

UNIVERSITÀ DEGLI STUDI DI MILANO
FACOLTÀ DI SCIENZE E TECNOLOGIE

DIPARTIMENTO DI INFORMATICA
GIOVANNI DEGLI ANTONI



Corso di Laurea triennale in
Informatica Musicale

UNA LIBRERIA C PER IL CARICAMENTO E LA
MANIPOLAZIONE DI DOCUMENTI IEEE 1599

Relatore: Prof. Federico Simonetta
Correlatore: Prof. Luca Andrea Ludovico

Tesi di Laurea di:
Alessandro Talamona
Matr. Nr. 895744

ANNO ACCADEMICO 2019-2020

Questo lavoro è dedicato a tutti gli studenti

*“Io studio,
ma studiate pure voi,
che se studio solo io non serve a un c. . . o”*

– Gli scarabocchi di Maicol & Mirco

*“No tale is so good
that it can’t be spoiled
in the telling”*

– Proverbio

Ringraziamenti

Questa sezione, facoltativa, contiene i ringraziamenti.

Indice

Ringraziamenti	ii
Indice	iii
1 Introduzione	1
1.1 Scopo Del Software	1
1.2 IEEE 1599	1
1.3 Language Binding	2
2 Tecnologie utilizzate	3
2.1 C	3
2.2 XML	3
2.2.1 DTD	3
2.3 Libxml2	3
2.4 Generalità	4
2.4.1 La scrittura WYSIWYG vs. WYSIWYM	4
2.4.2 Risorse e strumenti	5
2.5 Suggerimenti sull'uso di \LaTeX	5
2.5.1 Riferimenti incrociati	6
2.5.2 Ritorni a capo	6
2.5.3 Accenti	6
2.5.4 Spazi tra parole	7
2.5.5 Interlinea	7
2.5.6 Doppie virgolette	7
2.5.7 Ambienti per scrivere codice	8
2.5.8 Figure	8
2.6 \BibTeX	9
2.6.1 Generalità	9
2.6.2 Strumenti	9

3	Nome del Progetto	11
3.1	Panoramica del progetto	11
3.2	Implementazione	11
4	Test	12
4.1	Protocollo	12
4.2	Risultati	12
4.3	Osservazioni	13
5	Conclusioni	14
5.1	Conclusioni	14
5.2	Sviluppi futuri	14
A	Il tirocinio e la correzione	15
B	Il riassunto	16
C	La consegna	17
D	La presentazione	18
	Bibliografia	20

Capitolo 1

Introduzione

Il presente lavoro è organizzato come segue: nel Capitolo 1 ...

1.1 Scopo Del Software

Lo scopo di questo elaborato è quello di costruire una libreria che consenta il caricamento e quindi la manipolazione dei dati contenuti in documenti che rispettino lo standard IEEE 1599, un formato per la rappresentazione musicale sviluppato dal Laboratorio di Informatica Musicale (LIM) nel 2008 di cui verranno descritte le caratteristiche di interesse nelle successive sezioni. In particolare si vogliono creare le strutture dati per contenere ogni tipo di informazione ottenibile da tali documenti e le funzioni che permettano il caricamento di tali strutture. Ulteriore obiettivo è quello di permettere l'utilizzo di questa libreria da diversi linguaggi attraverso il language binding.

1.2 IEEE 1599

Ieee1599 è un formato basato su XML che descrive contenuti musicali eterogenei in modo completo. In un unico file, simboli musicali, spartiti, tracce audio, performance computer-driven, metadati per la catalogazione, testi e contenuti grafici relativi a una particolare brano musicale vengono collegati e reciprocamente sincronizzati all'interno della stessa struttura. Contenuti eterogenei vengono organizzati in una struttura multi-livello che supporta diversi formati di codifica e un numero di elementi digitali per ogni livello. [1]

I livelli definiti dal formato sono: [2]

- - General, che contiene i metadati relativi al brano in oggetto, tra cui le informazioni catalografiche su titolo dell'opera, autori e genere;

- - Logic, vero nucleo del formato, destinato alla descrizione simbolica dei contenuti musicali;
- - Structural, che identifica gli oggetti musicali su cui il brano è costruito e permette di evidenziarne i mutui rapporti;
- - Notational, che contiene le differenti rappresentazioni grafiche della partitura, ad esempio riferibili a diverse edizioni o trascrizioni;
- - Performance, che è dedicato ai formati per la generazione di esecuzioni sintetiche da parte dell'elaboratore;
- - Audio, che consente di legare al brano in oggetto le esecuzioni audio/video della partitura.

1.3 Language Binding

Con language binding si intende . . .

Capitolo 2

Tecnologie utilizzate

In questo capitolo vengono presentate le tecnologie utilizzate per lo sviluppo del software.

2.1 C

Il linguaggio di programmazione scelto è il C in modo da favorire futuri language binding, visto che molti dei linguaggi di alto livello più utilizzati sono basati su questo linguaggio, come per esempio Python, C++, Java, Javascript e Perl. Le funzionalità del linguaggio che sono state usate massicciamente per la creazione delle strutture dati adeguate sono le struct, puntatori, liste e allocazione dinamica della memoria. In alcuni casi è stato anche utile servirsi di typedef e enum.

2.2 XML

2.2.1 DTD

Il DTD che definisce la struttura dei documenti IEEE 1599, sul quale si basano le scelte di programmazione effettuate per questo elaborato, è disponibile sul sito ufficiale del Computer Society of the Institute of Electrical and Electronics Engineers <https://standards-ieee-org.pros.lib.unimi.it/downloads.html>, attualmente alla versione del 2008.

2.3 Libxml2

Il software creato si appoggia sulla libreria libxml2 (<http://www.xmlsoft.org/>) per il parsing dei file XML generici. Sul sito è presente un tutorial su come utilizzare

i metodi principali del linguaggio dal quale sono state estrapolate alcuni algoritmi ricorrenti per le operazioni di parsing. Questa libreria, tra le varie disponibili, è una tra le più utilizzate perché può facilitare la programmazione basandosi sulla creazione in memoria di un albero a partire dal document XML, Supporta XPath e permette la validazione tramite DTD durante il parsing.

2.4 Generalità

2.4.1 La scrittura WYSIWYG vs. WYSIWYM

L'acronimo WYSIWYG sta per “What You See is What You Get”, e si riferisce al concetto di ottenere sulla carta testo e immagini che abbiano una disposizione grafica equivalente a quella visualizzata a schermo dal software di videoscrittura. Un esempio classico di WYSIWYG è Microsoft Word, che mostra il testo impaginato e formattato come ci si aspetta di vederlo una volta stampato.

L'acronimo WYSIWYM sta per “What You See is What You Mean”, ed è il paradigma per la creazione di testi strutturati. \LaTeX è un ambiente che supporta tale paradigma. In realtà, anche Microsoft Word avrebbe la possibilità di strutturare il testo, principalmente attraverso il meccanismo degli stili, ma pochissimi utenti sfruttano tale funzionalità (ovviamente se sceglierete di scrivere la tesi in Word raccomandiamo caldamente l'uso di tali funzioni).

I principali svantaggi di un sistema WYSIWYM sono il tempo di apprendimento, dovuto a una minore intuitività degli strumenti software, e la necessità di invocare la compilazione del documento per vederne l'aspetto definitivo. Ad esempio, in \LaTeX l'intero documento viene scritto in testo semplice, che all'interno contiene ambienti e comandi con informazioni di layout, e solo la compilazione permette di scoprire eventuali errori di sintassi e giungere, infine, alla creazione del PDF.

Le difficoltà iniziali, però, sono ampiamente compensate dai vantaggi a medio e lungo termine. Infatti, il lavoro risulterà perfettamente impaginato e strutturato, e dunque avrà un aspetto professionale. Questo riguarda non solo gli stili, che vengono applicati al testo in modo coerente con il template prescelto, ma anche problemi tipicamente spinosi di Word, quali il posizionamento delle immagini e delle tabelle, la creazione di una bibliografia con relative citazioni nel testo, la creazione di un sommario (per cui esistono funzioni automatiche, ma sono piuttosto macchinose). Diventa automatico e molto semplice, ad esempio, aggiungere un indice delle figure o delle tabelle, oppure numerare le formule espresse nel testo. Un altro aspetto su cui \LaTeX è nettamente superiore a Word è proprio la scrittura di formule matematiche, come mostrato nell'esempio qui riportato:

$$x_i(n) = a_{i1}u_1(n) + a_{i2}u_2(n) + \cdots + a_{iJ}u_J(n). \quad (1)$$

2.4.2 Risorse e strumenti

Esiste una vastissima gamma di risorse online per avvicinarsi a \LaTeX . Un buon punto di partenza è la lista messa a disposizione sul sito del \TeX Users Group (TUG).¹ Tra queste si consiglia in particolare la “Not so Short Introduction to $\text{\LaTeX}2\epsilon$ ”,². Per chi volesse approfondire, uno dei riferimenti bibliografici più completi è il libro di Mittelbach *et al.* [?].

In alternativa a un’installazione locale sul proprio pc, è possibile utilizzare un editor \LaTeX online, con il vantaggio di avere immediatamente a disposizione l’ambiente di sviluppo e tutti i package necessari, nonché di potere condividere il proprio progetto con il relatore di tesi. Il più diffuso editor \LaTeX online è Overleaf,³ dove si può trovare anche ulteriore documentazione (in particolare la guida “Learn \LaTeX in 30 minutes”).⁴

Qualunque sia la risorsa utilizzata, ecco un elenco non esaustivo di argomenti di base nei quali con tutta probabilità ci si imbatte durante la stesura della tesi.

- Formattazione del testo (grassetto, italics, dimensioni font, ecc.) e del documento (paragrafi, comandi `\chapter`, `\section`, `\tableofcontents`, ecc.).
- Elenchi: ambienti *itemize* e *enumerate*, pacchetti rilevanti (*paralist*)
- Riferimenti incrociati: comandi `\ref`, `\pageref` e `\label`, etichette.
- Matematica: equazioni, modalità *inline* e *displayed*, pacchetti rilevanti (*amssymb*, *amsmath*).
- Figure: formati grafici, ambiente *figure*, pacchetti rilevanti (*graphicx*, *subfloats* per figure multiple).
- Tabelle: ambienti *table* e *tabular*, pacchetti rilevanti (*array*, *multirow*, *longtable*).
- Riferimenti e bibliografie (si veda più sotto la sezione 2.6).

2.5 Suggerimenti sull’uso di \LaTeX

Fatte salve le indicazioni generali fornite nella sezione precedente, di seguito si riportano alcune osservazioni puntuali sulle domande e gli errori più tipici degli studenti alle prime armi con \LaTeX .

¹<http://www.tug.org/interest.html>

²<http://mirrors.ibiblio.org/CTAN/info/lshort/>

³<http://www.overleaf.com>

⁴<https://www.overleaf.com/learn>

2.5.1 Riferimenti incrociati

Uno dei principali vantaggi di \LaTeX è la possibilità di impostare riferimenti automatici a molti elementi del documento, tra cui capitoli, sezioni, sottosezioni, tabelle, figure, equazioni, riferimenti bibliografici, e via dicendo.

Quindi il modo corretto per riferirsi, ad esempio, al secondo capitolo non è scrivere “Capitolo 2” bensì “Capitolo ??”. Il risultato apparente (nel PDF) è lo stesso, mentre ci sono differenze sostanziali a livello di codice. Il vantaggio è che, se il secondo capitolo diventasse il terzo, il riferimento incrociato continuerebbe a puntare alla posizione corretta. Si pensi, per estensione, alla numerazione delle immagini, o ai riferimenti alla bibliografia.

Sintatticamente, questo richiede di inserire dei comandi `\label{mia_label}` all’interno degli elementi cui ci si vuole riferire, e dei comandi `\ref{mia_label}` dove si vuole creare il riferimento. Fa eccezione la bibliografia (si veda più sotto la sezione 2.6).

2.5.2 Ritorni a capo

I ritorni a capo in \LaTeX possono essere effettuati in due modi: con la sintassi `\\` o con una doppia pressione del tasto di ritorno a capo. In generale, la soluzione corretta è la seconda, che equivale a usare il tasto Enter in Word. Il doppio Backslash, che corrisponde a Shift+Enter in Word, crea una nuova riga senza interruzione del paragrafo. Questo va usato solo in casi molto specifici, come nella frase seguente.

Il sito web ufficiale del Laboratorio di Informatica Musicale è:
<https://www.lim.di.unimi.it>.

In molti stili di \LaTeX , un nuovo paragrafo (dopo un doppio a capo) crea un rientro della prima riga. Non c’è nulla di male nel rientro, ma se proprio lo si vuole evitare la soluzione **non** è usare il doppio BackSlash! Esistono molte soluzioni più appropriate (ad esempio, dare una dimensione nulla al rientro tramite il comando `\setlength{\parindent}{0ex}`, da inserire nel preambolo della tesi).

2.5.3 Accenti

Scrivendo la tesi in italiano, l’uso di lettere accentate è frequente. I caratteri accentati immessi da tastiera non vengono però riconosciuti nativamente. Invece l’accento grave e acuto in \LaTeX si ottengono rispettivamente con i comandi `\`{a}` e `\’{a}`.

In alternativa è possibile specificare che si usa la codifica UTF-8, usando il comando `\usepackage[utf8]{inputenc}` nel preambolo del documento (già incluso in questo template). In questo modo i caratteri accentati immessi da tastiera verranno riconosciuti.

Nota ortografica: attenzione a non sbagliare gli accenti: si scrive “è”, ma si scrive “perché”.

2.5.4 Spazi tra parole

Riguardo la gestione della spaziatura tra parole, \LaTeX adotta una strategia molto elegante, che lascia uno spazio maggiorato dopo il punto di fine periodo. Un potenziale problema è che questo spazio extra viene introdotto dopo qualsiasi occorrenza del punto, indipendentemente dalla funzione sintattica, e dunque anche dopo i nomi puntati, quali “R. Schumann”, o dopo le formule “ad es.”, “Fig. n”, “ecc.” e via dicendo. Per evitarlo, questi spazi da non aumentare vanno sostituiti con alternative, quali un Backslash seguito da uno spazio (che immette un *control space*) o una tilde \sim (che introduce un *unbreakable space*, utile a impedire ritorni a capo intermedi).⁵

2.5.5 Interlinea

Per aumentare la leggibilità della tesi può essere utile usare un’interlinea maggiore di 1. Un modo per ottenerlo è usare il comando `\linespread{1.6}` nel preambolo del documento. Nota: il valore 1.6 indica interlinea doppia, il valore 1.3 indica interlinea 1.5. Don’t ask why.

2.5.6 Doppie virgolette

L’uso dell’unico carattere di doppie virgolette presente sulla tastiera è assolutamente sconsigliato, in quanto non viene correttamente interpretato da \LaTeX , soprattutto riguardo l’apertura delle virgolette.

La combinazione giusta da utilizzare è ‘ ‘ per l’apertura e ’ ’ per la chiusura. Si noti che in entrambi i casi si tratta di doppi apostrofi ravvicinati, e non di singoli caratteri. Se si utilizza come editor TeXstudio, c’è un’opzione per sostituire automaticamente le doppie virgolette con la versione corretta in \LaTeX : Opzioni → Configura TeXstudio... → Editore → Sostituisci i doppi apici: Inglese.

Le virgolette caporali, o francesi, si ottengono con i comandi `\guillemotleft` e `\guillemotright`, il cui risultato è «questo».

⁵Per una trattazione completa delle numerose varianti, si veda <https://tex.stackexchange.com/questions/74353/what-commands-are-there-for-horizontal-spacing>

2.5.7 Ambienti per scrivere codice

Il codice all'interno dell'elaborato va scritto con carattere monospaziato e rispettando, nell'ambito del possibile, le originali regole (o buone pratiche) di indentazione.

Per farlo, esiste innanzi tutto l'ambiente `verbatim`, che va aperto e chiuso con i comandi `\begin{verbatim}` ed `\end{verbatim}`.

Tra le alternative, si segnala l'ambiente `lstlisting`, che richiede innanzi tutto di aggiungere nel preambolo `\usepackage{listings}`, e quindi di aprire e chiudere l'ambiente con i comandi `\begin{lstlisting}` ed `\end{lstlisting}`. Un esempio, relativo al calcolo del massimo comun divisore attraverso l'algoritmo di Euclide in Python, è:

```
def MCD(a,b):  
    while b != 0:  
        a, b = b, a % b  
    return a
```

Se dopo l'apertura dell'ambiente si specifica tra parentesi quadrate il linguaggio adottato, ad esempio con la sintassi `\begin{lstlisting}[language=Python]`, si ottiene automaticamente l'evidenziazione del codice:

```
def MCD(a,b):  
    while b != 0:  
        a, b = b, a % b  
    return a
```

L'elenco dei linguaggi supportati e le opzioni avanzate per personalizzare la visualizzazione del codice si trovano all'indirizzo https://www.overleaf.com/learn/latex/Code_listing#Reference_guide.

Infine, si consideri la possibilità di importare interi file di codice attraverso la sintassi `\lstinputlisting[language=nome linguaggio]{filesorgente}`.

2.5.8 Figure

In tutti i casi in cui sia possibile (schemi a blocchi, plot di dati, ecc.), è opportuno che le figure siano in formato vettoriale (eps, pdf) per aumentarne la leggibilità. Nel caso di figure prodotte da software esterno (ad esempio, grafici esportati in eps o pdf da Matlab), è consigliabile conservare tutti i sorgenti e i dati utilizzati per generarle: in questo modo sarà possibile ricreare le figure quando necessario.

2.6 B_IB_TE_X

2.6.1 Generalità

Esistono più modi per inserire una bibliografia in L^AT_EX. Si consiglia fortemente l'utilizzo del sistema B_IB_TE_X. Questo consente di aggiungere, rimuovere e modificare voci di bibliografia in maniera efficiente, di formattarle, di riordinarle a piacere e aggiornare automaticamente i corrispondenti riferimenti nel testo, ecc.

Una guida introduttiva e completa è “Tame the BeaST”.⁶ In estrema sintesi, i passi per gestire una bibliografia tramite B_IB_TE_X sono essenzialmente tre.

1. Salvare i riferimenti bibliografici come entry di uno o più file con l'estensione .bib (si veda ad esempio il file `bibliografia.bib`, parte di questo template). Gli entry sono scritti in un formato specifico, in particolare ogni entry ha una propria etichetta testuale che lo identifica univocamente.
2. Creare la bibliografia alla fine del documento o dove desiderato, usando il comando `\bibliography{file1.bib,file2.bib,...}`. È possibile inoltre specificare uno stile bibliografico attraverso il comando `\biblographystyle{...}`.
3. All'interno del testo, riferirsi a una voce di bibliografia tramite il comando `\cite{etichetta_entry}`. Si noti che una voce bibliografica non viene inclusa in bibliografia in assenza di una citazione all'interno del testo (coerentemente con quanto discusso nella sezione ??).

È consigliabile cominciare a costruire la propria bibliografia in formato bib a mano a mano che si analizza lo stato dell'arte, invece che rimandare alla stesura finale della tesi.

2.6.2 Strumenti

Un file .bib è un file di testo e può quindi essere gestito con un qualsiasi text editor. Esistono comunque molti tool più evoluti per gestire bibliografie in formato bib. Un'applicazione installabile localmente sul proprio pc è JabRef.⁷ Oppure esistono tool online, come Zotero,⁸ che forniscono molte funzionalità tra cui l'esportazione di bibliografie in formato bib.

Peraltro, anche Google Scholar esporta automaticamente citazioni in formato bib cliccando sul link Cita (icona con doppie virgolette) e scegliendo l'opzione B_IB_TE_X nella parte bassa della finestra che si apre. **Attenzione** però: spesso i bib

⁶Accessibile da <http://www.tug.org/interest.html>

⁷<http://www.jabref.org/>

⁸<http://www.zotero.org/>

esportati da Scholar sono incompleti o sporchi, è sempre consigliabile controllarne la correttezza.

I più “smanettoni” possono addentrarsi a piacere in funzionalità avanzate. Ad esempio è possibile creare bibliografie multiple.⁹ Esiste anche il pacchetto `biblatex`, che fornisce una reimplementazione completa delle funzionalità bibliografiche di `LATEX-BIBTEX`, offrendo maggiore flessibilità e mantenendo compatibilità all’indietro con il formato `bib`.

⁹Un possibile modo viene illustrato qui: <https://bit.ly/2wI3Y7p>

Capitolo 3

Nome del Progetto

In genere il capitolo più corposo dell’elaborato è quello in cui si parla del lavoro svolto. Esistono alcune buone pratiche per rendere l’esposizione efficace, eccone alcune.

3.1 Panoramica del progetto

Prima di addentrarsi nei dettagli è bene fornire una panoramica (anche molto schematica, corredata da un diagramma) del lavoro svolto, in modo che il lettore abbia una mappa concettuale con cui orientarsi.

3.2 Implementazione

Una volta fornita una panoramica, è possibile addentrarsi nei dettagli, ricordando che si sta scrivendo un articolo scientifico e non un testo di narrativa. Vanno dunque evitate domande retoriche quali “*ma come è stato possibile risolvere questo problema? Ebbene ...*”.

Vale la pena riportare nel testo solo le parti di codice più importanti, demandando all’appendice il codice completo e altri extra come il manuale utente.

Capitolo 4

Test

Ogni lavoro scientifico richiede una validazione dei risultati ottenuti. Questo si può fare confrontando in modo sistematico il proprio lavoro con lavori concorrenti o misurando l'efficacia del lavoro mediante test con gli utenti. È fondamentale che questi test siano ripetibili, dovrete dunque fornire tutti i dettagli necessari nel testo per permettere a chi legge la tesi di replicare l'esperimento.

Progettare e condurre un test soggettivo con utenti è un lavoro complesso e lungo, che richiede pianificazione e competenza. Il bel libro di Lazar *et al.* [?] illustra in dettaglio i concetti principali della ricerca sperimentale e le metodologie correlate: ipotesi di ricerca, design sperimentale, analisi dei risultati sperimentali. Quelli che seguono sono alcuni consigli specifici sugli aspetti più importanti.

4.1 Protocollo

Uno degli errori più comuni è sottovalutare lo studio di un *protocollo* sperimentale. Affinché i risultati dei test siano significativi è necessario non trascurare i seguenti aspetti:

- eliminare distorsioni sistematiche involontarie;
- isolare le variabili oggetto dello studio;
- garantire una numerosità sufficiente del campione;
- confrontare gli effetti con un gruppo di controllo.

4.2 Risultati

I risultati dei test vanno presentati in modo chiaro e completo, possibilmente indicando la significatività statistica di quanto ottenuto.

Colonna 1	Colonna 2	Colonna 3
5	8	1
6	9	2
7	0	3

Tabella 1: Tabella di esempio.

È buona norma fornire sia i dati numerici (un esempio di come si fanno le tabelle in \LaTeX è visibile in Tab. 1), sia una rappresentazione grafica (a barre, a scatole e baffi, a violino, di dispersione, ecc.).

È inoltre consigliato riportare in appendice i dati grezzi completi, in modo da permettere al lettore di ripetere eventuali test statistici.

4.3 Osservazioni

Quando si traggono conclusioni dai dati bisogna prestare attenzione a non confondere la correlazione con un rapporto di causalità. Molto spesso accade che un test suggerisca la presenza di un fenomeno, ma non dica nulla sulla causa. In questo caso bisogna formulare delle ipotesi, calcolare le implicazioni, ed eseguire un test che valuti se e quali di queste implicazioni si verifichino. Se il nuovo test falsifica la teoria, non importa quanto questa sia elegante: è falsa. Se invece il nuovo test non falsifica la teoria, allora la si può dare per “vera fino a prova contraria”.

Per queste ragioni è necessario esporre le proprie osservazioni in maniera cauta, senza andare oltre ciò che suggeriscono i dati. È certamente possibile speculare sulle cause, ma va esplicitato chiaramente, e tali speculazioni vanno supportate dalla bibliografia.

Capitolo 5

Conclusioni

5.1 Conclusioni

Nelle conclusioni si tirano le somme di quanto realizzato, facendo un riassunto stringato del lavoro svolto. In particolare vanno dichiarati punti di forza e criticità della ricerca effettuata, nonché quali aspetti dello stato dell'arte siano stati superati dal lavoro in oggetto.

5.2 Sviluppi futuri

Tra gli sviluppi futuri in genere trovano posto quelle migliorie non realizzate per mancanza di tempo, la cui necessità è emersa solo dopo i test, e che riguardano il progetto ad un livello di astrazione più alto (nel caso di tesi che si inquadrano in una linea di ricerca).

Appendice A

Il tirocinio e la correzione

Lo studente che si appresta a svolgere il lavoro di tesi generalmente incontra relatore e correlatore a inizio progetto per definire una tabella di marcia, quindi al completamento di ogni obiettivo fissato si ripresenta per un aggiornamento e un controllo. Questo è molto importante per approcciare il problema in modo organico e per evitare di illudersi di aver raggiunto un risultato sufficiente ritenendo concluso il lavoro.

Durante il lavoro è consigliato di prendere appunti su ciò che si sta facendo, in modo da avere una base per la stesura dell'elaborato.

Infine, è richiesto agli studenti di consegnare la proposta di indice prima di iniziare la stesura, per poi consegnare ogni capitolo a mano a mano che questi vengono completati. La tesi completa va consegnata per la rilettura finale a relatore e correlatore almeno una settimana prima della consegna ufficiale, per permettere un ultimo *proof reading* e l'integrazione delle eventuali modifiche richieste.

Appendice B

Il riassunto

In prossimità della data di laurea, viene richiesto agli studenti di caricare in piattaforma anche il riassunto. La sua funzione è permettere alla commissione di visionare rapidamente gli argomenti trattati da ciascun elaborato.

L'estensione di tale documento non dovrebbe superare le due pagine, e spesso una pagina è più che adeguata allo scopo. Riguardo i contenuti, il riassunto riprende – in maniera molto succinta – lo schema dell'elaborato finale, e quindi, ad esempio, la scaletta proposta in ??.

Visto lo scopo del documento, è buona prassi bilanciare il testo in modo da dedicare poche righe allo stato dell'arte, in favore delle parti originali dell'elaborato.

Appendice C

La consegna

I membri del LIM in generale non richiedono agli studenti di consegnare una copia cartacea del proprio lavoro. In prossimità della discussione viene richiesto loro di condividere in formato digitale:

- la versione finale dell'elaborato;
- il riassunto caricato in piattaforma;
- il codice sviluppato;
- tutti i materiali di supporto utilizzati e/o prodotti;
- la presentazione predisposta per la discussione.

La consegna può avvenire su supporto ottico, via chiavetta USB o attraverso i sistemi di condivisione dei file (WeTransfer, Dropbox, Google Drive, ecc.).

Appendice D

La presentazione

Gli studenti delle lauree triennali hanno a disposizione al massimo 10 minuti per presentare il proprio elaborato, che diventano 15 per le lauree magistrali.

Si raccomanda di provare ripetutamente il discorso cronometrando e considerando i tempi morti per l'accensione del PC, l'avvio dei software, il passaggio da un'applicazione a un'altra, e via dicendo. Si segnala inoltre la possibilità di prenotare l'aula della discussione per provare in anticipo i collegamenti e verificare che immagini e audio vengano correttamente riprodotti.

Per via degli stringenti vincoli temporali, difficilmente una presentazione multimediale può superare la soglia di 15 slide. Questo valore è puramente indicativo, perché dipende dalla densità di informazioni della slide e dal tempo dedicato a ciascuna slide mentre si sta presentando.

La scelta tecnologica più comune è Microsoft PowerPoint, ma qualsiasi alternativa (Prezi, Acrobat PDF, L^AT_EX, ecc.) è ammissibile.

Gli errori più comuni da evitare sono:

- slide dal contenuto testuale troppo ricco;
- slide che verranno lette parola per parola in fase di presentazione.

Entrambi i problemi si risolvono impostando il testo in maniera estremamente schematica, tendenzialmente per elenchi puntati. Si presti attenzione, però, a usare una forma omogenea. Ad esempio, sarebbe poco elegante il seguente elenco, che include aggettivi, sostantivi e frasi di senso compiuto, e non fa un uso corretto di iniziali maiuscole:

- applicativo esteticamente molto gradevole;
- C'è la possibilità di interagire con altri utenti;
- l'utente può esportare file in formato XML.

Una possibile revisione, basata solo sui sostantivi, è:

- qualità estetica;
- interazione tra utenti;
- supporto dell'XML.

Un altro consiglio per rendere la presentazione cognitivamente più gestibile da parte del pubblico è quello di aggiungere in calce ad ogni slide un indicatore dello stato di avanzamento (ad esempio “Slide 3 di 15”). In questo modo, in caso di esaurimento del tempo concesso la commissione saprà se dare modo di concludere (“Slide 14 di 15”) o se fermare il candidato (“Slide 14 di 138”). In assenza di un indicatore la commissione tende a presupporre il secondo caso.

Per quanto riguarda gli argomenti da trattare durante la discussione, è opportuno esporre brevemente lo stato dell'arte per concentrarsi sul proprio lavoro, sui test e sui risultati. Una tecnica per capire quanto tempo dedicare a ogni argomento consiste nel dare un peso a ciascuna parte di cui si vuole parlare, quindi assegnare un numero di minuti proporzionale a detto peso in modo da raggiungere il tempo massimo a disposizione.

Video dimostrativi e *live demo* sono gradite, ma è sconsigliabile dedicare troppo tempo a questi contributi, a meno che al contenuto multimediale si sovrapponga una spiegazione da parte del candidato.

Bibliografia

- [1] Luca A Ludovico. Key concepts of the iee 1599 standard. In *Proceedings of the IEEE CS Conference The Use of Symbols To Represent Music And Multimedia Objects, IEEE CS, Lugano, Switzerland*, pages 15–26, 2008.
- [2] Adriano Baratè, Luca A Ludovico, and Davide A Mauro. Tecnologie basate su xml per la fruizione avanzata dei contenuti musicali. In *Conferenza GARR_09 Selected papers*, page 43, 2010.



Progetto sviluppato presso il Laboratorio di Informatica Musicale
<https://www.lim.di.unimi.it>