

Test online di matematica: il Progetto M.In.E.R.Va*

Grazia Messineo, Salvatore Vassallo

Sommario

Vengono presentati i pacchetti **esami-online**, per la creazione di prove d'esame con correzione automatica da somministrare al pc, e **minerva**, per la creazione di percorsi di recupero delle carenze in matematica per studenti delle scuole superiori.

Abstract

We present the packages **esami-online**, to create online exams with automatic corrections, and **minerva**, to create sets of exercises for high school students.

1 L'esame online

1.1 Introduzione

All'Università Cattolica, il progetto di somministrare una parte dell'esame di Matematica Generale in via automatica era nato già negli anni '80 da un'idea del professor Giovanni Melzi ed era stato sviluppato dalla professoressa Monica Bianchi (BIANCHI e MELZI, 1990). I mezzi, estremamente limitati sia per quanto riguarda l'hardware che il software a disposizione, sia per il numero di persone coinvolte nel progetto, non ne hanno permesso uno sviluppo completo.

Nell'A.A. 2008/09 il progetto è stato ripreso e si è pensato di modificare opportunamente il software sviluppato per la preparazione dei temi d'esame, in modo da ottenere test da somministrare online.

1.2 Le altre esperienze

Innanzitutto sono state analizzate le soluzioni adottate in altri contesti, sia utilizzando piattaforme specifiche, sia utilizzando $\text{\LaTeX}$ all'interno di pagine web create dinamicamente con linguaggi specifici (ASP, PHP, ecc.).

Le esperienze più significative e facilmente trasferibili sono sembrate:

- i pacchetti $\text{\LaTeX}$ **exerquiz** e **eqexam** del prof. Story (Università di Akron, Ohio) (STORY, 2012), su cui si basano anche i software presentati di seguito e quello da noi elaborato.
- Il software **MacQTeX** creato alla Macquarie University di Sidney, che attraverso l'uso di

una versione modificata del pacchetto **exerquiz**, di alcuni script in Perl e di un database MySQL, permette la creazione "al volo", tramite un'interfaccia web, di set di esercizi (scelti dal docente). Questo software viene utilizzato per attività di recupero autonomo o di tutorato, e anche per svolgere esercitazioni valutate, che contribuiscono al punteggio finale dell'esame. La piattaforma web consente due tipi di accesso: uno per lo studente, che sceglie quali gruppi di esercizi svolgere tra quelli proposti, e uno per il docente, che crea i set di esercizi da somministrare agli studenti ed esamina le statistiche e le valutazioni ad essi relative (GRIFFIN).

- Il software **ExOmatiK**, sviluppato dal prof. Nicolas Poulain del Collège Mallarmé di Parigi per la scuola superiore, ma adattabile anche per l'università. Questo software permette la creazione al volo di fogli di esercizi, scelti da un database e la cui anteprima è presentata online. Gli esercizi possono contenere anche elementi aleatori. Viene fornito, per ogni set di esercizi creato, anche il file sorgente $\text{\LaTeX}$ (POULAIN).
- Il software **AcroWeb** del professor Robert Mařík dell'Università Mendel di Brno, che usa una versione personalizzata del pacchetto **exerquiz**, alcuni altri pacchetti $\text{\LaTeX}$ e alcuni script in Perl, per consentire la creazione "al volo" di set di esercizi da svolgere online, e Cards di esercizi presentati in modo casuale, e altri strumenti didattici interattivi (MAŘÍK).
- Il software **MCQ-Xe $\text{\LaTeX}$** del prof. Davide Ferrario dell'Università Bicocca di Milano, che usa alcuni pacchetti $\text{\LaTeX}$ e alcuni script in Python per generare set di esercizi che possono essere fruiti in tre modi: a stampa (viene generato un file PDF che può essere stampato, completato dallo studente e letto da un lettore ottico), sul web (viene generato un output in HTML che usa MathJax per la parte matematica) e su una piattaforma dedicata (viene generato un output compatibile con la piattaforma e-learning gratuita Moodle in uso presso l'Università Bicocca) (FERRARIO).

Tutte le soluzioni esaminate sono state scartate a causa di vincoli tecnici che impedivano l'installazione sul server messi a disposizione del software necessario per la compilazione.

*Il lavoro si inserisce nel progetto di ricerca di interesse d'Ateneo dell'Università Cattolica "Progetto M.In.E.R.Va.: Creazione di esercizi di tipo interattivo con percorsi differenziati per il recupero delle carenze e la valorizzazione delle eccellenze in matematica".

Inoltre, come già è avvenuto per la gestione dei temi d'esame tradizionali, si è preferito creare un software basato esclusivamente su $\text{\LaTeX}$ sia per la formattazione del testo, sia, per quanto possibile, per la gestione della parte matematica e degli elementi aleatori presenti negli esercizi.

La scelta è stata quindi quella di adattare il pacchetto già creato per la versione standard degli esami (si veda (MESSINEO e VASSALLO, 2010)) affinché fosse possibile creare quiz (a risposta chiusa) da somministrare agli studenti in modalità online, utilizzando file PDF interattivi.

1.3 L'interfaccia

I quiz che costituiscono la prova d'esame sono presentati allo studente sotto forma di un file PDF reso interattivo con l'uso di codice JavaScript (ADOBE SOLUTIONS NETWORK, 2001).

Lo studente è identificato mediante il numero di matricola: se esiste sul server un file che ha come prima parte del nome quel numero, esso viene proposto allo studente. In tal caso, i campi cognome, nome e numero di matricola sono precompilati (fig. 2).¹

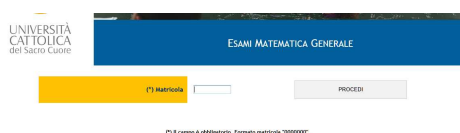


FIGURA 1: La maschera iniziale

Matematica Generale – **Test preliminare** – 29 ottobre 2011

Nome:

Cognome:

Matricola:

SCRIVERE NOME E COGNOME E CLICCARE SU "INIZIO QUIZ"

Inizio Quiz

14:59

FIGURA 2: La maschera di inserimento dati (normalmente già compilata)

Cliccando sull'apposito bottone “Inizio Quiz” lo studente può iniziare la prova. Nell'angolo in basso a sinistra dello schermo appare un segnatempo con avvisi visivi (rettangoli rossi o gialli) che cambiano colore in base al tempo rimanente. A tempo scaduto, lo studente viene avvisato mediante una finestra pop-up. Se, nonostante ciò, invia i dati al server oltre il tempo indicato (più un eventuale bonus) la prova viene annullata; sul server vengono però ugualmente memorizzati i dati.

1. Il sistema permette anche l'utilizzo di matricole "fittizie", cioè non corrispondenti ad alcuno studente per gestire casi eccezionali (studenti privi di matricola, iscrizioni tardive all'esame, ecc.). In tale caso lo studente riempirà i campi corrispondenti ai propri dati.

Lo studente può muoversi all'interno del file utilizzando gli appositi comandi (freccie) poste in basso (fig. 3).

Ogni domanda appare su una sola schermata e prevede una sola risposta esatta. È possibile cambiare una risposta già data, mentre non è possibile annullarla.²

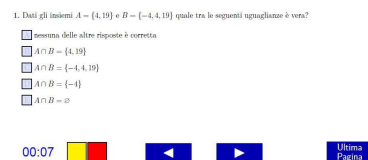


FIGURA 3: Una domanda, i pulsanti di controllo e il contatore con gli alert giallo e rosso

A prova terminata, lo studente clicca sul bottone “Fine Quiz” (fig. 4): tale azione determina l’esecuzione di una serie di istruzioni JavaScript:

- viene valutata la correttezza o meno delle risposte date;
- vengono mostrati allo studente il punteggio ottenuto e l'esito dell'esame;
- vengono inviati al server i dati dello studente, l'orario di inizio e di fine della prova, le risposte date, il punteggio ottenuto.³

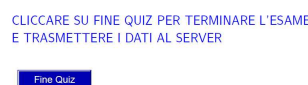


FIGURA 4: La pagina finale

Allo studente viene quindi mostrata un'altra pagina HTML che riepiloga i suoi dati e l'esito. Successivamente il sistema ritorna alla pagina iniziale o presenta allo studente un'altra parte di esame, se previsto.

Lo studente può modificare le risposte finché non ha cliccato su “Fine Quiz”: quando lo ha fatto non può più operare alcuna modifica e, se prova comunque a farlo, riceve un messaggio di errore.

1.4 La creazione dei file PDF

Le prove d'esame sono generate usando il linguaggio L^AT_EX. La scelta di L^AT_EX rispetto ad altri strumenti è dovuta ad alcuni fattori:

- a) la qualità necessaria dell'output;
- b) l'esistenza di un database di esercizi già predisposti (si veda (MESSINEO e VASSALLO, 2010));

2. La cosa non è comunque influente poiché nelle nostre modalità d'esame non sono previste penalizzazioni in caso di risposte errate. È comunque possibile prevedere un'alternativa "non risponde" per tali situazioni.

3. Tali dati vengono memorizzati in un file `.csv` e/o in un database.

- c) la possibilità di modificare secondo le nostre necessità il software, data la natura delle licenze (GPL o simili);
- d) la possibilità di operare senza installazione di software aggiuntivo sui server dell'università.

A differenza di tutte le altre esperienze a noi note di utilizzo di L^AT_EX per la creazione di quiz online (o anche offline) il software da noi creato non utilizza alcun programma esterno né per la randomizzazione degli esercizi né per la manipolazione delle espressioni matematiche (si veda (MESSINEO e VASSALLO, 2010)).

Anche questo pacchetto, che abbiamo chiamato **esami-online**, si basa su una versione altamente modificata di **exerquiz** di Donald P. Story (STORY, 2012) e prevede anche l'uso dei pacchetti **eq2db**, che consente la comunicazione di stringhe di dati dal file PDF al database sul server, e **insdljs**, che funge da interfaccia tra i comandi L^AT_EX e i comandi JavaScript.

Le principali modifiche (molte delle quali comuni con il pacchetto **esami**) riguardano:

- la completa randomizzazione nella scelta e nella presentazione degli esercizi, che si basa sul numero di matricola dello studente e sulla data dell'esame, minimizzando quindi la possibilità sia di avere versioni simili nella stessa data, sia di riproporre allo stesso studente gli stessi esercizi in date diverse.⁴

Questa scelta è diversa da quella effettuata per gli esami tradizionali (si veda (MESSINEO e VASSALLO, 2010)). Infatti, quando si somministrano le prove in forma cartacea è necessario assicurarsi che versioni contigue della stessa prova siano abbastanza diverse tra loro; quando le prove vengono somministrate online, invece, è impossibile sapere come si disporranno gli studenti. Questa differenza ci ha suggerito di scegliere per la modalità online una gestione totalmente casuale degli elementi aleatori.

- La possibilità di utilizzare esercizi in forma parametrica (come già avviene per gli esami tradizionali).
- La possibilità di utilizzare set di esercizi, scelti in modo casuale, per comporre diverse versioni della stessa prova (il pacchetto **eqexam** di D.P. Story usa esercizi "fissi" e prevedeva, quando abbiamo cominciato a lavorare su questo progetto, la possibilità di creare solo due varianti. Inoltre, gli elementi che devono variare tra una versione e l'altra devono essere modificati "a mano". Il pacchetto **exerquiz** non prevede invece varianti).

4. Il database dal quale vengono scelti gli esercizi viene integrato (aggiungendone di nuovi) solo occasionalmente.

- La modalità di lettura dei dati inviati al server (si veda la sezione 1.5).

La scelta degli esercizi da inserire in ogni prova è stata gestita in due modi:

- a) in una prima fase si è utilizzata la stessa modalità delle prove cartacee: gli esercizi vengono scelti da una persona che si preoccupa di comporre il tema d'esame, di valutare il livello di difficoltà, ecc. In questo caso, la scelta casuale riguarda i valori da attribuire ai parametri, la scelta della variante dell'esercizio, la rotazione delle domande e delle risposte.
- b) In una seconda fase (in particolare quando la modalità è stata estesa ai corsi con un maggior numero di studenti, come descritto nel paragrafo 1.6) si è deciso invece di rendere gli esercizi del database omogenei per livello di difficoltà e di lasciare quindi che la scelta degli esercizi da inserire fosse effettuata direttamente, in modo casuale, in fase di compilazione.⁵

Non potendo installare sui server software aggiuntivo, partendo dall'elenco degli iscritti alla prova abbiamo optato per creare un file PDF per ogni studente; tali file vengono caricati sul server e resi disponibili il giorno dell'esame.

Come già accennato nel paragrafo 1.3, il file viene nominato con il numero di matricola dello studente seguito da un codice identificativo della prova, così che solo gli studenti iscritti possono accedere al file.

Nel processo di compilazione vengono creati anche un file PDF della prova in forma stampabile e un file PDF con le soluzioni: questi file possono essere utili sia durante la discussione della prova con lo studente sia in casi di emergenza (ad esempio problemi sul server).

Per la creazione di questi tre file PDF è stato predisposto un file master, dal nome **master-csv.tex**, che utilizza, nella compilazione, un altro file, dal nome **master.tex**. Compilando il primo file, se ne creano altri, che vengono rinominati con il numero di matricola dello studente e il codice identificativo della prova. Il pacchetto fondamentale utilizzato in questo file è **datatool**, che permette di caricare il database (un file **.csv**) con i dati degli iscritti alla prova.⁶

Il codice usato nel file **master-csv.tex** è

5. In particolare, si è fatto uso del comando `\estraies`, già descritto nell'articolo "Il pacchetto **esami** per la creazione di prove scritte". Prossimamente, sarà sperimentata una modalità diversa, che consentirà di scegliere casualmente e in automatico gli esercizi dall'intero database, specificando solo quanti se ne vogliono per tipo.

6. Nel file **master.tex** viene caricato, ovviamente, anche il pacchetto **esami-online**, le cui opzioni fondamentali sono le stesse del pacchetto **esami** e sono state descritte nell'articolo "Il pacchetto **esami** per la creazione di prove scritte".

```

\usepackage{datatool}
...
\DTLloadddb{Iscritti}
{Elenco_studenti_xxx.csv}
%% il nome del database
\DTLforeach{Iscritti}
{\matricola=Matricola,\nome=Nome,%
\cognome=Cognome}
{\immediate\openout\dati=\matricola.dati
\immediate\write\dati
{\string\def\string\nomeesame{\nomeesame}}
\immediate\write\dati
{\string\def\string\nome{\nome}}
\immediate\write\dati
{\string\def\string\cognome{\cognome}}
\immediate\write\dati
{\string\date{\dataesame}}}

\immediate\closeout\dati

```

La seconda parte delle istruzioni serve a creare un file dal nome `matricola.dati`, nel quale sono contenute tutti i dati dello studente (nel nostro caso, cognome, nome, data della prova, nome della prova). Tali informazioni saranno poi usate nella creazione del file PDF, durante la quale verrà inserito automaticamente il nome dello studente (per evitare di dover gestire manualmente nel database i tanti errori compiuti nella digitazione).

Nello stesso file, viene poi eseguita una serie di istruzioni con il comando `\write18` che consentono di compilare, studente per studente, il file per il test cartaceo (con generazione delle eventuali figure), quello per il test online e, infine, quello con la stringa delle soluzioni. La scelta di eseguire istruzioni con il comando `\write18` comporta alcuni problemi di sicurezza e quindi non sarebbe utilizzabile se la compilazione avvenisse “al volo” su un server. È al momento l'unica soluzione che abbiamo trovato per far eseguire, in fase di compilazione, i comandi DOS necessari per creare cartelle, copiare o spostare i file, ecc.

Ad esempio, il seguente codice

```

\immediate\openout
\compilazione=compilazione.txt
\immediate\write\compilazione
{\string\def\string\ncompilazione{1}}
\immediate\closeout\compilazione

\immediate\write18
{copy \nomeesame.tex
\matricola-\modulo.tex}
\immediate\write18
{pdflatex -shell-escape
\matricola-\modulo.tex}

...

\immediate\write18
{move \matricola-\modulo.pdf

```

```

output/\matricola-\exte.pdf}
\immediate\write18
{move \matricola-\modulo.tex output}
...

```

consente di creare una copia del file master rinominata con il numero di matricola dello studente e il codice identificativo della prova, di compilarla, di spostare il file PDF e il file `.tex` rinominato in una cartella chiamata `output` (il file PDF viene anche rinominato durante lo spostamento).

Il file `compilazione.txt` che viene creato all'inizio della procedura, contiene un unico comando, `\ncompilazione`, che assume in sequenza i valori da uno a tre e consente di passare alcune opzioni al pacchetto `esami-online`, che variano a seconda del tipo di output che si vuole ottenere.

I file (sia il pacchetto `esami-online`, sia i file di configurazione) contengono anche un certo numero di istruzioni JavaScript che rendono interattivo il file PDF di output.

Ad esempio, il seguente codice

```

...
\ifthenelse{\equal{\nome}{}}
{\ifeqforpaper
\newcommand\FirstName[2]{\hrulefill}
\else
\newcommand\FirstName[2]
{\textField{IdInfo.Nome}{#1}{#2}}
\fi
\newcommand{\avvertenza}
{\noindent\textcolor{blue}
{\textsf{SCRIVERE NOME E COGNOME
E CLICCARE SU ‘‘INIZIO QUIZ’’}}}
}
{\newcommand\FirstName[2]{\nome}
\addHiddenTextField{IdInfo.Nome}{\nome}
\newcommand{\avvertenza}
{\noindent\textcolor{blue}
{\textsf{VERIFICARE I DATI INSERITI
E CLICCARE SU ‘‘INIZIO QUIZ’’}}}
}
...

```

consente di creare “spazi” diversi per il nome, in relazione al fatto che possa essere letto o no dal file `.csv` e in relazione al fatto che si stia producendo una prova interattiva o una prova cartacea: se la prova è interattiva e il nome non è acquisito dal file `.csv`, compare un campo nel quale si deve inserire il dato da tastiera, mentre se il nome è acquisito viene stampato nel campo corrispondente; se si sta producendo una prova cartacea, nel caso di nome non acquisito viene stampata una linea.

La prova somministrata allo studente può essere singola oppure divisa in più parti. In quest'ultimo caso, anche questo aspetto è gestito da comandi JavaScript, che consentono di indirizzare lo studente verso la parte successiva della prova, oppure di far-

lo uscire dalla procedura, sulla base del punteggio ottenuto.

1.5 La trasmissione dei dati e l'elaborazione lato server

L'applicazione online implementata per la gestione degli esami di Matematica Generale risiede su una partizione dedicata di un server che utilizza come web server Microsoft IIS e le pagine web sono scritte in linguaggio ASPX.

La piattaforma prevede due tipi di accessi:

- *Per i docenti:* è stata predisposta una sezione dedicata all'amministrazione dei test, a cui è possibile accedere solo come utenti autorizzati (sono previsti diversi livelli di autorizzazione). Mediante l'interfaccia è possibile compiere varie operazioni amministrative (modificare la visibilità del sistema, compiere verifiche, stabilire l'ordine in cui si devono svolgere i quiz nel caso di prova suddivisa in più parti, visualizzare, stampare ed archiviare gli elenchi degli esiti ed altro ancora).
- *Per gli studenti:* l'accesso avviene inserendo il numero di matricola e una password casuale generata dal sistema, come spiegato nel paragrafo 1.3. Il software cerca quindi se sul server sia presente un file che abbia come prima parte del nome il numero di matricola inserito. Tale file viene immediatamente rinominato per motivi di sicurezza.⁷ Lo studente svolge l'esame. Al termine, l'esito dell'esame viene inviato al sistema e salvato. Infine, viene disabilitato l'accesso dello studente, forzandone il logout.

La fase più delicata di tutto il processo è risultata essere quella finale. Infatti quando al termine della prova lo studente clicca su **Fine Quiz**, il codice JavaScript contenuto nel file PDF elabora le risposte, comunica mediante un apposito form allo studente il punteggio e l'esito e trasmette al server un file FDF (*Forms Data Format*) (ADOBE SOLUTIONS NETWORK, 1999) che contiene tutte le informazioni necessarie (dati dello studente, dati della prova, risposte date, esito, orario di inizio e fine della prova).

Nelle prime prove, i contenuti del file FDF venivano interpretati da uno script ASP e salvati sul server in un file in formato `.csv` che poteva essere scaricato dai docenti alla fine della prova. Questa soluzione aveva il vantaggio di un'enorme semplicità sia nell'implementazione che nell'utilizzo (lo script ASP è fornito insieme al pacchetto *exerquiz* e richiede una personalizzazione minima; il file `.csv` può essere scaricato e letto su qualsiasi macchina

7. Per esempio, impedire che un altro studente possa accedere da altra posizione allo stesso file per eseguire la prova al posto del candidato, o situazioni simili.

e sistema operativo e non necessita di particolari applicativi per la sua creazione e compilazione).

All'aumentare del numero di studenti coinvolti questa soluzione ha mostrato diversi problemi, in particolare nella scrittura dei dati nel caso di accessi contemporanei, poiché i dati potevano essere registrati con un certo ritardo.

Per questo motivo, nelle prove successive si è deciso di cambiare la modalità di registrazione dei dati che ora avviene in un database MSSql Server.

Il file FDF viene analizzato da uno script (attualmente in linguaggio ASPX) che provvede a scrivere i dati sia in un database che nel file `.csv` (che serve ora solo da backup). Dal database vengono generati automaticamente gli elenchi degli esiti suddivisi per gruppo e per data di svolgimento della prova ed è possibile ovviamente utilizzarlo anche per la ricerca di esiti relativi a un singolo studente, ecc.

Uno dei problemi che sembrava essere di difficile soluzione in fase di progettazione era quello della riproducibilità e conservazione della prova come svolta dallo studente per il periodo stabilito dalle norme. La prima soluzione adottata è stata conservare nel file `.csv` e nel database la stringa delle risposte data dallo studente, i suoi dati e tempi di esecuzione, unitamente al file PDF della sua prova (non compilato). Inoltre la modalità di creazione del file consente di ricrearlo identico se necessario anche a distanza di tempo. Un'altra e più efficace soluzione consiste nel memorizzare la coppia file PDF, file FDF in modo che venga salvato oltre al testo della prova anche l'insieme dei campi compilati dallo studente: quindi è possibile ricreare la prova come eseguita dallo studente importando i dati del modulo FDF o aprendolo direttamente con Acrobat Reader.

1.6 La sperimentazione

La prima somministrazione di una prova online è stata effettuata sugli studenti di un corso di Laurea in Economia dell'Università Cattolica il cui numero ridotto era ideale per testare il software.

È stata effettuata una simulazione, nel corso della quale si sono rilevate alcune criticità (layout e impaginazione non ottimali, necessità di un timer che indicasse il tempo residuo) risolte prima della sperimentazione vera e propria.

Gli esiti degli studenti nella prima parte (prova automatica) non si sono discostati significativamente da quelli degli altri gruppi di studenti dei corsi paralleli di Matematica Generale che hanno svolto, nella stessa data, una prova di pari contenuto e difficoltà.

Un questionario di gradimento (anonimo) è stato somministrato agli studenti una decina di giorni dopo la prova e ha mostrato che gli studenti coinvolti hanno apprezzato tale modalità d'esame e soprattutto la possibilità di conoscere immediatamente l'esito ottenuto.

I risultati del questionario sono i seguenti:

- il 51% degli studenti preferisce la modalità on-line rispetto a quella cartacea e il 19% è indifferente;
- il 51% degli studenti preferirebbe sostenere l'intera prova al PC e circa il 7% è indifferente;
- il 91% degli studenti ritiene molto positivo conoscere immediatamente il risultato della prima parte;
- gli aspetti tecnici della prova (interfaccia grafica, tempo a disposizione, chiarezza delle istruzioni, modalità di preparazione) sono stati giudicati positivamente in media da oltre il 70% degli studenti.

L'esito largamente positivo della sperimentazione, protratta per un intero anno accademico, sia dal punto di vista tecnologico che didattico, ha suggerito la possibilità di estendere questa modalità ad una parte dell'esame per tutti i corsi di Matematica Generale, coinvolgendo in alcuni appelli più di 1500 studenti.

1.7 Vantaggi e problemi delle prove automatizzate

La somministrazione delle prove di Matematica Generale con modalità in parte automatizzate presenta diversi vantaggi che possono essere così riassunti:

- il vantaggio principale per i docenti è quello dell'elaborazione automatica dei risultati che, considerando il numero complessivo di prove da correggere, è sicuramente considerevole;
- la votazione della prova (almeno della parte svolta al PC) è immediatamente disponibile per lo studente e, di conseguenza, i tempi di correzione della parte finale dell'elaborato (e quindi di attesa dell'esito per gli studenti) sono molto più brevi;
- la correzione automatica è priva di errori umani che sono sempre possibili valutando un numero molto elevato di prove;
- poiché la correzione automatica è oggettiva e spersonalizzata, dal punto di vista psicologico genera una minore ansia nello studente che riesce ad accettare con maggior serenità l'eventuale esito negativo, evitando inoltre le strumentali polemiche sulla maggiore o minore "bontà" di un docente rispetto ad un altro;
- il fatto che ogni studente abbia un compito diverso e che ogni studente visualizzi al computer solo un esercizio per volta diminuisce il rischio di "comunicazioni" tra studenti;

- la modalità è più facilmente accessibile anche agli studenti diversamente abili (ad esempio dislessici).

I principali problemi della somministrazione delle prove al PC sono invece:

- l'interattività di cui si dispone con i file PDF è limitata, anche se sufficiente per gli scopi attuali (una maggiore interattività potrebbe essere sviluppata usando ad esempio piattaforme di e-learning specializzate nella matematica);
- la messa a punto in anticipo di tutti i temi d'esame pone alcuni problemi di sicurezza che potrebbero essere agevolmente superati permettendo la creazione delle prove "al volo", ossia solo al momento della loro erogazione allo studente;
- la soluzione attualmente implementata non permette la possibilità di utilizzare *data set* di esercizi con la stessa struttura e contenuti di pari difficoltà, come prove di autovalutazione che gli studenti potrebbero svolgere a casa.

2 Il progetto M.In.E.R. Va.

2.1 Introduzione

Il progetto M.In.E.R. Va. (Matematica INterattiva Esercizi di Recupero e VALorizzazione)⁸ è un'evoluzione della gestione informatizzata delle prove d'esame e ha l'obiettivo di fornire uno strumento informatico flessibile che permetta agli insegnanti la creazione di percorsi, anche personalizzati, volti al recupero di debiti formativi o alla valorizzazione delle eccellenze.

Nella scuola secondaria, infatti, l'elevato numero di debiti formativi fatti rilevare dagli studenti in matematica — come le statistiche OCSE-PISA evidenziano — impone ai docenti di predisporre adeguati strumenti per il recupero.

Tradizionalmente, tali debiti vengono recuperati mediante corsi di recupero e assegnazione di compiti (personalizzati) a ciascuno studente. Se il numero di coloro che hanno carenze è elevato, si pone il problema della correzione dei compiti assegnati, spesso molto onerosa. Inoltre, nella maggior parte dei casi il recupero delle carenze passa anche per la ripetizione di molti esercizi tutti simili per acquisire automatismi di calcolo e risoluzione.

I test a risposta multipla con correzione automatica, nei quali vengono utilizzati esercizi scritti in forma parametrica, si affiancano agli interventi didattici tradizionali e mettono lo studente al centro del percorso didattico, permettendogli una maggiore autonomia nei tempi e nei modi di fruizione dei contenuti.

8. Il progetto si inquadra nell'ambito di una ricerca della durata di tre anni dell'Università Cattolica del Sacro Cuore.

Con lo stesso strumento, è possibile proporre percorsi di eccellenza per gli studenti interessati allo studio della matematica, anche in vista di una loro futura iscrizione a facoltà in cui la matematica è oggetto di studio.⁹

Esistono già software che propongono percorsi teorici e pratici utilizzati sia per il recupero del debito nelle scuole secondarie, sia come percorso in molte sedi universitarie.

Il software allo studio si differenzia da quelli già presenti sul mercato per i seguenti motivi:

- 1) la possibilità di fruire autonomamente offline del percorso;
- 2) un'ampia tipologia di esercizi oltre agli usuali test con domande a risposta chiusa singola: risposta aperta (con risposta numerica, vettoriale o in formato testo, anche matematico), completamento, risposta chiusa multipla;
- 3) la possibilità di fruizione senza installare alcun applicativo aggiuntivo se non pochi file su un server per la modalità online;
- 4) un'ottima resa del testo matematico (che è la parte nella quale molti altri software risultano carenti, poiché spesso le formule matematiche vengono importate come immagini, appesantendo la pagina e talvolta compromettendone la visualizzazione);
- 5) indipendenza dal sistema operativo (almeno dal lato utente).

Si è scelto di creare uno strumento informatico indipendente dal sistema operativo usato poiché molte scuole secondarie superiori si stanno orientando, come prevede la normativa, verso sistemi operativi *open source*. Questo aspetto permette da un lato di modificare liberamente il codice necessario, dall'altro di non avere spese per licenze, ma solo (eventualmente) per il materiale.

I docenti delle scuole coinvolte collaboreranno al progetto per quanto concerne la valutazione dei contenuti da proporre (sia in termini di efficacia didattica che in termini di difficoltà) la creazione del database di esercizi e dei percorsi teorici, la creazione e la gestione del supporto informatico per quanto di competenza della scuola, la sperimentazione degli strumenti proposti nelle classi.

2.2 La struttura del progetto

Il progetto prevede la realizzazione di quattro tipologie di contenuti:

- 1) schemi teorici per il rafforzamento o la creazione delle conoscenze;

9. Tali percorsi, pur previsti nel Piano dell'Offerta Formativa delle scuole, sono spesso di difficile realizzazione se gli studenti in questione sono inseriti in gruppi classe numerosi e disomogenei, nei quali le esigenze degli studenti in difficoltà prevalgono.

- 2) esercizi guidati per illustrare le principali modalità di risoluzione dei problemi;
- 3) esercizi di consolidamento e autovalutazione;
- 4) esercizi di verifica dei risultati ottenuti.

I contenuti possono presentare vari livelli di interattività e possono essere fruiti sia offline che online (tutti, tranne gli esercizi di verifica dei risultati ottenuti, che prevedono il tracciamento degli esiti da parte del docente).

Svolgendo gli esercizi in modalità offline, lo studente può ricevere automaticamente, a scelta del docente, i seguenti feedback:

- 1) attribuzione di un punteggio (valutazione);
- 2) segnalazione automatica della risposta esatta/sbagliata;
- 3) suggerimento per arrivare alla soluzione;
- 4) riferimenti ai contenuti teorici;
- 5) schema risolutivo.

Inoltre, nella modalità offline la fruizione dei contenuti avviene senza tracciare l'attività svolta dallo studente (come per le simulazioni delle prove ECDL). Ciò consente di minimizzare l'ansia da prestazione dello studente che può esercitarsi senza sentirsi valutato.

In modalità online è previsto, oltre a quanto già detto, anche il tracciamento, mediante la registrazione su un file che può restare sul server o essere inviato per email al docente e/o allo studente.

2.3 Il pacchetto minerva

Per la realizzazione del progetto, abbiamo predisposto un nuovo pacchetto che abbiamo chiamato *minerva*, mediante il quale è possibile realizzare gli esercizi, sia di consolidamento e autovalutazione, sia di verifica dei risultati ottenuti.¹⁰

Il pacchetto si caratterizza per un'estrema flessibilità, poiché, dallo stesso file di esercizi, consente di ottenere output differenziati che l'insegnante può offrire consequenzialmente o modularmente:

- 1) esercizio autonomo (con eventuale autovalutazione);
- 2) esercizio con monitoraggio degli esiti da parte del docente;
- 3) prova di accertamento.

10. Gli schemi teorici sono realizzati mediante presentazioni (usando *beamer* o strumenti simili) e dispense; gli esercizi svolti sono stati realizzati usando i pacchetti *slidyxx* e *mat-mak* del prof. Mařík, che ci ha messo gentilmente a disposizione anche tutto il materiale da lui predisposto.

Il pacchetto *minerva* può essere considerato un'evoluzione del pacchetto *esami-online*, di cui si è parlato nel paragrafo 1.4.

Le principali somiglianze riguardano il database di esercizi, che viene gestito in modo analogo, la presenza di un file master, dal quale vengono generati tutti gli output necessari per l'esecuzione della prova (file interattivo, file cartaceo, stringa delle soluzioni), i comandi matematici di semplificazione delle frazioni, degli esponenti, ecc. (già condivisi con il pacchetto *esami*). Inoltre, anche questo pacchetto permette di inserire parametri casuali negli esercizi: tale modalità consente allo studente con carenze di ripetere esercizi simili, ma con dati differenti, e di rafforzare la comprensione dei contenuti proposti.

Le principali differenze riguardano:

- le opzioni **compito** ed **esercitazione**: permettono di scegliere se si desidera costruire una prova di verifica vera e propria oppure un'esercitazione, che può essere tracciata oppure da svolgere offline. La principale differenza tra le due modalità consiste nel fatto che nella modalità **esercitazione**, oltre alla segnalazione della risposta giusta o sbagliata a test finito, è possibile inserire anche una traccia di soluzione che consenta allo studente di comprendere in modo autonomo i propri errori. Tale possibilità non è invece prevista nella modalità **compito**, nella quale si può scegliere se indicare allo studente quali risposte sono corrette e quali no. Inoltre, in modalità **esercitazione**, si può scegliere se somministrare a tutti gli studenti lo stesso set di esercizi (in questo caso il file non riporta all'inizio nome, cognome e classe dello studente, ma dei campi vuoti che di volta in volta devono essere compilati) oppure creare set personalizzati (in cui i campi nome, cognome e classe sono precompilati e gli esercizi vengono scelti e ruotati in modo casuale in base al compito che viene compilato).
- L'introduzione di un ambiente **solution** anche per i test a risposta multipla, come si diceva al punto precedente. Questa possibilità era già prevista nel pacchetto *exerquiz*, ma nel nostro caso comportava alcune difficoltà aggiuntive. Infatti, nel pacchetto *exerquiz* il contenuto dell'ambiente **solution** viene scritto *verbatim* su un file esterno, per poter poi essere messo in fondo a tutti i quiz ed essere reso visibile solo al termine della prova. Nel nostro caso ciò era impossibile perché il testo dell'esercizio e della sua soluzione è contenuto nel comando `\newproblem`, che, a nostra conoscenza, non poteva contenere *verbatim* come argomento. Questo problema non si presentava per gli esami cartacei, essendo stato gestito in modo

diverso: infatti il testo della soluzione viene scritto di seguito alla domanda a cui si riferisce se viene data l'opzione **solution** e invece viene cancellato se è data l'opzione **nosolution**. Nel pacchetto *minerva*, il problema dell'ambiente **solution** è stato risolto mediante l'uso del pacchetto *cprotect*.

- Il modo in cui vengono scelti gli esercizi all'interno del database: nei pacchetti *esami* ed *esami-online* gli esercizi vengono scritti ognuno in un file che contiene le n varianti (omogenee) dello stesso problema. L'inserimento di un esercizio in una prova viene fatto scegliendo una variante a caso da ogni file che si decide di inserire. Per le finalità del progetto, era invece opportuno che da uno stesso file si potessero scegliere più varianti dello stesso esercizio (si pensi per esempio ad una esercitazione sulle equazioni di secondo grado, nella quale l'apprendimento passa anche per la ripetizione di molti esercizi tutti simili). A tale scopo, è stato costruito l'ambiente **MNdb**, che ha un solo argomento obbligatorio, ovvero il nome di un file nel quale saranno scritti gli esercizi. All'interno di questo ambiente, il comando `\selectrandomlyn` permette di selezionare n varianti da un file (simile a quello costruito per gli esami tradizionali) e di scriverle nel file esterno definito nell'ambiente. La chiusura dell'ambiente fa scrivere, in ordine casuale o fissato, gli esercizi selezionati.

L'idea di base è quella di creare un database di esercizi che si adattino a diverse tipologie di necessità (esercizi autonomi, prove di valutazione, esercizi guidati). Il database sarà creato con la collaborazione dei docenti delle scuole coinvolte.

2.4 La sperimentazione effettuata nell'a. s. 2011-2012

La sperimentazione delle attività del progetto è iniziata nel settembre 2011. Sono stati coinvolti, nella prima fase, circa 80 studenti di due scuole secondarie di secondo grado, l'ITC "Giovanni Falcone" di Corsico e l'ITI "Giuseppe Lagrange" di Milano.

Nel corso del primo anno di attività del progetto è stato creato il pacchetto *minerva* e i file master necessari per la creazione delle prove: in particolare può essere considerata conclusa sia da un punto di vista "tecnico" che da un punto di vista "grafico" la parte relativa alle prove di esercitazione tracciata (ovvero con esiti salvati su server), di esercitazione autonoma non guidata (esercizi interattivi) e di verifica online con domande a risposta chiusa, valutata in modo automatico. Restano da completare le parti riguardanti le esercitazioni guidate (che vorremmo integrare nel pacchetto *minerva*) e la possibilità di utilizzare domande a risposta aperta o a risposta chiusa multipla.

Dal punto di vista didattico si è cominciato a sviluppare il percorso di recupero e consolidamento per le classi del triennio ITC, per le quali sono stati predisposti i materiali teorici e le esercitazioni (sia tracciate che non) su alcuni argomenti del programma. Parte del materiale predisposto è visibile sul sito web del progetto, all'indirizzo http://www.itcfalcone.it/portale/?page_id=4418.

I dati raccolti nella prima fase della sperimentazione sono in corso di elaborazione.

In alcune classi si è voluta anche sperimentare una modalità didattica alternativa: una parte degli esercizi e del materiale teorico è stata creata da alcuni studenti.

Questa sperimentazione ha dato esito largamente positivo in quanto ha permesso, tra l'altro, una riflessione da parte degli studenti sugli errori più comuni e ha dato la possibilità di far gestire loro, operativamente, problemi parametrici che di solito sono trattati in modo superficiale nello svolgimento "normale" del programma.

Il percorso ha permesso anche lo studio, seppure parziale, del linguaggio L^AT_EX, in collaborazione con i docenti di Laboratorio di Informatica.

L'utilizzo dei materiali prodotti ha reso necessaria la creazione di una piattaforma web attualmente ospitata sul sito dell'ITC "Falcone" e visibile all'indirizzo http://www.itcfalcone.it/portale/?page_id=4418. Viene utilizzato come web server Microsoft IIS. Le pagine sono scritte in linguaggio ASP.

La piattaforma prevede tre tipi di accesso:

- come utente *studente*: questo utente può partecipare agli eventi (prove, lezioni, ecc.) a cui è stato invitato dal docente. In futuro è previsto che possa visualizzare tutti i dati relativi alle sue prove (data di esecuzione, punteggio ottenuto, tempo di svolgimento, ecc.) e anche statistiche anonime relative alle prestazioni di studenti della sua classe o del suo gruppo di lavoro.
- Come utente *docente*: al docente vengono assegnate delle classi o dei gruppi; può creare eventi (prove, lezioni, ecc.) a cui invitare tutti gli studenti della classe/gruppo o solo alcuni; può visualizzare i risultati ottenuti da ciascun gruppo. In futuro potrà accedere a statistiche anonime relative alle prestazioni di tutti gli studenti della scuola. Inoltre, si prevede di integrare nella piattaforma un editor L^AT_EX che funzioni online, per consentire ai docenti la creazione di nuovi esercizi o la personalizzazione di quelli già esistenti. Sono stati valutati a tale scopo alcuni editor *open source*, il cui codice viene ora studiato per consentire la personalizzazione dell'interfaccia e l'integrazione nella piattaforma.

- Come utente *amministratore*: questo utente può creare eventi per tutti gli utenti della piattaforma; può visualizzare i dati relativi ad ogni utente; può compiere tutte le operazioni di amministrazione.

In futuro si prevede inoltre di migrare la piattaforma da server Microsoft IIS a server Apache (è necessario tradurre i file di comunicazione client-server dal linguaggio ASP al linguaggio PHP) e di creare un'interfaccia che consenta di gestire online il database degli esercizi, che sia simile alla piattaforma ExOmatiK (POULAIN).

Riferimenti bibliografici

- ADOBE SOLUTIONS NETWORK (1999). *FDF Toolkit Overview Technical Note # 5194*. ADOBE SYSTEMS INC.
- (2001). *Acrobat Forms JavaScript Object Specification, Version 5.0.5 Technical Note # 5186*. ADOBE SYSTEMS INC.
- BIANCHI, M. e MELZI, G. (1990). «Paut: un programma di autovalutazione». *Didattica delle Scienze*, n. 150.
- FERRARIO, D. «Mcq-xeL^AT_EX». URL <http://www.matapp.unimib.it/~ferrario/main.html>.
- GRIFFIN, F. «MacqT_EX». URL <http://rutherglen.ics.mq.edu.au/~macqtex/>.
- MAŘÍK, R. «Acroweb». URL <http://user.mendelu.cz/marik/index.php?item=41>.
- MESSINEO, G. e VASSALLO, S. (2010). «L^AT_EX per la creazione di temi d'esame. prima parte: la prova scritta tradizionale.» *Contributi di ricerca in Econometria e Matematica 113*, Università Cattolica del S. Cuore, Milano.
- POULAIN, N. «Exomatik». URL <http://www.exomatik.net/>.
- STORY, D. P. (2012). «Exerquiz & AcroT_EX». URL <http://www.acrotex.net/>.

- ▷ Grazia Messineo
Università Cattolica Milano – ITC "G. Falcone" Corsico
`grazia dot messineo at unicatt dot it`
- ▷ Salvatore Vassallo
Università Cattolica Milano
`salvatore dot vassallo at unicatt dot it`