

**UNIVERSITA' DEGLI STUDI DI PALERMO**  
**FACOLTA' DI SCIENZE MM.FF.NN.**

---

*Corso di Laurea in Informatica*

**TUTOR-SKY: UNO STRUMENTO PER LA**  
**DIDATTICA MULTIMEDIALE ON-LINE**

***Tesi di Laurea di***

Antonino Pace

***Relatore***

Prof. Marco Alfano

***Correlatore***

Prof. Biagio Lenzitti

---

**Anno accademico 2003/2004**

---

## Indice

<b>Introduzione.....</b>	<b>2</b>
<b>Capitolo 1 Didattica e multimedialità.....</b>	<b>5</b>
1.1La didattica e la multimedialità.....	5
1.2La didattica on line.....	6
1.3La didattica on line multimediale a supporto della didattica in presenza.....	9
1.4Esempi di didattica on-line.....	10
<b>Capitolo 2 Gli strumenti disponibili per creare un corso on-line .....</b>	<b>14</b>
2.1 Analisi di alcuni strumenti per la didattica on-line.....	15
2.2 Strumenti per rendere on-line l'Audio e il Video (streaming).....	30
2.3 Strumenti per la didattica on-line esistenti: vantaggi e svantaggi....	32
<b>Capitolo 3 La soluzione Tutor-Sky per creare corsi multimediale on-line .....</b>	<b>33</b>
3.1 Obiettivi.....	33
3.2 Scelte di Base.....	35
3.3 Architettura.....	38
3.4 Implementazione .....	41
<b>Capitolo 4 Esempio di utilizzo della soluzione Tutor-Sky.....</b>	<b>63</b>
<b>Conclusioni.....</b>	<b>73</b>
<b>Bibliografia.....</b>	<b>76</b>



## Introduzione

Le tecnologie informatiche e telematiche, che ormai si stanno affermando in tutti gli aspetti della vita quotidiana, hanno iniziato a fornire un supporto sempre più importante anche ai processi formativi. Il cambiamento del mondo didattico diventa un atto dovuto rispetto al discente, che di per sé è multimediale, trovandosi immerso in una realtà sociale sempre più informatizzata.

Se la didattica attuale si avvale sempre più della multimedialità per agevolare l'apprendimento ed offrire una formazione sempre più completa, a maggior ragione se ne avvale la didattica on-line dove gli strumenti multimediali hanno anche l'onere di rendere l'ambiente formativo il più possibile vicino a quello tradizionale, almeno dal punto di vista della comunicazione.

Attualmente è possibile trovare strumenti, gratis e a pagamento, che consentono la creazione e pubblicazione on-line di documenti didattici, ma pochi se non nessuno di questi permette al docente di inserire tutti i contenuti multimediali ritenuti necessari (si pensi ad esempio all'inserimento di più audio e video) nè consente allo studente la possibilità di scegliere la loro modalità d'utilizzo nel proprio percorso d'apprendimento.

Nel mondo didattico attuale, almeno in Italia ed in molte delle sue istituzioni didattiche, la predisposizione alla conoscenza degli strumenti necessari per una corretta didattica on-line, è poco diffusa o manca del tutto, quindi gli sforzi maggiori sono indirizzati nella creazione di strumenti con una forte impronta di flessibilità e facilità di fruizione nonché la possibilità di strutturare i propri corsi in maniera modulare e coerente.

Partendo da queste premesse in questo lavoro di tesi viene presentata una soluzione concettuale ed implementativa alle problematiche sopra esposte

costituita da un ambiente per la formazione on-line innovativo rispetto alla maggior parte degli strumenti oggi disponibili.

Infatti, l'ambiente consente al docente l'immissione di vari tipi di contenuti multimediali, consente allo studente la creazione di percorsi d'apprendimento personalizzati anche attraverso la scelta dei media a lui più adatti.

L'ambiente risulta di facile utilizzo sia per gli studenti che per i docenti che hanno la possibilità di pubblicare il proprio corso on-line e renderlo così disponibile agli studenti.

Il corso può essere strutturato su più livelli d'apprendimento, uno di base che presenta i contenuti minimi ed uno d'approfondimento in cui verranno affrontate in maniera più dettagliata i concetti generali esposti nella diapositiva di base.

Un corso sarà dotato, se il docente vorrà, di un insieme di contenuti multimediali che vanno dalla registrazione in video o sonora della lezione al sottofondo musicale che il docente ritenga adatto per creare un ambiente più confortevole per l'apprendimento. Sarà poi lo studente a scegliere quali contenuti utilizzare e la relativa modalità d'utilizzo.

In particolare nel primo capitolo saranno trattati i rapporti che legano la didattica in generale e la multimedialità, come questa possa stimolare l'apprendimento degli studenti, e le caratteristiche peculiari della didattica on-line rispetto a tutti gli altri tipi di FAD (formazione a distanza) ed, infine, come la didattica on-line possa essere usata per supportare la didattica in classe assieme a qualche esempio di contesto applicativo.

Nel secondo capitolo saranno descritti alcuni degli strumenti per la creazione e la pubblicazione di contenuti formativi on-line oggi esistenti, come ATutor e ADA Lesson Generator, se ne farà un confronto mettendo in evidenza le caratteristiche (pregi e difetti) di ogni strumento. Sarà descritta anche in maniera elementare la tecnica di streaming fondamentale per la sincronizzazione on-line dell'audio e del video.

Nel terzo capitolo verrà descritta “Tutor-Sky” una soluzione che permette la creazione e la gestione di un corso multimediale on-line, ne verranno illustrati gli obiettivi, le scelte fatte per implementarla, l’architettura della piattaforma e la sua implementazione.

Il quarto capitolo avrà come argomento l’uso di “Tutor-Sky” , si vedrà passo dopo passo come sia possibile creare un corso, e come possa essere seguito on line dagli studenti.

# Capitolo 1 Didattica e multimedialità

## *1.1 La didattica e la multimedialità*

Sempre più, oggi, in un mondo in cui le persone sono circondate da immagini, suoni e video, si avverte l'inadeguatezza del solo libro come strumento didattico e si sente forte il bisogno di affiancarlo con un'altri nuovi strumenti tra cui la multimedialità. Molti docenti in molte scuole integrano le loro lezioni con ipertesti e cd-rom multimediali che sempre più spesso vengono allegati ai libri di testo. Multimedialità ed ipertestualità non costituiscono saperi nuovi, ma forme nuove di organizzazione del sapere. Possiamo pensare alla multimedialità come ad una realtà didattica nuova, vale a dire qualcosa che permette di modificare il nostro metodo e modello di insegnamento, suscettibile di ampliare e potenziare tanto la capacità didattica quanto quella di apprendimento favorendo la percezione della complessità dei problemi e delle loro interrelazioni.

Essa non ci dà contenuti nuovi, ma nuovi modi di organizzare quelli che già abbiamo. La multimedialità, favorita dall'utilizzo di più media, promuove, in modo accelerato, le forme di intelligenza intuitiva che a loro volta aiutano lo studente a comprendere i concetti e le abilità che il corso vuole trasmettere sviluppando in loro maggiori capacità mnemoniche e di apprendimento. E' dimostrato infatti che coinvolgendo più sensi durante il processo di memorizzazione il cervello è facilitato ad apprendere più velocemente e a memorizzare meglio, dato che ha più punti di riferimento. La didattica di oggi per questi motivi si orienta sempre con maggiore convinzione a questo nuovo modello didattico che si prospetta come un cambiamento radicale per la didattica del futuro.

## **1.2 La didattica on line**

Al giorno d'oggi la didattica on-line è una realtà crescente per quanto riguarda i corsi di studi di livello avanzato, come quelli universitari o i corsi di aggiornamento professionale, mentre è ancora poco visibile la sua presenza per quanto riguarda l'istruzione di base. Questo è anche dovuto a caratteristiche intrinseche dell'e-learning, in quanto occorre avere sviluppato capacità dialettiche ma soprattutto auto organizzative oltre alle inevitabili conoscenze del mezzo di comunicazione, cioè la rete, affinché uno studente possa seguire proficuamente un corso on-line. Questo risulta più complesso per un bambino che va alla scuola elementare, non tanto per la mancanza di conoscenze del mezzo, che per alcuni bambini non è affatto vera, ma piuttosto per la mancanza della capacità di creare un percorso di apprendimento.

La didattica on-line rivoluziona il modo di insegnare e si differenzia non solo dalla didattica classica, ma anche degli altri tipi di didattica a distanza. Può essere, in qualche modo, paragonata alla rivoluzione copernicana dell'astronomia. Infatti, come Copernico "cambiò" il centro del sistema solare, così l'e-learning sta cambiando il centro gravitazionale dell'attività didattica che non è più il docente, ma diventa lo studente. Infatti con essa non è più il docente a trasmettere le sue conoscenze già confezionate al discente, ma è quest'ultimo che deve costruirsi un percorso didattico ricercando il materiale nella rete, ma soprattutto confrontandosi e dialogando con gli altri studenti e con i tutor per approfondire e criticare ciò di cui ha fatto esperienza di apprendimento. E' sempre lo studente che, inoltre, può testare la sua preparazione on-line attraverso quiz preparati dal docente. Sono queste caratteristiche, cioè l'interazione con gli altri studenti e con i tutori e la possibilità di auto testare la preparazione, che rendono la didattica on-line più efficace degli altri tipi di didattica a distanza come gli insegnamenti per

corrispondenza della scuola Radio Elettra o gli insegnamenti universitari a distanza del Consorzio Nettuno.

Un dialogo alla pari tra docente e studente è possibile su internet perché il mezzo pone tutti i fruitori su uno stesso piano e quindi elimina quella differenza di autorità che in un modo o nell'altro si viene a creare tra il professore e lo studente in un corso frontale. La rete permette infatti di incrementare il grado di interazione tra studenti e tutor e, soprattutto, fornisce le basi per instaurare processi di collaborazione effettiva anche tra gli studenti stessi durante il percorso formativo. Da tutto ciò nasce una nuova didattica cooperativa e distribuita che pone il discente al centro del processo formativo e lo considera parte attiva nella sua formazione.

All'interno dell'e-learning la funzione del docente si identifica più in quella di un facilitatore di conoscenze che non in quella di colui che infonde la sua conoscenza. Il docente può solo indicare e segnalare le linee guida del proprio corso ma non è più lui che gestisce in prima persona lo svolgimento della lezione.

Molti risultati sono stati raggiunti dalla didattica on line ma molto salda rimane la didattica frontale cui molti studenti non riescono ancora a farne a meno. Questo è soprattutto vero nel nostro paese dove l'e-learning è ancora in una fase principalmente sperimentale. La lezione in presenza fa parte della nostra cultura così tanto che lo studente è diffidente già all'idea di poter seguire un corso on-line [1]. Molti adducono come motivazione della loro preferenza la necessità durante la lezione di interagire con il professore, ma è pur vero che in molti casi le lezioni, soprattutto quelle universitarie, sono frequentate da un numero di studenti tale da rendere la lezione frontale impersonale, tanto quanto quella che potrebbe essere una lezione on-line. Nonostante ciò, molti studenti continuano a preferire la lezione frontale a quella virtuale, anche senza aver mai provato a seguire un corso in rete, ma solo per il pregiudizio che si ha, in generale, nei confronti della didattica on line.

Per cercare di rendere una lezione on-line il più simile possibile a quella reale si possono rendere i contenuti on-line multimediali, magari aggiungendo agli schemi e alle immagini, che oramai fanno spesso parte dei contenuti creati per le lezioni anche da docenti poco esperti, la voce del docente che descrive il materiale o un video della lezione stessa valorizzando così le molteplici forme e modalità di comunicazione. Questo aumenta certamente il livello di interesse dello studente e la sua attenzione durante la lezione. Utilizzando questi mezzi infatti il discente è tenuto non solo a leggere i testi delle diapositive o ad interpretare i grafici, ma anche ad ascoltare le parole dell'insegnante e quindi coinvolgere più sensi contemporaneamente nel processo d'apprendimento. Inoltre la multimedialità on-line apporta tutti quei vantaggi prima descritti che già portava alla didattica in presenza. Quindi la possibilità di rendere l'ambiente formativo più vicino a quello di una lezione frontale in rete non serve solo a dare un maggiore peso al corso on-line, ma anche, e soprattutto, a motivare ed aiutare lo studente ad apprendere così come la multimedialità aiuta la didattica frontale.

Questi strumenti uniti a quelli propri di Internet, come le chat i forum o anche le mailing list, definiscono un vero e proprio nuovo ambiente d'insegnamento, la classe virtuale, che sembra sempre di più essere il futuro della nostra istruzione superiore e che può interessare anche quella inferiore, soprattutto per l'integrazione della didattica con materiali multimediali in classe.

La formazione on-line, per raggiungere gli obiettivi prefissati, deve presentare alcune caratteristiche di base, cioè essere reticolare, interattiva, cooperativa e multimediale [2].

### **RETICOLARE**

Per ottimizzare la produzione, occorre che le stesse informazioni siano fruibili da utenti diversi. D'altra parte, per personalizzare il corso, deve essere possibile l'attivazione di link diversi da uno stesso nodo. La struttura deve quindi essere composta da unità collegate con link non statici. Questo vuol dire

che il percorso può cambiare ad ogni consultazione, sia che si tratti di utenti diversi, sia che a navigare sia lo stesso utente.

### **INTERATTIVA**

Sarebbe auspicabile avere una possibile sorgente di interazione. Ma questo non deve tradursi in una forzatura che allontana chi è semplicemente 'in visita' e non ha ancora deciso di impegnarsi.

### **COOPERATIVA**

Uno studente dovrebbe avere la possibilità di poter interagire con gli altri studenti attraverso l'uso di chat e forum in cui possano accedere solo gli iscritti del corso e che hanno deciso di partecipare attivamente al corso. Uno studente o un tutor all'interno del forum dovrebbe poter aggiungere un commento sul corso o su quello di un altro studente.

### **MILTIMEDIALE**

La multimedialità diventa indispensabile per un corso integrato e completo, visto che, come già mostrato, aiuta l'apprendimento e facilita in qualche modo anche l'insegnamento perché rende on-line i materiali della lezione. Lo studente, quindi, prima di chiedere chiarimenti al docente potrà rivedersi da solo la lezione e chiarire magari i suoi dubbi indipendentemente.

## ***1.3 La didattica on line multimediale a supporto della didattica in presenza***

La didattica on-line non deve essere vista solo come qualcosa di separato dal resto del mondo dell'istruzione o addirittura in opposizione al mondo della didattica classica. Infatti essa è molto efficace anche come integrazione ad un corso frontale, creando un corso "misto" che sfrutti l'e-learning come strumento a supporto per insegnamento, non disprezzando naturalmente lo sviluppo e l'utilizzo dei mezzi multimediali.

Un corso misto soddisfa la necessità degli studenti di un avere un riferimento nel docente ed aggiunge ad un corso normale tutti gli strumenti collaborativi e interattivi di un corso on-line, creando inoltre all'interno dell'ambiente virtuale la possibilità di creare rapporto maggiormente equilibrato tra docente e studenti.

Lo sfruttamento della didattica on-line all'interno di un corso frontale alleggerisce anche il lavoro del docente e permette un recupero dello studente rispetto alle lezioni più veloce e dinamico anche durante il corso.

Lo studente avendo a disposizione il materiale on-line ha l'opportunità di affrontare preventivamente la lezione, cioè può, volendo, leggere prima di andare in aula la lezione on-line in maniera tale da avere già un'idea degli argomenti che verranno trattati e seguire meglio la lezione in presenza. Inoltre il materiale può essere usato dallo studente per rileggere e riaffrontare gli argomenti che magari sono stati meno chiari durante la lezione in aula, ed ancora lo studente potrebbe approfondire la lezione seguendo dei link che l'insegnante può aggiungere alle diapositive del corso on-line.

Altra caratteristica di un corso on-line, che è sicuramente molto utile per un corso tradizionale, è la possibilità dell'auto valutazione da parte dello studente stesso che così durante lo svolgimento del corso può verificare il suo grado di apprendimento e, se è il caso, cambiare approccio al corso in maniera tale da colmare le possibili lacune. Le attività di un corso on-line possono quindi in qualche modo compensare le possibili inefficienze del corso in presenza.

#### ***1.4 Esempi di didattica on-line***

Su Internet molti sono gli esempi di ambienti di didattica on-line vista sia come corso a se stante sia come supporto per un corso in presenza.

Ambienti del primo tipo li troviamo su "corsi.altrascuola.it" [3], un completo ambiente di e-learning dove è possibile registrarsi per seguire alcuni corsi gratuiti, ma dove la maggior parte dei corsi è a pagamento. L'approccio ai corsi

si ispira ad un modello costruttivista della formazione dove l'insegnamento non consiste nella trasmissione di nozioni e concetti dal docente al discente, ma nella costruzione di ogni studente del suo sapere [4]. Una volta iscritti, la piattaforma mette a disposizione chat, forum ma soprattutto la presenza di un tutor che segue una classe, cioè gli iscritti al corso, oltre ad un insieme di contenuti in realtà poco multimediali.

Altra interessante esperienza di didattica on-line è quella di un Corso di Laurea interamente on-line. Esistono alcune Università in Italia che già forniscono questo servizio e in particolare esiste il Corso di Laurea in Scienze e Tecnologie per l'Ambiente ed il Turismo (STAT) [5] presso la Facoltà di Scienze MM.FF.NN dell'Università di Palermo.

Questo corso è un vero e proprio corso di laurea triennale a tutti gli effetti ed è indirizzato a tutti coloro che hanno intenzione di conseguire la laurea, ma che non hanno il tempo per seguire le lezioni. Infatti in questo Corso di Laurea tutte le lezioni dei corsi sono on-line, ma la verifica finale, cioè l'esame, è rimasto frontale.

Questo progetto è realizzato in collaborazione con l'Istituto per le Tecnologie Didattiche [6], del Consiglio Nazionale delle Ricerche, sezione di Palermo, che ha curato gli aspetti tecnologici e metodologici del corso on-line. Le lezioni sono distribuite mediante un ambiente di apprendimento a cui possono accedere soltanto gli studenti iscritti, dotati di un nome d'accesso ed una password. Da qui è possibile non solo seguire i corsi ma anche iscriversi agli esami, altra cosa molto utile non solo in un corso on-line ma per qualsiasi corso di laurea.

Non solo le università si stanno attrezzando in questa direzione, ma anche sempre più scuole si affacciano al mondo della didattica on-line. Un interessante esempio è il corso di inglese del liceo classico Garibaldi di Palermo [7]. Uno studente può qui valutare la sua preparazione e il suo livello di conoscenza della grammatica, dei vocaboli grazie a quiz, parole crociate ed altri esercizi proposti.

Se molti sono ormai i corsi on-line ancora di più sono i corsi creati da docenti di istituti scolastici e accademici per supportare i corsi frontali. Tra questi ricordiamo "Cone" [8], il supporto on-line al corso di Reti di calcolatori del Corso di Laurea in Informatica dell'Università di Palermo. Da quest'anno, infatti, viene utilizzato direttamente il materiale on-line per fare lezione in aula. Oltre al materiale on-line (che può anche essere scaricato), il sito contiene notizie utili sul corso, per esempio il programma dettagliato, e permette un'autovalutazione passo dopo passo tramite delle domande a risposta multipla di cui è possibile avere il riscontro immediato. E' presente anche una sezione in cui gli studenti possono esprimersi e dare una loro interpretazione su Internet.

La parte didattica vera e propria del sito è divisa in due parti: una teorica e un'altra di laboratorio dove è presente un interessante laboratorio on-line che permette di esercitarsi nella configurazione di un router e dove si trovano alcune proposte di esercitazioni numeriche comprensive delle soluzioni. Alle lezioni del docente, divise modularmente secondo una gerarchia protocollare, si affiancano ulteriori letture e approfondimenti che lo studente è libero di scegliere. Questo rende il corso più coinvolgente e permette allo studente di assecondare i propri ritmi di apprendimento.

Altro esempio di corso che si può seguire on-line, preso però stavolta, dal mondo aziendale è il corso della Alcatel, corso comunque pubblico, sulle tecnologie e sui principi delle reti telematiche e di internet (The Fundamentals of Communications [9]).

Questo corso si avvale di un player lesson in flash, che unisce agli strumenti testuali anche quelli multimediali, come la voce di un docente che illustra i contenuti esposti.

Il corso è modulare e reticolare e in più dà allo studente la possibilità di affrontare una parte del corso o tutto, secondo i suoi bisogni. Per fare questo è disponibile una mappa dell'intero corso dove sono presenti i link alle varie parti del corso, sia intese come moduli, sia come lezioni.

La parte più interessante rimane comunque il lesson player. Completamente realizzato in flash, potrebbe da solo rappresentare l'applicazione, date le numerose operazioni possibili in esso e l'integrazione della mappa del corso nello stesso player.

Anche la BBC [10], propone in rete un corso di base sui fondamenti della rete, però a differenza di quello dell'Alcatel, questo corso è molto meno tecnico dato che si riferisce ad un bacino di utenti di inesperti fruitori di internet. Lo scopo del corso è proprio quello di illustrare i principali strumenti che si hanno navigando in internet come il browser, l'email, i motori di ricerca. E' proprio un corso per chi muove i primi passi e l'interfaccia infatti risulta semplice ed efficace. Manca però l'integrazione con contenuti multimediali, che non sono integrati nel lesson player che anche qui è implementato in flash. Al termine di ogni unità è presente anche un quiz a cui viene dato un feed-back immediato e grazie al quale un utente può mettere alla prova quello appena appreso.

## Capitolo 2 Gli strumenti disponibili per creare un corso on-line

Navigando nella rete oggi non è difficile incontrare piattaforme che permettono ad un utente di seguire corsi on-line, più o meno strutturati come una lezione frontale. Alcune offrono anche la possibilità di inserire contenuti, magari creati on-line, costruendo così un intero corso. Tra queste esistono ATutor [11], Spaghettilearning [12], Moodle [13] e altre ancora. Esiste anche un'altra categoria di strumenti software che crea corsi strutturati anche in maniera complessa off-line, che poi si possono anche pubblicare in appositi server. Tra questi ricordiamo ADA [14] della LynxLab, Breeze [15] della Macromedia e Producer della Microsoft [16].

Gli strumenti della prima categoria appartengono agli LCMS (Learning Content Management System) [17] cioè piattaforme dedicate all'erogazione di corsi. Spesso un LCMS consente anche la creazione e la gestione dei contenuti e l'importazione di contenuti conformi allo standard SCORM (*Sharable Content Object Reference Model*) [18]: lo standard internazionale per l'E-learning. Lo SCORM rappresenta il modello di riferimento per la realizzazione di piattaforme e contenuti per l'e-learning ed è un insieme di specifiche adattate da molte fonti per fornire una suite completa di strumenti di e-learning che permettono interoperabilità, accessibilità e riusabilità dei contenuti formativi basati su web, alcuni degli obiettivi stessi dell'e-learning. L'architettura di SCORM è composta da quattro elementi essenziali:

1. Learning Object: la cellula minima della quale si compone un corso che comprende in se una nozione autosufficiente dal resto. Uno stesso Learning Object se compatibile con lo standard SCORM, può essere utilizzato all'interno di corsi diversi (architettura modulare);

2. Learning Management System (LMS): il sistema di gestione del corso che ne consente la fruizione;
3. Course Structure Format (CSF): file d'interscambio in grado di tradurre lo stesso corso in LMS differenti;
4. Runtime, il sistema che avvia il corso, soddisfacendo le richieste dell'utente finale.

### ***2.1 Analisi di alcuni strumenti per la didattica on-line***

Dopo aver affrontato in generale alcune tipologie di piattaforme ed aver descritto in sintesi gli standard a cui queste fanno riferimento, adesso confrontiamo alcuni degli strumenti più utilizzati (ATutor, Moodle, ADA Lesson Generator, Producer e Breeze) descrivendo le caratteristiche di ogni applicazione e analizzando i mezzi che questi strumenti mettono a disposizione ad un docente che voglia creare un corso da pubblicare on-line.

**ATutor** [11]. ATutor è un LCMS (quindi un componente dello standard SCORM) e, oltre alle caratteristiche normali di uno strumento del suo genere, ATutor permette[19] (come si vede anche nella figura 2.1):



**Fig. 2.1.** Alcuni strumenti forniti ad uno studente da ATutor.

- Interfaccia diretta per l'amministrazione del corso;
- Possibilità di costruire corsi progettando una serie di pagine di contenuti attraverso un editor HTML visuale on-line;
- Possibilità di importare contenuti conformi allo standard SCORM;
- Possibilità di esportare i contenuti;
- Funzione integrata di posta attraverso messaggi interni con possibilità di contattare simultaneamente tutti gli iscritti ad un particolare corso ;
- Forum e Chat;
- Help contestuale e possibilità di accedere ad ulteriori aiuti nel sito;
- Possibilità di creare un glossario;
- Possibilità di gestire l'upload dei file sul server per i docenti e di gestire i file zip;

ATutor è un programma open-source elaborato da un gruppo di studiosi dell'Università di Toronto, in particolare dall'[20][Adaptive Technology Resource Centre \(ATRC\)](#). E' distribuito con licenza GPL(GENERAL PUBLIC LICENSE) [21] e può essere installato in qualsiasi server che supporti il linguaggio php e il database Mysql. L'installazione è abbastanza semplice basta avere una conoscenza di base di Mysql e php per configurare la figura dell'amministratore. E' un ambiente formativo che può essere configurato in modo da supportare gli stili di apprendimento e le abilità di coloro che lo utilizzano. E' stato progettato per essere adattabile e accessibile. La documentazione fornita nel sito è completa e facile da utilizzare e in più esiste anche un forum ufficiale di supporto on-line, solamente in inglese, dove porre quesiti e ricevere risposta dalla comunità sia su come sfruttare ATutor sia per risolvere problemi durante la configurazione. In questo ambiente virtuale sono gli utenti-istruttori a creare i corsi, assemblando pacchetti informativi già esistenti o creandoli appositamente, magari on-line, e a gestire l'amministrazione del corso e dello svolgimento delle discussioni all'interno dei forum creati appositamente. Per accedere alla piattaforma è prevista una identificazione tramite login e password. Questo è indispensabile per una gestione curricolare dei corsi oltre che utile per una gestione oculata del sistema.

Un utente-istruttore può dare ad un suo corso l'attributo di:

- pubblico: corsi visibili a tutti, anche da coloro che non si sono iscritti ad ATutor;
- protetto: corsi a cui hanno accesso chiunque abbia un account;
- privato: corsi a cui hanno accesso solo coloro che sono iscritti e hanno l'approvazione del proprietario del corso.

Naturalmente questa resta una scelta del docente e non è detto che vi siano dei corsi pubblici nelle piattaforme esistenti. Un docente può chiedere all'amministratore di attivare la tracciabilità degli accessi del corso. In questo

modo il proprietario di un corso può vedere il numero di accessi e per ognuno di essi può sapere quando e quanto è durato un accesso nonché l'identità delle persone che accedono. E' anche possibile visualizzare un istogramma riassuntivo degli accessi ad ogni strumento del corso (es. numero degli accessi alla chat).

Sarà sempre il docente a creare il forum del suo corso. Solo egli ne ha l'autorità. Lo studente può accedere e partecipare al forum, naturalmente sempre rispettando le regole di accessibilità ai corsi pubblici, protetti e privati.

Lo studente potrà anche scaricare file grazie a link inseriti dall'istruttore e utilizzare il servizio di e-mail interna per inviare messaggi agli altri utenti che sono iscritti allo stesso corso e per dialogare e discutere degli argomenti del corso.

Gli istruttori possono creare test on-line che gli studenti potranno eseguire entro il tempo di vita del test settato dagli istruttori. ATutor però non permette un feed-back immediato e neanche la possibilità di sapere immediatamente le risposte esatte, ma deve essere il docente a controllare il test e ad inviare il risultato.

Manca in questa piattaforma la possibilità di integrare in modo semplice contenuti multimediali audio e video alle lezioni. Il modo più semplice è quello di aggiungere dei link a questi contenuti in maniera che poi sia il browser a gestire il file, eventualmente attraverso applicazioni esterne, ma in questo modo comunque i contenuti rimangono esterni alla lezione e non è possibile ascoltare, ad esempio, la voce dell'istruttore che illustra i contenuti mentre avanzano le diapositive. Altro metodo per inglobare realmente audio e video, almeno così è spiegato nella guida, è quello di scrivere il codice sorgente HTML con l'editor on-line usando i tag HTML "object" ed "embed" che però non sono certo semplici da usare per un utente medio data l'enorme numero di parametri che è possibile configurare e la scarsità di manuali sull'argomento.

Inoltre non sempre l'integrazione funziona a dovere, io stesso ho provato ad utilizzarla con scarso successo.

**Moodle** [13]. Anche Moodle come ATutor è un LCMS ed è ha molti punti in comune con quest'ultimo. In particolare, sono comuni le piattaforme utilizzate per la loro creazione (php, MySql) e anche il tipo di licenza con cui vengono rilasciati (GPL), per non parlare delle funzioni pressoché identiche.

La parola Moodle è l'acronimo di Modular Object-Oriented Dynamic Learning Environment (Ambiente di Apprendimento Dinamico Modulare Orientato agli Oggetti) [22]. Moodle è volto a creare un ambiente educativo basato sul costruttivismo. Le sue funzioni principali sono:

- la gestione dei corsi e degli utenti;
- la gestione dei contenuti;
- la gestione e la tracciabilità dell'interazione tra studenti e contenuti, tra studenti e studenti e tra studenti e docenti,

Naturalmente, essendo stato progettato con il linguaggio di scripting Php, Moodle è adattabile a qualsiasi piattaforma che supporti tale linguaggio come Unix, Windows o Linux. La scelta del database dove memorizzare i dati, a differenza di ATutor, non è obbligatoria, ma si può scegliere fra MySql, PostgreSQL, o altri ancora. Tra le caratteristiche più importanti di Moodle sicuramente possiamo considerare:

- l'accesso previo autenticazione (utente, password) secondo tre livelli: amministratore, docente e studente;
- la possibilità di importare un'anagrafica utenti esistente;
- la possibilità di attivare e gestire corsi in tre diversi modalità: per settimana(fig. 2.3), per argomento (fig. 2.2) o con un forum generale(fig. 2.4);

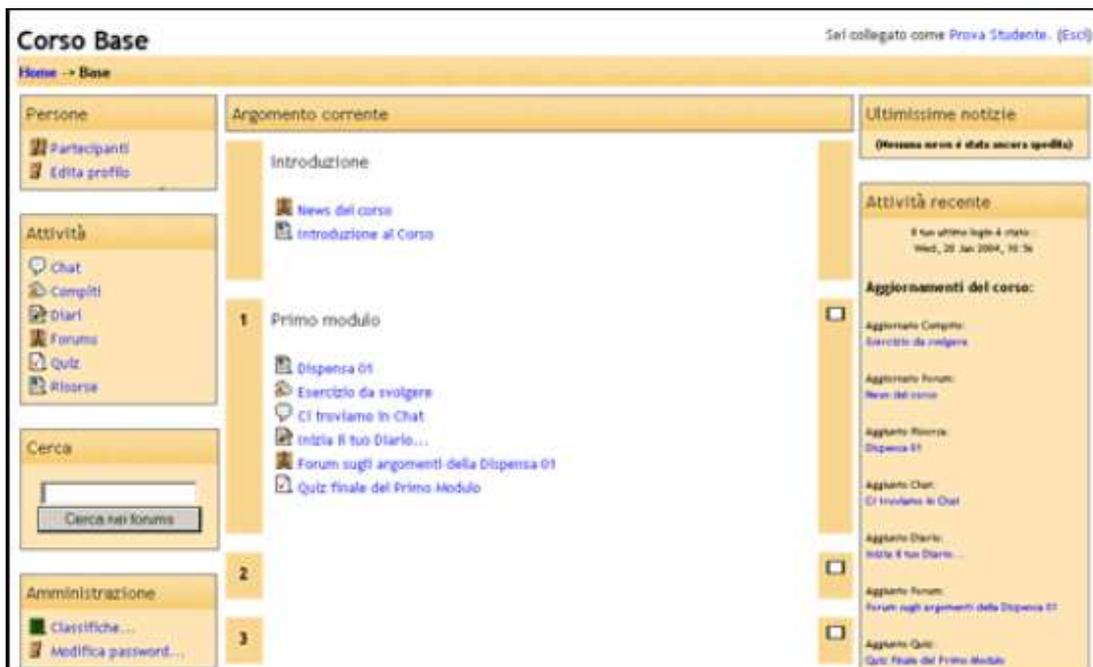


Fig. 2.2. Gestione per argomenti.

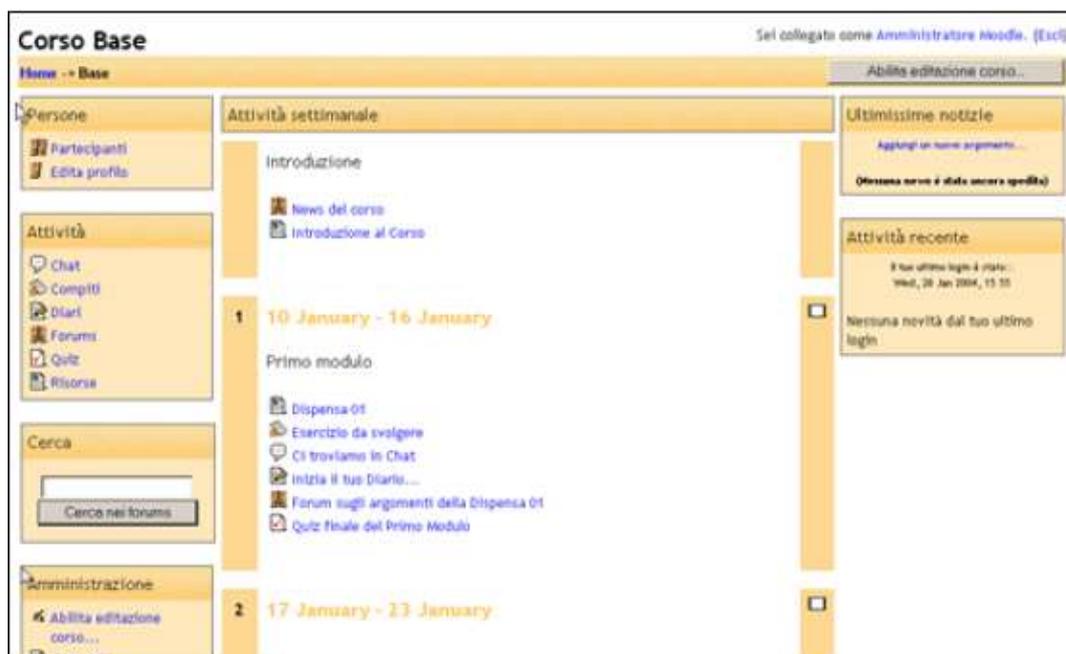


Fig. 2.3. Gestione per settimana.



Fig. 2.4. Gestione con un forum generale.

L'installazione risulta semplice come quella di ATutor. Notevole è invece la differenza di dimensione dei file per l'installazione: ATutor occupa 1,06 MB mentre Moodle occupa 11 MB. Le principali caratteristiche di Moodle sono:

- personalizzazione del layout per i singoli corsi;
- creazione di forum;
- possibilità di settare annunci settimanali;
- la gestione di file di log, ossia documenti in cui vengono registrati le ultime attività, divisi per utenti;
- possibilità per i docenti di assegnare compiti agli studenti da svolgere entro una certa data;
- possibilità di tenere traccia dei compiti assegnati e delle valutazioni effettuate dai docenti (devono essere però i docenti ad inserirle manualmente, il sistema non valuta in modo autonomo);
- possibilità di creare forum di discussione fra insegnanti;
- possibilità di aggiornare i contenuti dei corsi;

- possibilità per gli studenti di porre domande e tenere un diario del corso.

La modalità di creazione di un account è decisa dall'amministratore il quale può scegliere tra la conferma via e-mail, l'inserimento manuale, che è molto utile per controllare gli accessi alla piattaforma e l'opzione senza autenticazione in cui gli utenti possono registrarsi e creare iscrizioni valide immediatamente senza autenticazione di un server esterno e senza conferma tramite email.

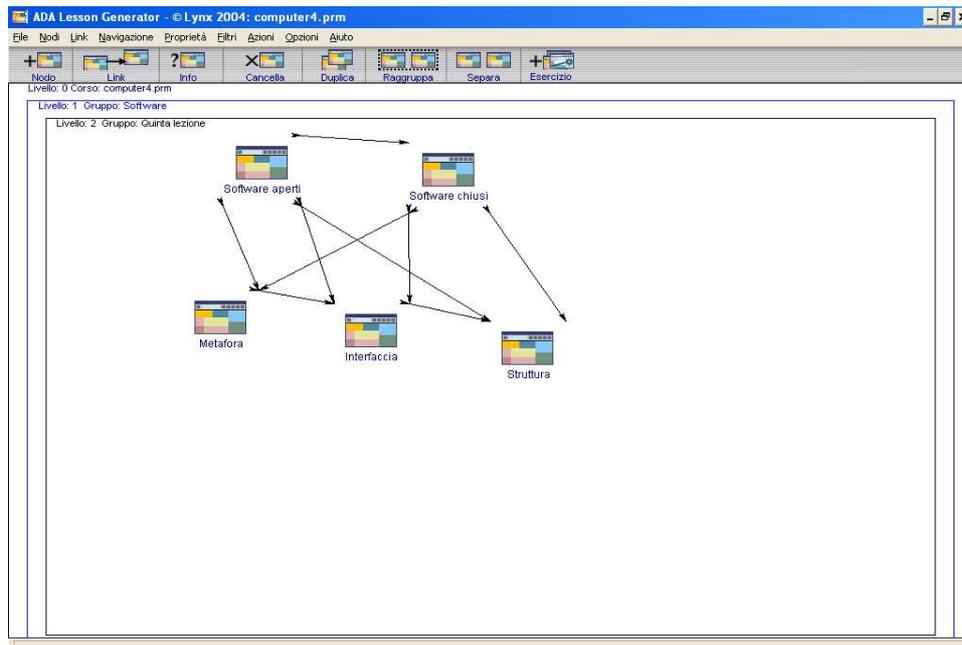
Il docente può gestire l'accesso ai propri corsi scegliendo i propri studenti, abilitando l'accesso anche ad altri insegnanti ed infine potendo anche abilitare un utente visitatore mediante una password. Sia l'amministratore che l'istruttore possono visualizzare i log delle lezioni. Naturalmente gli istruttori possono visualizzare i log soltanto dei propri corsi. All'interno dei log sono contenuti: il numero di iscritti, l'identità dell'utente e le azioni che ha eseguito.

La documentazione è fornita nel sito ufficiale [www.moodle.org](http://www.moodle.org) e contiene una guida per l'installazione, un manuale docente e un manuale studente. Inoltre è anche presente sulla rete un'attiva comunità italiana che partecipa a diversi forum.

Molto interessante e completa è la gestione dei file. Oltre all'upload sul server e alla gestione delle cartelle e dei singoli file, infatti, Moodle permette di gestire archivi compressi di file nel formato zip, permettendo di aprirli ed estrarli. Gli istruttori possono creare test on-line e, come in ATutor, dargli una scadenza, ma a differenza di ATutor è possibile avere un feedback immediato se il docente inserisce un commento ad ogni risposta tipo del tipo "risposta esatta". Così come in ATutor, Moodle permette la visualizzazione dei contenuti multimediali sotto due forme. O come link visualizzabili all'esterno della piattaforma oppure scrivendo il codice HTML per includere gli oggetti, che, come illustrato in precedenza, non è un'operazione semplice per l'utente medio. E anche se si riuscisse a scrivere il codice comunque la piattaforma non permetterebbe di avere un audio in sincrono con il contenuto testuale perché

per far ciò è necessario che i contenuti audio o video siano distribuiti da un server di streaming apposito,.

**ADA Lesson generator** [14]. ADA Lesson Generator o ADA L.G. è uno strumento differente dai due precedenti. E' stato creato da Lynx, un'azienda italiana, e distribuito con licenza (GPL). La differenza principale consiste nel fatto che ADA L.G. non è un'applicazione web ma un'applicazione tradizionale ed è precisamente il modulo autore del sistema ADA che è una piattaforma software per l'apprendimento on-line. Il punto di forza di questo strumento è che non bisogna essere esperti informatici per poterla usare. ADA Lesson Generator permette di creare corsi strutturati in maniera anche complessa senza perdere la visione d'insieme del corso. I contenuti dei corsi vengono organizzati in moduli, gruppi e nodi. Il corso viene visualizzato grazie ad una mappa grafica e ad un indice generato automaticamente. E' possibile strutturare il corso a livelli: nel livello principale sono visibili i moduli che compongono un corso. Ogni modulo può essere a sua volta composto da gruppi di nodi e da nodi singoli come è mostrato nella Fig. 2.5. I moduli sono la principale unità informativa. Essi sono completi ed esprimono un argomento quasi a se, "autosufficiente". I corsi, realizzati con ADA Lesson Generator, hanno una struttura modulare tale da facilitare l'accesso alle unità formative (nodi).

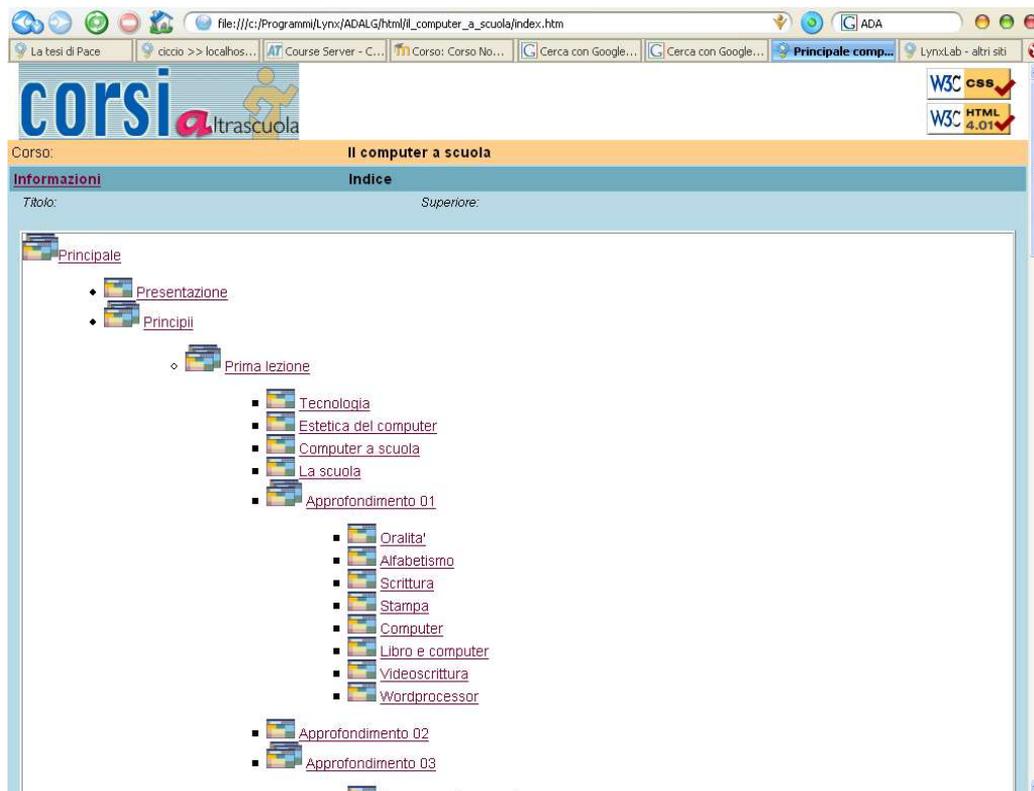


**Fig. 2.5.** Un esempio di come organizzare un modulo.

È possibile fornire informazioni metatestuali sui corsi e sugli oggetti che li formano grazie ai livelli, ai collegamenti e alle proprietà.

E' possibile aggiungere ai nodi dei corsi alcuni contenuti multimediali, ma semplicemente come sussidi al nodo o meglio come link scaricabili o gestibili da qualche applicazione esterna. Non si possono invece inserire contenuti multimediali come oggetti integranti della risultante pagina HTML. Anche ADA L.G. permette di creare test. Il docente può specificare una o più risposte esatte dando un valore di correttezza percentuale. Il sistema fornirà un feedback immediato informando lo studente se ha dato la risposta giusta o sbagliata.

Questa comunque rimane l'unica fonte di interazione tra il corso e lo studente. Infatti all'interno dell'anteprima HTML mancano decisamente luoghi virtuali di discussione come una chat o un forum in cui gli studenti possano chiedere spiegazione ai docenti e dialogare con gli altri studenti. Una volta creato il corso è possibile esportarlo in formato XML con tutti i riferimenti ai media presenti ad un server ADA oppure lo si può visualizzare in un'anteprima in HTML.



**Fig. 2.6.** Pagina iniziale dell'anteprima HTML.

E' da notare, nella figura 2.6, come la pagina iniziale contenga un indice cliccabile che porta ai relativi nodi i quali possono contenere sia altri nodi che contenuti veri e propri. Questa mappa viene creata secondo le specifiche del corso e ricalca l'andamento voluto dal docente. Lo studente può seguire l'andamento voluto dal docente e suggerito dalla struttura stessa oppure può semplicemente leggere ciò che più lo interessa e seguire un percorso personale, ma non può in alcun modo personalizzare l'ambiente di studio, dato che l'ambiente è completamente deciso dal docente.

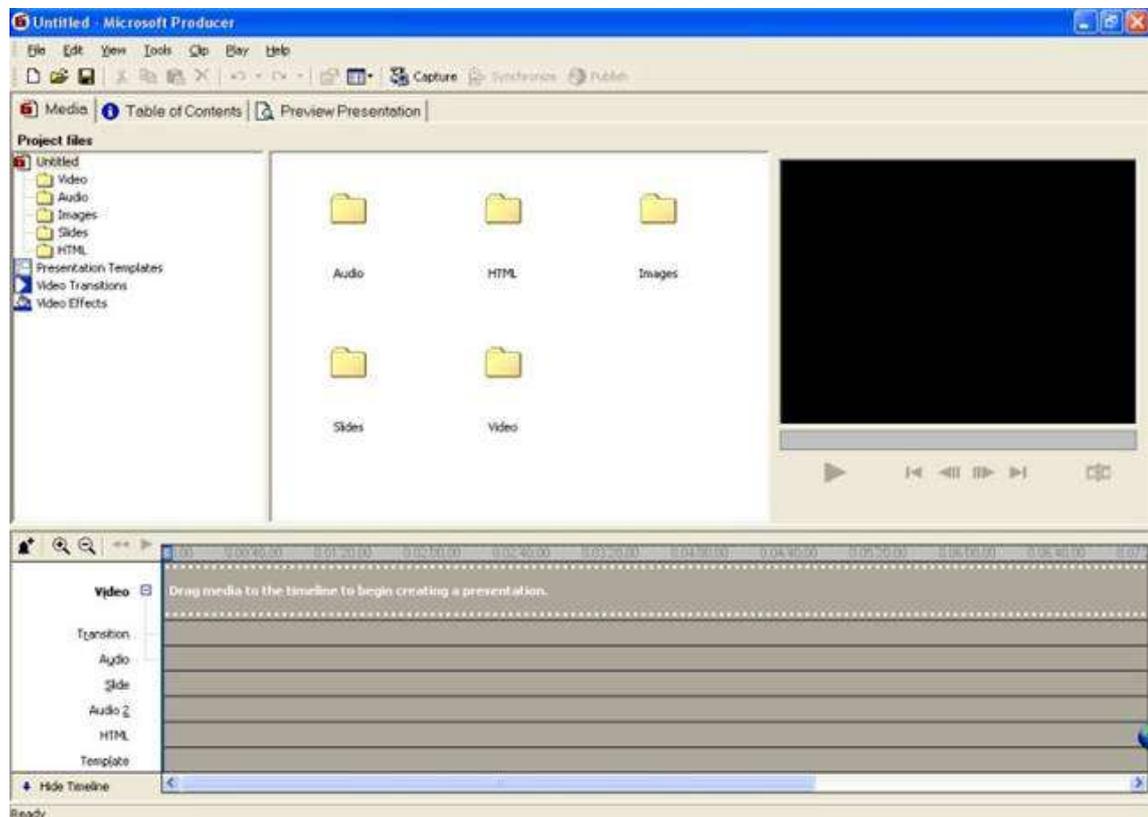
**PRODUCER**[16]. Producer è un'applicazione Microsoft distribuita gratuitamente a tutti i possessori autorizzati di Microsoft PowerPoint. E' una componente aggiuntiva di quest'ultimo grazie al quale è possibile trasformare diapositive, file audio, video e immagini in presentazioni on-line multimediali. Microsoft Producer si integra, inoltre, in Office XP. Richiede Windows 2000 Professional

o XP. In <http://www.microsoft.com/office/powerpoint/producer/EvalGuide.doc> è disponibile una guida in inglese.

Producer è stato concepito essenzialmente per integrare file multimediali (filmati, audio, pagine HTML, immagini) con un insieme di diapositive PowerPoint, sincronizzando queste due componenti e permettendo di mixare tali contenuti.

Il prodotto finale può essere salvato sia sul disco rigido, che in rete basta che i file risultanti non siano particolarmente pesanti. Tali file sono comunque visibili solo attraverso Internet Explorer 5 o versione superiore.

Dal punto di vista didattico si prospetta come un ambiente molto interessante. Infatti, visto che l'uso più immediato che si può immaginare è quello di un'integrazione audio-video di una classica presentazione in stile Powerpoint, si può pesare di creare relazioni più ampie come la realizzazione di lezioni multimediali.



**Fig. 2.7.** Producer visto dal lato di chi crea i contenuti.

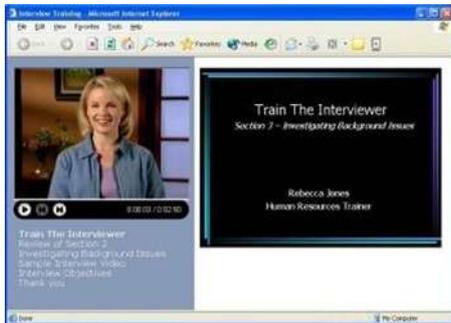
L'ambiente di sviluppo, come vediamo in figura 2.7, è molto simile ad altri programmi di editing video, quindi per coloro che hanno avuto esperienze similari è abbastanza intuitivo.

Le fasi fondamentali del suo uso sono le seguenti:

- si importano gli elementi che si intendono “miscelare” nelle rispettive cartelle (Audio, HTML, Images, Slides, Video);
- si “montano” i contenuti utilizzando la Timeline posta in basso;
- si pubblica il prodotto finale mediante il pulsante Publish (a destra in alto) o mediante File/Publish Presentation. Da lì in poi basta seguire le istruzioni a video. Per pubblicare su Internet, sarà sufficiente pubblicare in locale e poi trasferire la struttura generata via FTP.

Altre funzioni importanti sono le seguenti:

- la possibilità di editare la descrizione del materiale e di vedere in anteprima il risultato finale;
- la possibilità di dividere un video in clips;
- la possibilità di aggiungere Transizioni e Effetti ai video;
- la possibilità di usare dei Templates ed una serie molto ampia di modelli di Layout. Usando la Timeline se ne possono assegnare più di uno a ciascuna presentazione, in modo che le sue componenti vengano di volta in volta proposte nel modo più opportuno dal punto di vista comunicativo. Sarà l'esperienza a farci imparare sempre meglio quali templates utilizzare nelle varie situazioni. In assenza di indicazioni viene usato il Template Standard.

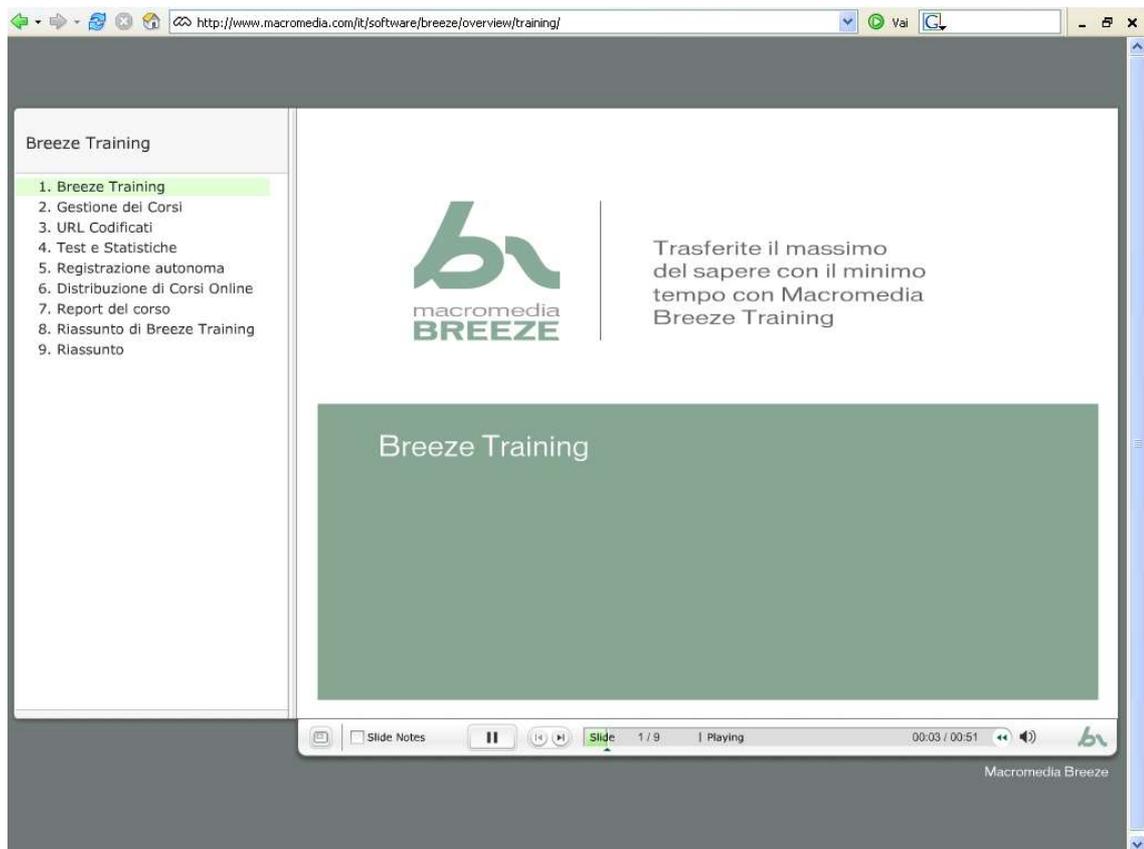


**Fig. 2.8.** Prodotto di Producer.

Nella Fig. 2.8 possiamo vedere il risultato di un prodotto di Producer. La pagina visualizzata si divide in due parti, una in cui sono visualizzate le slide (parte destra), mentre nell'altra viene visualizzato il video ed i controlli della presentazione.

Comunque Producer rimane un prodotto che può solo creare lezioni anche se questa volta multimediali, ma non da nessuno strumento per creare un intero corso nè alcun tipo di strumento per gestirlo, ne tanto meno da la possibilità allo studente di cambiare la forma o i contenuti della lezione, rendendoli più adatti ai suoi interessi.

**BREEZE**[15]. Macromedia Breeze è composto da una piattaforma centrale e da moduli che offrono funzionalità specifiche per diverse applicazioni, tra cui ricordiamo Breeze Presentation, Breeze Training e Breeze Live. Il primo serve per integrare le proprie presentazioni PowerPoint con l'aggiunta di sonoro, con Breeze Live invece è possibile partecipare a meeting on-line. L'ultimo di questi moduli è invece quello che interessa di più perché permette di creare interi corsi on-line multimediali.



**Fig. 2.9.** Esempio di come Breeze si presenta graficamente.

Breeze Training comprende naturalmente tutte le funzioni di Breeze Presentation oltre alle seguenti funzioni aggiuntive:

- **Gestione dei Corsi** cioè progettare corsi, invitare ed iscrivere partecipanti, analizzare statistiche e molto altro.
- **Amministrazione e report** cioè permette di creare report dettagliati su studenti, corsi, e risultati degli esami ed impostare criteri automatici per la certificazione e la promozione o la bocciatura.

In parole semplici Breeze fa un lavoro simile a Producer, cioè aggiunge alle presentazioni PowerPoint contenuti multimediali, con la differenza però che il prodotto finale usa la tecnologia Flash [23], che lo rende facilmente esportabile su qualunque browser, al contrario di Producer che si può visualizzare solo con Internet Explorer versione 5.0 o superiore. In figura 2.8 vediamo un esempio di

come Breeze si presenta agli utenti finali, all'interno del famoso player flash. Breeze è un prodotto che punta molto sulle aziende che vogliono risparmiare sui costi di gestione ed organizzazione di meeting e di corsi d'aggiornamento. Già ora una significativa percentuale delle aziende utilizza Breeze combinando, attraverso Flash, i tradizionali strumenti di presentazione con il commento audio in applicazioni streaming. In questo modo è possibile amalgamare contenuti di diverso genere (slide, audio, animazioni, indagini, domande, ecc.) in soluzioni che, tra l'altro, permettono di controllare i risultati ed essere integrate in sistemi di management preesistenti. Manca però la possibilità di personalizzare l'ambiente di apprendimento da parte dello studente, che è costretto a seguire il corso tale e quale come dato dal docente. Così si impoverisce la possibilità dello studente di apprendere più facilmente, aiutato dall'ambiente di studio. Si noti che Breeze, come tutti gli altri prodotti della Macromedia non è gratuito.

## ***2.2 Strumenti per rendere on-line l'Audio e il Video (streaming)***

Si è parlato in precedenza di fornire contenuti multimediali su Internet, ma questo non è immediato. Infatti chi tenta di usare soltantoactivex e plug-in per trasmettere audio o video ad un browser capisce ben presto che è impossibile ottenere e mantenere una temporizzazione dato che, prima di poter eseguire un file multimediale, l'applicazione deve scaricare l'intero file e questo richiede un tempo variabile e non gestibile con la programmazione. Questo è dovuto alla natura di Internet, che è nata per distribuire contenuti asincroni e quindi ha problemi a far viaggiare al suo interno contenuti in real-time o comunque sensibili al tempo di propagazione. Questo problema si risolve con lo streaming, cioè utilizzando un "flusso" di dati che viaggia dalla fonte (di solito un server adatto ) ad un client (il quale può essere una applicazione media player oppure un plugin del browser) non appena diventa disponibile.

Lo streaming (letteralmente “fluire”), come già detto, è la tecnologia che permette in rete la trasmissione continua e progressiva di flussi audio e video e contenuti multimediali. Un sistema di streaming è composto da tre elementi fondamentali:

- un *encoder*, cioè un codificatore di file multimediali; un sistema in grado di generare ed elaborare i contenuti destinati alla diffusione;
- un *Media Server*, ovvero la macchina che ha il compito di diffondere il contenuto mediando le informazioni di banda provenienti dal client ricevente con le indicazioni impostate dall’encoder in fase di programmazione;
- un *client*, destinato a ricevere il contenuto della trasmissione direttamente dal Media Server, secondo le modalità definite dalle proprie impostazioni di banda.

Il concetto chiave che caratterizza il processo di streaming è la possibilità, per un contenuto di adattarsi alle impostazioni (in questo caso di banda disponibile) riconosciute sul client, così da mettere a disposizione la prestazione migliore predisposta dall’encoder. Lo streaming è inoltre caratterizzato da una struttura che consente la visualizzazione di contenuti erogati, ma non la loro memorizzazione. Eliminando la necessità di download infatti non è più possibile salvare i contenuti, né visualizzarli al di fuori del processo. Questo è molto importante per le aziende che vogliono trasmettere on-line contenuti coperti da copyright e che quindi non vogliono che questi vengano memorizzati dagli utenti.

In altre parole, lo streaming permette la frammentazione di un flusso dati (sia video che audio) continuo (o precedentemente registrato) e la sua trasmissione e fruizione remota attraverso Internet evitando di dover fare il download completo prima di eseguire il file, il che consente una decisa riduzione dei tempi di attesa e permette la sincronizzazione e temporizzazione di elementi multimediali all’interno delle pagine internet e non solo e quindi è la soluzione al

problema di poter aggiungere e sincronizzare file multimediali a diapositive con testi e immagini. Bisogna usare allora un sistema di streaming per trasmettere lezioni multimediali complete.

### **2.3 Strumenti per la didattica on-line esistenti: vantaggi e svantaggi**

Riassumiamo l'analisi, schematizzando vantaggi e svantaggi delle singole applicazioni in una tabella:

<b>PRODOTTO</b>	<b>Caratteristiche</b>	<b>Vantaggi</b>	<b>Svantaggi</b>
ATUTOR	<ul style="list-style-type: none"> <li>• LCMS</li> <li>• On-line</li> <li>• Gestione e creazione dei contenuti</li> <li>• Servizi aggiuntivi</li> </ul>	Gratuito; Facile da usare; Personalizzazione della foma;	Niente integrazione multimedia; Non usa streaming;
MOODLE	<ul style="list-style-type: none"> <li>• LCMS</li> <li>• On-line</li> <li>• Gestione e creazione dei contenuti</li> <li>• Servizi aggiuntivi</li> </ul>	Gratuito; Comunità in italiano; Personalizzazione della foma;	Niente integrazione multimedia; Non usa streaming;
ADA Lesson Generator	<ul style="list-style-type: none"> <li>• Off-line</li> <li>• Gestione modulare dei contenuti</li> <li>• Pubblicazione materiale</li> </ul>	Facile da usare; Facile da esportare;	Niente integrazione multimedia; Non usa streaming; Nessuna personalizzazione da parte dell'utente;
PRODUCER	<ul style="list-style-type: none"> <li>• Off-line</li> <li>• Simile ad un programma di video editino</li> </ul>	Integrazione slide ppt con audio e video; Gratuito(per chi già possiede PowerPoint);	Nessuna personalizzazione da parte dell'utente; Poco supportato dai browser;
BREEZE	<ul style="list-style-type: none"> <li>• Off-line</li> <li>• Simile ad un programma di video editino</li> </ul>	Crossbrowser; Integrazione slide ppt con audio e video; Gestione di corsi;	Nessuna personalizzazione da parte dell'utente;

**Tabella 2.1** Riassunto delle caratteristiche dei prodotti.

Durante l'analisi di questi strumenti abbiamo potuto notare pregi e difetti di ognuno di essi, ma soprattutto ci siamo accorti della mancanza di uno

strumento completo che permetta la creazione e la gestione di corsi multimediali on-line, sia indipendenti, sia visti come strumenti di supporto per un corso in presenza. Infatti nessuno di questi prodotti riesce a creare e gestire sia contenuti tradizionali che multimediali e sono ancora di meno quelli che riescono a gestire una struttura modulare dei corsi e la personalizzazione dei contenuti da parte degli studenti. Questo ci ha portati a proporre una nostra soluzione che presentiamo nel prossimo capitolo.

## **Capitolo 3 La soluzione Tutor-Sky per creare corsi multimediale on-line**

### ***3.1 Obiettivi***

Scopo della soluzione implementata, che prende il nome di Tutor-Sky, è quello di creare un ambiente online di e-learning totalmente integrato, in cui un docente possa facilmente inserire in rete il proprio corso integrandolo con contenuti audio e video opzionali in modo tale che siano gli studenti stessi a decidere quali contenuti utilizzare e il loro formato. Questa decisione può dipendere da scelte personali, ad esempio si decide di non vedere il video della lezione perché si è convinti che questo non possa apportare alcun miglioramento all'apprendimento, o perché le risorse sono limitate e, ad esempio, si usa una connessione alla rete a 56Kbps perché non si vuole appesantire il collegamento con la trasmissione del video.

Gli obiettivi che si è pensato di conseguire con il lavoro qui di seguito descritto si possono riassumere nei seguenti punti:

- un ambiente facile da usare, sia da parte del docente sia da parte dello studente;
- possibilità per il docente di creare lezioni on-line;

- integrazione automatica dei contenuti multimediali ( il docente deve solo caricare i file sul server);
- possibilità per lo studente di personalizzare la lezione ed il livello di approfondimento (quali contenuti multimediali visionare);
- gestione modulare delle lezioni e degli argomenti del corso;

La facilità di utilizzo degli strumenti dell'ambiente web e della piattaforma stessa è importante poiché, come già detto in precedenza, bisogna dare a tutti gli insegnanti, o almeno alla maggior parte di essi, la possibilità di utilizzare tale piattaforma senza dovere avere conoscenze di Php, MySQL o di streaming o di quant'altro l'ambiente integra. L'utente dovrà naturalmente avere esperienza di navigazione in Internet dato che l'applicazione è basata sul web. Per rendere ancora più facile l'utilizzo della piattaforma e per evitare che un docente debba installare altri programmi aggiuntivi e quindi, per rendere Tutor-Sky autonomo almeno per i compiti basilari, si è integrato un editor HTML visuale così anche chi non ha mai avuto esperienza di scrittura di codice HTML possa utilizzare Tutor-Sky (questa possibilità è comunque presente anche in altre piattaforme di e-learning).

La caratteristica principale di Tutor-Sky è la possibilità per il docente di aggiungere facilmente alle diapositive delle lezioni (composte da testo e immagini) la sua voce, un video che lo ritrae mentre spiega (o con altro contenuto) e una musica di sottofondo (o più musiche tra cui l'utente possa scegliere) per agevolare il processo di apprendimento. Come detto in precedenza, sarà poi l'utente a decidere cosa utilizzare assieme alla diapositiva nel suo percorso di apprendimento.

Altro obiettivo altrettanto importante è la possibilità di creare corsi modulari e strutturati su due livelli. Modulare in quanto ogni corso è diviso in più argomenti i quali possono essere considerati come unità didattiche indipendenti. L'indipendenza sta nella capacità di queste unità di esprimere esaurientemente

i concetti senza bisogno di ulteriori spiegazioni. I concetti possono comunque essere approfonditi grazie ad un livello di approfondimento delle lezioni. Infatti un docente può strutturare le lezioni del corso su due livelli, uno che deve essere la base per tutti gli studenti ed un altro di approfondimento, che non è obbligatorio per tutti, ma che esplora con più attenzione i concetti trattati nelle slides della lezione standard.

Questa scelta è stata fatta per dare ai docenti la possibilità di scegliere la strutturazione del corso secondo le proprie idee, così che i risultati non siano tutti piatti ed unidirezionali, ma dinamici lasciando anche allo studente la possibilità di scegliere come affrontare il corso tenendo conto dei propri interessi e delle proprie necessità.

### ***3.2 Scelte di Base***

Scelta fondamentale del progetto è l'utilizzazione di software "open source". Tutti i programmi usati per crearlo sono open source: il server web (Apache), il linguaggio di scripting (Php), il database (MySQL) e il server di streaming (Icecast2).

La scelta di questi componenti dà la possibilità di eseguire Tutor-Sky sia su un server Windows che su server Linux (o qualunque altro server) senza alcuna modifica.

La scelta di utilizzare Php e MySQL è la più semplice e naturale nel momento in cui si vuole creare un applicazione web multiplatforma facile da installare. [24] Php è un linguaggio di script che interagisce fortemente con l'HTML ed è un linguaggio interpretato "server side". Molta della sua sintassi è presa in prestito dai linguaggi C, Java e Perl, a cui sono state aggiunte alcune specifiche caratteristiche. L'obiettivo del linguaggio è di semplificare il lavoro dei webmaster nella realizzazione di pagine dinamiche. Questo lo si può fare

anche con javascript che però è un linguaggio client side, anche se non tutto ciò che si riesce a fare in Php può essere fatto in javascript. Ma in cosa si caratterizza un linguaggio di scripting server side? Quando un server web predisposto al Php, molto spesso Apache, riceve la richiesta di una pagina “.php”, questa viene prima letta dal server che interpreta il codice, genera il relativo codice HTML e lo spedisce al client.

Al contrario, uno script client side non viene interpretato dal server web ma dal browser. Questo porta delle differenze di prestazioni, in quanto la pagina con lo script javascript è già pronta e viene immediatamente inviata al browser, mentre una pagina php deve essere interpretata prima di poter essere inviata. C'è anche da dire che scrivere uno script client side “crossbrowser” non è certo facilissimo in quanto essendo interpretato dal browser lo script può a volte essere interpretato diversamente. Naturalmente questo tipo di problema non esiste con uno script server side.

MySQL invece è il DBMS (DataBase Management System) scelto. La prima versione esce nel 1995, quando uno sviluppatore della TcX decise di aggiungere ad un altro DBMS già esistente, mSQL(miniSQL), la gestione degli indici che non erano supportati [25]. Utilizza un'architettura client-server, cioè una macchina server contiene i dati e gestisce gli accessi, ed i client si connettono al server e attraverso questo accedono ai dati. MySQL è classificabile come un "database relazionale", in quanto non conserva i dati in una unica grande entità, ma in tabelle separate che sono legate fra loro da riferimenti tra attributi e in quanto lo schema concettuale delle tabelle è indipendente dalla realizzazione delle tabelle.

Naturalmente il linguaggio supportato da MySQL è proprio SQL, un linguaggio sia DDL (Data Definition Language) che DML (Data Manipulation Language) [26]. Questo vuol dire che questo linguaggio permette sia la creazione delle strutture che formano un database, quindi tabelle, record, attributi database stessi, sia l'inserimento e la gestione dei dati. Sql ha una struttura molto

semplice che si rifà spesso alla lingua parlata, così se noi cerchiamo il numero di telefono di una persona, in una tabella dove ci sono *nome,cognome,recapito* basterà scrivere: “select *recapito* from *agenda* where nome=“NOME””, il che è come dire “*seleziona il recapito nell’agenda dove il nome è “NOME”*”. Questo semplicità, unità ad ottime prestazioni ha fatto di MySQL il DBMS più usato per la gestione dei dati di moltissimi siti e non solo.

Apache è un server web open source che si trova già installato in praticamente ogni distro linux, la piattaforma che abbiamo usato per sviluppare la nostra soluzione. E’ uno dei server più usati nel web e questo è sicuramente indice di stabilità e di buone performances. Poi c’è anche da dire che nelle versioni installate nelle distro linux, apache supporta nativamente Php.

L’elevata efficienza di questo web server permette di ottenere buone prestazioni anche da macchine meno recenti: anche un 486 può essere “riciclato” come web server [27]. Il nucleo principale dei programmatori di Apache è composto di circa 20 persone, l’Apache Group che dedica il proprio tempo libero per lo sviluppo di questo software e quindi del progresso del mondo web.

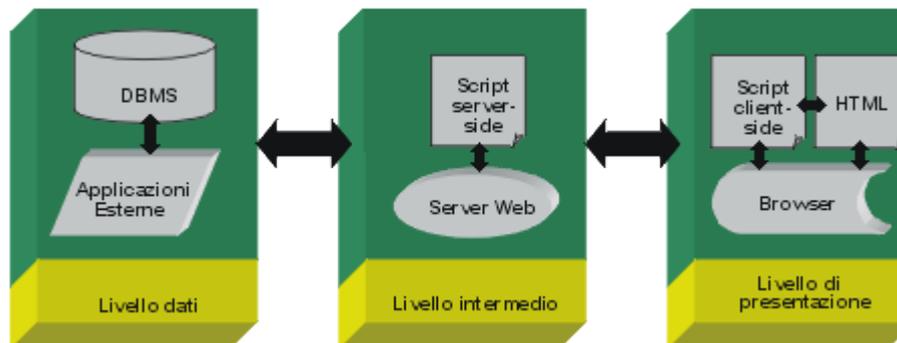
Icecast è un sistema di streaming audio che gestisce file sia MP3 che nel formato Ogg Vorbis [28]. È disponibile sotto i termini della licenza GNU GPL. Oggi è uno dei server di streaming audio più diffusi per creare radio su internet, ma che può anche essere usato per servizi “on demand” come serve a noi. Icecast2 è disponibile sia sotto la piattaforma linux, sia Windows. La sua interfaccia per la versione Linux è solo con linea di comando, mentre sotto windows è disponibile una semplice interfaccia visuale, anche se poi è sempre necessario usare un editor di testi per configurare il file di configurazione.

### **3.3 Architettura**

Tutor-Sky è una “web application three tier” cioè un’applicazione web a tre livelli (client, server, database) e si differenzia dalle classiche applicazioni client-server per l’aggiunta di un livello fondamentale, il database [29]. Generalmente una web application three tier funziona così: un client chiede informazioni al server il quale interroga il database e costruisce il risultato in base ai contenuti del database e all’elaborazione del codice di scripting della web application. Un classico esempio è un sistema di calcolo distribuito, dove la complessità di computazione è relegata ad alcuni host remoti della rete, i server. Quindi i tre livelli si possono definire come:

- **livello di presentazione** – è il livello più vicino all’utente, costituisce proprio l’interfaccia dell’applicazione. Il suo compito è quello di inviare le richieste al livello intermedio e mostrarne i risultati all’utente. Non è altro che il browser;
- **livello intermedio** – questo livello è il punto d’incontro tra le richieste dell’utente e i dati nel livello sottostante. E’ questo livello che si occupa di elaborare i dati e di renderli in un formato tale che il livello di presentazione sia in grado di rappresentare. Questo compito è svolto dai server web (tipo Apache) o meglio da un modulo del server web come php, asp, perl o altri;
- **livello dati** – è più che altro un insieme di informazioni che il livello intermedio analizza secondo il programma che sta elaborando. Questi dati sono contenuti in applicazioni indipendenti dal Web, come ad esempio un gestore di database, un sistema di gestione di posta elettronica, ecc.

Una descrizione del rapporto tra questi tre livelli è visibile nella figura 3.1.



**Fig. 3.1.** Web application three tier.

Un utente o un browser non si rendono conto della differenza che c'è tra contattare un server web che distribuisce pagine web statiche o un server web che distribuisce pagine generate da un'applicazione web. L'introduzione delle "web application" non ha solo reso più gradevole e interattivo il web ma ha anche permesso una gestione e manutenzione dei

siti da parte dell'amministratore più semplice e automatizzata senza dover controllare ogni pagina del sito ogni volta che questo viene aggiornato.

Un'applicazione web è facile da distribuire e aggiornare, infatti, poiché essa si trova interamente sul server, la pubblicazione sul server coincide con la distribuzione e l'aggiornamento effettuato è automaticamente reso disponibile a tutti gli utenti.

Altro vantaggio è l'accesso multiplatforma in quanto l'accesso all'applicazione è indipendente dall'hardware e dal sistema operativo utilizzato dagli utenti e può quindi essere di qualsiasi tipo.

Quando si crea un'applicazione web si incontrano problematiche sconosciute ai programmatori di applicazioni tradizionali. Questo è dovuto al fatto che il web non è nato per eseguire applicazioni ma per condividere documenti. Un classico problema deriva dalla mancanza di persistenza della connessione nel Web. Questo problema rende più complicato la gestione del cosiddetto *mantenimento dello stato dell'applicazione*. Infatti ogni richiesta del client corrisponde ad una connessione e appena la richiesta viene esaudita, la

connessione viene chiusa in accordo allo standard del protocollo http (v1.0) il quale è stateless cioè non mantiene lo stato. Quindi il programmatore per risolvere il problema si deve affidare ai dati inviati durante le richieste delle pagine o immagazzinati in alcuni files, i cosiddetti cookie, o ancora, si può affidare alle sessioni. Le sessioni consentono di identificare un determinato client per un determinato periodo di tempo e di associargli dei dati, che vengono serializzati e salvati in un file di testo avente come nome una stringa alfanumerica di 32 caratteri, sufficienti ad evitare il rischio di due sessioni con identico nome, passata in un cookie al client o "appesa" ad ogni URL, quindi anche con la gestione dei cookie è necessario l'uso dei cookies.

Altra problematica propria delle applicazioni web è il limite dell'ampiezza di banda di trasmissione a disposizione e della quantità di lavoro che il server deve sviluppare. Questo porta a limitare i dati in risposta ad una richiesta e a volte a dover rinunciare ad una grafica accattivante per migliorare il contenuto informativo.

Inoltre, la presenza di applicazioni esterne particolarmente complicate, come un database nella stessa macchina del server comporta a volte un calo delle prestazioni del server web. Per questo, spesso le due applicazioni si trovano su macchine differenti, distribuendo sempre di più il carico elaborativo, velocizzando le interazioni e quindi limitando i tempi di attesa.

Naturalmente anche la gestione degli script client-side come javascript o active-x è un problema perché essendo questi script client-side, il codice sarà interpretato dai browser e, non esistendo uno standard definito, è possibile che vengano interpretati diversamente tanto che molto spesso è obbligatorio scrivere il codice più volte per tutti i tipi di browser che si vogliono supportare.

Anche nella progettazione di Tutor-Sky queste problematiche sono state affrontate e vedremo come sono state risolte nel prossimo paragrafo dove parleremo della sua implementazione

### **3.4 Implementazione**

Viene ora illustrata la struttura implementativa della web application Tutor-Sky. In particolare viene mostrato come sono stati implementati il livello dati e quello intermedio. Successivamente viene discusso come sono stati affrontati i problemi di crossbrowser del livello presentazione.

#### **LIVELLO DATI**

Per prima viene discussa l'organizzazione del terzo livello, il livello dati. Questo livello è rappresentato da un DBMS MySQL 4 o superiore e, nel caso in questione, è stato creato un database che si chiama "corsi\_online".

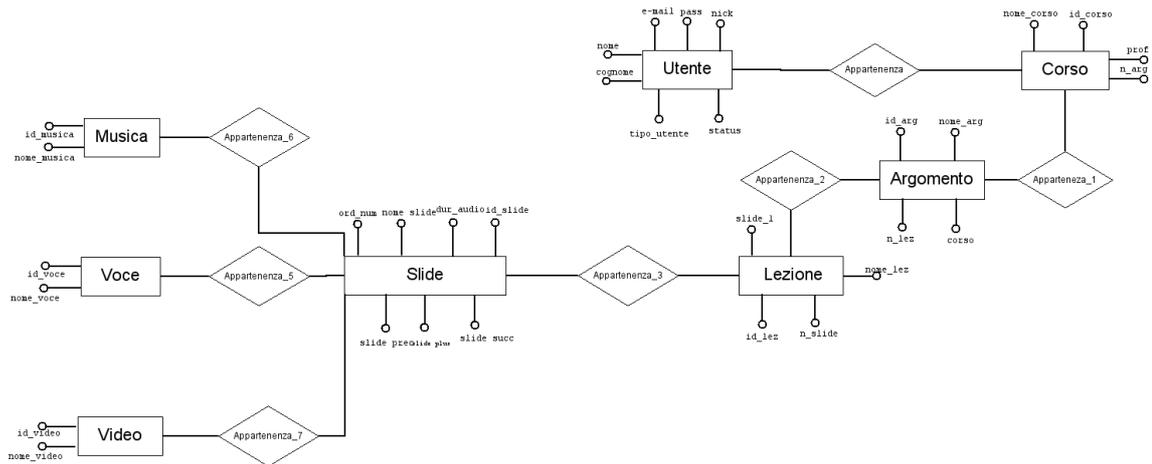
Questo database è formato da 10 tabelle:

1. **Utente**, dove sono memorizzati i dati dell'utente;
2. **Corso**, dove sono memorizzati i dati del corso;
3. **Argomento**, dove sono elencati tutti gli argomenti e le informazioni relative;
4. **Lezione**, dove sono memorizzate le informazioni sulle lezioni;
5. **Slide**, dove sono memorizzate le informazioni delle diapositive ;
6. **Appartenenza**, dove vengono messe in relazione i corsi con gli utenti;
7. **Voce**, dove vengono archiviate le informazioni sui file mp3 dove è registrata l'eventuale voce del docente;
8. **Musica**, dove vengono archiviate le informazioni sui file mp3 dell'eventuale musica di sottofondo;
9. **Video**, dove sono memorizzate le informazioni sui file video;

Nella Fig. 3.2 possiamo vedere la struttura concettuale dei dati rappresentati nel database utilizzando uno schema "entità-relazioni" (entity-relationship). In questo schema sono indicati i concetti coinvolti nell'applicazione e il legame che li lega a partire dalle definizioni di relazione ed entità:

**Entità (rettangoli)** : rappresentano classi di oggetti (fatti, persone) che hanno proprietà comuni ed esistenza “autonoma” per un’applicazione.

**Relazione (rombi)** : rappresentano legami logici, significativi per l’applicazione, tra due o più entità.



**Fig. 3.2** Schema concettuale del database “corsi\_online”.

Dopo l’analisi concettuale nel processo di progettazione del database, questi concetti vengono tradotti dallo schema entity-relationship nelle vere e proprie tabelle del database attraverso regole standard di conversione delle entità e delle relazioni a seconda il tipo di associazione che lega le entità con le relazioni che possono essere:

- Associazioni Molti a Molti in cui la cardinalità massima con cui ogni occorrenza di entrambe le entità coinvolte è uguale ad N;
- Associazioni Uno a Molti in cui soltanto una delle due entità coinvolte partecipa con cardinalità massima uguale ad N, mentre l’altra partecipa con cardinalità massima pari ad 1;
- Associazioni Uno ad Uno in cui entrambe le entità coinvolte partecipano con cardinalità massima pari ad 1.

Viene ora descritto ogni concetto che è stato tradotto in tabelle:

UTENTE: è un entità che viene tradotta direttamente in una tabella dove ogni suo attributo diventa un campo. UTENTE rappresenta, come il nome stesso dice, l'utente dell'applicazione. E' rappresentato da: *nome*, *cognome*, *nick* (attributo identificativo e chiave primaria nella tabella), *pass* (password), *e-mail*, *tipo utente* che identifica se l'utente è uno studente o un docente ed è settato dall'utente in fase di registrazione, e *status* una variabile che indica se l'account di quell'utente è stato attivato via e-mail.

Il codice SQL per creare la tabella è il seguente:

```
CREATE TABLE `utente` (  
  `nick` varchar(20) NOT NULL default ",  
  `pass` varchar(255) NOT NULL default ",  
  `nome` varchar(20) default NULL,  
  `cognome` varchar(20) default NULL,  
  `email` varchar(60) NOT NULL default ",  
  `tipo_utente` int(1) NOT NULL default '0',  
  `status` int(1) NOT NULL default '1',  
  PRIMARY KEY (`nick`)  
);
```

CORSO: entità che rappresenta il corso che viene tradotta anch'essa direttamente in una tabella. I suoi attributi sono: *id\_corso* (attributo identificativo), *nome\_corso*, *n\_arg* cioè il numero di argomenti, *prof*, il nome del docente che è legato con un vincolo referenziale alla tabella utente.

Il codice SQL per crearla è il seguente:

```
CORSO CREATE TABLE `corso` (  
  `nome_corso` varchar(255) NOT NULL default ",
```

```

`id_corso` int(3) NOT NULL auto_increment,
`prof` varchar(20) default NULL,
`n_arg` int(2) NOT NULL default '0',
PRIMARY KEY (`id_corso`),
UNIQUE KEY `nome_corso` (`nome_corso`)
);

```

APPARTENENZA: questa è la relazione che lega “UTENTE” e “CORSO”. Essendo questa associazione di tipo *molti a molti* la relazione si traduce in una tabella a parte che contiene i campi chiave delle due tabelle, in questo caso *nick* ed *id\_corso* che sono stati tradotti in *utente* e *corso*. Questa relazione specifica chi appartiene ad un corso. Con appartenenza si intende sia i docenti con i corsi da loro creati, sia gli studenti con i corsi da loro seguiti. Il codice SQL per la creazione delle tabelle è il seguente:

```

CREATE TABLE `appartenenza` (
  `utente` varchar(20) NOT NULL default "",
  `corso` varchar(15) NOT NULL default "",
  PRIMARY KEY (`utente`,`corso`)
);

```

ARGOMENTO: questa entità racchiude il concetto di *argomento* di un corso, cioè quella parte di corso autosufficiente che esprime già al suo interno un insieme di concetti ben definiti. Gli attributi dell’entità sono: *id\_arg* (attributo identificativo e poi primary key), *n\_lez* cioè il numero di lezioni per argomento, *nome\_arg* cioè il nome dell’argomento. Questa entità viene tradotta come al solito in una tabella che, però, stavolta avrà un campo in più rispetto agli attributi dello schema concettuale, l’attributo che lo lega al corso di

appartenenza e che non è altro che la traduzione della relazione *uno a molti* che lega l'entità CORSO con ARGOMENTO e che noi abbiamo chiamato "corso". Quindi il codice SQL per creare la tabella è:

```
CREATE TABLE `argomento` (  
  `id_arg` int(3) NOT NULL auto_increment,  
  `nome_arg` varchar(255) NOT NULL default '',  
  `n_lez` int(2) NOT NULL default '0',  
  `corso` int(3) NOT NULL default '0',  
  PRIMARY KEY (`id_arg`)  
);
```

LEZIONE: per lezione si intende una successione di diapositive che possono essere mostrate in sequenza dal *Lesson Player*, cioè da quella parte dell'applicazione che permette allo studente di navigare tra le diapositive. Questa entità è caratterizzata dagli attributi *id\_lez* (attributo identificativo e poi primary key), *nome\_lez* in altre parole il titolo della lezione, *n\_slide* cioè il numero di slide della lezione, ed infine *slide\_1* che memorizza l'id della prima slide della lezione in ordine d'esecuzione. LEZIONE è in relazione *uno a molti* con l'entità ARGOMENTO e quindi per tradurre l'associazione aggiungiamo un campo nella tabella *lezione* in maniera tale da poter legare ogni lezione con il proprio argomento. Naturalmente il nuovo campo avrà il nome di *argomento*. Il codice SQL per la creazione della tabella è il seguente:

```
CREATE TABLE `lezione` (  
  `id_lez` int(4) NOT NULL auto_increment,  
  `nome_lez` varchar(255) default NULL,  
  `n_slide` int(2) NOT NULL default '0',
```

```

`slide_1` int(4) NOT NULL default '0',
`argomento` int(3) NOT NULL default '0',
PRIMARY KEY (`id_lez`)
);

```

SLIDE: la diapositiva o slide è il contenitore informativo basilare. E' definito da sette attributi che sono: *id\_slide* (attributo identificativo e poi primary key), *durata* che definisce la durata di una slide, *nome\_slide* che identifica il nome del file slide, *slide\_prec* dove s'indica l'id della slide precedente e *slide\_succ* che identifica attraverso l'id la slide successiva, *slide\_plus* che identifica una eventuale slide di approfondimento, *ord\_num* cioè il posto della slide nella sequenza della lezione. La traduzione di questa entità in tabella porta ad avere un campo in più, *lezione*, derivante dall'associazione **uno a molti** tra SLIDE e LEZIONE. Quindi il codice per creare la tabella è il seguente:

```

CREATE TABLE `slide` (
  `id_slide` int(4) NOT NULL auto_increment,
  `lezione` int(4) NOT NULL default '0',
  `durata` int(8) NOT NULL default '0',
  `nome_slide` varchar(255) NOT NULL default "",
  `slide_plus` int(4) NOT NULL default '0',
  `slide_prec` int(4) NOT NULL default '0',
  `slide_succ` int(4) NOT NULL default '0',
  `ord_num` int(2) NOT NULL default '0',
  PRIMARY KEY (`id_slide`)
);

```

VOCE: questa entità è caratterizzata da *nome\_voce* il nome del file audio e da *id\_voce* (attributo identificativo e poi primary key) a cui dopo la traduzione in tabella si aggiunge il campo *slide* dove è memorizzato l'id della slide cui è associato il file audio in questione. Il codice SQL per creare la tabella è il seguente:

```
CREATE TABLE `voce` (  
  `id_voce` int(4) NOT NULL auto_increment,  
  `slide` int(4) NOT NULL default '0',  
  `nome_voce` varchar(255) NOT NULL default '',  
  PRIMARY KEY (`id_voce`)  
);
```

MUSICA: questa entità contiene le informazioni delle musiche di sottofondo. È caratterizzata da due attributi *nome\_musica* ed *id\_musica*, mentre un terzo è aggiunto durante la traduzione in tabella per lo stesso motivo di VOCE ed il nuovo campo sarà stavolta *lezione*, perché la musica di sottofondo non è legata alla singola slide, altrimenti potrebbe distogliere l'attenzione piuttosto che aumentarla. Il codice SQL per creare questa tabella è il seguente:

```
CREATE TABLE `musica` (  
  `id_musica` int(3) NOT NULL auto_increment,  
  `lezione` int(3) NOT NULL default '0',  
  `nome_musica` varchar(255) NOT NULL default '',  
  PRIMARY KEY (`id_musica`)  
);
```

VIDEO: l'ultima entità, ma anche l'ultimo concetto che viene tradotto in tabella. Serve per contenere le informazioni (*id\_video*, *nome\_video*) dei file video

relativi alle slide (se esistono). Quando viene tradotta in tabella, anche questa entità aumenta il numero dei suoi campi per mantenere l'informazione dell'associazione. Il codice SQL per creare questa tabella sarà:

```
CREATE TABLE `video` (  
  `video` int(4) NOT NULL auto_increment,  
  `slide` int(4) NOT NULL default '0',  
  `nome_video` varchar(255) NOT NULL default '',  
  PRIMARY KEY (`video`)  
);
```

### **Livello intermedio**

Dopo aver descritto il *livello dati* viene ora illustrata la parte più significativa dell'applicazione web nella sua componente attiva cioè le pagine “.php”. Esiste ed ha un ruolo fondamentale un file di configurazione che si chiama “*tesi.conf*” che è all'interno della cartella “*tesi.lib*” nella quale sono definite le impostazioni generali del server, dell'amministratore e del sito.

E' fondamentale impostare correttamente questi dati per garantire il corretto funzionamento dell'applicazione in questo file sono contenuti sia informazioni di carattere generale, come il nome del sito, il suo URL, il recapito del web-master, sia informazioni legate all'implementazione, come il path delle cartelle pubbliche dei server coinvolti(web server streaming server audio e poi video), sia infine informazioni proprie del progetto come i file header e footer, cioè l'intestazione e la chiusura di molte pagine, o il file di stile. Vi sono anche definite informazioni sul database e sulla sessione:

### **TESI.CONF**

```
<?php
```

```

$TESI_SITE_NAME = "Tutor SKY"; // Nome del sito
$TESI_SITE_URL = "http://tutorsky.no-ip.com"; // indirizzo
del sito
$TESI_LANG_FILE = "italiano.lang"; // file della lingua
del sito.
$TESI_SITE_HOME = "index.php";

// definizione dei file comuni
$TESI_HEADER_FILE = "../templates/header.php";
$TESI_FOOTER_FILE = "../templates/footer.php";
$TESI_CSS_STYLE_FILE = "../style.css"; // definizione del
nome del file di stile

// dati dell'amministratore
$TESI_WEBMASTER_NAME = "tesista sky";
$TESI_WEBMASTER_EMAIL = "dancing.tony@gmail.com";

//database settings $TESI_DATABASE_HOST = "localhost";
$TESI_DATABASE_NAME = "corsi_online";
$TESI_DATABASE_USER = "root";
$TESI_DATABASE_PASS = "";
$TESI_TABLE_MEMBERS = "utente";

//server streaming settings ( indica in quale cartelle
posizionare i file multimediali)
$TESI_SERVER_STREAM = "C:\Programmi\Icecast2 Win32\web";
$TESI_SERVER_STREAM_VIDEO = "C:\temp\video";

// definizione degli indici dei valori che consentono
l'identificazione
$SESSION_NAME_USER = "user";

```

```
$SESSION_NAME_PASS = "pass";
```

?>

Questo file verrà poi incluso in molti file dell'applicazione rendendo la modifica delle impostazioni semplice e rapida.

La struttura del livello intermedio può essere divisa in:

- **Gestione utenti;**
- **Gestione contenuti;**
- **Lesson Player;**

### **Gestione utenti**

Iniziamo parlando della gestione **degli utenti**. Questa sezione riguarda essenzialmente due importanti funzioni: l'identificazione e la registrazione degli utenti.

La prima cosa che ogni utente deve fare prima di utilizzare l'applicazione è la registrazione, la quale avviene tramite la compilazione di un form in cui si specifica anche se l'utente è uno studente o un docente. Una volta compilato e inviato il form, il server controlla che i dati siano stati inseriti correttamente e se la verifica ha esito positivo allora viene crittografata la password con la funzione MD5 e conservata in questo formato. MD5 (message digest 5) è un particolare algoritmo di crittografia, o meglio di hashing ed in particolare è un algoritmo hash a 128 bit cioè indipendentemente dall'input e dalla sua lunghezza restituisce un output lungo 128bit. Salvando nel database le password nel formato restituito dalla funzione MD5 aumenta leggermente la sicurezza: difatti, anche avendo accesso manuale al database, non si avrà alcun modo per conoscere la password relativa ad un dato utente visto che la funzione MD5 essendo una funzione di hashing è anche una funzione iniettiva, questo vuol dire che per ogni valore in ingresso è definito un unico risultato, chiamato impronta (fingerprint) e che, dato il risultato, non è possibile risalire al valore

d'ingresso. Questo significa anche che se un utente dimentica la password non esiste un modo per recuperarla, ma egli stesso dovrà sceglierne una nuova. L'account appena creato non risulta ancora attivo. Dopo aver archiviato i dati personali nella tabella UTENTE, viene inviata automaticamente una e-mail di conferma dove si richiede l'attivazione dell'account. Finché questa operazione non viene effettuata, all'utente non sarà consentito l'ingresso all'applicazione. Una volta che l'utente si è registrato, ogni volta che usufruirà dell'applicazione dovrà autenticarsi. Durante questa operazione vengono richieste username e password. Queste vengono inviati al server, il quale interroga il database e cerca un utente con quel username che sia attivo:

```
"SELECT nick, pass, status
FROM $TESI_TABLE_MEMBERS
WHERE nick='$user' AND status!=0"
```

I risultati di questa query vengono poi confrontati con i dati forniti dall'utente ed in particolare si verifica che la password, opportunamente criptata, sia uguale alla password che si trova nel database. Se username e password non coincidono, l'utente viene avvisato e rimandato alla pagina di login per ripetere l'operazione. Quando invece l'operazione ha esito positivo username e password vengono salvate nella sessione e l'utente viene inviato alla sua pagina personale.

```
<?php
    session_start();
    .
    .
    .
    if (($pass == $db_pass) && ($db_status !=
    0)) {
```

```
        $_SESSION[$SESSION_NAME_USER] =
$db_user;
        $_SESSION[$SESSION_NAME_PASS] =
$db_pass;
        return true;
    }else{
        return false;
    }
?>
(codice dell'autenticazione)
```

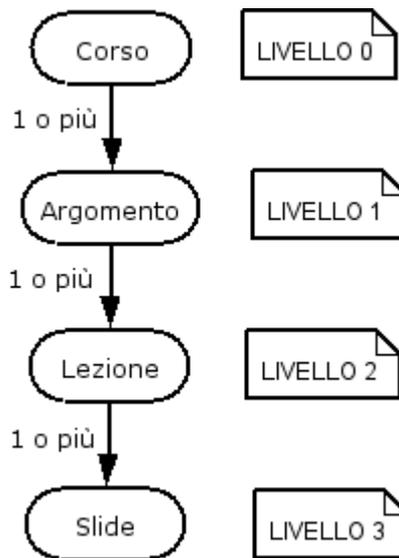
Una volta che l'utente ha terminato con l'applicazione è sempre tenuto a fare il logout per evitare che un nuovo utente che utilizzi quella macchina possa entrare con il suo account nel programma. Ciò avviene perché nelle impostazioni della sessione non è stato implementato alcun timeout quindi i dati salvati rimangono tali fino a quando si esegue la funzione `session_destroy()`, che poi è quella utilizzata nella funzione `logout`, oppure si chiude il browser. A queste operazioni di base si affiancano poi alcune funzioni per la gestione dell'account come, la cancellazione di un account, la modifica della password o il ricordo della password. La cancellazione dell'account comporta la cancellazione dei dati dell'utente dal database e, se si è un docente, anche dei contenuti inseriti da questi. Quando un utente chiede di aver ricordato la password, in realtà, non viene inviata la sua password, ma gli si dà la possibilità di sceglierne un'altra. Infatti, poiché le password sono memorizzate in formato cifrato come specificato prima, è impossibile recuperarle se l'utente le dimentica in quanto lui è l'unico a conoscerle.

### **Gestione contenuti**

Per prima cosa descriviamo la struttura delle cartelle nel server. Per ogni corso esiste una cartella di nome “corso”+ id\_corso che viene creata non appena un docente crea un corso e questo viene inserito nel database. All’interno di questa cartella troviamo la cartella “immagini” e una cartella per ogni lezione del corso. La cartella immagini viene creata quando viene creato il corso e serve per contenere le immagini che un docente inserisce per utilizzarle con gli altri contenuti. I nomi delle cartelle delle lezioni sono “lezione”+id\_lezione. Questi tipi di nomi, sia per le directory dei corsi che per quelle delle lezioni, servono ad identificare immediatamente il percorso dove è conservata una diapositiva. Per esempio, la “slide 1” della lezione “introduzione” (id = “5”), del corso di Inglese (id = “12”), si troverà in :

*SERVER\_DIR/corso12/lezione5/slide 1.html*

dove “SERVER\_DIR” è il percorso della cartella utilizzata dal server web per contenere le pagine che distribuisce. Si noti che il formato delle slide deve essere un formato web (cioè in html). I contenuti come detto in precedenza sono organizzati modularmente, ed esiste una gerarchia che li regola, come mostrato in figura 3.3. Il corso è in cima alla scala gerarchica e risulta il contenuto più vasto. Ogni corso si suddivide in argomenti, gli argomenti in lezione e questi ultimi in diapositive o slide che a loro volta possono essere formati da audio, video, immagini o solo testo.



**Fig. 3.3.** Struttura a livelli dei contenuti.

Poiché sia l'audio che il video vengono associati alle slide è necessario che il docente spezzetti questi media in piccoli file, ognuno corrispondente ad una slide, in maniera tale che l'upload non diventi troppo pesante o addirittura fallisca per il superamento dei limiti di upload di php che normalmente è di soli 2 MB (ma anche se questa applicazione ha impostato il valore a 8 MB). Una lunga trasmissione di dati ha anche una maggiore probabilità di incorrere in errori di trasmissione che porterebbero al fallimento di tutta l'operazione.

Esistono due modi per importare le singole slide all'interno della lezione: crearle on-line oppure salvarle in formato "zip". Il primo metodo utilizza uno script client-side in javascript chiamato HTMLArea. Questo script non fa altro che sostituire i textarea richiesti di un form con l'equivalente HTMLArea. Questi nuovi spazi permettono di avere un editor HTML WYSIWYG (What You See Is What You Get) che consente sia di scrivere direttamente il codice HTML sia di usare la modalità visuale. Per far funzionare l'editor basta aggiungere le righe seguenti alla pagina che conterrà l'HTMLArea:

```

        <script type="text/javascript">
// url di dove si trova lo script dell'editor
        _editor_url = "./htmlarea/";
        _editor_lang = "it";
</script>
<script type="text/javascript"
        src="./htmlarea/htmlarea.js">
</script>

```

ed `onload="HTMLArea.replaceAll()"` all'interno del body. Una volta fatta questa sostituzione, il componente del form che corrispondeva alla textarea adesso conterrà il codice HTML della slide. Quando il form viene inviato la pagina che lo riceve apre un file in scrittura e scrive al suo interno un'intestazione HTML + contenuto della textarea + chiusura della pagina HTML nel modo seguente:

```

// aggiungo l'estensione alla slide
$slidename = $_POST['title'].".html";
// creo l'intestazione
$header="<html><head><title>$TESI_SITE_NAME</title></head>
<body>";
// creo la chiusura della pagina
$footer = "</body></html>";
// creo la pagina vera e propria
$page = $header.stripSlashes($_POST['page']).$footer;
// creo il file o più precisamente lo apro in scrittura
$fh = fopen($uploaddir.$slidename, "w");
// ed infine scrivo nel file
fwrite($fh, $page);
fclose($fh);

```

E' anche possibile inserire immagini nelle slide create on-line in quanto è sufficiente inserire l'URL dell'immagine attraverso la finestra di dialogo di HTMLArea . Poiché non tutti hanno a disposizione uno spazio web in cui inserire le proprie immagini è stato implementato un sistema per l'upload dei file "jpeg", "gif", "jpg" ,"bmp", "png", "ico" direttamente nel server web così da integrare totalmente l'HTMLArea. L'upload delle immagini è gestito tramite l'upload php ed è quindi soggetto a tutti i vincoli che lo caratterizzano. Per implementare l'upload php basta creare un form *enctype="multipart/form-data"* ed inserirvi un input di tipo *file*. Poi quando il form viene inviato il file viene copiato dalla cartella temporanea di lavoro di php, alla directory di destinazione scelta utilizzando la variabile globale *\$\_FILES* e la funzione *move\_uploaded\_file* nel modo seguente:

```
<?php
if(!(move_uploaded_file($_FILES['image']['tmp_name'],
"../corso$id_corso/immagini/".$_FILES['image']['name']))) {
    switch($_FILES['image']['error'])
    {case 1: print "<p> Il file ".$_FILES['image']['name']
        ." supera la dimensione massima possibile
        concessa dal server php qui installato</p>";
        break;
    case 2: print "<p> Il file ".$_FILES['image']['name'].
"
        supera la dimensione massima possibile concessa
        dalla
        form</p>";
        break;
    case 3: print "<p> Solo una parte di questo file
        ".$_FILES['image']['name']. " è stata caricata dal
        server</p>";
```

```

        break;
    case 4: print '<p> Nessun file è stato caricato</p>';
            break;
    default: print " <p> Tipo di errore sconosciuto
                "$_FILES['image']['error']."</p>";
            break;
    }
    die("Errore nell'upload dei file");
}
?>

```

Il modo alternativo di aggiungere slide ad una lezione è l'upload diretto della slide. Supponendo che il docente scriva una pagina HTML off-line, potrà trasferirla racchiusa in un file zip insieme alla cartella degli oggetti referenziati dalla pagina. La creazione dell'archivio è necessaria perché una pagina HTML quasi mai è costituita da solo testo, ma ha dei riferimenti ad altri oggetti come immagini, fogli di stile etc. Per far in modo che sia la pagina HTML che gli oggetti referenziati vengano caricati come un unico oggetto, è opportuno mettere gli oggetti referenziati in un'unica cartella e la pagina HTML all'esterno di essa ed archiviare tutto in formato compresso in modo anche da alleggerire il trasporto. Poi sarà il server o meglio il codice php a estrarre file e cartella dopo aver eseguito l'upload del file così come mostrato per le immagini.

Per gestire i file zip in php è necessario utilizzare una libreria esterna. Tutor-Sky utilizza il file "zip.php" in cui è definito la classe "*Archive\_Zip*". Per questa classe sono implementati metodi per ottenere la lista dei files, per estrarli, per aggiungere file all'archivio e altri ancora. L'unico metodo che qui interessa è quello per estrarre i file dall'archivio. Questa funzione (*function extract(\$path)*) estrae l'archivio nel percorso specificato da path. Così abbiamo il seguente codice :

```

<?php
    require "zip.php";
    // istanzio un oggetto di quella classe
    $zip = new Archive_Zip("$uploadaddir$zipfile");
    // estraggo il file nella directory di upload
    $list_file = $zip->extract($uploadaddir);
    // recupero il nome del file web
    $slidename = find_htlm($list_file);
?>

```

Una volta che le slide sono state aggiunte alla lezione, qualunque sia stata la modalità, vengono inseriti i dati della slide nelle varie tabelle e per la precisione:

1. vengono prima inseriti solo i valori “lezione” e “ord\_num”;

a. *"INSERT INTO slide (lezione,ord\_num) VALUES ('\$id\_lez','\$n\_slide')";*

2. viene calcolata la durata della slide;

a. per calcolare la durata della slide in realtà viene calcolata la durata del file audio associato tramite una funzione di una libreria esterna per la gestione degli mp3;

```

$MP3=new MP3("$TESI_SERVER_STREAM/audio$id_slide.mp3");
$MP3->get_info();
// aggiungo "000" perché la funzione setTimeout prende in
ingresso i millisecondi
$durata = $MP3->calculate_length_sec()."000";

```

3. completo l'inserimento dei dati;

a. *"UPDATE slide SET  
dur\_audio = '\$durata',  
nome\_slide = '\$slidename',*

```

ord_num = '$n_slide',
slide_prec='$slide_prec'
WHERE id_slide = '$id_slide';
b. "UPDATE slide SET
slide_succ = '$id_slide'
WHERE id_slide = '$slide_prec'"

```

Tutor-Sky non accetta solo le slide come contenuti ma anche i file audio e video in quanto è una piattaforma multimediale. Questi file vengono caricati, se il docente li inserisce, con il solito metodo dell'upload php e vengono spostati nelle cartelle del server di streaming con il nome *nome\_file+\_id\_slide.mp3* creando così un legame immediato tra i files e le slide.

Tutti questi contenuti vengono poi riuniti nel “**Lesson Player**”. Questa parte dell'applicazione web si occupa di fare lo slideshow sincronizzato con l'esecuzione dei contenuti multimediali.

### **Lesson Player**

Il Lesson Player è costituito da tre parti principali: il menù laterale a sinistra in cui sono elencate tutte le lezioni e le slide, raggruppate per argomento a seconda del corso che si sta seguendo, il frame di visualizzazione ed il frame che comanda la navigazione della lezione e dei contenuti multimediali. Queste tre parti sono tutte riunite in un file “main.php” dove vengono inizializzate le variabili di controllo e organizzati i frame in dimensione e posizione.

Si descrivono ora le tre parti attive cominciando dal main.php. Ogni volta che questa pagina viene richiesta viene passato con il metodo GET il valore della id della slide che il player deve visualizzare e per prima cosa main.php memorizza il valore della variabile globale `$_GET['id_slide']` in una variabile locale `$id_slide`. Avendo già il valore dell'id del corso che è memorizzata nella sessione e avendo `$id_slide` si è in grado di recuperare il file della slide. Per fare questo serve la query:

```
SELECT *  
FROM slide  
WHERE id_slide='$id_slide'
```

Adesso si conosce la lezione a cui appartiene la slide e qual è il nome del file che v'è visualizzato per primo. Questi valori vengono conservati per tutta la durata della sessione in maniera tale che finché la finestra del lesson player rimane aperta si è in grado di usare questi dati. Nella sessione viene anche salvato il valore "ord\_num" della slide, un valore che serve per navigare fra le slide.

Quando si richiama il Lesson Player, come prima cosa appare un form che chiede quali file multimediale visionare (questo è implementato in main.php). La struttura del form non è fissa ma dipende dalla presenza nella lezione scelta di quali contenuti multimediali siano presenti.

Nel frame di sinistra contenuto in main.php, *leftFrame*, si trova menu\_lez.php cioè il menù delle lezioni. Anche qui per prima cosa viene salvato in una variabile locale il valore dell'id\_corso memorizzato nella sessione. Avendo questo valore si possono ottenere tutte le informazioni necessarie per mostrare l'elenco delle lezioni e degli argomenti. Per creare il menu ad albero o *tree menù* bisogna prima creare un file che definisca la struttura dell'albero. Per costruire questo file si utilizza un altro script php: "meun\_file.php". Naturalmente questo file deve rispettare la struttura gerarchica dei contenuti mostrata in Fig. 3.3, quindi va eseguita una serie di query a cascata. In pratica, per ogni argomento si ricavano tutte le lezioni collegate e per ogni lezione tutte le slide. I risultati verranno salvati in un file "menu.txt" che verrà interpretato da un altro script php che visualizzerà il menù e controllerà tutti gli "expand" e "collapse". Selezionando le voci del menu che identificano le slide nel frame di visualizzazione, viene mostrata la slide prescelta e viene anche eseguito uno

script php che imposta le variabili di controllo come `id_slide` e `ord_num` ai loro valori iniziali.

L'ultima parte che rimane da descrivere è il frame di navigazione che ricopre un ruolo di grande importanza per navigare tra i contenuti. Alla pagina viene sempre fornito il valore della slide corrente, cosicché si ricavino tutti i dati necessari per fare visualizzare correttamente la slide. Questo valore viene fornito tramite la variabile globale `$_GET`. Non vi è differenza in questa fase tra la gestione delle slide di approfondimento e quelle normali, entrambi i tipi hanno lo stesso trattamento. Questo è possibile perché i dati delle slide di approfondimento vengono conservate nella stessa tabella `SLIDE` e perché queste contengono molti dati della slide principale a cui sono associate.

Ogni volta che lo script `nav.php` viene eseguito, si ricava l'id dal database della slide precedente, successiva e di approfondimento e si memorizzano tali valori nella sessione. Se la slide corrente è una slide di approfondimento allora i valori di slide successiva e di slide di approfondimento coincideranno con quelli della slide principale.

La funzione di `play` dello slide show in sintesi non è altro che una serie di `refresh` sincronizzati dall'istruzione `setTimeout('show_plus()', '$durata');` dove la funzione `show_plus()` mostra il collegamento alla slide di approfondimento nel momento in cui tutto il tempo di durata della slide corrente è trascorso.

```
function plus() {
document.getElementById("plus_td").
style.visibility="hidden";
parent.mainFrame.location.href="/corso<?php
echo$id_corso; ?>/lezione<?php echo $id_lez;?>/<?php echo
$nome_slide_plus;?>";
document.location.href = "nav.php?type=3";
}
```

La prima volta che si preme sul play, oltre ad essere avviato il conto alla rovescia con la funzione `setTimeout()` che controlla quando è trascorso il tempo di visualizzazione della slide corrente in modo che alla fine venga visualizzato il collegamento alla slide di approfondimento, vengono impostati come visibile l'object e l'embed con il parametro auto play settato a true in maniera tale che al successivo refresh i componenti media player ricomincino l'esecuzione. In pratica il pulsante play non fa altro che fare il refresh della pagina mentre i pulsanti next e prev oltre al refresh modificano la slide presente nel frame di visualizzazione. Vengono infine eseguiti altri controlli per vedere se si è raggiunta la prima slide o l'ultima in maniera che non sia permesso superare tali limiti.

#### **LIVELLO PRESENTAZIONE.**

Questa parte dell'applicazione è quella meno trattata, dato che non è altro che il browser. L'unica parte di cui ci siamo interessati è quella di rendere la visualizzazione dell'applicazione il più crossbrowser possibile. La maggiore difficoltà si è avuta nella gestione del componente media player dato che i due ad oggi più diffusi browser, cioè Internet Explorer e Mozilla, usano codici javascript differenti per includere i plugin. Infatti mentre IE usa il tag `<object>`, Mozilla utilizza il tag `<embed>`. Per risolvere questo problema si ricorre ad uno stratagemma già collaudato in altri siti, come 105.net ed altri ancora. Il trucco sta nell'inserire all'interno nel tab `<object>` destinato ai browser della famiglia Microsoft, il tab `<embed>` destinato ai browser successori di Netscape, come mostrato nel listato d'esempio seguente, così che l'interprete del linguaggio client-side riconosca solo il codice a lui appartenente.

```
<object id="MediaPlayer1" type="application/x-oleobject">
  <param name="fileName" value="File.mp3">
  <param name="autoStart" value="true">
```

```
<param name="showControls" value="true">
<embed src="File.mp3" type="video/x-ms-asf-plugin"
      autostart="1" showcontrols="True">
</embed>
</object>
```

Si può notare che ogni parametro dato per il tag <object>, è ripetuto per il tag <embed> proprio perchè sono uno il sostituto dell'altro e quindi devono essere indipendenti.

Per il resto dell'applicazione il livello presentazione non è stato ulteriormente affrontato dato che la maggior parte degli script è di tipo server-side.

## **Capitolo 4 Esempio di utilizzo della soluzione**

### **Tutor-Sky**

In questo capitolo viene descritto come usare Tutor-Sky, portando l'esempio del corso di Reti di Calcolatori del Corso di Laurea in Informatica dell'Università di Palermo che è in fase di trasferimento sulla nuova piattaforma. Si mostrerà prima un esempio di utilizzazione da parte del docente per la creazione e pubblicazione del corso e poi si mostrerà come lo studente può seguire il corso pubblicato.

Preventivamente viene descritta la struttura generica della web application vista da un utente. Nella parte superiore di ogni pagina si trova un menù che contiene al centro (o quasi) il link alla home page, personale o generica (dipende se si è già stati autenticati o meno), alla destra una serie di collegamenti per gestire il proprio account o per crearne un nuovo e indica se si è "loggati". Nella parte sinistra di ogni pagina invece c'è un menù che se non si è già autenticati inviterà a farlo oppure a registrarci, mentre se si è già

effettuata la fase di login, mostrerà i dati personali e le possibili operazioni da compiere ( figura 4.1).

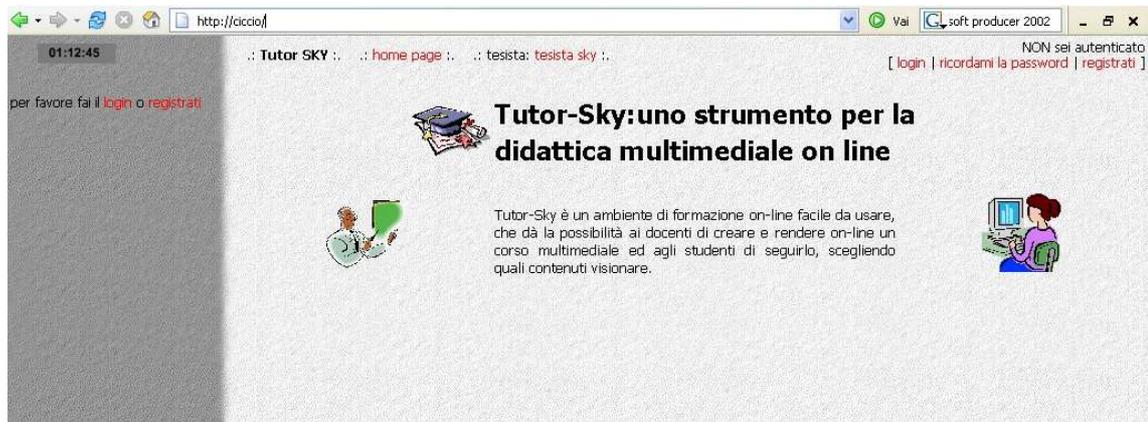


Fig. 4.1 Home page generica.

Viene ora mostrato come il docente interagisce con l'applicazione. Iniziamo con la creazione del corso. Per prima cosa dobbiamo registrarci come docenti inserendo i nostri dati (vedi Fig. 4.2).

The image shows a registration form titled 'Registrazione utente'. The form contains the following fields and values:

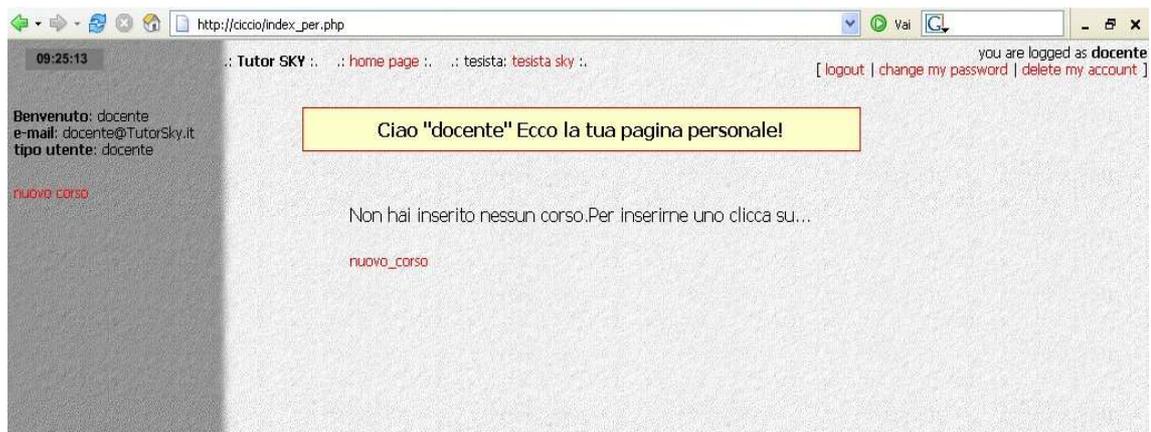
- nick: docente
- password: \*\*\*\*\*
- conferma password: \*\*\*\*\*
- indirizzo email: docente@TutorSky.it
- nome: docente
- cognome: docente
- tipo utente: docente (dropdown menu)

At the bottom of the form, there are two buttons: 'registrami' and 'pulisci'.

Fig. 4.2. Registrazione.

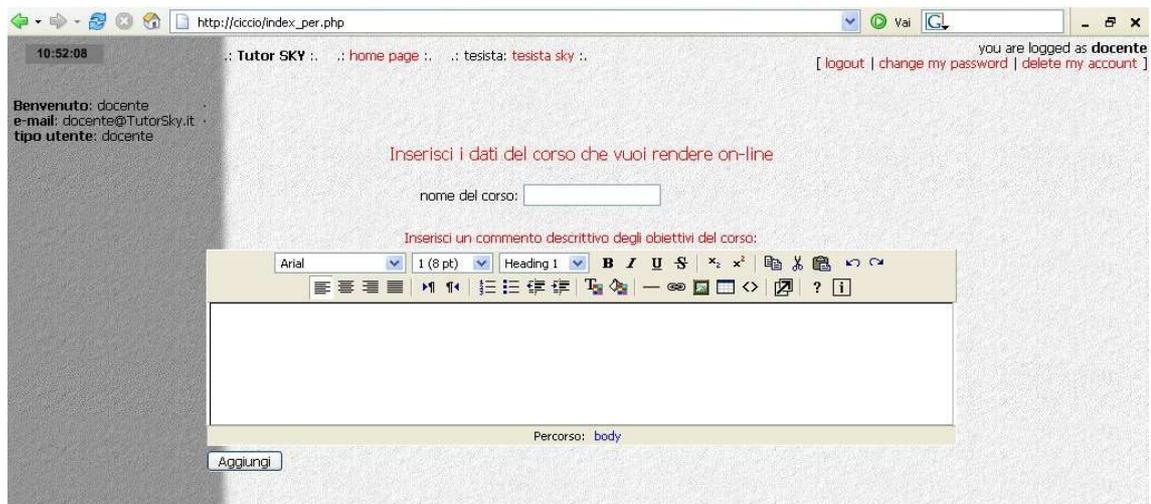
In questo caso utilizzeremo un utente fittizio che si identificherà con “docente”, “docente” con un indirizzo mail falso, anche se è importante inserirne uno valido perché l’attivazione dell’account avviene via e-mail come già descritto precedentemente.

Una volta creato e attivato il nostro account possiamo finalmente autenticarci e muovere i nostri primi passi dentro Tutor-Sky. La prima schermata che ci si presenta in quanto nuovi docenti ci informa che non abbiamo ancora creato nessun corso e quindi l’unica cosa che ci è permessa fare è proprio quella di crearlo (vedi Fig. 4.3).



**Fig. 4.3** Primo ingresso.

Come detto precedentemente possiamo notare che nel menu di sinistra sono comparsi i nostri dati ed i possibili strumenti. Adesso seguiamo il link [nuovo corso](#) e creiamo il nostro corso “Reti di Calcolatori”.



**Fig. 4.4** Creazione corso.

Per creare il nostro corso, come è mostrato in figura 4.4, ci viene richiesto un “nome del corso” e se vogliamo, anche un commento in cui descrivere gli obiettivi del corso, il programma o i destinatari del corso e quindi quali requisiti occorrono per seguirlo. Tutto questo apparirà nella pagina iniziale del corso che viene creata automaticamente in questo momento.

Fatto questo, la web application continua con il chiederci se vogliamo inserire un nuovo argomento. Non è indispensabile farlo immediatamente. Infatti è anche possibile aggiungere lezioni ed argomenti anche in un secondo momento. Supponiamo di aggiungere il nuovo argomento “**Computer Networks and Internet**” e subito dopo continuiamo con l’aggiunta della prima lezione. Per farlo basta inserire il titolo della lezione, nel nostro esempio “**Course presentation**”, e scegliere l’argomento tra quelli disponibili (vedi figura 4.5).



Fig. 4.5 Creazione di una lezione.

Fatto ciò possiamo inserire le slide della lezione, come si può vedere nella figura 4.6, che, come abbiamo già descritto dal punto di vista implementativo nel capitolo precedente, può avvenire in due modi o attraverso l'upload delle slide o attraverso la creazione on-line.

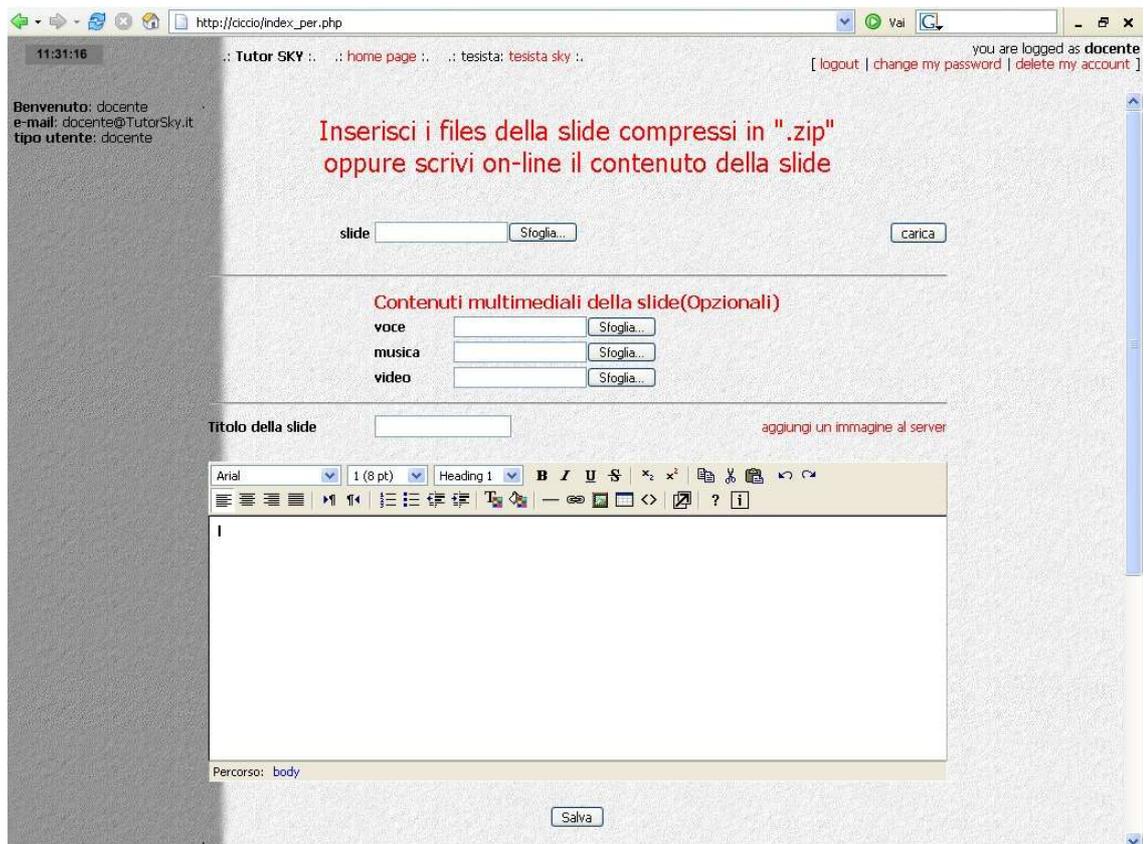
Quando si crea una slide on-line se si vuole aggiungere un'immagine bisogna prima caricare l'immagine sul server.



Fig. 4.6 Carica immagine

A tal fine si usa una finestra di pop-up, mostrata in figura 4.6, di facile utilizzo. Basta selezionare l'immagine desiderata e poi cliccare su carica. L'immagine verrà caricata e salvata nella cartella "immagini" del corso. In basso spunteranno tutte le immagini contenute nella cartella.

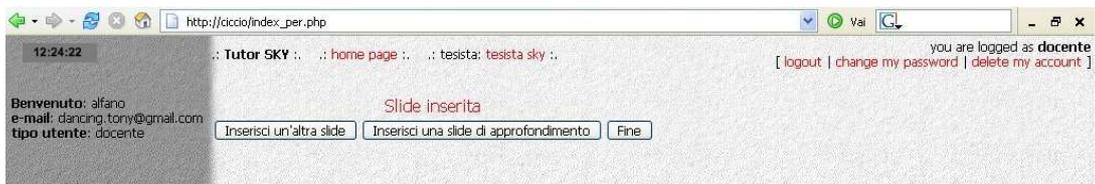
Una volta caricata l'immagine nel server è possibile inserirla nella slide. Basta cliccare sul pulsante . Si aprirà una finestra di dialogo in cui già comparirà l'URL del sito con tutto il percorso dove sono conservate le immagini. Basterà inserire il nome dell'immagine caricata oppure, se si sta inserendo un'immagine che non si trova sul server, cancellare quello che c'è scritto già e riscrivere l'intera URL dell'immagine che si desidera aggiungere.



**Fig. 4.7** Inserire una slide.

Se si vuole caricare una slide direttamente, questa deve essere in un archivio compresso nel formato “zip” (come illustrato precedentemente). Questo fa risparmiare tempo di connessione e permette di creare slide più complesse non contenenti solo testo. In entrambi i casi è possibile inserire i contenuti multimediali attraverso le finestre di dialogo che si aprono cliccando sui relativi bottoni “sfoggia”. Il formato supportato da TutorSky per i file audio è l’mp3”, mentre ancora non esiste una versione definitiva per i file video la cui implementazione in streaming sarà implementata più avanti. Comunque una volta inserita la slide si potrà scegliere tra tre opzioni (vedi figura 4.8):

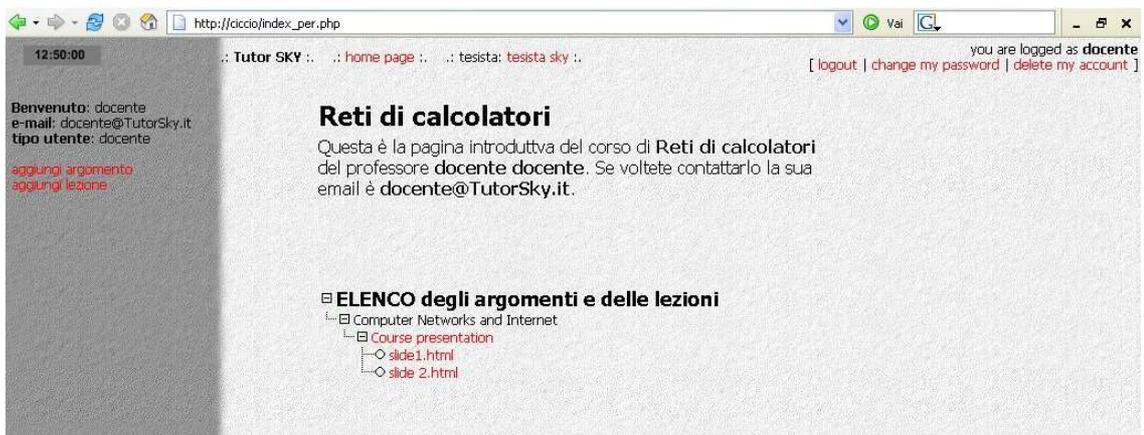
- inserire un'altra slide;
- inserire la slide di approfondimento;
- finire gli inserimenti;



**Fig. 4.8** Fine inserimento.

Se si sceglie la prima opportunità si ripresenta esattamente la stessa pagina e lo stesso succede se si sceglie di aggiungere la slide di approfondimento. Mentre se si sceglie di terminare gli inserimenti allora si tornerà alla pagina principale del corso.

Nel nostro esempio supponiamo di inserire tutte le slide da file creati off-line una di seguito all'altra. Quindi la pagina iniziale del nostro corso dopo il primo inserimento sarà come nella figura 4.9.

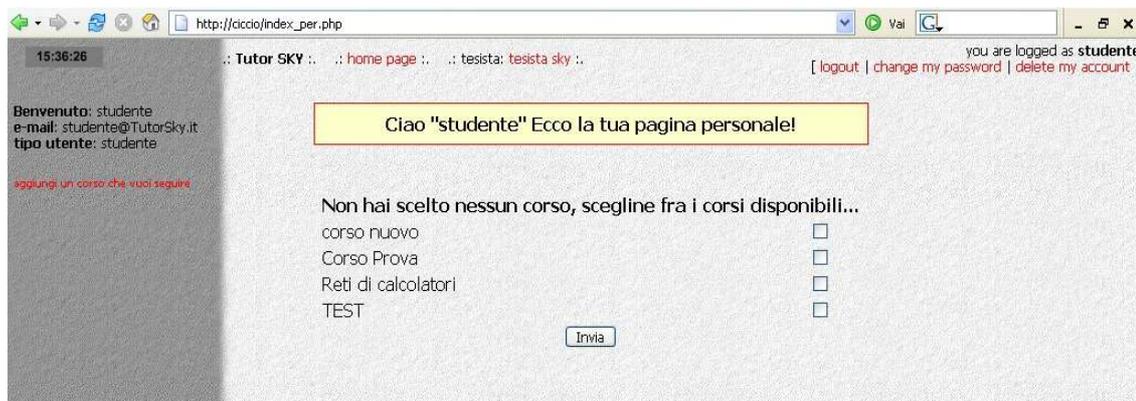


**Fig. 4.9** Indice del corso.

Una volta arrivati a questo punto il docente potrà continuare ad aggiungere lezioni ed argomenti modificando l'intero corso oppure potrà anche aggiungere altri corsi.

Consideriamo adesso la modalità studente e vediamo quali operazioni può compiere. Creiamo un nuovo utente di tipo studente ed identificato con

“studente”, “studente”. La prima volta che accediamo stavolta troveremo la lista di tutti i corsi disponibili fino a quel momento (vedi Fig. 4.10).



**Fig. 4.10** Prima schermata da studente.

Questa pagina verrà riproposta anche quando lo studente vorrà scegliere un altro corso da seguire. Per aggiungere un corso uno studente deve “cliccare” sull’apposito link che c’è sulla colonna di sinistra. Il nostro “studente” adesso sceglierà “Reti di calcolatori” e tornerà alla sua pagina personale (vedi figura 4.11) dove troverà l’elenco di tutti i corsi scelti.



**Fig. 4.11** Pagina personale di “studente”.

Le impostazioni della pagina sono esattamente le stesse di quelle di un docente, quello che cambia sono le possibili operazioni sulla colonna di sinistra in quanto, naturalmente, uno studente non può creare un corso. Una volta che

“studente” si è iscritto al corso di “docente”, egli può visualizzare la pagina indice del corso e seguire le lezioni.

Anche questa volta la pagina visualizzata dallo studente non differisce da quella visualizzata dal docente, soltanto che lo studente non avrà la possibilità di modificare in alcun modo i contenuti ma può soltanto usare il lesson player per seguire le lezioni.

Al Lesson Player si accede tramite la pagina principale del corso cliccando sulla slide o sulla lezione che ci interessa. Dopo aver cliccato su un collegamento viene visualizzata la schermata di scelta dei contenuti multimediali disponibili. Lo studente sceglie secondo i propri gusti e poi può finalmente seguire il corso.

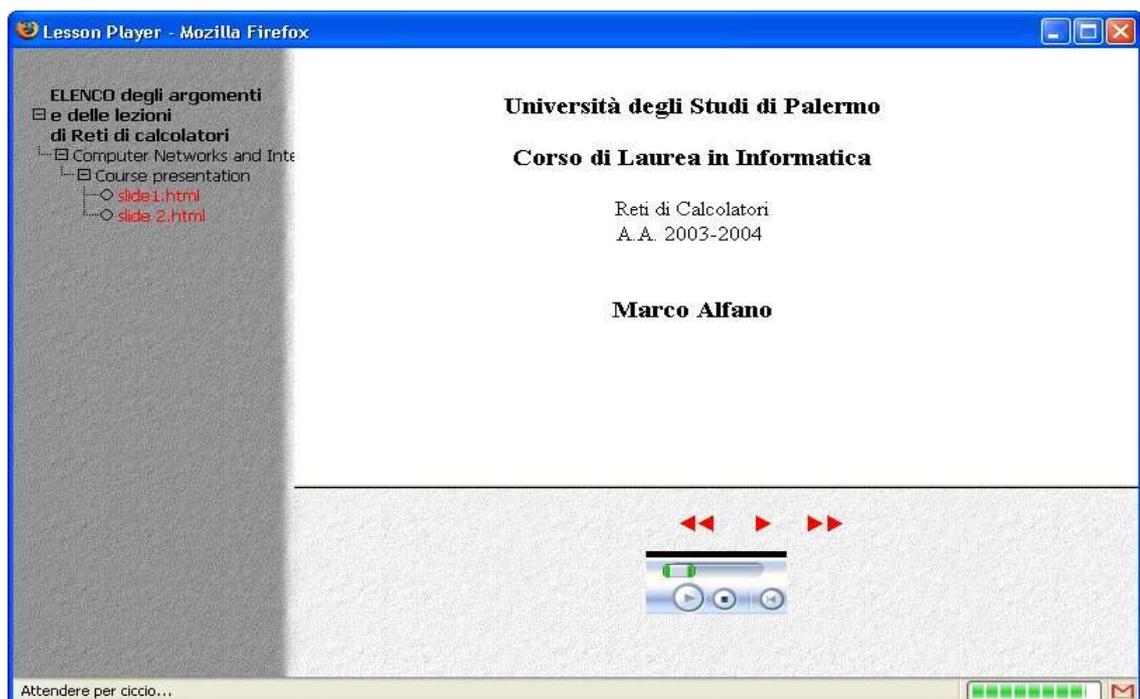


Fig. 4.12 Lesson Player.

La Fig. 4.12 è un esempio di cosa vede lo studente che segue una lezione. Si può notare come nel Lesson Player oltre ad essere visibili i comandi dello “slide show” sono visibili anche i comandi del plugin Media Player che in questo caso esegue in streaming la voce del docente sull’introduzione e la presentazione

del corso. Un altro plugin di cui si dovrebbero vedere i comandi è il plugin del video, mentre l'object per la musica rimarrà invisibile all'utente. E' in definitiva questa la parte che sfrutta lo streaming e che rende l'ambiente veramente multimediale.

## Conclusioni

In questo lavoro di tesi si è partiti dal concetto di didattica a distanza e si è visto come oggi questo si sia evoluto ed abbia dato la vita alla didattica on-line la quale, grazie all'utilizzo di nuove tecniche e di nuovi meccanismi, ha fatto fare un salto di qualità alla didattica a distanza ed è riuscita ad ottenere in vari ambiti il riconoscimento di didattica a tutti gli effetti. Successivamente si è trattata la didattica multimediale on-line, strumento che migliora e integra l'insegnamento in rete. Questo tipo di didattica utilizza al suo interno contenuti multimediali che rendono più completa e interessante la lezione e, soprattutto, aumentano la capacità di memorizzare dello studente stimolando le diverse modalità di apprendimento.

Dopo aver esaminato alcuni strumenti già esistenti per creare e rendere on-line un corso si è mostrato quali strumenti supplementari occorrono per gestire contenuti multimediali in rete (streaming).

Infine si è mostrata l'architettura e l'implementazione di un'applicazione web a tre livelli, Tutor-Sky, che permette di creare e gestire corsi multimediali on-line. Dopo la descrizione del tipo di architettura software sviluppata e di come sia stata implementata, con la descrizione di parte di codice, si è illustrato un esempio di utilizzo di Tutor-Sky.

Il progetto si è basato sui seguenti punti chiave:

- modularità;
- multimedialità;
- utilizzo efficiente delle risorse
- facilità d'uso (sia come creatore che utente del corso);
- flessibilità;
- Customizzazione dell'ambiente;

La modularità si è sviluppata dando la possibilità di creare corsi strutturati in diversi argomenti, i quali formano unità didattiche indipendenti le une dalle altre. Nella ricerca della multimedialità sta comunque una delle parti più originali del progetto. La facile integrazione dei contenuti multimediali con i contenuti "tradizionali", permette anche a docenti non esperti nell'informatica di elaborare corsi completi e più piacevoli, della solita lezione analitica e monomediativa. L'introduzione dei contenuti multimediali può anche essere uno strumento in più alla lotta delle difficoltà che le persone diversamente abili, come gli udientesi, incontrano nel momento in cui vogliono usufruire di un mezzo come internet, per conoscere ed apprendere.

Durante l'implementazione si è seguita anche una logica di risparmio delle risorse, lasciando allo studente la possibilità di scegliere quali contenuti multimediali visionare, a seconda delle risorse che l'utente vuole impiegare. Anche la facilità d'uso è stato uno dei nostri obiettivi durante tutto lo sviluppo. La semplicità e l'intuitività dell'applicazione è necessaria dato la poca predisposizione alla conoscenza degli strumenti necessari per una corretta didattica on-line che si riscontra nella realtà didattica italiana. Naturalmente anche l'interfaccia dello studente risulta semplice ed intuitiva, in maniera tale che la piattaforma possa essere utilizzata al massimo delle sue possibilità anche da utenti, sia docenti che studenti, inesperti.

Infine altre caratteristiche sono la flessibilità e la customizzazione dell'ambiente. Flessibile nel senso che uno studente non è costretto a seguire un andamento già designato dal docente, ma può egli stesso decidere anche il grado di approfondimento delle lezioni. Con il termine customizzazione invece intendiamo la possibilità di scegliere i contenuti e la forma dell'ambiente.

Naturalmente ci sono ancora alcuni punti aperti da esplorare sia concettualmente che dal punto di vista implementativo:

- aggiunta di ulteriori contenuti multimediali (ad es. animazioni e video);

- aggiungere ulteriore flessibilità e customizzazione alle future implementazioni;
- una gestione dei file sul server da parte dei docenti più completa;
- la gestione da parte del lesson player di file creati direttamente con PowerPoint, Word o altri strumenti capaci di creare contenuti web;

Certamente questi ed altri punti aperti verranno meglio alla luce utilizzando l'ambiente per alcune sperimentazioni con docenti della scuola secondaria e universitari. Grazie a queste sperimentazioni sarà possibile valutare le prestazioni in termini di apprendimento e di risorse occupate in modo da ottimizzarle e migliorare l'architettura di Tutor-Sky.

In questa prima realizzazione vengono solo accennate tutte quelle problematiche inerenti alla interazione docente-studente, in particolare quelle riguardanti il trasferimento dei contenuti, che dovrebbe, per quanto possibile, assecondare il ritmo di apprendimento degli studenti e dovrà, principalmente, fare in modo che lo studente riesca a "sintonizzarsi" su tali contenuti in modo da facilitarne l'apprendimento. Queste saranno le linee guida di un possibile sviluppo dell'ambiente che dovrà immancabilmente essere integrato con delle occasioni di verifica sia per lo studente, che potrà valutare il livello di apprendimento raggiunto, che per il docente, che potrà comprendere il livello e i tempi di apprendimento degli studenti e apportare le modifiche opportune, se necessario, alla metodologia o ai contenuti dell'insegnamento.

## Bibliografia

- [1] Antonio Calvani e Mario Rotta ed. Erickson: "Fare formazione in Internet. Manuale di didattica online";
- [2] Stefano Penge, "La rete per l'apprendimento e la didattica on line", sito ["http://www.infoaccessibile.com/penge.htm"](http://www.infoaccessibile.com/penge.htm);
- [3] "Altrascuola", sito ufficiale: ["http://corsi.altrascuola.it"](http://corsi.altrascuola.it);
- [4] "Filosofia dei nostri corsi on-line", sito ["http://corsi.altrascuola.it/modules.php?name=Sections&sop=viewarticle&artid=123"](http://corsi.altrascuola.it/modules.php?name=Sections&sop=viewarticle&artid=123);
- [5] "Scienze e Tecnologie per l'Ambiente ed il Turismo", sito ufficiale: ["http://webstat.pa.itd.cnr.it"](http://webstat.pa.itd.cnr.it);
- [6] "Istituto per le Tecnologie Didattiche", sito ["http://www.pa.itd.cnr.it"](http://www.pa.itd.cnr.it) ;
- [7] Liceo Classico Garibalidi di Palermo, "Corso d'inglese on-line", sito: ["http://www.provincia.palermo.it/~istpa052/englishonline/"](http://www.provincia.palermo.it/~istpa052/englishonline/);
- [8] "CoNe, **CO**mputer **NE**tworks", sito: ["http://www.math.unipa.it/~malfano/cone/"](http://www.math.unipa.it/~malfano/cone/);
- [9] Alcatel, "The Fundamentals of Communications", sito: ["http://www.ind.alcatel.com/fundamentals/index2.html?pass=true"](http://www.ind.alcatel.com/fundamentals/index2.html?pass=true);
- [10] BBC, "Webwise, the BBC's guide to using the internet", sito : ["http://www.bbc.co.uk/webwise/index.shtml"](http://www.bbc.co.uk/webwise/index.shtml)
- [11] Sito ufficiale di ATutor. ["http://www.atutor.ca"](http://www.atutor.ca)::;
- [12] Sito ufficiale di Spaghetti Learning: ["http://www.spaghettilearning.com/"](http://www.spaghettilearning.com/)::;
- [13] Sito ufficiale di moodle: ["http://moodle.org"](http://moodle.org);
- [14] Lynx, "Sito ufficiale della Lynx dove trovare ADA Lesson Generator": ["http://www.lynxlab.com/adalg/adalg\\_it.php"](http://www.lynxlab.com/adalg/adalg_it.php);

- [15] Macromedia, "Sito ufficiale di Breeze":  
["http://www.macromedia.com/software/breeze/";](http://www.macromedia.com/software/breeze/)
- [16] Microsoft, "Sito ufficiale di Producer":  
["http://www.microsoft.com/windows/windowsmedia/technologies/producer.aspx";](http://www.microsoft.com/windows/windowsmedia/technologies/producer.aspx)
- [17] Grifomultimedia, "Glossario definizione LCMS", sito:  
["http://www.grifomultimedia.it/t\\_prodotti.asp?url=GlossarioFL.asp#"](http://www.grifomultimedia.it/t_prodotti.asp?url=GlossarioFL.asp#)
- [18] Grifomultimedia, "Glossario: definizione SCORM,", sito:  
["http://www.grifomultimedia.it/t\\_prodotti.asp?url=GlossarioMZ.asp#";](http://www.grifomultimedia.it/t_prodotti.asp?url=GlossarioMZ.asp#)
- [19] E-Learning Lab, Università degli studi di Bologna, "La scelta di un Learning Management System Open Source: Atutor vs. Iliais vs. Moodle";
- [20] Adaptive Technology Resource Centre, sito ufficiale:  
["http://www.utoronto.ca/atrc/"](http://www.utoronto.ca/atrc/)
- [21] Free Software Foundation "GNU General Public License", sito:  
["http://www.gnu.org/copyleft/gpl.html";](http://www.gnu.org/copyleft/gpl.html)
- [22] Amici Formazione, "Brochure moodle", sito ["http://www.amiciformazione.it";](http://www.amiciformazione.it)
- [23] Macromedia , sito ufficiale dell'azienda produttrice di Flash:  
["http://www.macromedia.com/software/flash/?promoid=home\\_prod\\_flash\\_082403";](http://www.macromedia.com/software/flash/?promoid=home_prod_flash_082403)
- [24] Php: introduzione: ["http://php.it/manual/it/";](http://php.it/manual/it/)
- [25] Mario Protto, "Introduzione a MySQL", sito :  
["http://www.openskills.info/view/boxdetail.php?IDbox=1115";](http://www.openskills.info/view/boxdetail.php?IDbox=1115)
- [26] P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone: "Basi di dati: concetti, linguaggi e architetture", Seconda edizione; McGraw-Hill Italia, 1999;
- [27] "Introduzione APACHE", sito:

["http://www.dia.unisa.it/~ads/corso-security/www/CORSO-0102/sito\\_sicuro/Apache.htm"](http://www.dia.unisa.it/~ads/corso-security/www/CORSO-0102/sito_sicuro/Apache.htm);

[28] Kerry J. Cox, Ph.D. "What is Icecast?", sito:

["http://www.gnuware.com/icecast/sect\\_01\\_02.html"](http://www.gnuware.com/icecast/sect_01_02.html);

[29] "Architettura Applicazioni Web-Based" , sito:

["http://www.prometheo.it/a\\_applicazione\\_web.htm"](http://www.prometheo.it/a_applicazione_web.htm);