'interno dell'applicazione stessa.

E con questa si concludono anche gli sviluppi futuri che sono in progetto.

Ringraziamenti

Vorrei ringraziare di cuore, innanzitutto, la mia famiglia e i miei genitori per avermi sostenuto in ogni singola scelta universitaria presa fin qui.

Un altro grazie va ai miei amici che mi hanno accompagnato durante questo percorso, che mi hanno ascoltato e capito fin'ora, soprattutto in momenti difficili.

Un sentito ringraziamento anche alle due ragazze, Ersilia e Teresa , che mi hanno aiutato e consigliato durante tutto il percorso di tesi.

Un ringraziamento speciale va fatto anche a tutti i ragazzi che come me si stanno per laureare.

E infine un grazie anche all'Ateneo di Bologna che ci ha permesso di affrontare questo iter scolastico.

Bibliografia

- [1] Paola Italia e Claudia Bonsi. Edizioni critiche digitali. *Università editrice*, pages 76–77, 2016.
- [2] Friedrich Nietzsche. Aurora. pensieri sui pregiudizi morali. 1881.
- [3] Autori di Wikipedia. Filologia classica. https://it.wikipedia.org/wiki/Filologia_classica.
- [4] Luca Carlo Rossi. Finalità e metodi della filologia. *ICoN – Italian Culture on the Net*, pages 22–26, 2003.
- [5] Michelangelo Zaccarello. L’edizione critica del testo letterario: Glossario filologico. *academia.edu*.
- [6] Paola Italia e Claudia Bonsi. Edizioni critiche digitali. *Università editrice*, page 4, 2016.
- [7] Paola Italia e Claudia Bonsi. Edizioni critiche digitali. *Università editrice*, page 5, 2016.
- [8] Matthew James Driscoll. Levels of transcription. in: Electronic textual editing. *New York: The Modern Language Association of America*, pages 254–261, 2006.
- [9] Paola Italia e Claudia Bonsi. Edizioni critiche digitali. *Università editrice*, pages 78–79, 2016.
- [10] Sara Mori. Febbraio 1883: escono “le avventure di pinocchio”. *Portale Storia di Firenze*, 2013.

- [11] Paola Italia e Claudia Bonsi. Edizioni critiche digitali. *Università editrice*, pages 79–83, 2016.
- [12] Autori di Wikipedia. Text encoding initiative. https://it.wikipedia.org/wiki/Text_Encoding_Initiative.
- [13] Thomas Powell. Html xhtml: the complete reference. page 25, 2003.
- [14] Autori di Wikipedia. Markup language. https://it.wikipedia.org/wiki/Linguaggio_di_markup.
- [15] Sergio Monteleone. Markdown. <https://www.html.it/articoli/markdown-guida-al-linguaggio/>.
- [16] Autori di Wikiversity. Php. <https://it.wikiversity.org/wiki/PHP>.
- [17] Autori di Wikipedia. Javascript. <https://it.wikipedia.org/wiki/JavaScript>.
- [18] Autori di Wikipedia. JQuery. <https://it.wikipedia.org/wiki/JQuery>.
- [19] Autori di Wikipedia. Wikied diff. <https://en.wikipedia.org/wiki/User:Cacycle/diff>.
- [20] Autori di Wikipedia. Bootstrap. [https://it.wikipedia.org/wiki/Bootstrap_\(informatica\)](https://it.wikipedia.org/wiki/Bootstrap_(informatica)).

Elenco delle figure

3.1	Statistiche PhiloEditor 2.0	17
3.2	Versioni e Stili PhiloEditor 2.0	18
3.3	Testi a Confronto PhiloEditor 2.0	19
3.4	TEI PhiloEditor 3.0 parte 1	21
3.5	TEI PhiloEditor 3.0 parte 2	22
3.6	PhiloEditor 3.0	23
3.7	PhiloEditor 6.0	24
3.8	Opere e Contributi	25
3.9	Esempio SearchBar	26
3.10	Vista e Modifica	26
3.11	Statistiche: Dati Numerici	27

FINE