

Unelte Interactive pentru Dezvoltarea Lecțiilor în Sisteme Colaborative de eLearning

Mircea Mărginean

Universitatea Tehnică din Cluj-Napoca,
Departamentul Calculatoare

Str. Barițiu, nr. 26-28, Cluj-Napoca,
România

margineanmircea@yahoo.com

Teodor Ștefanuț

Universitatea Tehnică din Cluj-Napoca,
Departamentul Calculatoare

Str. Barițiu, nr. 26-28, Cluj-Napoca,
România

teodor.stefanut@cs.utcluj.ro

REZUMAT

Dirrecția înspre care s-a îndreptat atenția în ultima vreme în cazul sistemelor eLearning este aceea de lecții colaborative bazate pe unelte interactive. Cu alte cuvinte, profesorul creează lecția folosind unelte interactive, după care o prezintă studenților în mod colaborativ, în timp real, asemeni unei lecții desfășurate în sala de clasă, unde profesorul interacționează cu studenții. În materialul de față am studiat tipuri de unelte interactive, precum și modalitatea de dezvoltare a acestora și integrarea lor în sisteme de eLearning deja existente pentru compunerea de lecții. Au fost studiate, dezvoltate și integrate următoarele tipuri de unelte: vizualizare text, vizualizare imagine, vizualizare video. Fiecare unealtă are două module: modulul de editare – disponibil profesorului și modulul de vizualizare – disponibil atât studenților cât și profesorului. Una din problemele pe care le-am abordat a fost aceea de maximizare a spațiului ocupat de modulul de vizualizare al uneltei și minimizarea spațiului ocupat de modulul de editare al uneltei. O altă problemă abordată a fost de propunere a unui standard de bază pentru dezvoltarea modulară a uneltelor astfel încât acestea să poată fi dezvoltate independent de sistemul în care vor fi integrate sub formă de module.

Cuvinte cheie

Unelte interactive, sisteme eLearning, sisteme colaborative, lecții interactive.

Clasificare ACM

H5.2. Information interfaces and presentation: User interfaces.

INTRODUCERE

Sistemele eLearning și-au făcut în ultima vreme din ce în ce mai tare simțită prezența în majoritatea mediilor academice, deoarece facilitează procesul de învățare fără a fi nevoie prezența fizică în aceeași încăpere a profesorului și studenților. Desigur că, pe lângă numeroasele avantaje evidente cum ar fi re folosirea materialului didactic, învățământul la distanță fără a fi necesară prezența fizică în aceeași locație, reducerea costurilor asociate procesului de predare etc., astfel de sisteme au neajunsurile lor față de metodele clasice de predare a materialului didactic. Unul dintre aceste neajunsuri este interacțiunea dintre profesor și studenți care poate ridica probleme datorită limitărilor tehnologice existente. Această interacțiune ar

trebui să aibă loc în mediul virtual cât mai asemănător cu mediul real, spre exemplu profesorul și studenții ar trebui să aibă la dispoziție o tablă de scris, sau profesorul să poată modifica “din mers” o lecție, iar actualizarea să se facă automat la studenți.

În sistemele eLearning, dispozitivul corespunzător tablei de scris din sălile de curs este monitorul, respectiv fereastra în care se vizualizează sau editează lecția. Această fereastră are dimensiuni mult mai reduse decât o tablă reală, de aceea trebuie găsite metode de gestionare cât mai bună a spațiului disponibil pentru lecție, ceea ce presupune folosirea unor metode de interacțiune cât mai inovative.

Desigur, o soluție simplă și la îndemână ar fi folosirea scrollbar-ului orizontal, însă interacțiunea cu acest scroll poate îngreuna procesul de eLearning, datorită mișcărilor multiple pe care trebuie să le execute utilizatorul pentru această interacțiune. Pe măsura evoluției tehnologice aceasta nu va mai fi o problemă atât de importantă, deoarece tendința actuală este spre monitoare cu diagonală din ce în ce mai mare. De asemenea, apariția mouse-ului cu scroll și pentru orizontală ar putea facilita navigarea pe pagina lecției prin simpla mișcare a degetului pe scroll, fără a necesita poziționarea mouse-ului pe scroll-ul orizontal, interacționând asupra lui prin “drag&drop”.

Un sistem de eLearning modern trebuie să cuprindă facilități de vizualizare și formatare a textului, vizualizare imagini, vizualizare video, vizualizare obiecte 3D, vizualizare galerie de imagini (slideshow). Toate aceste facilități pot fi văzute ca unelte care pot fi incluse în lecții. Uneltele sunt modulare și trebuie să pună la dispoziția utilizatorului diverse tehnici de interacțiune cum ar fi adnotare grafică sau modificare diverși parametri (înălțime, lățime etc.). Aceste interacțiuni trebuie să fie realizabile într-o manieră cât mai simplistă, deoarece cei mai mulți dintre profesori nu au aptitudinile tehnice necesare pentru o utilizare avansată a calculatorului.

Această lucrare nu se va ocupa de adnotarea grafică, în schimb va aborda modul de dezvoltare, integrare și interacțiune pentru diverse tipuri de unelte specifice sistemelor de eLearning.

STUDII SIMILARE

În Era Informațională, pentru a face față volumului imens de informații disponibile pe internet și a timpului liber disponibil din ce în ce mai puțin, a fost necesară apariția sistemelor de eLearning. Un sistem de eLearning robust ar

trebuie să permită atât crearea și vizualizarea de lecții prin re folosirea materialelor existente, cât și colaborarea între profesor și studenți. Colaborarea poate fi atât sincronă (live chat, white board, conferințe video și audio, sesiuni colaborative etc.) cât și asincronă (emailuri, forumuri de discuții, linkuri pentru download-ul materialelor etc.). Desigur că metoda de dorit este cea sincronă deoarece se aseamănă mai mult cu metodele de predare clasice.

Datorită avansului tehnologic din ultima perioadă, a evoluției domeniului educațional și a abundenței informațiilor din toate domeniile, multe organizații s-au ocupat de dezvoltarea de sisteme de eLearning, unele axându-se pe partea colaborativă, iar altele axându-se pe îmbunătățirea designului și a tehnicilor de interacțiune. Aplicații precum eTrace [1], Web-Trace [2] sau Dokeos [3] pune la dispoziție utilizatorilor tehnici de interacțiune prin adnotare grafică pe materialele didactice, în timp ce alte proiecte [4] experimentează colaborarea bazată pe vorbire.

În ciuda dezvoltării puternice a sistemelor de eLearning, uneltele disponibile pentru crearea lecțiilor sunt foarte puține și nu foarte ușor de folosit, datorită faptului că majoritatea sistemelor și-au concentrat cercetările pe aplicații care necesită abilități tehnice din partea utilizatorului, cum este cazul LOC Tool [5], care este o unealtă de authoring și care include o metodologie de dezvoltare a materialelor didactice online bazată pe template-uri pedagogice.

UNELTE INTERACTIVE

O unealtă este o componentă a unei lecții. Unealta poate fi un Image Viewer, Video Player, Slideshow Viewer, Rich Text Editor, File Uploader, o unealtă de administrare a proceselor executate pe grid (folosită în procesarea imaginilor de dimensiuni foarte mari) etc. O unealtă poate avea una sau mai multe instanțe.

O instanță a unei unelte are două moduri de rulare:

- modul editare – poate fi folosit numai de către autorul uneltei pentru a edita proprietățile acesteia (cum ar fi URL-ul resursei, înălțime, lățime etc.);
- modul vizualizare – poate fi folosit de către oricine (student, profesor, administrator) pentru a vizualiza instanța uneltei.

Avantajele folosirii uneltelor pentru dezvoltarea lecțiilor în cadrul unui LMS (Learning Management System), în comparație cu alte soluții tehnice bazate doar pe *rich text editors*, sunt următoarele:

- procesul de dezvoltare și integrare a uneltelor în sistem este independent și poate fi realizat mult mai ușor;
- tipurile de interacțiune pot fi variate, spre exemplu interacțiuni cu creionul grafic (adnotări grafice), interacțiuni vocale, interacțiuni cu mouse-ul și tastatura etc.;
- flexibilitatea în alegerea și folosirea unei tehnologii pentru dezvoltarea uneltelor – diverse tehnologii disponibile la ora actuală în domeniul web (PHP, Flash, Flex) pot fi folosite atâta timp cât este respectat un anumit standard impus de sistemul de eLearning pentru care se dezvoltă unealta respectivă;

- flexibilitatea în re folosirea materialului educațional prin mijloace de repurposing (repurposing-ul presupune folosirea unor obiecte pre-existente create într-un anumit context inițial și cu un anumit scop inițial și adaptarea acestora astfel încât să servească unui nou scop în același context sau într-un nou context). Mai multe informații despre tipurile de repurposing pot fi găsite în [6].

Definiția unui standard de bază

În urma cercetărilor efectuate de echipa noastră, a ieșit la iveală necesitatea exprimării câtorva reguli de bază ce trebuie respectate pentru dezvoltarea uneltelor, reguli pe care le-am strâns sub forma unui standard de bază.

Pentru ca uneltele să poată fi dezvoltate într-o manieră independentă, un standard trebuie definit și respectat pentru fiecare sistem de eLearning în parte. Spre exemplu, pentru aplicația noastră am definit următorul set de reguli pentru fiecare unealtă:

Regula 1: Unealta trebuie să garanteze că de fiecare dată când pagina este reîmprospătată, o copie proaspătă a uneltei este încărcată și nu o copie rămasă în memoria cache. Este lucru este necesar pentru partea colaborativă în timp real a sistemului, spre ex. dacă profesorul editează o instanță a unei unelte în timp ce studenții o vizualizează, aceștia trebuie să fie siguri ca prin refresh-ul paginii vor vizualiza cea mai actuală versiune a instanței.

Regula 2: Unealta trebuie să implementeze un mecanism de creare de noi instanțe ale ei, astfel încât fiecare instanță să aibă un identificator unic în sistem iar acest identificator trebuie returnat sistemului într-o oarecare formă.

Regula 3: Unealta trebuie să își administreze singură instanțele din baza de date (prin operații de Insert, Update, Delete).

Regula 4: Unealta trebuie să poată face distincție între cele două moduri de rulare (modul editare și modul vizualizare) în funcție de modul în care este apelată instanța uneltei. De asemenea, unealta trebuie să verifice dacă utilizatorul care apelează instanța într-unul din moduri are drepturi pentru acel tip de apel (de ex. dacă instanța este apelată în modul editare, unealta trebuie să verifice dacă utilizatorul apelant este autorul instanței).

Regula 5: Unealta trebuie să aibă implementat propriul mecanism de sincronizare pentru a putea comunica cu diferiți clienți care rulează aceeași instanță a uneltei. De exemplu, dacă unealta este Slideshow Viewer, există diferite moduri de colaborare între clienți. Unul dintre moduri ar fi ca profesorul să poate rula, opri, pauza slideshow-ul, trece la următorul slide, în timp ce studenții pot doar vedea slideshow-ul comandat de profesor. Un alt mod ar fi acela în care atât profesorul cât și studenții pot manevra independent și individual slideshow-ul.

Regula 6: Unealta trebuie să implementeze propriul mecanism de actualizare/colaborare, astfel încât atunci când autorul editează instanța unei unelte, toți clienții care vizualizează instanța respectivă la acel moment să fie notificați de actualizarea instanței, fie prin refresh automat al paginii clienților, fie printr-un mesaj care îi avertizează că versiunea instanței lor nu este cea mai actuală și că ar

trebuie să facă refresh pentru a vedea versiunea cea mai nouă.

Trebuie menționat faptul că aceste reguli de mai sus reprezintă mai degrabă niște linii de ghidare în dezvoltarea de unelte și nu un standard bine definit care trebuie neapărat respectat. Acestea reguli au rezultat din problemele cu care ne-am confruntat în procesul de dezvoltare a uneltelor, iar o direcție de cercetare ulterioară ar putea fi asupra îmbunătățirii acestora sau a adăugării de noi reguli.

Exemplu de unealtă

În liniile ce urmează vom prezenta procesul de dezvoltare a unelei ImageViewer (folosită pentru vizualizarea unei imagini) ce respectă regulile de mai sus.

În figura 1 am prezentat o versiune a unelei ImageViewer din perspectiva autorului (în modul editare), iar în figura 2 am prezentat aceeași unealtă din aceeași perspectivă, dar interacțiunea cu formularul de editare se face diferit, astfel că acest formular nu mai este vizibil imediat sub unealtă. În schimb poate fi făcut vizibil cu un click dreapta al mouse-ului și făcut invizibil prin click în orice altă parte a unelei. Prin această metodă nu numai că am redus spațiul ocupat de unealtă, ci am îmbunătățit și maniera de schimbare între modurile editare și vizualizare. În timp ce în figura 1 autorul trebuia să schimbe între modurile de vizualizare printr-un click pe linkul corespunzător, în figura 2 autorul interacționează cu unealta într-o manieră WYSWYG (What You See is What You Get), nefiind necesare mișcări de interacțiune suplimentare pentru schimbarea între moduri. În cazul figurei 2, formularul de editare apare din punctul unde se dă click cu mouse-ul, acesta putând fi suprapus complet peste imagine sau poziționat aproximativ în afara imaginii pentru a putea păstra contactul vizual și cu imaginea.

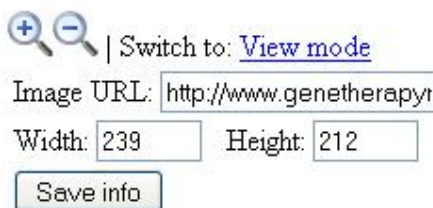
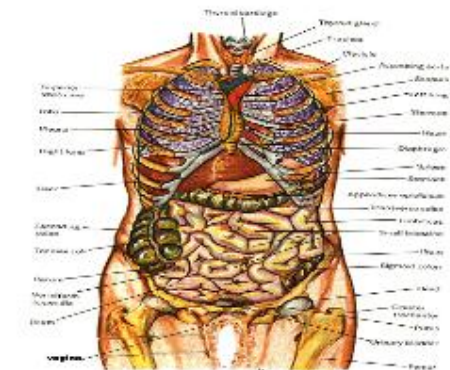


Figura 1. Unealta ImageViewer în modul editare – versiunea 1

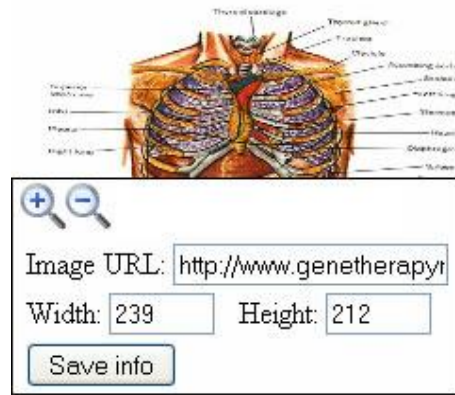


Figura 2. Unealta ImageViewer în modul editare – versiunea 2

Între cele două versiuni de unealtă prezentate mai sus, se vede clar că a doua este superioară din punct de vedere al utilizării spațiului disponibil. În plus, după cum vom arăta mai jos în lucrare, se îmbunătățește considerabil și modul de interschimbare între modurile editare și vizualizare ale instanței.

Notă: În cazul uneltelor de tip embed cum ar fi Video Player de video-uri de pe youtube, unde nu poate fi accesat evenimentul de click dreapta, am folosit tehnici de interacțiune alternative, cum ar fi mouse over, tehnici destul de evidente pentru utilizatorul obișnuit.

Unealta ImageViewer este dezvoltată în tehnologia PHP și este folosită pentru vizualizarea unei imagini. Proprietățile de bază editabile ale unelei sunt: înălțimea, lățimea, URL-ul imaginii (poate fi local sau remote). Autorul instanței unelei poate de asemenea să facă operațiuni de zoom in/zoom out cu un pas de 10%.

Fiecare tip de unealtă are atașată o tabelă a instanțelor. Proprietățile instanței unelei sunt salvate direct în baza de date corespunzătoare unelei respective.

Parametrii de apel a instanței unelei se transmit prin URL, după cum arătăm mai jos, unealta integrându-se în sistemul de eLearning prin iframe-uri, făcând astfel posibilă auto-administrarea unelei.

Pentru a respecta regulile definite anterior, am definit o clasă abstractă numită Tool.php care definește următoarele metode publice și abstracte pe care trebuie să le implementeze orice unealtă: `createInstance()`, `displayInstance($instID)`, `deleteInstance($instID)`.

Clasa ImageViewer extinde clasa Tool și implementează metodele de mai sus. Primele linii de cod din clasă sunt pentru pornirea sesiunii (`@session_start()`), ca să putem verifica drepturile de acces și pentru încărcarea unei copii proaspete de fiecare dată când pagina este reîmprospătată (`@header("Cache-Control: no-cache");`).

Unealta implementează următoarele metode:

- `create()` – această metodă este apelată când apelul la clasa ImageViewer.php se face fără nici un alt parametru în URL. Această metodă creează o nouă instanță a unelei cu un identificator unic, inițializează proprietățile default și inserează instanța în tabelă, împreună cu autorul instanței unelei, după care redirecționează pagina către URL-ul `ImageViewer.php?id=12345`, unde `id` este

identificatorul unic al instanței. Metoda `createInstance()` returnează `this->create()`.

- `show()` – această metodă afișează instanța în modul corespunzător (editare sau vizualizare). Înainte de afișarea într-unul din moduri, unelta verifică dacă userul care apelează instanța are drepturi pentru modul în care o apelează. Metoda `displayInstance($instId)` returnează `this->show()`.
- `delete()` – această metodă șterge instanța din tabela uneltei. Metoda `deleteInstance($instId)` returnează `this->delete()`.

Unelta pune la dispoziție un mecanism de interschimbare între modul editare și modul vizualizare, disponibil doar autorului instanței respective. În cazul `ImageViewer`-ului, interschimbarea se face prin click dreapta cu mouse-ul peste imagine, însă modul de interschimbare diferă de la unelță la unelță în funcție de limitările tehnologice ale uneltei, de exemplu la `Video Viewer` interschimbarea se face cu click dreapta (deoarece obiectele de tip `embed` nu permit apelarea metodei `javascript onmouseover`), la `Rich Text Editor` interschimbarea se face prin dublu click pe textul respectiv.

Inițial, modul de afișare a instanței se făcea în funcție de parametrii cu care se apela unelta, astfel:

- `ImageViewer.php?id=12345` – se afișează instanța cu id-ul dat în modul editare dacă userul care o apelează este autorul instanței, sau în modul vizualizare în caz contrar;
- `ImageViewer.php?id=12345&mode=edit` – se afișează instanța cu id-ul dat în modul editare dacă userul care o apelează are drepturi pentru acest mod, în caz contrar afișând o eroare;
- `ImageViewer.php?id=12345&mode=view` – se afișează instanța cu id-ul dat în modul vizualizare.

Uterior, după îmbunătățirea modului de afișare a uneltei, s-a simplificat și modul de apelare a instanței, astfel că unelta se apelează doar cu `ImageViewer.php?id=12345`. Dacă userul care apelează instanța este autorul ei, atunci el are posibilitatea interschimbării între modurile editare și vizualizare prin tehnicile prezentate mai sus (mouse over, click dreapta, dublu click etc.), iar dacă userul nu e autorul instanței, atunci nu are această posibilitate.

Prin modificările făcute am obținut creșterea utilizabilității uneltei, avantajele pentru această unelță fiind următoarele:

- simplificarea modului de interacțiune la interschimbare între modurile de afișare;
- minimizarea spațiului ocupat de unelță și de formularul de editare a uneltei, ceea ce a dus la maximizarea randamentului de folosire a spațiului de pe ecran;
- reducerea timpului de așteptare la comutarea între modurile de afișare, deoarece javascript-ul se execută la client, pe când apelurile de tip GET și POST prin URL se execută printr-o cerere la server.

Un dezavantaj ar fi inconsistența tipurilor de interacțiune între diverse unelte, utilizatorul putând fi derutat de faptul că la `ImageViewer` trebuie să ducă mouse-ul peste unelță, la `Video Viewer` trebuie să dea click dreapta, iar la `Rich Text Editor` trebuie să dea dublu click pe text.

Un alt dezavantaj ar fi chiar utilizarea javascript-ului care poate fi dezactivat din browser-ul clientului, nemaifiind astfel posibilă interschimbarea între modurile de afișare ale uneltei.

Metodologia de implementare și dezvoltare a celorlalte unelte este asemănătoare, de aceea vom prezenta în continuare doar câteva imagini care să prezinte interfața uneltelor, fără a mai intra în detalii.

Figura 3 prezintă unelta numită `Video Viewer`, folosită pentru vizualizarea unui filmuleț. URL-ul video-ului poate fi fie al unui filmuleț de pe site-ul `youtube.com`, caz în care unelta construiește codul `embed` corespunzător pentru player-ul `youtube`, fie al unui filmuleț de pe un server, caz în care unelta folosește media player-ul default în care include filmulețul respectiv.



Figura 3. Unelta `Video Viewer` în modul editare

Figura 4 prezintă unelta numită `Text Viewer` (care este un `Rich Text Editor`) și care permite vizualizarea și formatarea unui text.

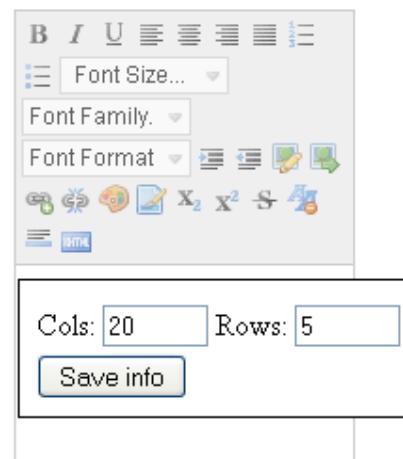


Figura 4. Unelta `Text Viewer` în modul editare

Celelalte unelte, precum `Slideshow Viewer` sau unelta pentru managementul execuției proceselor pe grid sunt în curs de dezvoltare.

În continuare, pentru a exemplifica cele susținute mai sus, vom prezenta un studiu de caz și anume crearea unei lecții. Vom arăta cum se crează uneltele, cum se editează, cum se adaugă în lecție de către autor și apoi cum este vizualizată de către studenți lecția finală cu toate uneltele incluse.

STUDIUL DE CAZ – CREAREA UNEI LECȚII

Profilul utilizatorului-profesor

Un utilizator al sistemului de eLearning, care dorește să creeze material educațional refolosibil trebuie să aibă următoarele caracteristici:

- este un specialist în domeniul în care dorește să creeze materialul educațional;
- cunoaște tipurile de bază de interacțiune utilizator în sistemele eLearning;
- nu are cunoștințe tehnice despre tehnologiile folosite precum HTML, Javascript, AJAX sau despre concepte precum baze de date distribuite, servicii web etc.;
- are cunoștințe medii de utilizare a calculatorului, care includ: navigare pe internet, tehnoredactare, cunoștințe de bază despre tipurile de fișiere, despre operațiile disponibile în cadrul unui sistem de operare.

Pașii premergători unei lecții (presupunând că profesorul și studenții sunt înscriși deja în sistem) ar fi ca profesorul să creeze un nou curs, în cadrul căruia va crea lecții. După crearea cursului, fie studenții se înscriu la cursul respectiv și profesorul le aprobă înscrierea, fie profesorul îi înscrie direct pe studenți (ambele modalități sunt implementate în sistemul nostru). O dată parcurși acești pași, profesorul poate trece la crearea unei lecții în cadrul unui curs.

Lecții interactive

Una dintre marile probleme apărute în procesul de creare a materialelor didactice este dată de modul de descriere al layout-ului vizual al elementelor ce fac parte din cadrul aceleiași resurse (în cazul nostru – lecția).

De obicei pentru formatarea unui layout al unei lecții se folosesc limbaje ca HTML și CSS, pentru care există o multitudine de aplicații online care generează astfel de cod (ex. rich text editors), dar aceste aplicații nu au fost dezvoltate în scopuri educaționale, de aceea funcționalitățile de formatare sunt relativ limitate. Aplicația dezvoltată de noi asigură un mecanism de control al layout-ului unei lecții, precum și mecanisme de creare și vizualizare a lecțiilor.

Structura unei lecții este formată din unul sau mai multe pattern-uri, unde fiecare pattern reprezintă un rând din lecție, un astfel de rând putând conține una sau mai multe unelte înșiruite pe orizontală. Profesorul poate folosi unul din pattern-urile predefinite sau poate crea pattern-uri noi, pe care le poate salva și refolosi ulterior.

O lecție odată creată poate fi salvată ca un template (grup de pattern-uri) și refolosită de către alți profesori în procesul de creare de noi materiale educaționale.

Crearea unei lecții din medicină

Pentru acest studiu de caz am ales să prezentăm crearea unei lecții din medicină.

Scenariul: Profesorul dorește să creeze o lecție cu titlul “Anevrismul Abdominal” care să conțină text, câteva imagini reprezentative, un model 3D (cu un nivel de detaliere grafică foarte mare) și un video. Pentru acestea are nevoie de uneltele *TextViewer*, *ImageViewer*, *VideoViewer* și de o *unealtă capabilă să afișeze informații*

prin streaming de pe internet, și care se poate conecta cu un cluster grafic (pentru modelul 3D).

Acțiunile: Profesorul accesează modulul de creare a unei noi lecții și adaugă pattern-uri la lecție și unelte la pattern-uri. După adăugarea uneltelor în pattern-urile lecției, profesorul poate modifica proprietățile fiecărei unelte în parte prin intermediul formularului de editare al fiecărei unelte (vezi Fig. 5). O dată finalizate toate modificările asupra uneltelor, lecția poate fi salvată și este automat accesibilă utilizatorilor autorizați pentru a o vizualiza.

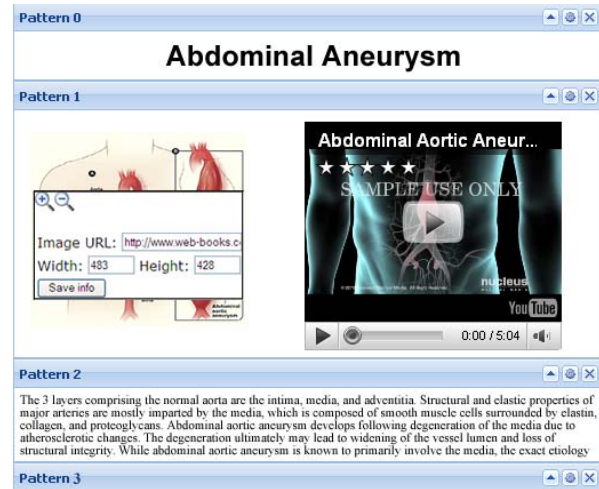


Figura 5. Exemplu de lecție cu unelte interactive

Lecția poate fi salvată sub formă de template, iar template-ul respectiv poate fi făcut fie public (caz în care îl pot include și alți profesori în lecțiile lor, fie privat (caz în care doar autorul template-ului îl poate folosi pentru crearea altor lecții).

CONCLUZII

Câteva din avantajele folosirii de unelte pentru dezvoltarea de material educațional în sisteme eLearning, în comparație cu soluțiile tehnice bazate doar pe rich text editors, sunt:

- adăugarea de noi funcționalități ale sistemului (cum ar fi noi tehnici de interacțiune, noi unelte, noi tipuri sau surse de date etc.) poate fi făcută mult mai ușor;
- uneltele pot fi dezvoltate folosind diferite tehnologii, cum ar fi: PHP, Javascript, Flash, Flex etc. în funcție de care se potrivește cel mai bine scopului pentru care este creată uneltea respectivă;
- posibilitatea de includere în aceeași lecție, de date din surse diferite (interne sau externe) și folosirea de protocoale de comunicare diferite.

Fiecare unelță dezvoltată este responsabilă cu implementarea interfeței utilizator necesară pentru setarea proprietăților specifice uneltei (sursa de date, tipurile de interacțiune, dimensiuni etc.) și cu implementarea corectă și completă a celor două moduri de rulare: editare și afișare. Mai mult, fiecare unelță trebuie să implementeze un API standard care să permită:

- schimbul de informații dintre unelță și sistem;
- controlul comportamentului uneltei, pentru a restricționa drepturile de acces în conformitate cu rolul utilizatorului (de ex. dacă utilizatorul este un

student, el nu poate să acceseze interfața de editare a uneltei);

- comutarea între modurile de lucru editare și vizualizare.

Prin aplicația noastră și unelte dezvoltate de noi, am reușit să aducem următoarele îmbunătățiri sistemelor LMS (Learning Management Systems):

- am propus noi metode de interacțiune pentru interschimbarea între modurile editare și vizualizare (click dreapta, mouse over, dublu click) care au condus la o mai bună utilizare a spațiului de pe ecran disponibil lecției;
- am dezvoltat un set de reguli de bază care să ușureze procesul de dezvoltare a uneltelor și integrarea acestora în sistem;
- dezvoltarea uneltelor se poate face într-o manieră independentă și individuală;
- sistemul facilitează salvarea lecțiilor (grupuri de pattern-uri, care la rândul lor sunt grupuri de unelte) ca și template-uri pentru a putea fi refolosite ulterior;

Ca dezvoltări ulterioare ne-am propus următoarele scopuri:

- dezvoltarea de noi unelte cum ar fi: editor de grafuri, manager de execuții de procese pe grid, slideshow etc.;
- îmbunătățirea setului de reguli pentru dezvoltarea uneltelor, prezentat în această lucrare, prin rafinarea regulilor existente și adăugarea de reguli noi pe măsura cercetării și dezvoltării a noi tipuri de unelte;
- strângerea regulilor propuse sub forma unui standard general;
- cercetarea aspectului colaborativ sincron/asincron al uneltelor și adaptarea uneltelor în acest sens astfel încât profesorul și studenții să poată colabora într-un mod cât mai asemănător cu cel din viața reală;
- posibilitatea de adnotare grafică cu creionul grafic sau cu mouse-ul atât pe fiecare unelă în parte cât și pe toată lecția;

CONFIRMARE (MULȚUMIRI)

Activitățile din spatele acestei lucrări au fost susținute de către:

- ESA PECS Contract no. 98061 GiSHEO – On Demand Grid Services for High Education and Training in Earth Observation – dezvoltarea platformei de bază;
- mEducator project – Multi-type Content Repurposing and Sharing in Medical Education, funded by European Community, under the Contract 418006/2009, eContentPlus section – dezvoltarea uneltelor;

REFERINȚE

1. Gorgan D, Stefanut T, Gavrea B. Pen Based Graphical Annotation in Medical Education in Proc, 20th IEEE International Symposium on Computer-Based Medical Systems(Maribor, Slovenia):681–686 2007.
2. Giordano D, Leonardi R. Web-Trace and the Learning of Visual Discrimination Skills in Proc, 1st International Workshop on Pen-based Learning Technologies (Catania, Italy):CD Published by IEEE Computer Society 2007.
3. Dokeos eLearning at <http://www.dokeos.com/>
4. Li W, Zhang Y, Fu Y. Speech emotion recognition in e-learning system based on affective computing in Proc, 3rd Conf. on Natural Computation; 5 (Haikou, China) 2007.
5. Watson J, Dickens A, Gilchrist G. The LOC Tool: Creating a Learning Object Authoring Tool for Teachers in Association for the Advancement of Computing in Education (<http://eprints.soton.ac.uk/52551/01/LOCTool.pdf>) 2008.
6. Kaldoudi E., Dovrolis N., Konstantinidis S., Bamidis P.D., "Social Networking for Learning Object Repurposing in Medical Education", The Journal on Information Technology in Healthcare, vol. 7(4), pp. 233–243, 2009.