

UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

FACOLTÀ DI SCIENZE DELLA COMUNICAZIONE E DELL'ECONOMIA

**CORSO DI LAUREA IN ECONOMIA, RETI, INFORMAZIONE
(CURRICULUM IN ECONOMIA DELLE RETI E DELLA COMUNICAZIONE)**

TESI DI LAUREA

L'integrazione di un modulo per la condivisione di
"students' notes" all'interno del portale LMS Moodle

LAUREANDO:
Fabio Ruini

RELATORE:
Prof. Tommaso Minerva

ANNO ACCADEMICO 2003-2004

Sommario

1. INTRODUZIONE.....	6
2. MOODLE.....	13
3. SMOOTHFILEOPERATOR.....	17
4. L'INTEGRAZIONE DI SMOOTHFILEOPERATOR ALL'INTERNO DI MOODLE.....	20
4.1. Creazione di un nuovo modulo.....	20
4.2. Installazione del nuovo modulo: interazione con il file system.....	23
4.3. Installazione del nuovo modulo: interazione con il DB di Moodle.....	24
4.4. Aggiunta di una nuova istanza del modulo.....	26
4.5. Aggiunta di una nuova istanza del modulo: interazione con il file system.....	29
4.6. Aggiunta di una nuova istanza del modulo: interazione con il DB di Moodle.....	33
4.7. Aggiornamento di un'istanza.....	35
4.8. Cancellazione di un'istanza.....	37
4.9. Visualizzazione di un'istanza.....	39
4.10. Utilizzo del modulo.....	45
4.11. Upload di un file.....	48
4.12. Creazione di una nuova directory.....	49
4.13. Creazione di un nuovo file di testo.....	50
4.14. Visualizzazione dei dettagli di un file o di una directory.....	51

4.15. Modifica della descrizione di un file o di una directory.....	52
4.16. Ridenominazione di un file o di una directory.....	52
4.17. Cancellazione di un file o di una directory.....	53
4.18. Copia e spostamento di un file o di una directory.....	54
4.19. Modifica di un file.....	56
4.20. Aspetti grafici.....	57
5. CONCLUSIONI.....	59
6. RINGRAZIAMENTI.....	62
7. BIBLIOGRAFIA.....	64

A mio padre

1. INTRODUZIONE

Basta sfogliare le pagine di un qualsiasi quotidiano per trovarsi di fronte a terminologie che fino a pochi anni fa risultavano assolutamente sconosciute ai più. “E-learning”, “distance learning”, “training on line”. Sono queste, nell’era di Internet e dei calcolatori, le nuove frontiere dell’apprendimento.

Il mondo sta cambiando. Il bagaglio di conoscenze di cui noi tutti disponiamo sembra destinato a segnare il passo di fronte alle recenti sfide imposte da un mondo globalizzato ed in continuo divenire. I tempi in cui la formazione professionale garantiva le conoscenze necessarie per un’intera vita lavorativa appaiono ora semplici reminescenze di un passato remoto. Il mondo verso il quale stiamo tendendo presuppone un aggiornamento rapido, continuo ed efficace delle proprie competenze. Esattamente ciò che l’e-learning è in grado di offrirci.

Ma che cos’è effettivamente l’e-learning?

Quello che comunemente identifichiamo con il termine “e-learning” deriva dalla fusione di due diversi campi di sperimentazione nelle tecnologie didattiche: la formazione a distanza (FAD) ed il “*Computer Based Training*” (CBT)¹. Se la formazione a distanza può vantare una lunga tradizione, in particolar modo all’interno della cultura anglosassone, dove già nel diciannovesimo secolo varie università ed istituzioni si occupavano di “vendere” formazione veicolata tramite i servizi postali dell’epoca², il Computer Based Training rispecchia un settore di attività decisamente più recente. Il CBT, ossia lo studio basato sull’uso del computer quale tecnologia didattica di auto-istruzione, inizia a diffondersi nel corso degli anni ‘70³, trovando applicazione

¹ <http://www.acidlearning.it/storia.asp>

² Stefania Fraccavento, *Teorie, modelli e sviluppi del mercato, a livello nazionale e internazionale, riguardanti i processi di e-learning* (<http://www.studiotaf.it/teoriemodellifad1.htm>).

³ <http://www.crimpa.it/centroeccellenza/ricerca/wp2.asp>

soprattutto nell'ambito delle discipline informatiche e nell'addestramento del personale all'utilizzo di software specifici.

Il primo avvicinamento tra le due metodologie didattiche avviene negli anni '90, quando sul mercato inizia a diffondersi la tecnologia del CD-ROM. Tale innovazione, che rende possibile la memorizzazione di un'enorme (almeno per l'epoca) quantità di informazioni su supporti facilmente trasportabili, spiana la strada all'affermazione del CBT in nuovi settori della formazione. Nascono e si affermano così le prime enciclopedie multimediali⁴, nonché i primi corsi di lingua su CD-ROM.

Nel frattempo i mass-media si accorgono della prepotente esplosione di un nuovo fenomeno. Si tratta di Internet. Nato, cresciuto e sviluppatosi sino a diventare quell'immenso agglomerato dello scibile umano che conosciamo oggi, grazie alla libera circolazione delle idee e delle informazioni che ha avuto ed ha tuttora luogo al suo interno. E' naturale che, trascorso un breve lasso di tempo, qualcuno abbia iniziato a vedere il World Wide Web come un ambiente naturale all'interno del quale collocare forme innovative di apprendimento a distanza.

Nascono così i primi sistemi di e-learning, grazie ai quali formazione a distanza e Computer Based Training giungono ad una definitiva fusione. Il successo di questi nuovi ambienti di apprendimento basati su Internet è straordinario. E si tratta di un fenomeno la cui entità non accenna a diminuire. Recenti stime affermano che, entro il 2005, il giro d'affari relativo al "training on line" raggiungerà la cifra record di 46 miliardi di dollari⁵. Cifra destinata ad aumentare ulteriormente, man mano che l'intero

⁴ A farne le spese fu in particolar modo l'Enciclopedia Britannica, le cui vendite subirono un autentico tracollo dopo l'ingresso sul mercato di Microsoft Encarta (Carl Shapiro, Hal R. Varian, *Information Rules*, pagg. 25-26).

⁵ <http://www.labitalia.com/articles/Approfondimenti/1529.html>

mondo occidentale procederà nel suo cammino verso l'ormai imminente era della "formazione continua"⁶.

Gli ambienti di e-learning hanno ormai raggiunto una discreta maturità. Sono stabili, efficienti e vengono commercializzati a costi tutto sommato sostenibili per le imprese operanti nel settore dell'erogazione di formazione a distanza. Solitamente, questi ambienti sono però rigidi. Bloccati su di una struttura "verticale" dell'insegnamento, tanto antiquata quanto difficile da modificare nelle menti di chi gli insegnamenti deve impartirli.

"In generale, è possibile affermare che attualmente un buon numero di corsi online sono organizzati come corsi per corrispondenza, con l'unica differenza rappresentata dal fatto di utilizzare Internet come mezzo di comunicazione. Vengono messe a disposizione raccolte di dispense da scaricare, stampare e leggere esattamente come un libro, insieme a test a scelta multipla da compilare: una modalità educativa che continua a presupporre un tipo di apprendimento passivo"⁷.

Sistemi e-learning di successo, quali ad esempio Evolution⁸, Training Partner 2003⁹ e Top Class¹⁰, sono fossilizzati proprio su una struttura didattica di questo tipo. Per rendersene conto è sufficiente osservare con un po' di attenzione le pagine web dove questi programmi vengono presentati al pubblico.

I produttori di Evolution, quale caratteristica di punta del software, ne mettono in risalto la capacità di offrire contenuti personalizzati per i diversi studenti: "*Personalizes*

⁶ http://www.governo.it/GovernoInforma/Dossier/formazione_continua_03/rapporto.html

⁷ Pierluigi Farace, *Strategie nell'e-learning: l'impatto del modello open source nelle scelte tecnologiche e funzionali*, pag. 327.

⁸ <http://www.outstart.com>

⁹ <http://trainingpartner.com/TP2003/tp2000web.dll/Home>

¹⁰ <http://www.wbtsystems.com/>

learning by determining the right training and assembling the right content for each learner¹¹”.

La brochure di Training Partner si focalizza principalmente sull’aderenza dell’ambiente di apprendimento a standard quali AICC¹² e SCORM¹³, accennando brevemente agli studenti, che possono navigare all’interno dei corsi, iscriversi alle varie classi e visualizzare il calendario degli impegni futuri: *“Learners can browse course catalogs, view calendars, enroll or waitlist for classes, review training history and track progress¹⁴*”.

L’home page di Top Class, con toni trionfalistici, annuncia invece che gli utenti possono collaborare con gli insegnanti attraverso forum ed e-mail: *“Learners and trainers collaborate through built-in discussion groups, class announcements, and internal or external email¹⁵*”.

Già ad un primo sguardo la situazione appare addirittura paradossale. I tanto decantati sistemi di e-learning, teoricamente in grado di garantire un apprendimento personalizzato, *“student-oriented”* come si sente dire spesso dai media, sembrano considerare gli studenti un semplice “contorno” dell’attività di apprendimento on-line. Semplici comparse, insomma, che devono limitarsi a ricevere passivamente quanto viene loro offerto dai gestori dei corsi. Giusto un forum e qualche indirizzo e-mail per sfogare il loro desiderio di comunicare, di interagire con i docenti ed i compagni di corso.

¹¹http://www.outstart.com/portal/index.jsp?epi_menuItemID=fd03713ce90331777ea129d4301000a0&epi_menuID=d882a74e6f6e33677ea129d4301000a0&epi_baseMenuID=7a3fa5f3cecaa06cce8e76b6301000a0

¹² <http://www.readygo.com/aicc/aic02/00aic02.htm>

¹³ <http://www.adlnet.org/index.cfm?fuseaction=scormabt>

¹⁴ <http://trainingpartner.com/TP2003/tp2000web.dll/LMS%20Online%20Elearning>

¹⁵ <http://www.wbtsystems.com/products/topclass>

Cosa c'è di realmente innovativo in tutto questo? Non si tratta forse del tradizionale approccio insegnante-studenti, tipico delle istituzioni scolastiche di ogni ordine e grado, prelevato in blocco e trasferito sul web?

Internet è in grado di offrire molto di più. Ad esempio può permettere di ribaltare la tradizionale concezione secondo cui gli insegnamenti debbano giungere solo ed esclusivamente dall'alto: da un docente o da chi per lui. In un ambiente di apprendimento elettronico a distanza, docenti e studenti convivono fundamentalmente sullo stesso piano. Entrambi rappresentano, in ultima istanza, un insieme di bit visualizzati sullo schermo di un computer. Qualcuno potrebbe addirittura arrivare a chiedersi per quale motivo uno studente dovrebbe ritenere che certi bit (inviati da un insegnante) abbiano un contenuto formativo maggiore rispetto ad altri (inviati da un compagno di corso).

Ciò che manca ai più diffusi ambienti di e-learning sono in sostanza gli strumenti necessari affinché gli studenti possano costituire, per i loro colleghi, attuali e futuri, una fonte di apprendimento complementare a quella "istituzionale". Gli studenti, da semplici utenti passivi, possono e devono diventare parte attiva del processo di formazione. Ritengo sia questa la grande sfida che il mondo dell'apprendimento elettronico a distanza dovrà affrontare nel prossimo futuro.

La piattaforma LMS¹⁶ Moodle può svolgere un ruolo cruciale in questo senso. L'epistemologia costruttivista che ne sta alla base, secondo cui il processo di creazione della conoscenza è un processo infinito, in cui ogni nuova creazione avviene sulla base di conoscenze precedenti¹⁷, rappresenta un'ottima base teorica di partenza per capovolgere i tradizionali schemi di insegnamento. E' basandosi su di un approccio del genere che diventa possibile sfruttare al meglio ogni potenziale fonte di apprendimento,

¹⁶ Acronimo che sta ad indicare "Learning Management System".

¹⁷ Joseph D. Novak, *Apprendimento significativo e Mappe Concettuali alla base di un Nuovo Modello Educativo*, paragrafo: "Come gli esseri umani creano la conoscenza".

comprese quelle provenienti dagli stessi studenti. Non a caso, Moodle dispone di un gran numero di funzionalità dedicate all'interazione tra studenti, affinché essi possano trarre il massimo beneficio possibile dalla condivisione delle rispettive esperienze formative. Quello che manca è però uno strumento che permetta loro di condividere agevolmente il materiale digitale di propria produzione.

Il modulo che ho sviluppato per questa tesi, denominato "Students' Notes", mira a colmare, almeno in parte, questa lacuna della piattaforma, offrendo agli studenti la possibilità di mettere in condivisione, mediante un'interfaccia grafica piacevole e di semplice utilizzo, i propri appunti delle lezioni.

In ambito universitario, lo scambio degli appunti delle lezioni tra studenti è forse la tradizione che più si è conservata col passare degli anni. In qualsiasi facoltà, umanistica o scientifica che sia, è assolutamente normale veder passare in continuazione studenti armati di quaderni e fogli volanti, diretti verso le macchine fotocopiatrici. Con un qualsiasi motore di ricerca si può ad ogni modo avere una conferma oggettiva dell'entità del fenomeno. Ricercando la stringa "student notes" con Google¹⁸, ad esempio, si ottengono circa 26'000 risultati. Una cifra considerevole, che diventa ancora più significativa nel momento in cui si opera una ricerca sul corrispondente italiano "appunti universitari". In questo caso, l'algoritmo di Google restituisce i collegamenti a 23'100 pagine web nelle quali viene affrontata la questione. Segno che nel nostro Paese l'interesse sull'argomento si attesta su livelli assoluti.

Il fatto che il modulo sia stato scritto per consentire lo scambio di appunti tra studenti, non pregiudica comunque un suo differente utilizzo. Esso può essere usato per condividere qualsiasi tipo di informazione digitale, dai testi scritti alle fotografie, dal software ai contenuti multimediali dei generi più disparati. In particolare, grazie alle sue

¹⁸ <http://www.google.com>

buone capacità di logging, “Students’ Notes” può essere efficacemente sfruttato come “*repository*” comune per eventuali lavori di gruppo.

Questo mio lavoro, ovviamente, non pretende di essere la panacea ai numerosi mali degli attuali sistemi di e-learning. E’ un primo passo, che risponde ad un’esigenza ben precisa. Quella di offrire agli utenti della piattaforma Moodle un sistema attraverso il quale condividere agevolmente materiale digitale di qualsiasi genere. E, facendo questo, riavvicinarli a quell’ideale di libera condivisione delle informazioni che ha portato Internet ad essere il fenomeno di portata mondiale che noi tutti conosciamo. Un fenomeno dal futuro incerto, contrastato com’è dalle logiche del profitto messe in atto da aziende che, nel World Wide Web, non vedono altro che un mercato globale, dagli elevati livelli di profitto e dalle pressoché inesistenti barriere all’ingresso.

Ribaltare l’approccio tradizionale all’apprendimento e rendere gli studenti parte attiva del processo formativo. Il futuro dell’e-learning deve passare da questa strada.

2. MOODLE

Parlare della storia di Moodle, almeno nelle sue fasi iniziali, significa parlare della vita di Martin Dougiamas, ideatore della piattaforma ed attuale leader del team di sviluppo.

Il progetto nacque negli anni '90, all'epoca in cui Martin svolgeva il lavoro di webmaster presso la Curtin University of Technology¹⁹ ed iniziò a scontrarsi con l'ambiente WebCT²⁰ utilizzato all'interno dell'ateneo. La sua forte convinzione, secondo cui l'educazione "*Internet-based*" potesse contare su potenzialità decisamente maggiori rispetto a quelle offerte da WebCT, lo spinse a conseguire un PhD in pedagogia. Affascinato in particolar modo dall'epistemologia del costruzionismo sociale e forte del suo background informatico, Martin varò la versione 1.0 di Moodle, acronimo di "*Modular Object-Oriented Dynamic Learning Environment*", il 20 agosto 2002. L'ambiente LMS²¹ fu testato all'interno di alcuni corsi universitari frequentati da un ristretto numero di studenti. Esso divenne oggetto di studi sullo sviluppo in relazione all'ambiente, che si focalizzarono sulla natura delle collaborazioni e delle riflessioni registrate all'interno di questi piccoli gruppi di partecipanti adulti.

Il resto è storia recente. L'aver rilasciato il proprio lavoro sotto licenza GPL²² ha fatto sì che centinaia di utenti e sviluppatori sparsi per il pianeta si avvicinassero a Moodle. Il suo sviluppo prosegue oggi con una velocità sorprendente anche in confronto ad altri importanti progetti Open Source²³ e la sua diffusione diventa sempre più ampia a livello mondiale²⁴. Il merito è da attribuire in parte al team di sviluppo, per gli sforzi profusi nel tentativo di rendere Moodle compatibile con il più ampio numero possibile di web

¹⁹ <http://www.curtin.edu.au/>

²⁰ WebCT è un ambiente di e-learning molto diffuso (<http://www.webct.com>).

²¹ Acronimo di "Learning Management System".

²² <http://www.gnu.org/copyleft/gpl.html>

²³ Proprio nel momento in cui sto scrivendo questa tesi la versione 1.4 è stata dichiarata stabile (<http://moodle.org/mod/forum/discuss.php?d=11782>).

²⁴ <http://moodle.org/sites/>

server e DBMS²⁵, ed in parte ai traduttori, grazie ai quali la piattaforma può oggi vantare di essere localizzata in oltre 40 lingue diverse, compresi arabo e cinese.

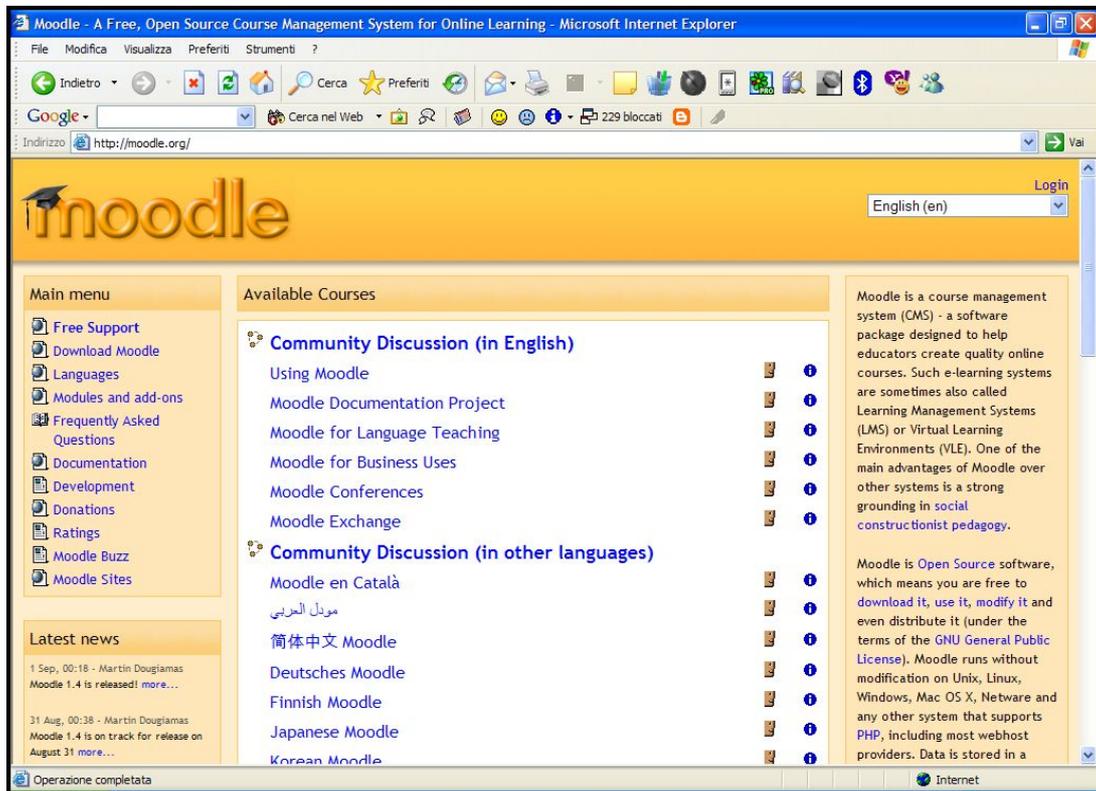


Figura 1 – L’home page di Moodle

La configurazione generale di Moodle viene gestita da un utente-amministratore definito durante la fase di installazione della piattaforma. Egli può scegliere tra i vari temi disponibili per quanto riguarda l’interfaccia grafica, nonché i vari moduli di attività didattica da integrare nell’installazione e le modalità di fruizione dei singoli corsi. I corsi possono infatti essere organizzati secondo tre formati distinti: settimanale (impostata la durata temporale del corso, il materiale didattico viene organizzato in

²⁵ “Moodle runs without modification on Unix, Linux, Windows, Mac OS X, Netware and any other system that supports PHP, including most webhost providers. Data is stored in a single database: MySQL and PostgreSQL are best supported, but it can also be used with Oracle, Access, Interbase, ODBC and others” (<http://moodle.org>).

chiave settimanale), per argomento (il corso viene suddiviso in diverse macro-aree tematiche all'interno delle quali trova posto il rispettivo materiale) o relazionale (l'intero corso viene gestito come se fosse un forum di discussione).

Moodle consente inoltre un'avanzata gestione degli utenti, che vengono suddivisi in tre categorie: studenti, docenti ed amministratori. Queste tipologie di account si differenziano tra loro in funzione del tipo di operazioni in grado di eseguire. Mentre gli studenti possono soltanto usufruire degli strumenti che vengono messi a loro disposizione, i docenti possono modificare liberamente i corsi di propria competenza. Ad un livello più alto stazionano gli amministratori, che possono creare nuovi corsi ed assegnarvi i rispettivi docenti.

Moodle ha infine il pregio di essere stato concepito come un ambiente modulare. Questo tipo di struttura permette agli sviluppatori di creare moduli ad hoc per rispondere a specifiche necessità che possono presentarsi durante l'utilizzo della piattaforma.

I moduli standard presenti in Moodle 1.4 sono, nel dettaglio, i seguenti:

- compito: consente al docente di assegnare agli studenti un particolare compito da svolgere. Questi effettueranno l'upload sulla piattaforma del compito svolto, che il docente provvederà a correggere assegnandovi al tempo stesso una valutazione personale;
- chat: offre ai partecipanti un'interfaccia sincrona per la comunicazione;
- diario: fornisce agli studenti una zona riservata all'interno della quale possono registrare annotazioni personali relative al corso. Il diario può essere visualizzato solo dallo studente che ne è curatore e dal docente, il quale può eventualmente commentare le riflessioni presenti;
- etichetta: consente di aggiungere testo o grafica all'interfaccia del corso;

- forum: offre ai partecipanti un'interfaccia asincrona per la comunicazione;
- glossario: dà la possibilità di aggiungere un glossario all'interno dei corsi, nel quale annotare la terminologia che si ritiene particolarmente utile;
- lezione: permette di creare vere e proprie lezioni, suddivise per pagine, al termine di ciascuna delle quali vi possono essere delle domande relative ai contenuti mostrati;
- quiz: consente al docente di creare dei test (domande aperte, a scelta multipla o del tipo vero/falso) da sottoporre agli studenti;
- risorsa: permette di visualizzare all'interno del corso qualsiasi contenuto elettronico (Word, PowerPoint, Flash, video, suoni, ecc...), collegandosi a files già presenti sul server o creati sul momento attraverso la generazione di documenti di testo semplice o HTML;
- scelta: offre al docente la possibilità di porre delle domande agli studenti, che potranno scegliere la risposta da un elenco pre-impostato di alternative;
- scorm: permette di creare nuovi contenuti aderenti allo standard SCORM per i learning objects²⁶;
- sondaggio: del tutto simile al modulo "scelta", ha la limitazione di consentire esclusivamente la creazione di quesiti la cui risposta sia selezionabile dall'elenco impostato dal docente;
- wiki: fornisce agli studenti l'accesso ad una sezione Wiki²⁷ all'interno della quale è possibile lavorare su documenti comuni;
- workshop: offre agli studenti la possibilità di assegnare una valutazione dei documenti presenti all'interno del corso.

²⁶ <http://www.eduworks.com/LOTT/tutorial/>

²⁷ <http://wiki.org/wiki.cgi?WhatIsWiki>

3. SMOOTHFILEOPERATOR

SmoothFileOperator²⁸, giunto alla versione 0.8.26, è uno script realizzato in PHP²⁹ appartenente alla categoria dei cosiddetti “*File Managers*”³⁰, letteralmente “gestori di files”. Il suo creatore è Olaf Wodrich, giovane programmatore bavarese, che ha deciso di distribuire il suo lavoro sotto licenza GPL.

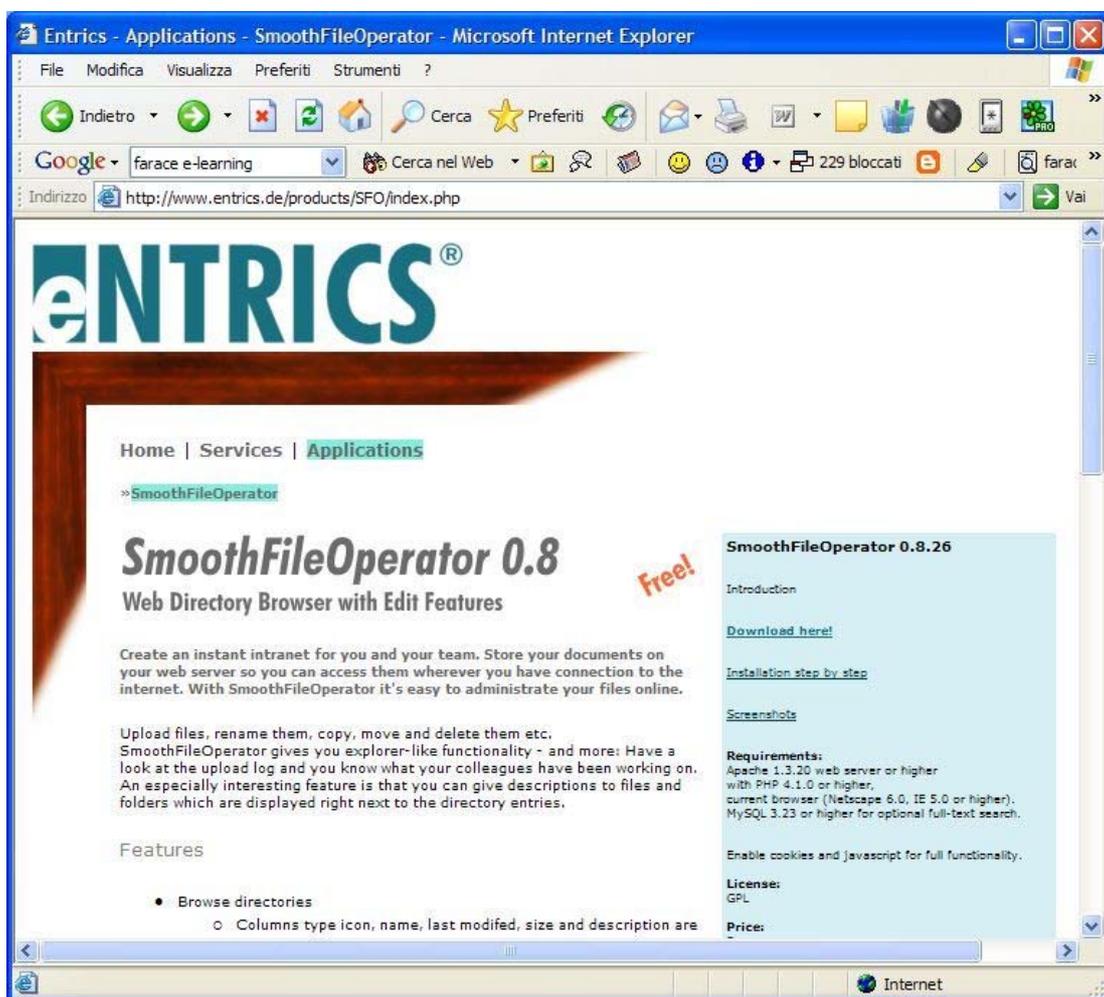


Figura 2 – L’home page di SmoothFileOperator 0.8

²⁸ <http://www.entrics.de/products/SFO/index.php>

²⁹ <http://www.php.net/>

³⁰ Per un elenco di alcuni file managers scritti in PHP, si veda <http://www.zend.com/apps.php?CID=263>

Per essere eseguito, SFO (abbreviazione di SmoothFileOperator) necessita sul lato server di un web server Apache³¹, con installato il modulo PHP³². Sul lato client è invece sufficiente un browser di ultima generazione³³, nel quale sia abilitato il supporto a Javascript e che sia in grado di accettare i cookies provenienti dal server.

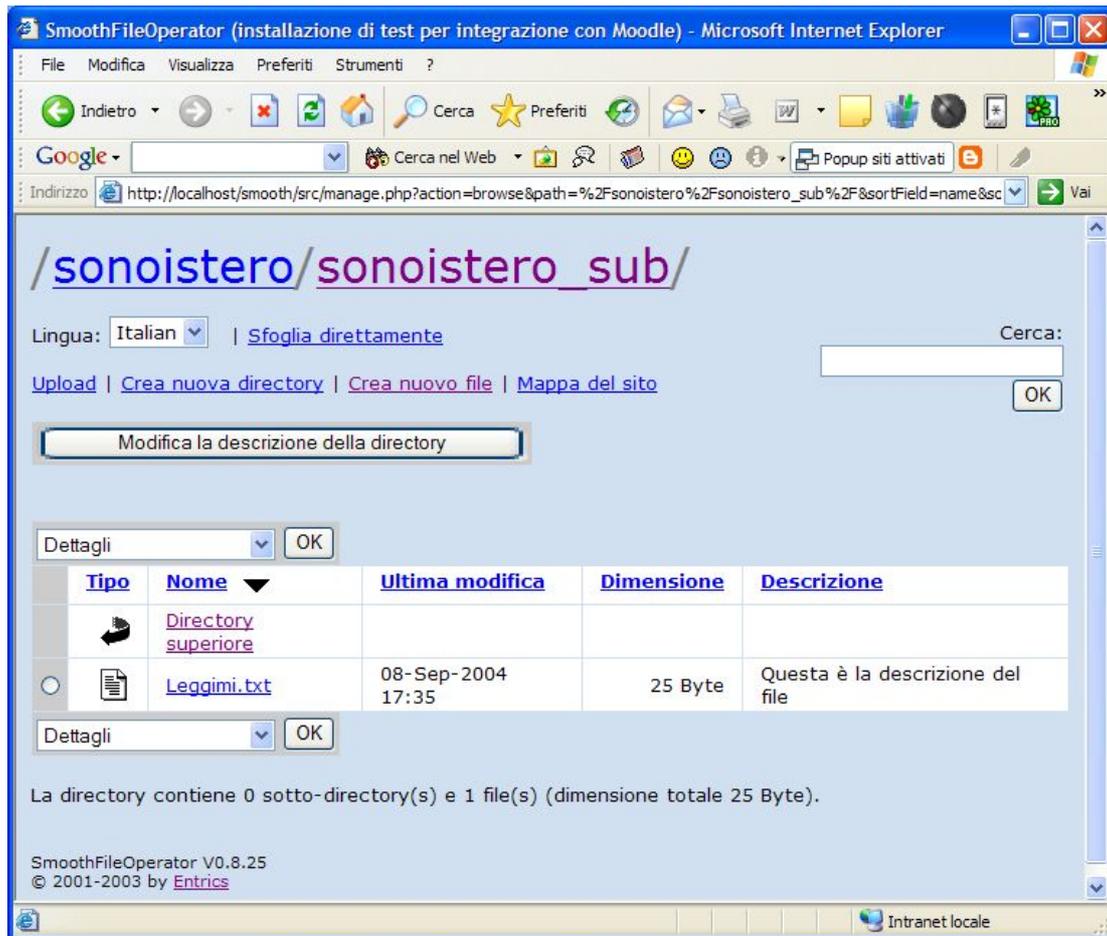


Figura 3 – L’interfaccia principale di SmoothFileOperator

Lo script mette a disposizione degli utenti un’interfaccia grafica mediante la quale effettuare tutte le operazioni tipiche di un file manager: upload di files, creazione di directory, modifica dei contenuti, ecc...

³¹ Nella versione 1.3.20 o superiore (<http://www.apache.org>).

³² Nella versione 4.1.0 o superiore.

³³ Netscape 6.0, Internet Explorer 5.0 o superiori.

Decisamente interessanti sono le possibilità di arricchire, con una descrizione testuale, qualunque file o cartella presente sul server e di tenere traccia di tutti gli upload effettuati. Tali caratteristiche rendono lo script particolarmente appetibile all'interno di ambienti di lavoro condivisi, nei quali può essere fortemente sentita l'esigenza di saper associare, in qualunque momento, ciascun contenuto al rispettivo proprietario.

Tra le decine di file managers disponibili in rete e che sarebbe stato possibile integrare all'interno di Moodle, la scelta è ricaduta su SmoothFileOperator per vari motivi. La maggior parte di questi riguardano il codice, fortemente basato sulla direttiva "*include()*" di PHP, che ha permesso una sua più agevole implementazione³⁴.

³⁴ Salvo poi presentare altre controindicazioni, come illustrato nel paragrafo 4.9: "Visualizzazione di un'istanza".

4. L'INTEGRAZIONE DI SMOOTHFILEOPERATOR ALL'INTERNO DI MOODLE

4.1. Creazione di un nuovo modulo

Per creare un nuovo modulo di attività didattica (*learning activity module*) il metodo più pratico è quello di utilizzare il template³⁵ messo a disposizione degli sviluppatori all'interno della sezione “*Developers Manual*”³⁶ di Moodle.org³⁷.

Il template in questione non è altro che un archivio compresso (in formato ZIP³⁸), contenente al suo interno i files fondamentali attorno ai quali costruire il nuovo modulo. Questi files sono, nel dettaglio, i seguenti:

- *mod.html*: semplice form HTML che viene visualizzato nel momento in cui si opera l'aggiunta (o l'aggiornamento) di un'istanza del modulo di attività didattica in questione;
- *version.php*: contiene alcune meta-informazioni riguardanti il modulo, quali la sua data di creazione ed il numero di versione;
- *icon.gif*: icona (di dimensioni 16 x 16 pixels), che identifica graficamente le istanze del modulo all'interno delle pagine relative ai vari corsi;
- *mysql.php*: contiene il codice PHP necessario per operare in maniera automatica un eventuale aggiornamento del modulo;
- *mysql.sql*: contiene lo schema, in linguaggio SQL, delle tabelle del database necessarie per il corretto funzionamento del modulo;
- *index.php*: contiene il codice PHP che permette di visualizzare tutte le istanze del modulo presenti all'interno di un determinato corso;

³⁵ http://moodle.org/mod/newmodule_template.zip

³⁶ <http://moodle.org/doc/?frame=developer.html>

³⁷ <http://moodle.org>

³⁸ http://www.fact-index.com/z/zi/zip__file_format_.html

- *view.php*: contiene il codice PHP che consente di visualizzare una particolare istanza del modulo;
- *lib.php*: file libreria, contenente le funzioni PHP richiamate più frequentemente dalle varie istanze del modulo.

Tutti questi files sono pre-impostati per un ipotetico modulo chiamato “NEWMODULE”. Il primo passo reso necessario è stato dunque quello di editarli, sostituendo tutte le occorrenze³⁹ della stringa “NEWMODULE” con “note”. Il nome “note” è stato scelto in accordo con quanto suggerito dal file *readme.txt*⁴⁰ che accompagna il template.

In aggiunta ai succitati files è necessario crearne manualmente uno nuovo, avente lo stesso nome del modulo ed estensione “.php” (nel nostro caso, dunque, un file dal nome “*note.php*”). Al suo interno devono essere collocate tutte le stringhe e le pagine di testo che verranno poi richiamate dal modulo.

Per la memorizzazione del testo all’interno del file, la forma utilizzata è quella dell’array associativo. Esiste infatti un array di questo tipo, denominato “*\$string*”, i cui valori vengono impostati mediante una sintassi molto semplice:

```
$string['modulename'] = 'Appunti';
$string['modulenameplural'] = 'Condivisione appunti';
```

I valori “modulename” e “modulenameplural” dell’array sono gli unici due obbligatori, in quanto vengono utilizzati rispettivamente per l’identificazione del modulo nel momento in cui ne deve venire aggiunta una nuova istanza e per la costruzione dell’interfaccia di navigazione, parte integrante dell’header delle pagine dei corsi.

³⁹ Per una definizione formale di “occorrenza” in ambito linguistico, si veda: http://www.tlab.it/it/allegati/help_it_online/gcoocor.htm

⁴⁰ “*This name should be a single English word, if possible, all lowercase and with only a-z characters*”.

Ovviamente è possibile “farcire” queste stringhe di tags HTML (in modo tale da agevolarne la resa grafica), come nell’esempio qui sotto riportato:

```
$string['errore_include'] = '<b>Si è verificato un errore sul server</b><br>'.  
'<br>Si consiglia di provare ad aggiornare la pagina dal browser.<br>'.  
'Nel caso in cui si dovesse riscontrare nuovamente lo stesso problema, '  
"si prega di contattare l'amministratore di sistema.";
```

Le stringhe verranno successivamente richiamate, all’interno del codice, mediante la funzione di Moodle “*get_string()*”, caratterizzata dalla seguente sintassi:

```
get_string("nome_della_stringa","nome_del_modulo");
```

Per stampare la stringa “*error_include*” vista qui sopra, ad esempio, si utilizzerà il codice seguente:

```
print get_string("errore_include","note");
```

Tale sistema, che a primo impatto potrebbe sembrare un’inutile complicazione, si rende necessario per garantire una facile “internazionalizzazione⁴¹” del modulo. Nel caso, infatti, in cui occorra tradurlo in una nuova lingua, colui che effettua la traduzione non ha la necessità di mettere mano al codice, ma può limitarsi a modificare in maniera adeguata il testo contenuto nel file “*note.php*” e collocarlo successivamente all’interno della cartella appropriata⁴².

⁴¹ “Moodle has been designed for internationalisation. Each 'string' or 'page' of text that is displayed as part of the interface is drawn from a set of language files. Each language is a subdirectory of the directory 'lang'” (<http://moodle.org/doc/?file=developer.html#languages>).

⁴² Quale sia la cartella appropriata, verrà analizzato nel paragrafo 4.2: “Installazione del nuovo modulo: interazione con il file system”.

4.2. Installazione del nuovo modulo: interazione con il file system

Una volta preparati i files elencati al paragrafo precedente è possibile procedere all'installazione del nuovo modulo. Per farlo occorre posizionarsi all'interno della cartella principale dell'installazione di Moodle⁴³, individuare la cartella “mod” e crearvi al suo interno una nuova subdirectory avente lo stesso nome del modulo che si desidera installare (nel nostro caso, come succitato, “note”). In questa nuova cartella devono essere inseriti i files: *icon.gif*, *index.php*, *lib.php*, *mod.html*, *version.php* e *view.php*, mentre *mysql.php* e *mysql.sql* sono da collocare all'interno di un'ulteriore sottodirectory, chiamata “db”. Il file contenente le stringhe di testo relative alla lingua (*note.php*) deve invece essere inserito all'interno della sottocartella “lang/it⁴⁴” della directory principale di Moodle.

La struttura risultante del file system è quindi la seguente:

- Directory principale dell'installazione di Moodle;
 - subdirectory “\mod”
 - subdirectory “\note”
 - *icon.gif*
 - *index.php*
 - *lib.php*
 - *mod.html*
 - *version.php*
 - *view.php*
 - subdirectory “\db”

⁴³ A titolo esemplificativo, nella mia installazione locale di Moodle (su piattaforma Windows), la cartella principale risulta essere: “C:\Programmi\EasyPHP1-7\www\moodle”.

⁴⁴ Questo, poiché il modulo è stato sviluppato originariamente in lingua italiana. Nel caso in cui il file “*note.php*” fosse stato redatto in una lingua diversa, avrebbe dovuto trovare posto nella cartella appropriata. Per una panoramica dei linguaggi supportati da Moodle, si veda: <http://moodle.org/download/lang/>

- *mysql.php*
- *mysql.sql*
- subdirectory “**lang**”
 - subdirectory “**it**”
 - *note.php*

Terminata la copia dei files, affinché il nuovo modulo possa essere considerato a tutti gli effetti installato, occorre che l’amministratore di Moodle (o comunque un utente che possa godere delle medesime credenziali) effettui il login sul sito e visiti la pagina di amministrazione.

4.3. Installazione del nuovo modulo: interazione con il DB di Moodle

Alla prima occasione in cui un utente di Moodle, con credenziali “amministrative”, accede alla pagina di amministrazione del portale, viene innescato un duplice procedimento.

Da un lato viene aggiunto un nuovo record alla tabella “*mdl_modules*⁴⁵”, in modo tale che Moodle possa considerare il nuovo modulo come tale. Il record inserito in coda alla tabella è costituito dai seguenti campi:

- *id*: identificativo univoco del modulo;
- *name*: nome del modulo;
- *version*: meta-informazioni su data di creazione e numero di versione del modulo, prelevate dal file *version.php*;

⁴⁵ Il nome della tabella è arbitrario per quanto riguarda il prefisso, che viene infatti impostato durante la fase di installazione di Moodle. In questo caso, avendo lasciato il prefisso impostato sul valore standard “mdl_”, il nome risultante della tabella è “mdl_modules”.

- *cron*: intervallo di secondi, passato il quale viene eseguita un'operazione di cron⁴⁶ sul modulo;
- *lastcron*: data e ora dell'ultima operazione di cron effettuata;
- *visibile*: variabile booleana, che assume i valori 1 o 0, a seconda che il modulo sia visibile o meno agli utenti.

Dall'altro lato viene generata una nuova tabella, destinata a contenere al suo interno il riferimento a tutte le future istanze del modulo. In accordo con quanto specificato dalla “*Coding Guide*”⁴⁷, la tabella è stata costruita con al proprio interno i tre campi seguenti:

- *id* : identificativo univoco dell'istanza;
- *course* : identificativo del corso in cui è contenuta l'istanza del modulo;
- *name* : nome completo dell'istanza.

Le istruzioni SQL in grado di creare una tabella di questo tipo (ottenute costruendo la tabella mediante l'interfaccia grafica *phpMyAdmin*⁴⁸ ed effettuando un'operazione di copy&paste sul codice generato) sono state incluse all'interno del file *mysql.sql*:

```
CREATE TABLE `prefix_note` (
  `id` int(10) unsigned NOT NULL auto_increment,
  course int(10) unsigned NOT NULL default '0',
  name varchar(255) NOT NULL default "",
  PRIMARY KEY (`id`)
) TYPE=MyISAM COMMENT='Each of these is a students notes instance';
```

⁴⁶ “Cron is the name of program that enables unix users to execute commands or scripts (groups of commands) automatically at a specified time/date. It is normally used for sys admin commands, like *makewhatis*, which builds a search database for the *man -k* command, or for running a backup script, but can be used for anything. A common use for it today is connecting to the internet and downloading your email” (<http://www.unixgeeks.org/security/newbie/unix/cron-1.html>).

⁴⁷ <http://moodle.org/doc/?file=coding.html>

⁴⁸ http://www.phpmyadmin.net/home_page/

Si noti che il campo “id” costituisce la chiave primaria della tabella. E’ questo il motivo per cui si è reso necessario aggiungervi l’attributo “auto_increment⁴⁹”.

4.4. Aggiunta di una nuova istanza del modulo

Una volta che il nuovo modulo è stato installato, gli utenti di Moodle possono aggiungerne un’istanza a ciascuno dei corsi presenti sul sito.

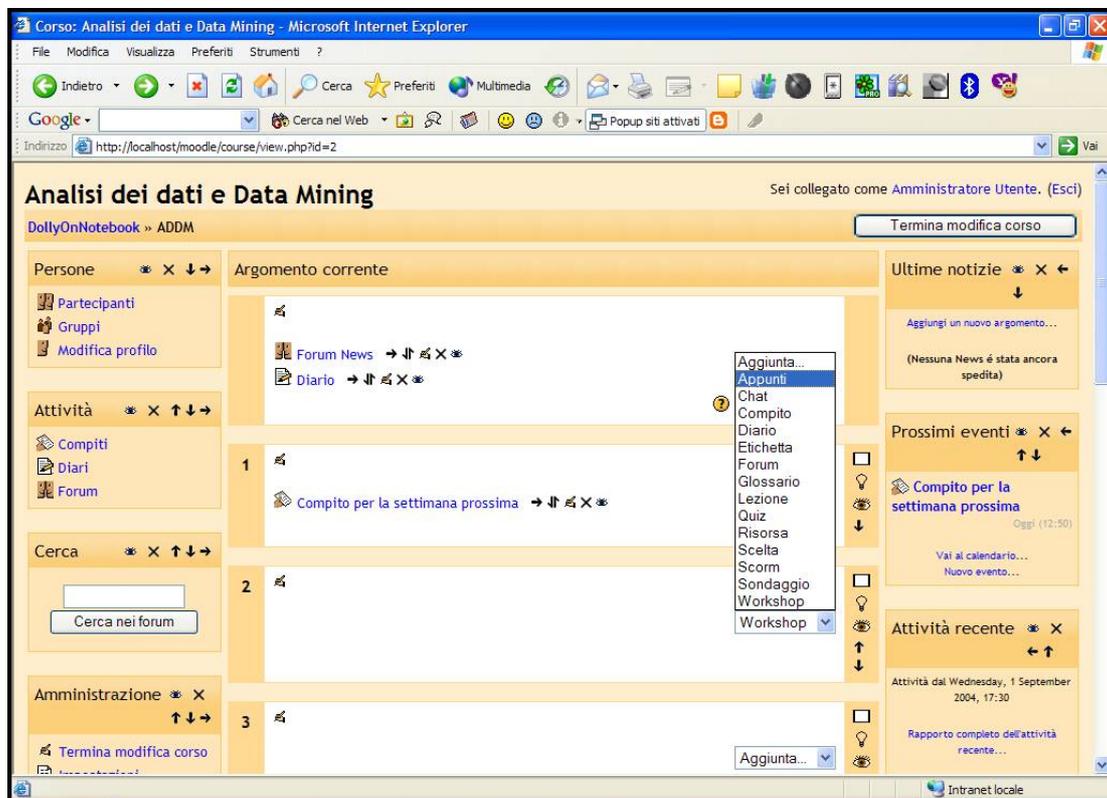


Figura 4 – L’interfaccia a disposizione di insegnanti ed amministratori per aggiungere al corso corrente un’istanza del modulo

⁴⁹ “The AUTO_INCREMENT column modifier automatically increases the value of a column by adding 1 to the current maximum value. It provides a counter that is useful for creating unique values. The value of a newly inserted row into an AUTO_INCREMENT column starts at 1 and increases by 1 for every record that is inserted into the table” (Mark Maslakowski, *Sam’s Teach Yourself MySQL in 21 Days*, pagg. 52-53).

Per farlo (a patto, inutile sottolinearlo, che l'utente corrente sia in possesso dei necessari privilegi di "editing"⁵⁰ sul corso prescelto) è sufficiente sfruttare la tradizionale interfaccia grafica per l'aggiunta di nuove attività didattiche ai corsi. In seguito all'installazione del nuovo modulo, il menu a discesa che presenta tutte le possibilità di inserimento si è infatti arricchito della voce "Appunti".

Una limitazione che si è resa obbligatoria (a causa delle modalità di implementazione scelte per questo modulo) è quella relativa al numero di istanze aggiungibili a ciascun corso. Per ragioni che diventeranno ovvie in seguito⁵¹ è infatti possibile aggiungere una sola istanza di "Students' Notes" per ciascun corso configurato sul portale. A tale mancanza (se così può essere definita) sopperisce comunque la possibilità, offerta di default da SmoothFileOperator, di creare varie cartelle all'interno delle quali archiviare il materiale da condividere.

Nel momento in cui l'utente, all'interno di un corso, decide di aggiungere un'istanza del modulo, Moodle richiama lo script contenuto all'interno del file "*course/mod.php*".

Questo script svolge una duplice funzione:

- in primo luogo propone all'utente il form (il cui codice è contenuto all'interno del file "*mod.html*"), attraverso il quale egli può scegliere il nome da dare alla nuova istanza (tale nome sarà visualizzato nella pagina del corso, accanto all'icona corrispondente⁵²);
- in secondo luogo, a patto che l'utente non rinunci a creare la nuova istanza (premendo il pulsante "indietro" del proprio browser o sfruttando l'interfaccia di

⁵⁰ In Moodle, i privilegi di editing (ossia di modifica) relativi ad un corso sono solitamente appannaggio degli amministratori del sito e degli insegnanti attribuiti ai singoli corso.

⁵¹ Si veda il paragrafo 4.5: "Aggiunta di una nova istanza del modulo: interazione con il file system".

⁵² L'icona che identifica l'istanza del modulo è "*icon.gif*", cui si è accennato nel paragrafo 4.1: "Creazione di un nuovo modulo".

navigazione dell'header), ma al contrario le assegna un nome e faccia successivamente clic sul pulsante “Salva modifiche⁵³”, lo script esegue la sequenza di istruzioni necessarie alla sua creazione. Queste possono essere suddivise in due categorie: quelle che vanno ad interagire con il file system del server su cui è installato Moodle (creando le cartelle destinate a contenere al proprio interno il materiale da condividere) e quelle che si occupano invece di modificare in maniera opportuna le tabelle del database.



Figura 5 – Il form dove assegnare il nome all’istanza di “Students’ Notes” da aggiungere al corso corrente

⁵³ Inviando in questo modo il form visualizzato sullo schermo.

4.5. Aggiunta di una nuova istanza del modulo: interazione con il file system

Lo script verifica prima di tutto quale tipo di operazione sia in corso, effettuando un controllo sulla variabile “*\$mod->mode*”⁵⁴ mediante la sintassi:

```
switch ($mod->mode) {
    case “update”:
        // codice per l’update dell’istanza
    case “add”:
        // codice per l’aggiunta di una nuova istanza
    case “delete”:
        // codice per la rimozione di un’istanza
}
```

Nel momento in cui si aggiunge un’istanza di “Students’ Notes”, la porzione di codice che viene eseguita è ovviamente quella contenuta all’interno della condizione *case* “add”. Essa consiste sostanzialmente nell’aggiornamento del database di Moodle⁵⁵, con le informazioni relative alla nuova istanza ed alla posizione esatta in cui essa deve essere visualizzata.

Siccome il modulo in questione, durante la fase di creazione di un’istanza, ha un comportamento differente rispetto a quello tipico di tutti gli altri moduli di Moodle (questo vale almeno per quanto riguarda i moduli “standard”⁵⁶), all’interno del *case*

⁵⁴ I valori che può assumere la variabile “*\$mod->mode*” sono: “update” (nel caso in cui si stia aggiornando un’istanza già esistente), “add” (nel caso in cui se ne stia aggiungendo una nuova) o “delete” (nel caso se ne stia cancellando una già esistente).

⁵⁵ L’aggiornamento del database, in corrispondenza dell’aggiunta di una nuova istanza, è spiegato al paragrafo 4.6: “Aggiunta di una nuova istanza del modulo: interazione con il DB di Moodle”.

⁵⁶ I moduli standard di Moodle 1.4, alla data del 5/9/2004, sono: “Assignment”, “Chat”, “Choice”, “Forum”, “Glossary”, “Journal”, “Label”, “Lesson”, “Quiz”, “Resource”, “Scorm”, “Survey”, “Wiki”, “Workshop” (si veda a riguardo la pagina <http://moodle.org/download/modules/>).

“add” è stato inserito un controllo sulla variabile “*\$mod->mode*”⁵⁷. Tale controllo viene realizzato mediante un’istruzione “if”⁵⁸ di questo tipo:

```
switch ($mod->mode) {
    case “add”:
        if ($mod->modulename == "note") {
            // Codice eseguito nel caso di aggiunta di
            // una nuova istanza di “Students’ Notes”
        }
    }
}
```

Nel caso in cui si stia effettuando l’aggiunta di un’istanza di “Students’ Notes”, l’espressione ‘*\$mod->modulename == “note”*’ risulta essere vera, e viene pertanto eseguito il codice appropriato.

Le cartelle in cui verranno archiviati gli appunti inviati dagli studenti sono contenute all’interno del percorso “mod/note”. Esse sono nel dettaglio:

- “*notedirs*”: cartella contenitore, che conterrà al suo interno le subdirectory relative ai singoli corsi, dove saranno memorizzati i files inviati dagli utenti;
- “*notedirs_logs*” e “*notedirs_Team*”: a loro volta cartelle contenitori, conserveranno al proprio interno le subdirectory relative ai singoli corsi, dove verranno registrate le operazioni effettuate dagli utenti.

Come prima cosa vengono dunque impostate le variabili contenenti i percorsi delle cartelle da creare o di cui occorre verificare l’esistenza:

⁵⁷ La variabile “*\$mod->mode*” contiene una stringa di testo, corrispondente al nome del modulo su cui si sta lavorando (nel nostro caso, “note”).

⁵⁸ “*An if statement tests conditions, executing a block of statements when a condition is true*” (Janet Valade, *PHP 5 For Dummies*, pag. 140).

```

$directory_contenitore = $CFG->dirroot . "/mod/note/notedirs";
$directory_logs = $CFG->dirroot . "/mod/note/notedirs_logs/";
$directory_Team = $CFG->dirroot . "/mod/note/notedirs_Team/";
$directory_corso = $CFG->dirroot . "/mod/note/notedirs/" .
    $course->shortname;
$directory_corso_logs = $CFG->dirroot . "/mod/note/notedirs_logs/" .
    $course->shortname;
$directory_corso_Team = $CFG->dirroot . "/mod/note/notedirs_Team/" .
    $course->shortname;

```

Successivamente viene effettuato un controllo sulla presenza delle tre cartelle “contenitori” sul server in cui è installato Moodle, mediante il richiamo alla funzione “*is_dir()*”⁵⁹. Nel caso in cui non siano presenti, esse vengono create mediante un’altra funzione di PHP, chiamata “*mkdir()*”⁶⁰,

```

if (!is_dir($directory_contenitore)) {
    mkdir("$directory_contenitore");
}
if (!is_dir($directory_logs)) {
    mkdir("$directory_logs");
}
if (!is_dir($directory_Team)) {
    mkdir("$directory_Team");
}

```

Queste cartelle vengono create, generalmente, nel momento in cui si aggiunge per la prima volta un’istanza di “Students’ Notes” all’installazione corrente di Moodle.

⁵⁹ <http://www.php.net/manual/en/function.is-dir.php>

⁶⁰ <http://www.php.net/mkdir>

Questo, a condizione che non siano state effettuate rimozioni “manuali” delle cartelle in questione da parte dell’amministratore di sistema.

Il passo successivo svolto dallo script è quello di controllare se, all’interno di queste cartelle “contenitori”, siano già presenti le directories relative ai singoli corsi. Queste cartelle, generate contestualmente all’aggiunta di una nuova istanza del modulo all’interno dal corso, assumono lo “short name⁶¹” del corso in questione.

Ad esempio, per un corso denominato “Analisi dei Dati e Data Mining”, caratterizzato dal nome breve “ADDM”, le cartelle “specifiche” che verranno create sul server saranno:

- “*mod/note/notedirs/ADDM*”;
- “*mod/note/notedirs_logs/ADDM*”;
- “*mod/note/notedirs_team/ADDM*”.

Il controllo che si occupa di verificare la loro esistenza⁶² (e crearle, nel caso in cui esse non siano esistenti, mediante la solita funzione “*mkdir()*”) è il seguente:

```
// La cartella esiste?
if (is_dir($directory_corso)) {
    // Cartella già esistente
    print get_string("errore_gia_esistente","note");
    die();
} elseif (!is_dir($directory_corso)) {
    mkdir("$directory_corso");
}
```

⁶¹ Lo “short name” (o “nome breve del corso”) viene impostato dall’amministratore di Moodle nel momento in cui crea un nuovo corso. Esso è accessibile attraverso la variabile “*\$course->shortname*”.

⁶² In realtà, il codice mostrato si occupa di controllare soltanto l’esistenza della cartella. “*\$directory_corso*”. Questo perché si parte dal presupposto che, se esiste la cartella “*\$directory_corso*”, esisteranno anche le cartelle “*\$directory_corso_logs*” e “*\$directory_corso_Team*” (e viceversa).

```
        mkdir("$directory_corso_logs");
        mkdir("$directory_corso_Team");
    }
```

Nel caso in cui la cartella “*\$directory_corso*” risulti già presente sul server, viene visualizzato un messaggio di errore e la procedura di aggiunta di una nuova istanza viene immediatamente arrestata mediante l’istruzione “*die()*”⁶³.

Questo tipo di implementazione che, come accennato in precedenza, ha lo svantaggio di permettere l’inserimento di una sola istanza del modulo all’interno di ciascun corso, d’altro canto si caratterizza per la facilità di realizzazione e per il limitatissimo numero di accessi al database di Moodle.

4.6. Aggiunta di una nuova istanza del modulo: interazione con il DB di Moodle

Nel momento in cui l’utente invia il form mediante il pulsante “Salva modifiche”, (ammesso che venga superato con successo il controllo sulla validità del nome attribuito alla nuova istanza) al database vengono aggiunti tre nuovi records.

Il primo, inserito in coda alla tabella “*mdl_note*”, é una tupla⁶⁴ composta da tre soli valori⁶⁵. Questi possono essere ad esempio:

(5,2,”Riassunti degli studenti”)

⁶³ <http://www.php.net/manual/en/function.die.php>

⁶⁴ Si definisce “*tupla su un insieme di attributi X, una funzione t che associa a ciascun attributo A appartenente ad X un valore del dominio DOM(A)*” (Atzeni P., Ceri S., Paraboschi S. e Torlone R., *Basi di dati*, pag. 20).

⁶⁵ L’identificativo univoco dell’istanza, quello del corso ed il nome dell’istanza; si veda, a riguardo, il paragrafo 4.3: “Installazione del nuovo modulo: interazione con il DB di Moodle”.

Gli altri due records, invece, rappresentano la registrazione dell'avvenuta aggiunta della nuova istanza all'interno del corso. Tale registrazione ha luogo all'interno della tabella "mdl_log⁶⁶" e viene effettuata mediante una duplice chiamata alla funzione *add_to_log()* di Moodle:

```
add_to_log($course->id, "course", "add mod",
          "../mod/$mod->modulename/view.php?id=$mod->coursemodule",
          "$mod->modulename $mod->instance");
```

```
add_to_log($course->id, $mod->modulename, "add",
          "view.php?id=$mod->coursemodule",
          "$mod->instance", $mod->coursemodule);
```

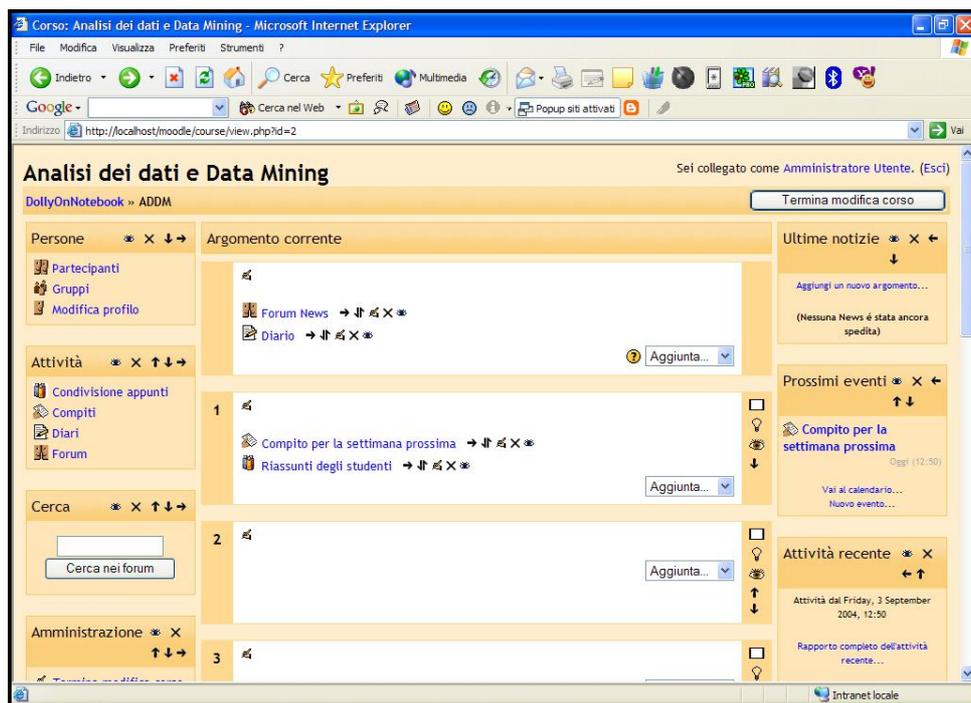


Figura 6 – La pagina principale del corso, con la nuova istanza di “Students’ Notes”, chiamata “Riassunti degli studenti”

⁶⁶ Vale il discorso fatto precedentemente (nota nr. 9) a proposito del nome della tabella (da questo punto in poi verranno omesse le annotazioni a riguardo). La struttura della tabella “mdl_log” è la seguente: mdl_log = {id, time, userid, ip, course, module, cmid, action, url, info}.

Questi records possono essere ad esempio:

(1141, 1094224255, 1, 127.0.0.1, 2, course, 0, add mod,
../mod/note/view.php?id=23, note 18)

(1142, 1094224256, 1, 127.0.0.1, 2, note, 23, add, view.php?id=23, 18)

4.7. Aggiornamento di un'istanza

Una volta che il nuovo modulo “Students’ Notes” è stato correttamente installato e che ne sono state aggiunte una o più istanze all’interno dei corsi, potrebbe rendersi necessario modificare il nome di una di queste⁶⁷. Lo stesso risultato è ottenibile percorrendo due strade diverse:

- modificando manualmente il campo “*name*” della tabella “*mdl_note*” di Moodle (mediante un’interfaccia grafica quale *phpMyAdmin*, o direttamente da console digitando le opportune istruzioni SQL⁶⁸);
- utilizzando l’interfaccia di editing di Moodle.

Tale interfaccia, accessibile facendo clic sull’icona “aggiorna” (che compare, una volta attivata la modalità di editing di Moodle, accanto a ciascuna istanza presente nel corso), è richiamata dal solito script “*course/mod.php*”.

⁶⁷ I motivi possono essere dei più svariati: errore di battitura durante la creazione, mutate esigenze grafiche, ecc...

⁶⁸ La sintassi dovrebbe risultare simile alla seguente:
UPDATE mdl_note SET name = ‘*nome_corretto*’ WHERE id = *id_istanza*.

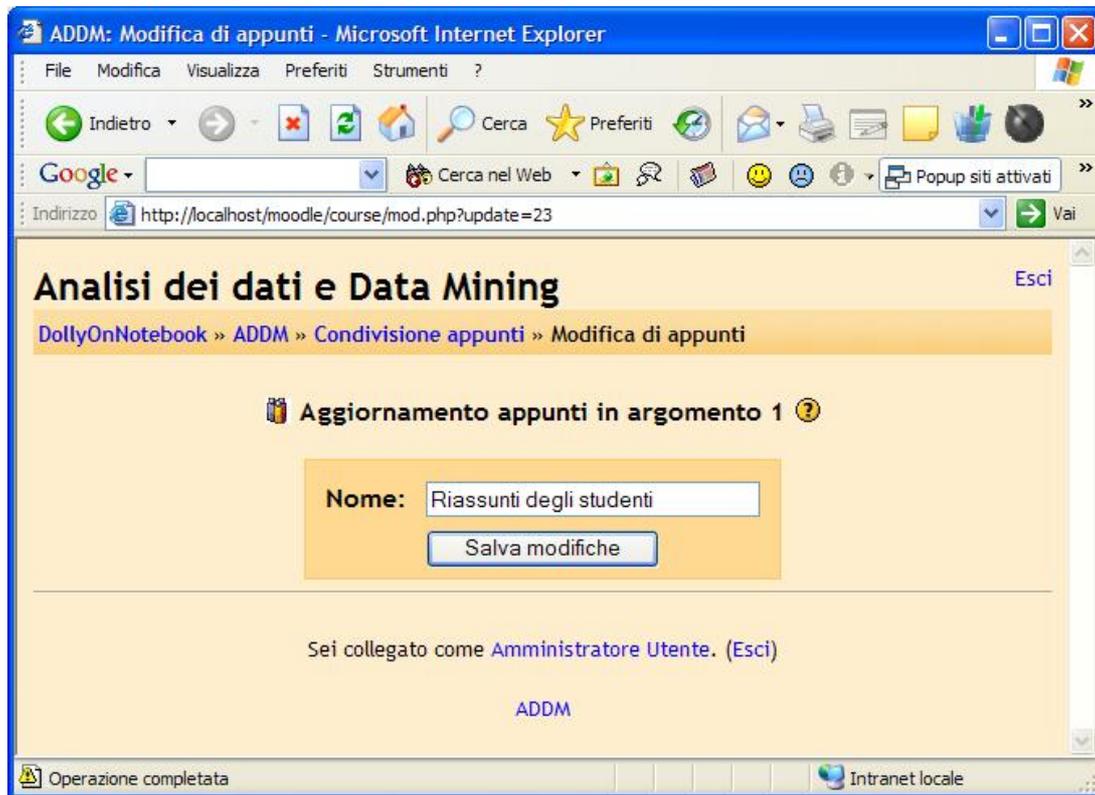


Figura 7 – Il form per la modifica del nome di un’istanza di “Students’ Notes” già esistente

Il valore di “ $\$mod->mode$ ”, che viene passato allo script⁶⁹ attraverso l’URL, è questa volta “update”. Lo switch provvede dunque ad eseguire la porzione di codice contenuta all’interno della condizione *case* “update”: essa fa in modo che all’utente venga presentato un form dal quale modificare agevolmente il nome dell’istanza (pre-impostato sul nome corrente dell’istanza stessa). Una volta modificato il nome ed inviato il form con la pressione del pulsante “Salva modifiche”, lo script provvede ad aggiornare il database, memorizzando all’interno della tabella “mdl_note” il nuovo nome dell’istanza modificata.

⁶⁹ La struttura dello script è quella evidenziata all’inizio del paragrafo 4.5: “Aggiunta di una nuova istanza del modulo: interazione con il file system”.

4.8. Cancellazione di un'istanza

Il procedimento adottato per la cancellazione di un'istanza ripercorre, in senso contrario, quello seguito per la sua creazione. Nel momento in cui un utente, in possesso dei privilegi di editing sul corso in questione, fa clic sull'icona “cancella”, si trova davanti ad una schermata che gli chiede di confermare l'operazione di rimozione dell'istanza.

Il tutto ruota ancora attorno allo script “*course/mod.php*”, il cui valore di “*\$mode*”, ricevuto via URL, corrisponde in questo caso a “delete”. In maniera analoga a quanto avviene nel momento dell'aggiunta di una nuova istanza di “Students' Notes”, anche in questo caso è stato inserito un controllo all'interno del *case* “delete”, che si occupa di verificare quale tipo di istanza si sia in procinto di rimuovere.



Figura 8 – La schermata che si presenta all'utente nel momento in cui sceglie di eliminare un'istanza

La struttura dello script risulta dunque essere la seguente:

```
switch ($mod->mode) {
    case "delete":
        if ($mod->modulename == "note") {
            // Codice eseguito nel caso di rimozione
            // di un'istanza di "Students' Notes"
        }
    }
}
```

Nel caso in cui l'espressione `'$mod->modulename == "note"'` risulti essere vera, viene eseguita la porzione di codice che provvede ad eliminare le tre cartelle all'interno delle quali sono memorizzati gli eventuali files condivisi dagli utenti e tutti i relativi logs.

Date le limitazioni insite nella funzione PHP `"rmdir()"`⁷⁰, che consente di eliminare una cartella solo a condizione che essa sia vuota, si è scelto di utilizzare la più performante funzione `"deldir()"`⁷¹, che provvede a svuotare preventivamente la cartella il cui path è ricevuto in input:

```
function deldir($dir) {
    $current_dir = opendir($dir);
    while($entryname = readdir($current_dir)) {
        if(is_dir("$dir/$entryname") and ($entryname != "." and
        $entryname!="..")) {
            deldir("$dir/$entryname");
        } elseif($entryname != "." and $entryname!="..") {
            unlink("$dir/$entryname");
        }
    }
}
```

⁷⁰ <http://it.php.net/manual/en/function.rmdir.php>

⁷¹ [http://groups.google.it/groups?q=flexer%40cutephp.com+deldir\(\)&hl=it&lr=&ie=UTF-8&selm=note-31418%40php.net&rnum=1](http://groups.google.it/groups?q=flexer%40cutephp.com+deldir()&hl=it&lr=&ie=UTF-8&selm=note-31418%40php.net&rnum=1)

```

    }
    closedir($current_dir);
    rmdir(${dir});
}

```

Tale funzione viene richiamata all'interno del codice con la sintassi seguente:

```

deldir($directory_corso);
deldir($directory_corso_logs);
deldir($directory_corso_Team)72;

```

Il codice restante provvede ad eliminare dalla tabella “*mdl_note*” il record corrispondente all’istanza rimossa, nonché a “loggere” l’operazione in coda alla tabella “*mdl_log*”.

4.9. Visualizzazione di un’istanza

Nel momento in cui si è iniziato il lavoro di integrazione di SmoothFileOperator quale nuovo modulo di Moodle, il problema principale che si è dovuto affrontare è stato quello legato al passaggio delle variabili tra i due ambienti. SmoothFileOperator è infatti strutturato per ricevere gli input necessari mediante URL. Tale scelta, se da un lato rende abbastanza agevole la stesura e la comprensione dello script, dall’altro presenta grossi problemi in un caso come questo. Effettuando l’inclusione del file principale di SFO via HTTP, vengono infatti perse tutte le variabili di Moodle necessarie per la sua corretta integrazione all’interno del portale.

Per ovviare a questo inconveniente si è fatto ricorso ad un file di testo temporaneo, dove, precedentemente al caricamento del modulo per la condivisione degli appunti, vengono memorizzate tutte le variabili che serviranno a quest’ultimo. Le istruzioni per

⁷² I valori di “*\$directory_corso*”, “*\$directory_corso_logs*” e “*\$directory_corso_Team*” sono gli stessi definiti all’interno del paragrafo 4.5: “Aggiunta di una nuova istanza del modulo: interazione con il file system”.

la sua creazione sono contenute all'interno del file `“mod/note/view.php”`, che viene invocato nel momento in cui un utente sceglie di visualizzare un'istanza di “Students' Notes”.

```
$fh = fopen("variabili.txt","w")
      or die(get_string("errore_scrittura","note"));
```

Come si può notare dal codice riportato qui sopra, il file di testo (a cui è assegnato il nome `“variabili.txt”`) viene aperto mediante la funzione di PHP `“fopen()”`⁷³, in modalità `“w”`. Questo sta a significare che, nel caso in cui il file non esista, esso viene creato; nell'ipotesi contraria, invece, il file esistente viene completamente sovrascritto⁷⁴.

All'interno del file, le informazioni vengono memorizzate tramite la funzione `“fwrite()”`⁷⁵. Le variabili di cui sarà necessario servirsi in seguito vengono disposte in quantità di una per ciascuna riga, grazie all'aggiunta del carattere `“\n”`⁷⁶ in coda al secondo parametro di `“fwrite()”`: questa soluzione rende decisamente più agevoli le successive operazioni di lettura del file.

```
fwrite($fh,$course->id."\n");           // Identificativo del corso
fwrite($fh,$course->shortname."\n");     // Nome breve del corso
fwrite($fh,$course->fullname."\n");     // Nome complete del corso
fwrite($fh,$note->name."\n");           // Nome dell'istanza del modulo
fwrite($fh,$CFG->wwwroot."\n");         // Directory principale di Moodle
fwrite($fh,$CFG->theme."\n");           // Nome del tema di Moodle in uso
fwrite($fh,$CFG->pixpath."\n");         // Directory delle immagini
fwrite($fh,$site->shortname."\n");     // Nome breve del sito
fwrite($fh,$USER->firstname."\n");     // Nome dell'utente corrente
```

⁷³ <http://it.php.net/fopen>

⁷⁴ “If the file does not exist, PHP attempts to create it. If the file exist, PHP overwrites it” (Janet Valade, *PHP 5 For Dummies*, pag. 250).

⁷⁵ <http://it.php.net/manual/en/function.fwrite.php>

⁷⁶ <http://www.zend.com/zend/spotlight/shellsriptingp1.php>

```

fwrite($fh,$USER->lastname."\n");           // Cognome dell'utente corrente
fwrite($fh,$USER->id."\n");                 // Identificativo dell'utente corrente
fwrite($fh,get_string("loggedinas","moodle")."\n"); // Stringa "loggato come"
fwrite($fh,get_string("loggedinnot","moodle")."\n"); // Stringa "non loggato"
fwrite($fh,get_string("logout","moodle")."\n"); // Stringa "Esci"
fwrite($fh,$THEME->body."\n");             // Colore del body delle pagine
fwrite($fh,$THEME->cellheading."\n");      // Intestazione delle tabelle (1)
fwrite($fh,$THEME->cellheading2."\n");     // Intestazione delle tabelle (2)
fwrite($fh,$THEME->cellcontent."\n");      // Colore delle tabelle (1)
fwrite($fh,$THEME->cellcontent2."\n");     // Colore delle tabelle (2)

```

In aggiunta a queste variabili, in coda al file ne vengono memorizzate due di tipo booleano, che indicano rispettivamente se l'utente corrente è un amministratore del sito e/o un insegnante del corso corrente. Il doppio controllo che viene effettuato prima di memorizzare queste informazioni si basa sulle funzioni "*isadmin()*" ed "*isteacher()*" di Moodle.

```

// L'utente corrente è un amministratore del sito?
if (isadmin()) {
    fwrite($fh,"TRUE"."\\n");
} else {
    fwrite($fh,"FALSE"."\\n");
}

// L'utente corrente è un insegnante del corso corrente?
if (isteacher($course->id)) {
    fwrite($fh,"TRUE"."\\n");
} else {
    fwrite($fh,"FALSE"."\\n");
}

```

Una volta ultimate le operazioni di scrittura, il file viene chiuso mediante l'uso della funzione “*fclose()*”⁷⁷

```
fclose($fh);
```

Per quanto riguarda la successiva fase di lettura, siccome le variabili memorizzate dovranno essere utilizzate all'interno di diversi files di SFO, è stato creato un apposito script (denominato “*leggi_file.php*”) contenente tutte le istruzioni per leggere il files e trasformare in variabili di PHP le informazioni contenute al suo interno.

Tale script provvede innanzitutto ad aprire, in sola lettura⁷⁸, il file “*variabili.txt*”.

```
$fh = fopen("variabili.txt","r")  
or die(get_string("errore_lettura","note"));
```

Successivamente, un ciclo while si occupa di scorrere tutto il file⁷⁹, copiando all'interno delle diverse caselle di un array ciascuna riga del testo. Per ottenere questo risultato, all'interno del ciclo viene utilizzata la funzione “*fgets()*”⁸⁰, inserita come input della funzione “*rtrim()*”⁸¹, che provvede a ripulire la stringa prelevata dai caratteri finali di formattazione o da eventuali spazi bianchi⁸².

```
while (!feof($fh)) {  
    $array_variabili[] = rtrim(fgets($fh));  
}
```

⁷⁷ <http://it.php.net/fclose>

⁷⁸ L'apertura in sola lettura è la prerogativa della modalità “r”, della funzione “*fopen()*”.

⁷⁹ Nel momento in cui un file viene aperto in modalità “read only”, il puntatore viene posizionato al suo inizio. E' così possibile creare un ciclo che scorra il file fino a quando la condizione “*feof*” (“end of file”) non risulti essere vera.

⁸⁰ <http://it.php.net/manual/en/function.fgets.php>

⁸¹ <http://it.php.net/rtrim>

⁸² Soluzione adattata da: Janet Valade, *PHP 5 For Dummies*, pag. 253.

L'array "*\$array_variabili*" viene quindi scomposto in singole variabili, con le istruzioni seguenti:

```
$file_id_corso = $array_variabili[0];
$file_nome_breve_corso = $array_variabili[1];
$file_nome_completo_corso = $array_variabili[2];
$file_nome_groupware = $array_variabili[3];
$file_moodle_root = $array_variabili[4];
$file_tema = $array_variabili[5];
$file_pix_root = $array_variabili[6];
$file_nome_breve_sito = $array_variabili[7];
$file_nome_utente = $array_variabili[8];
$file_cognome_utente = $array_variabili[9];
$file_id_utente = $array_variabili[10];
$file_loggedinas = $array_variabili[11];
$file_loggedinnot = $array_variabili[12];
$file_logout = $array_variabili[13];
$file_body = $array_variabili[14];
$file_cell_heading = $array_variabili[15];
$file_cell_heading2 = $array_variabili[16];
$file_cell_content = $array_variabili[17];
$file_cell_content2 = $array_variabili[18];
if ($array_variabili[19] == "TRUE") {
    $file_amministratore = True;
} else {
    $file_amministratore = False;
}
if ($array_variabili[20] == "TRUE") {
    $file_insegnante = True;
} else {
```

```
        $file_insegnante = False;
    }
```

Lo script termina quindi con la normale chiusura del file di testo, ottenuta per mezzo della direttiva “*fclose()*”.

Chiusa la parentesi su come funziona il sistema di lettura delle variabili memorizzate su file di testo, è possibile tornare ora ad analizzare il file “*mod/note/view.php*”. Immediatamente in coda alle istruzioni che effettuano la scrittura del file di testo, è presente il codice che gestisce l’inclusione del file “*SFO_manage.php*”. Il file “*SFO_index.php*”, che costituiva la pagina principale di SmoothFileOperator⁸³ è infatti stato eliminato e le istruzioni contenute al suo interno sono state inglobate in “*view.php*”.

```
$result = include($CFG->wwwroot.$appPath.'SFO_manage.php?action=' .
    'browse&path=rawurlencode(dirname($_SERVER['PHP_SELF']).'/').
    '&sortField=name&sortDir=asc&nome_breve_corso=' .
    $course->shortname.'&primo_caricamento=true');
```

Sulla riuscita dell’operazione di inclusione viene successivamente effettuato un controllo.

```
if ($result == 0) {
    print get_string("errore_include","note");
}
```

⁸³ Ma che, di fatto, si limitava ad impostare il valore di un paio di variabili ed a richiamare il file “*SFO_manage.php*” con i parametri necessari per effettuare il browsing della cartella principale specificata all’interno del file di configurazione di SFO.

In questo caso non è necessario aggiungere un'istruzione di tipo “*die()*”, poiché, nel caso in cui l'inclusione non dovesse andare a buon fine, lo script terminerebbe comunque la sua esecuzione non essendovi altre operazioni da eseguire⁸⁴.

4.10. Utilizzo del modulo

L'intero modulo ruota attorno al file “*SFO_manage.php*”, che gestisce tutte le diverse funzionalità a disposizione dell'utente. Lo script riceve, mediante URL, la variabile “*\$action*”, che varia in funzione del tipo di operazione sul modulo che l'utente intende effettuare.

La struttura del file, basata sul controllo “*switch(\$action)*”, è la seguente:

```
switch ($action) {
    case 'disp_upload':
        // Codice per l'upload di un file
    case 'disp_makeDir':
        // Codice per la creazione di una nuova directory
    case 'edit_file':
        // Codice per la creazione o la modifica di un file di testo
    case 'disp_details':
        // Codice per la visualizzazione dei dettagli di un
        // file o di una directory
    case 'edit_item_descr':
        // Codice per la modifica della descrizione di un
        // file o di una directory
    case 'disp_rename':
        // Codice per la ridenominazione di un file o di
```

⁸⁴ Eccezion fatta per la stampa del footer della pagina.

```

        // una directory
    case 'disp_delete':
        // Codice per la cancellazione di un file o di
        // una directory
    case 'disp_copy':
        // Codice per la copia di un file o di una directory
    case 'disp_move':
        // Codice per lo spostamento di un file o di una directory
    }

```

Le operazioni⁸⁵ che possono essere effettuate da tutti gli utenti sono le seguenti:

- upload di un file;
- creazione di una nuova directory;
- creazione di un nuovo file di testo;
- visualizzazione dei dettagli relativi ad un/a file/directory selezionato/a;
- modifica della descrizione relativa ad un/a file/directory selezionato/a;

Altre operazioni, appannaggio dei soli amministratori del sito e dei docenti del corso corrente, sono invece:

- ridenominazione di un/a file/directory selezionato/a;
- cancellazione di un/a file/directory selezionato/a;
- copia di un/a file/directory selezionato/a;
- spostamento di un/a file/directory selezionato/a;
- modifica di un file di testo selezionato.

⁸⁵ Le prime tre operazioni dell'elenco (upload di un file, creazione di una nuova directory e creazione di un nuovo file di testo) sono richiamate mediante collegamenti presenti nella parte superiore della pagina. Tutte le altre, comprese quelle riservate ad insegnanti ed amministratori, vengono selezionate da una listbox.



Figura 9 – L’interfaccia principale di SmoothFileOperator, integrato all’interno di Moodle

Affinché queste ultime operazioni fossero visualizzabili soltanto dagli utenti in possesso delle credenziali adeguate, sono state effettuate alcune modifiche all’interno del file “*SFO_BrowseDirForm.php*”. In particolare, la porzione di codice contenuta nella funzione “*_formSelectItem()*”, che provvede a stampare gli elementi “amministrativi” della listbox in questione, è stata condizionata al superamento di un controllo “if”.

```

if (($file_amministratore) || ($file_insegnante)) {
    $output .= '<option value="disp_rename">.RENAME.</option>'. "\n";
    $output .= '<option value="disp_delete">.DELETE.</option>'. "\n";
    $output .= '<option value="disp_copy">.COPY.</option>'. "\n";
    $output .= '<option value="disp_move">.MOVE.</option>'. "\n";
    $output .= '<option value="edit_file">.EDIT.</option>'. "\n";
}

```

4.11. Upload di un file

Nel momento in cui l'utente corrente tenta di effettuare l'upload di un file, lo script "*SFO_manage.php*" viene richiamato mediante il passaggio della variabile "*\$action*" con valore "*disp_upload*". Questo provoca la visualizzazione del form per l'upload, il cui codice è contenuto all'interno del file "*SFO_UploadForm.php*".



Figura 10 – Il form per l'upload di un file all'interno del portale

All'interno del form sono presenti: una textbox all'interno della quale digitare il percorso ed il nome del file da caricare sul portale; il pulsante "Sfoggia", per ricercare tramite l'interfaccia grafica del sistema operativo in uso (in alternativa alla digitazione manuale dei dati) il file destinato all'upload e la checkbox per effettuare la sovrascrittura, nel caso un file avente lo stesso sia già presente all'interno della cartella di destinazione.

La dimensione massima consentita, per quanto riguarda il file di cui effettuare l'upload, viene visualizzata sullo schermo, dopo essere stata prelevata dal file di configurazione di PHP⁸⁶ mediante la funzione “*ini_get()*”⁸⁷.

```
ini_get('upload_max_filesize')
```

4.12. Creazione di una nuova directory

L'operazione di creazione di una nuova directory prevede che lo script “*SFO_manage.php*” venga richiamato mediante il passaggio della variabile “*\$action*” con valore “*disp_makeDir*”. Questo provoca la visualizzazione del form per la creazione della nuova directory, il cui codice è contenuto all'interno del file “*SFO_MakeDirForm.php*”.

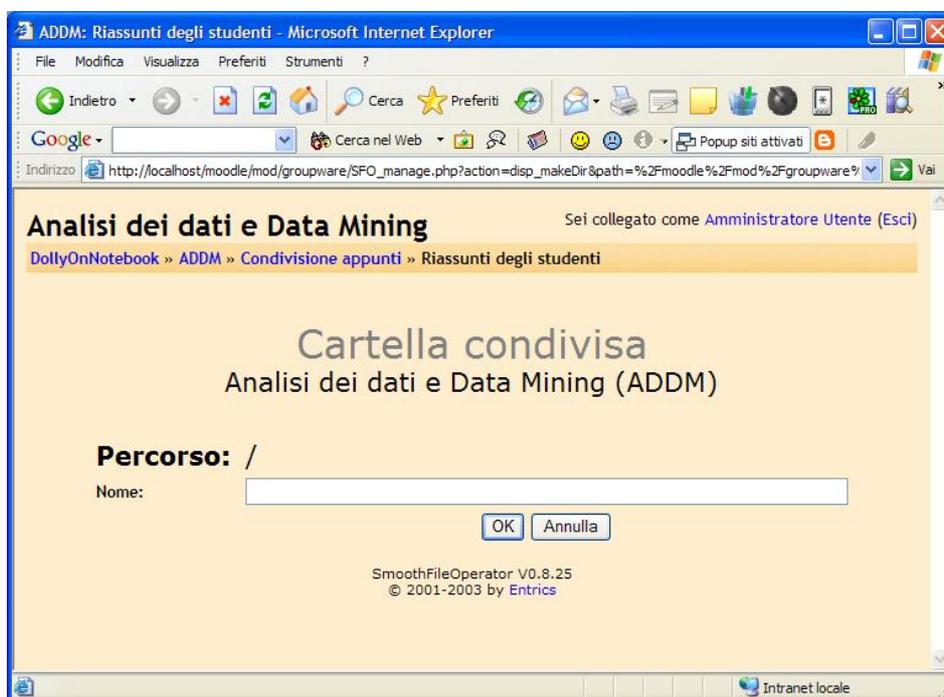


Figura 11 – Il form per la creazione di una nuova directory

⁸⁶ Si tratta dell'istruzione “*upload_max_filesize*” contenuta nella sezione “*File Uploads*” del file “*php.ini*”.

⁸⁷ http://it.php.net/ini_get

L'interfaccia è molto simile rispetto a quella che viene presentata all'utente nel momento in cui tenta di effettuare l'upload di un file. Mancano in questo caso il pulsante per "sfogliare" tra le cartelle locali, nonché la checkbox per l'eventuale sovrascrittura di cartelle aventi lo stesso nome di quella che si sta creando⁸⁸.

4.13. Creazione di un nuovo file di testo

Per creare un nuovo file di testo, lo script "*SFO_manage.php*" viene richiamato mediante il passaggio della variabile "*\$action*" con valore "*disp_edit_file*". Ne consegue la visualizzazione del form il cui codice di creazione è contenuto all'interno del file "*SFO_EditFileForm.php*".

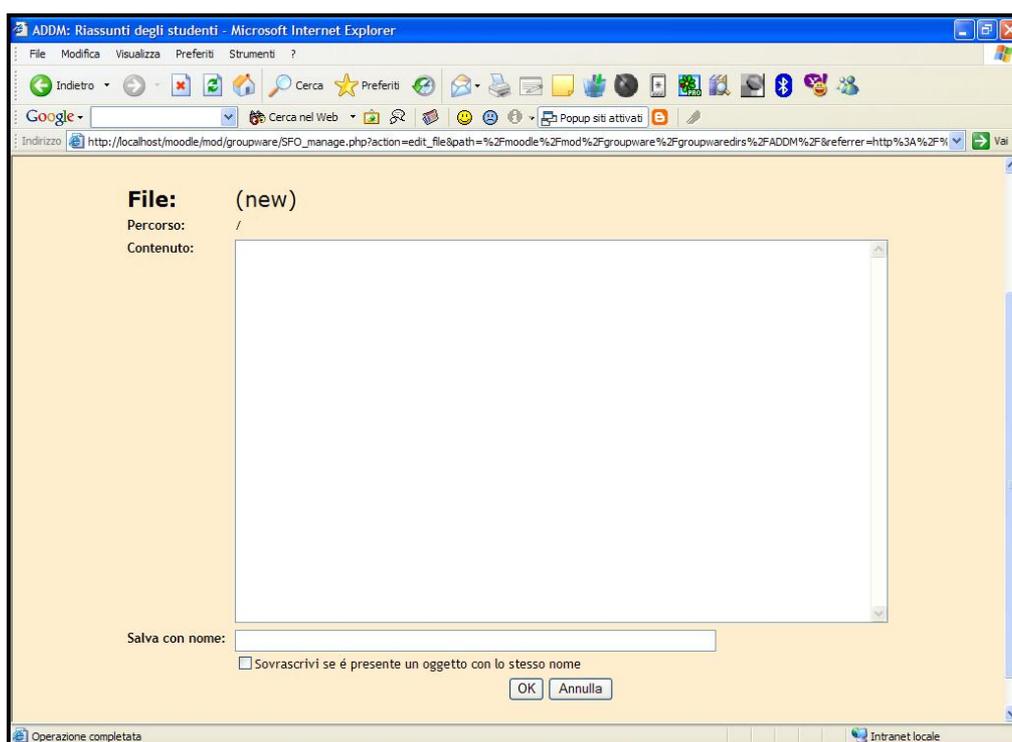


Figura 12 – Il form per la creazione di un nuovo file di testo

⁸⁸ Nel caso in cui si tenti di creare una directory avente lo stesso nome di una già esistente, all'utente viene nuovamente presentato il form, corredato da un messaggio di errore.

L'interfaccia comprende: una textarea di ampie dimensioni (24 righe per 80 colonne⁸⁹) nella quale scrivere le informazioni da memorizzare all'interno del file; una textbox, al cui interno dovrà essere impostato il nome da dare al nuovo documento, ed una checkbox per sovrascrivere un eventuale file, avente lo stesso nome di quello in fase di creazione, contenuto nella cartella di destinazione.

4.14. Visualizzazione dei dettagli di un file o di una directory

La visualizzazione dei dettagli relativi ad un file o ad una directory prevede che lo script “*SFO_manage.php*” venga richiamato mediante il passaggio della variabile “*\$action*” con valore “*disp_details*”. Ne consegue la visualizzazione di una schermata, il cui codice per la creazione è contenuto all'interno dello script “*SFO_ItemDetail.php*”.

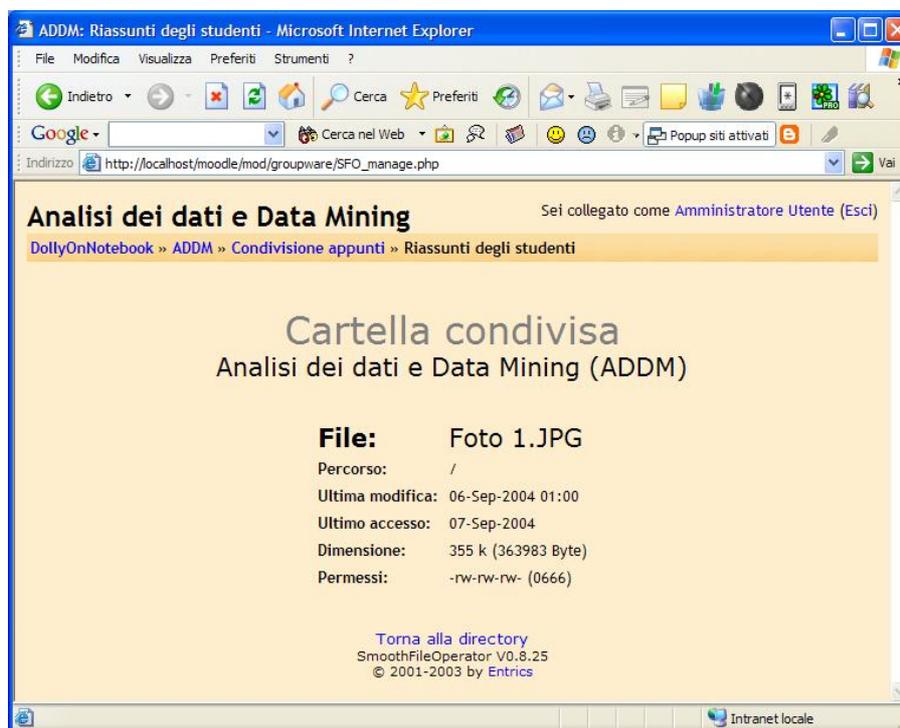


Figura 13 – La schermata di visualizzazione dei dettagli relativi ad un file di esempio

⁸⁹ Questa è la dimensione della textarea visualizzata a video. E' comunque possibile aggiungere un numero qualsiasi di righe e sfruttare la scrollbar verticale per la navigazione all'interno del box.

In questa schermata vengono riepilogate le seguenti informazioni:

- nome del file o della directory;
- percorso relativo⁹⁰;
- ultima modifica effettuata;
- ultimo accesso effettuato;
- dimensione (solo nel caso in cui l'oggetto visualizzato sia un file);
- permessi (nel formato “*UGO*” dei sistemi Unix).

4.15. Modifica della descrizione di un file o di una directory

La modifica della descrizione di un file o di una directory ha luogo tramite il richiamo dello script “*SFO_manage.php*”, con il passaggio della variabile “*\$action*” il cui valore è impostato su “*edit_itemDescr*”. Viene così generato un form, il cui codice è contenuto all'interno del file “*SFO_ItemDescForm.php*”.

L'interfaccia è del tutto simile a quella visualizzata per la creazione di una nuova directory. Cambia naturalmente il comportamento del pulsante di invio, che provvede a modificare la descrizione del file o della directory selezionati, con il nuovo valore impostato all'interno della textbox.

4.16. Ridenominazione di un file o di una directory

L'operazione di ridenominazione di un file o di una directory avviene mediante la chiamata allo script “*SFO_manage.php*”, con il passaggio della variabile “*\$action*” il

⁹⁰ Per motivi di sicurezza (ovvero per non mostrare agli utenti l'esatta collocazione dell'oggetto in esame nel file system del server), è stata aggiunta una porzione di codice che nasconde la visualizzazione della parte iniziale del percorso (http://moodle_root/mod/note/notedirs/nome_corso). Tale soluzione è stata adottata anche in tutti gli altri script di SFO integrati in Moodle.

cui valore è impostato su “*disp_rename*”. Viene così generato un form, il cui codice è contenuto all’interno del file “*SFO_RenameForm.php*”.

Ancora una volta l’interfaccia è apparentemente uguale a quella che si presenta nel momento in cui si avvia il procedimento di creazione di una nuova directory o quello di modifica della descrizione di un file o di una cartella.

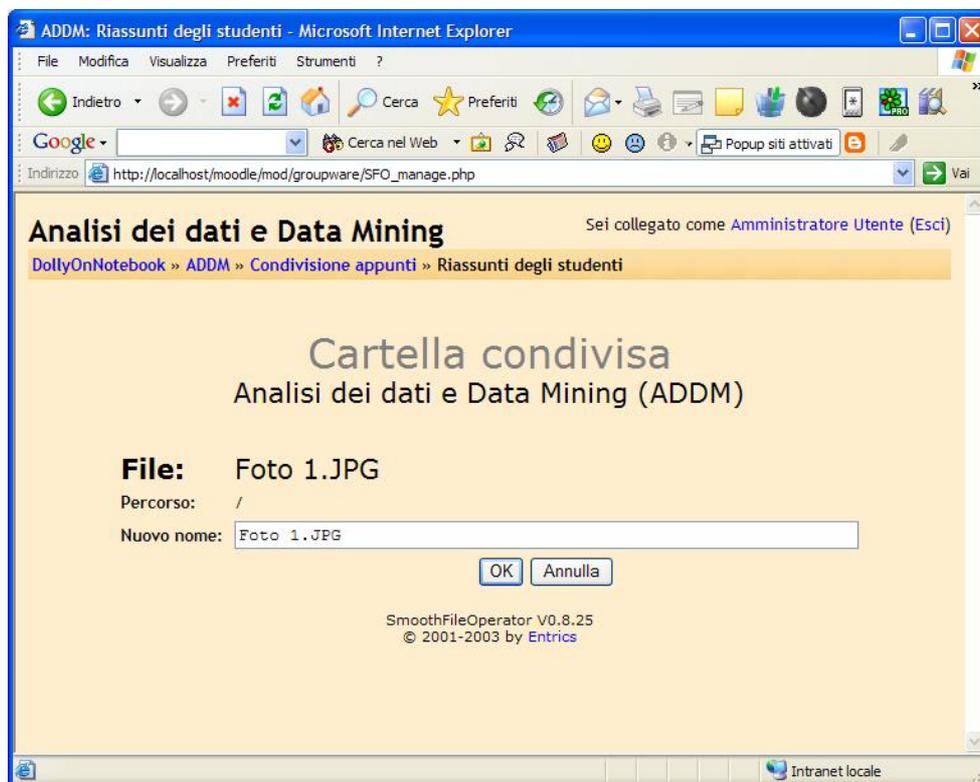


Figura 14 – Il form per la ridenominazione di un file

4.17. Cancellazione di un file o di una directory

L’operazione di cancellazione di un file o di una directory prevede che lo script “*SFO_manage.php*” venga richiamato mediante il passaggio della variabile “*\$action*” con valore “*disp_delete*”.

In seguito a questo evento viene visualizzata una schermata (il codice per la sua creazione è contenuto all'interno del file “*SFO_DeleteForm.php*”) dove si richiede all'utente di dare conferma dell'operazione in corso. La pressione del pulsante “OK” ed il conseguente invio del form danno il la alle operazioni di rimozione “fisica” del file o della cartella selezionati.

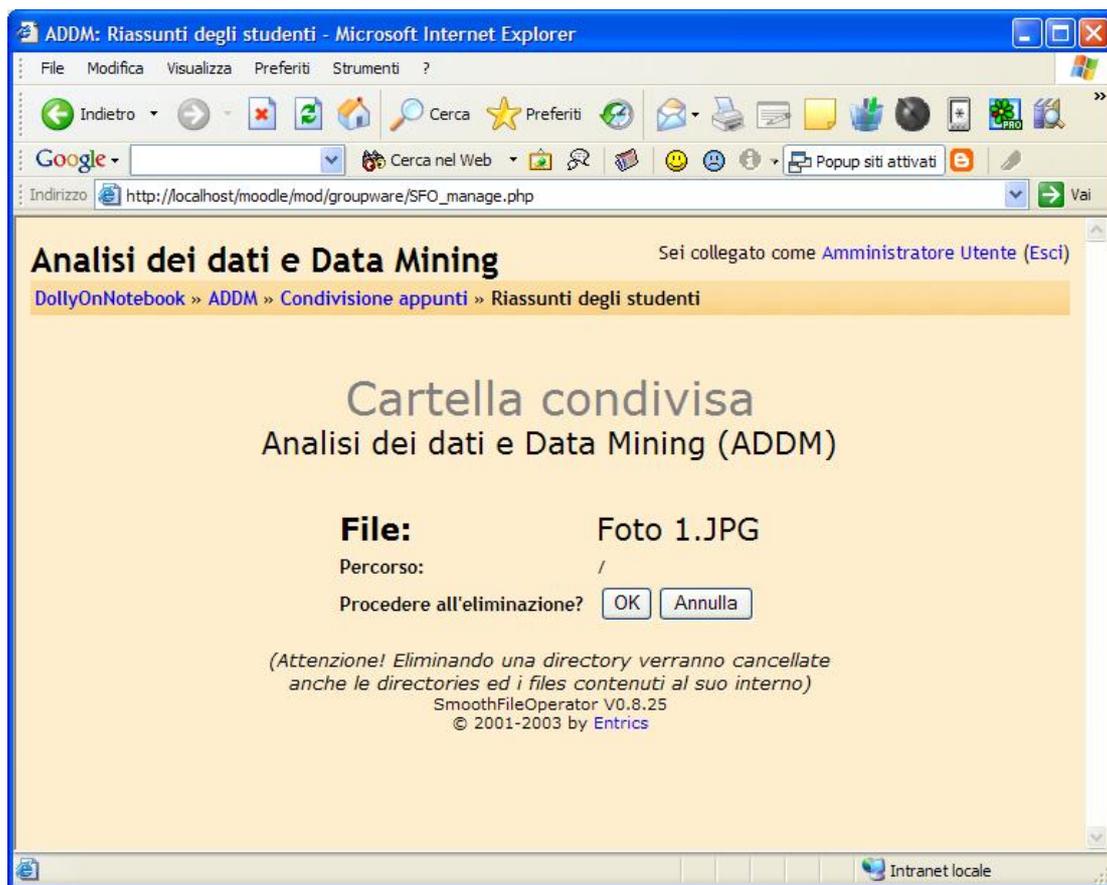


Figura 15 – Il form dove viene chiesta la conferma per la cancellazione di un file

4.18. Copia e spostamento di un file o di una directory

I procedimenti di copia e di spostamento di un file o di una directory, pur essendo decisamente simili tra di loro, differiscono molto rispetto a quelli relativi alle altre

operazioni viste finora. In entrambi i casi, nel momento in cui viene selezionata l'operazione da effettuare, ad essere richiamato è il solito file “*SFO_manage.php*”, il cui parametro “*\$action*” viene passato rispettivamente con i valori “*disp_copy*” (copia) o “*disp_move*” (spostamento).

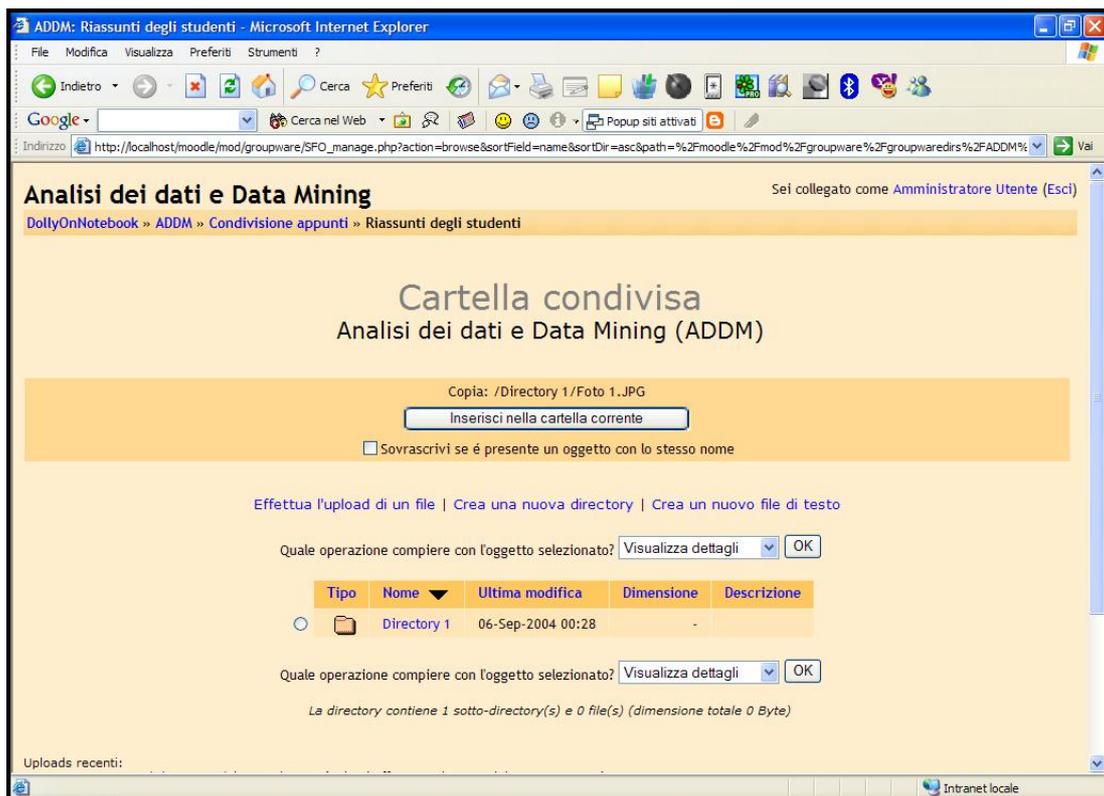


Figura 16 – La schermata che si presenta all'utente dopo aver selezionato un file da copiare

In entrambi i casi, lo script provvede a creare alcune variabili di sessione, attraverso le quali memorizza le informazioni relative al file (o alla directory) che si è in procinto di copiare o di spostare.

```
$_SESSION['clipbAction'] = 'copy'; // oppure 'move' a seconda dell'operazione
$_SESSION['clipbItemName'] = $params['itemName'];
$_SESSION['clipbPath'] = $path;
```

Una volta memorizzate queste variabili, mentre nella parte inferiore della pagina rimane visualizzata l'interfaccia di navigazione di SmoothFileOperator, nella parte superiore viene inserito un form. Il pulsante di invio di questo form permette di inserire all'interno della cartella corrente l'oggetto precedentemente selezionato.

L'unica differenza tra le due interfacce consiste esclusivamente nella diversa stringa di testo visualizzata nella parte alta della pagina ("Copia" nel primo caso, "Muovi" nel secondo).

4.19. Modifica di un file

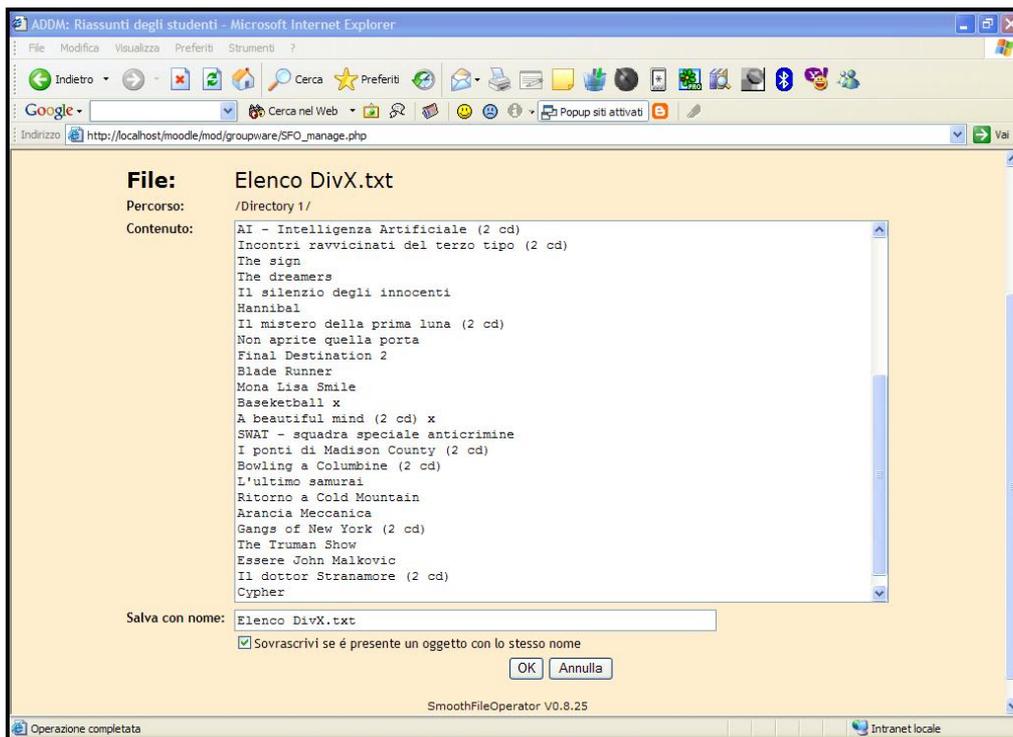


Figura 17 – Il form per la modifica di un file

L'ultima operazione che è possibile effettuare all'interno del modulo "Students' Notes" è quella relativa alla modifica di un file già presente all'interno della cartella condivisa.

Anche in questa circostanza ad occuparsi dell'operazione è lo script “*SFO_manage.php*”, che riceve in input il parametro “*\$action*” con valore “*edit_file*”. Questo provoca la visualizzazione del form, il cui codice per la creazione è contenuto all'interno del file “*SFO_EditFileForm.php*”, che permette di modificare il file selezionato.

L'interfaccia è la stessa che viene visualizzata nel momento in cui si sceglie di creare un nuovo file di testo. L'unica differenza è ovviamente il fatto che tutti gli elementi testuali della pagina (sia la textarea contenente il testo, sia la textbox sottostante) sono pre-compilati con i dati provenienti dal file in esame. Anche la checkbox per la sovrascrittura del file appare automaticamente spuntata.

4.20. Aspetti grafici

Come avrà sicuramente notato chi ha prestato attenzione agli screenshots pubblicati nel paragrafo precedente, l'aspetto grafico di SmoothFileOperator è stato drasticamente modificato rispetto alla sua implementazione standard.

Le modifiche, alcune delle quali hanno implicato leggeri ritocchi al codice sorgente, si sono concentrate in particolare su:

- l'eliminazione della listbox da cui scegliere la lingua del modulo⁹¹;
- l'eliminazione del form che permette di effettuare ricerche tra i files condivisi;
- l'eliminazione del pulsante che permette di modificare la descrizione della directory corrente;
- l'eliminazione del link per la navigazione “tradizionale” all'interno della cartella⁹²;

⁹¹ Tale scelta avviene ora in modo del tutto trasparente all'utente: il linguaggio corrente viene infatti prelevato direttamente dalla configurazione di Moodle.

- l'eliminazione del link che mostra una sorta di “mappa” delle cartelle condivise;
- l'eliminazione del campo in cui viene mostrata la descrizione della directory corrente;
- la modifica dell'intestazione della pagina;
- un “restyling” della tabella contenente l'elenco degli oggetti condivisi
- lo spostamento, al di sotto di detta tabella, dei logs;

A livello di codice, la quasi totalità delle modifiche sono state effettuate all'interno dei file “*SFO.css.php*” e “*SFO_BrowseDirForm.php*”, sfruttando (per quanto riguarda i colori di sfondo, delle celle, ecc...) le informazioni memorizzate all'interno del file temporaneo “*variabili.txt*”.

⁹² Dove, per “navigazione tradizionale” si intende la navigazione mediante la tradizionale interfaccia fornita dal browser in uso.

5. CONCLUSIONI

Su tutti gli ambienti in cui è stato testato, il modulo “Students’ Notes” ha evidenziato un funzionamento sostanzialmente corretto, pur non mancando di evidenziare alcune lacune (“*bug*” in gergo informatico) in particolari circostanze di utilizzo.

Il modulo in questione, stando alle intenzioni dei responsabili dell’Ateneo, nel corso dell’anno accademico 2004/05 diventerà parte integrante del portale “Dolly⁹³” del Corso di Laurea On Line in Comunicazione e Marketing⁹⁴, erogato dalla Facoltà di Scienze della Comunicazione e dell’Economia⁹⁵ dell’Università degli Studi di Modena e Reggio Emilia⁹⁶.

Sarà un’opportunità molto interessante non solo per gli studenti (i quali avranno a disposizione un portale più ricco di funzionalità rispetto a prima), ma anche per il sottoscritto. Se non posso essere certo del fatto che il nuovo modulo riscontrerà i favori degli utenti (pur coltivando la speranza che ciò accada), sicuramente si tratterà di un’esperienza oltremodo formativa se vissuta all’interno del BackOffice informatico.

Lavorare su di un server web che deve gestire migliaia di accessi al giorno, pone il programmatore di applicazioni Internet dinanzi a problematiche ben diverse rispetto a quelle che si è soliti affrontare in ambiente domestico o SOHO⁹⁷. L’ottimizzazione del codice, intesa sia come capacità di scrivere il minor numero di righe di codice possibile, sia come abilità di ridurre al minimo l’utilizzo della CPU del server, diventa un aspetto cruciale. Per non parlare dei problemi inerenti la sicurezza. In un ambiente quale è

⁹³ “Dolly”, acronimo di “Didattica On Line”, è l’installazione di Moodle dedicata agli studenti che frequentano i corsi erogati dalla Facoltà di Scienze della Comunicazione e dell’Economia dell’Università degli Studi di Modena e Reggio Emilia (<http://dolly.ceionline.unimo.it>).

⁹⁴ <http://www.ceionline.unimore.it/>

⁹⁵ <http://www.cei.unimore.it>

⁹⁶ <http://www.unimore.it>

⁹⁷ Acronimo di “Small Office – Home Office”: con questo termine ci si riferisce in sostanza a tutti quegli ambienti (domestici o aziendali) di piccole o medie dimensioni.

Moodle, anche un minimo errore di programmazione all'interno di un suo modulo potrebbe pregiudicare l'integrità dell'intero sistema. Ragione per cui occorre prestare estrema attenzione all'integrazione di "Students' Notes" all'interno di Dolly.

A livello di codice, il modulo è ancora aperto a notevoli miglioramenti. A causa del poco tempo a disposizione per lo sviluppo, diversi problemi manifestatisi durante la fase di "coding" sono stati risolti "alla meno peggio", focalizzandosi machiavellicamente sul fine (il funzionamento degli script), piuttosto che sul mezzo (la bontà del codice).

Ritengo che vi siano in particolare tre macro-aree di problemi alle quali assegnare la priorità nella scaletta degli interventi futuri:

- l'interfaccia "amministrativa" del modulo. Come analizzato nel capitolo 3, rispetto agli studenti, gli amministratori del portale ed i docenti del corso in cui è presente un'istanza di "Students' Notes" hanno la possibilità di svolgere alcune operazioni aggiuntive all'interno della cartella condivisa. In realtà non vi è un vero e proprio controllo su quali siano i privilegi dell'utente che sta cercando di eseguire una certa operazione: lo script si limita infatti a fornire a docenti ed amministratori l'interfaccia grafica per effettuare le operazioni a loro riservate. In questo modo, però, qualunque utente potrebbe replicare le medesime operazioni, anche senza avere accesso all'interfaccia grafica, semplicemente modificando in maniera appropriata l'URL verso cui far puntare il browser;
- la compatibilità tra diverse piattaforme e DBMS⁹⁸. Il modulo, fino a questo momento, è stato testato in maniera abbastanza approfondita soltanto su macchine con sistemi operativi della famiglia "Microsoft Windows⁹⁹", con web server Apache e DBMS MySQL. Occorre verificarne attentamente il corretto

⁹⁸ Per una definizione di "DBMS", si veda: http://www.html.it/sql/sql_02.htm

⁹⁹ Sono state usate le versioni: 98, 2000 Professional, XP Home ed XP Professional.

funzionamento su sistemi Unix/Linux¹⁰⁰, con web server e DMBS alternativi (ad esempio IIS¹⁰¹ e PostgreSQL¹⁰²)

- la mancata aderenza alla Coding Guide di Moodle, da parte di molte porzioni di codice. Sebbene non si tratti di un problema di carattere funzionale, adeguare il codice agli standard (piuttosto rigidi) di Moodle si rende necessario per condividere con l'intera comunità dei “*moodlers*” il lavoro svolto.

Queste lacune, e le prime due in particolare, dovranno essere necessariamente colmate prima di procedere al definitivo inserimento del modulo all'interno di “Dolly”. Si tratta di questioni abbastanza spinose, che richiederanno sicuramente molte ore di lavoro per essere risolte nel migliore dei modi. Lavoro che sarà presumibilmente svolto dal sottoscritto nelle settimane a venire.

Ma il codice sorgente è accessibile a tutti. Chiunque può studiarlo, scovare dei bug al suo interno, riutilizzarlo in ambiti diversi e compiere altre mille operazioni diverse. Chiunque può proporre delle modifiche, dei miglioramenti, delle correzioni al software originale. Chiunque può decidere di implementare in prima persona queste modifiche e renderle patrimonio della comunità Internet.

Benvenuti, o meglio, bentornati¹⁰³ nel mondo dell'Open Source.

¹⁰⁰ Al momento sono stati condotti soltanto alcuni test di funzionamento utilizzando Linux Slackware 10.0. I risultati sono stati positivi, ma è necessario approfondire lo studio, utilizzando anche altre distribuzioni.

¹⁰¹ <http://www.microsoft.com/windowsserver2003/iis/default.msp>

¹⁰² <http://www.postgresql.org/>

¹⁰³ <http://www.opensource.org/docs/history.php>

http://eu.conecta.it/paper/brief_history_open_source.html

6. RINGRAZIAMENTI

I miei più sentiti ringraziamenti vanno a tutti coloro che, in qualunque modo ed a qualsiasi titolo, hanno collaborato alla programmazione del modulo “Students’ Notes” ed alla stesura di questa tesi.

Tra tutti questi, meritano una citazione particolare:

- il professor *Tommaso Minerva*, per avermi affidato questo lavoro e per essersi sempre dimostrato ottimista riguardo alle mie capacità di portarlo a termine nel migliore dei modi;
- *Martin Dougiamas*, ideatore ed attuale leader del progetto Moodle, e tutto il team di sviluppo, le cui discussioni sui vari forum sono state per me un’impareggiabile fonte di apprendimento;
- *Olaf Woldrich*, creatore di SmoothFileOperator, per avere progettato questo ottimo script e per avermi dato, via e-mail, tutta la collaborazione necessaria alla sua piena comprensione;
- *disasta, nathan e char8n*, assidui frequentatori del canale IRC “#PHP”¹⁰⁴, per aver saputo risolvere la maggior parte dei miei dubbi relativi alla programmazione in PHP. Ringrazio in particolare *char8n*, giovanissimo programmatore polacco, per avere passato con me un lungo pomeriggio nel tentativo di comprendere il comportamento (apparentemente assurdo) dell’interprete PHP, di fronte ad alcune condizioni di inclusione files piuttosto particolari;
- *Setec Astronomy*, autentico leader del newsgroup: “it.comp.www.php”, per avermi definitivamente chiarito le idee riguardo alla famigerata funzione “include()” di PHP;

¹⁰⁴ Raggiungibile, tra gli altri, attraverso il server IRC di Tin: “irc.tin.it”.

- *Merkur, John, Ornis, Bluecow* e tutti gli altri sviluppatori che hanno dato origine al progetto eMule¹⁰⁵, per avermi dotato di uno strumento che mi ha consentito di reperire tutta la manualistica e la strumentazione software necessarie alla realizzazione di questa tesi;
- *Emmanuel Faivre, Laurent Abbal, Thierry Murail* e tutto il team di sviluppo di “EasyPHP”¹⁰⁶, per avermi messo a disposizione un ambiente di sviluppo stabile, performante e, soprattutto, installabile e configurabile in soli pochi minuti;
- *Giordano Lanzi*, studente di ingegneria informatica, per avermi dato supporto durante l’installazione e la successiva configurazione, sul mio notebook, del sistema operativo Linux Slackware 10.0¹⁰⁷, necessario per testare la compatibilità del modulo “Students’ Notes” con piattaforme non-Windows.

¹⁰⁵ <http://www.emule-project.net/>

¹⁰⁶ <http://www.easyphp.org/>

¹⁰⁷ <http://www.slackware.org>

7. BIBLIOGRAFIA¹⁰⁸

- Valade J. (2004), *PHP 5 For Dummies*, Wiley Publishing Inc., Indianapolis;
- Novak J. D. (2003), *Apprendimento significativo e Mappe Concettuali alla base di un Nuovo Modello Educativo*,
<http://81.113.59.234/novakrimini/novakrimini.htm>;
- Fraccavento S. (2003), *Teorie, modelli e sviluppi del mercato, a livello nazionale e internazionale, riguardanti i processi di e-learning*,
<http://www.studiotaf.it/teoriemodellifad1.htm>;
- Farace P. (2003), *Strategie nell'e-learning: l'impatto del modello open source nelle scelte tecnologiche e funzionali*,
http://www.farace.it/files/tesi701482_.doc;
- Branca G. (2002), *PHP*, The McGraw-Hill Companies (Publishing Group Italia), Milano;
- Greenspan J. e Bulger B. (2001), *MySQL/PHP Database Applications*, M&T Books, Foster City;
- Welling L. e Thomson L. (2001), *PHP and MySQL Web Development*, Sams Publishing, Indianapolis;
- Maslakowski M. (2000), *Sam's Teach Yourself MySQL in 21 Days*, Sams Publishing, Indianapolis;
- Shapiro C., Varian H. R. (1999), *Information Rules*, Harvard Business School Press, Boston;
- Atzeni P., Ceri S., Paraboschi S. e Torlone R. (1999), *Basi di dati* (seconda edizione), McGraw-Hill Libri Italia, Milano.

¹⁰⁸ http://www.unisi.it/grotti/come_scrivere_una_bibliografia.htm