

## INTRODUCERE

În ultima vreme se vorbește tot mai mult despre *noua economie*, care este bazată pe tehnologia IT precum și de faptul că societatea umană trece într-o nouă fază denumită **Societatea Cunoașterii sau Societatea Informațională**. Trecerea la Societatea cunoașterii reprezintă o nouă economie în care procesul de inovare (capacitatea de a asimila și converti cunoașterea nouă pentru a crea noi servicii și produse) devine determinant. Inovarea, în societatea cunoașterii, urmărește a îmbunătăți productivitatea, nu numai productivitățile clasice în raport cu munca și capitalul, ci și productivitățile noi în raport cu resursele energetice și materiale naturale, cu protecția mediului.

Într-un raport al Comisiei Comunităților Europene din anul 2001 se arată că pentru a obține beneficii din noua economie sunt necesare un Internet din ce în ce mai performant și reforme structurale adecvate în societate, administrație și economie. Influența Internetului ca piață în societatea informațională și recunoașterea importanței valorii bunurilor (activelor, assets) intangibile, în special cunoașterea, reprezintă caracteristici ale noii economii. Richard Boulton caracterizează astfel diferența dintre vechea și noua economie: în prima contează bunurile tangibile, în a doua, activele intangibile care crează valoare. Intangibilul este nematerial, greu de descris și mai ales de quantificat și măsurat. Activul (bunul) intangibil are valoare și crează valoare.

În acest context, al noii economii, nevoia de instruire continuă a proprietarilor angajați devine o condiție esențială de supraviețuire și dezvoltare a oricărei întreprinderi. Instruirea continuă este privită ca o condiție necesară menținerii statusului profesional al specialistului în orice organizație. Una din modalitățile de instruire continuă o reprezintă și **învățământul virtual sau e-learning**. Aceasta este reprezentat de interacțiunea dintre procesul de predare/învățare și tehnologiile informaționale – **ICT (Information and Communication Technology)**. Materiale educaționale electronice au devenit o veritabilă sursă bibliografică și imagistică pentru prezentarea lecțiilor sau pregătirea temelor, profesorii fiind antrenați în module de pregătire specială în domeniul ICT. La nivel mondial, există tot mai multe instituții educaționale care funcționează exclusiv prin intermediul Internetului, oferind fie un ciclu complet de învățământ, fie cursuri de specializare adresate unei palete foarte largi de utilizatori.

Învățământul tradițional își revendică prerogativa dialogului direct dintre profesor și studenți, prin care se creează un cadru optim pentru modelarea personalității, menținerea mai pregnantă a spiritului de competiție, dar și a suportului psihologic reciproc. Fără a pretinde exclusivitate, e-learning-ul oferă noi oportunități de instruire și de comunicare, devenind tot mai popular. Studii recente indică faptul ca nu mai există practic nici o activitate socială (educație, cercetare, cultură, economie, transporturi sau divertisment) care să nu fie secundată de o prezentare în format electronic concretizată în pagini Web pe Internet, baze de date multimedia, prezentări interactive etc. Pregătirea cadrelor didactice pentru a utiliza eficient TIC în activitatea didactică apare ca deziderat major în coordonarea procesului instructiv-educativ. Accesul la Internet înlătură barierele geografice și temporale, permitând colaborări la distanță, accelerând corespunzător ritmul de obținere și de comunicare a ideilor și rezultatelor. Noile

tehnologii educaționale constituie astfel de rezultate și se propagă prin Internet pentru a fi folosite în activitatea didactică.

Majoritatea marilor universități a adoptat obligativitatea prezentării cursurilor pe Web (tematica, cuprins, bibliografie), oferindu-se online toate materialele de curs în cadrul intranetului propriu. Au fost elaborate, se dezvoltă și se extind tot mai mult produse program specializate pentru realizarea de cursuri electronice interactive, iar un rol important continuă să îl joace materialele de instruire pentru servicii și instrumente de lucru în rețea, necesare pentru inițierea scolarilor, studenților și profesorilor în folosirea accesului la Internet, obținerea de informații utile sau găsirea unor parteneri. Rețele specializate în teleeducație (cum sunt KidLink, ThinkQuest sau K12), acoperind o arie tot mai mare și înglobând tot mai multe școli din lumea întreagă, stimulează comunicația electronică a școlarilor în cadrul procesului de globalizare, fiind folosite adesea ca un motor important în procesul educațional. De mai mulți ani, pe lângă cercetare, învățământ, știri și divertisment, Internet-ul a devenit și mediu de afaceri, ultimele studii raportând aproape 300 de milioane de utilizatori (mai mult de o treime a acestora sunt din SUA).

Internet-ul permite închegarea și desfacerea rapidă a unor echipe de cercetare, indiferent de locul unde activează partenerii, dar și realizarea unor legături de durată, bazate pe afinități și pasiuni comune, legături umane care apar și se mențin prin mesaje transmise prompt prin poșta electronică (corespondență greu de imaginat cu mijloacele tradiționale). Deși modul de învățare tradițional este în continuare scump și relativ lent, noile tehnologii îl vor transforma într-un proces mult mai productiv.

Schimbările majore din sfera învățării asistate de calculator au condus la obținerea unei noi paradigmă educaționale: în locul realizării unui flux informațional cu un singur sens (de exemplu de la profesor către un grup pasiv de cursanți), noile tehnici de învățare sunt, ca și Internet-ul, bidirectionale, colaborative și interdisciplinare. Pătrunderea noilor tehnologii în sălile de clasa a schimbat radical relația care era fundamentată între profesori și elevi. Astfel, profesorii au trecut de la poziția de conducători atotștiitori la cea de ghizi ai infosferei, iar materialele instructive au evoluat de la cărțile rigide la programele administrabile. Informația a devenit mult mai accesibilă, iar utilizatorii își aleg singuri ceea ce doresc, fiecare devenind un creator de conținut. Mediul educațional al viitorului va fi un campus electronic: locul în care se vor desfasura orele de clasa, în care se va consulta biblioteca și desigur, unde vor avea loc contactele sociale.

În sens restrâns, elearning-ul reprezinta un tip de educatie la distanta, ca experienta planificata de predare-învatare organizata de o institutie ce furnizeaza mediat materiale într-o ordine secventiala si logica pentru a fi asimilate de studenti în maniera proprie. Medierea se realizeaza prin noile tehnologii ale informatiei si comunicarii - în special prin Internet. Internetul constituie atât mediul de distributie al materialelor, cât și canalul de comunicare între actorii implicați. Funcțional deocamdata doar la nivelul învățământului superior și în educatia adultilor, sistemul de instruire prin Internet replica și adapteaza componentele demersului didactic traditional/ fata-în-fata: planificare, continut specific și metodologie, interacțiune, suport și evaluare.

# CAPITOLUL I : Studiul și Analiza Sistemului Existență

## **1.1 Prezentarea societății**

Societatea Comercială „Evolution” S.R.L. cu sediu în Satu Mare pe strada Capricornului, numărul 8 s-a înființat în anul 2006, ca organizatie având drept specific producerea materialelor pentru învățamânt deschis și la distanță.

Principalul obiectiv al fimei, înscris în statutul ei de înființare este educația bazată pe tehnologiile Internet și Web (e-learning) folosite ca : mediu colaborativ de instruire, mediu de comunicare profesori-studenții și studenți-studenți, mediu de administrare cursuri și evaluare studenți . Firma oferă cursanților ultimele materiale disponibile în materie de specialitatea aleasă (specialitățile disponibile sunt cele din domeniul IT ele fiind : PHP, Flash, C++, Java, HTML, SQL, XML, Dreamweaver, UNIX/Linux, J2EE).

## **1.2 Rolul și poziția unității în economie**

Obiectivele unității sunt:

- dezvoltarea modelului calității în e-learning orientat pe student prin care se specifică procesele, activitățile, produsele, resursele, serviciile, precum și caracteristicile de calitate, indicatorii de măsurare și criteriile de evaluare;
- elaborarea metodologiei de evaluare a calității și a recomandărilor de aplicare în mediile e-learning, în concordanță cu standardele internaționale de măsurare și evaluare și cu cele mai bune practici utilizate pentru evaluarea calității în elearning;
- dezvoltarea metodelor și indicatorilor de măsurare și evaluare a calității resurselor digitale și a serviciilor de e-learning din perspectiva utilizatorului (studentului);
- dezvoltarea modelelor de evaluare a acceptanței tehnologiilor e-learning prin extinderea și adaptarea la domeniul e-learning a modelelor conceptuale de acceptanță a tehnologiilor;
- evaluarea calității în e-learning prin metode și instrumente software luând în considerare procesele de dezvoltare a sistemelor de e-learning, resursele (informațiile, conținutul digital etc.) și procesele de furnizare a serviciilor elearning;
- evaluarea e-learning prin luarea în considerare a caracteristicilor și activităților utilizatorului (studentului), a tipologiei sistemelor de e-learning și a gradului de disponibilitate a serviciilor furnizate;
- dezvoltarea metodelor și soluțiilor software prin care se asigură corelarea calității serviciilor de e-learning cu performanțele și strategia furnizorului de e-learning.

Pentru realizarea acestor obiective, firma are la îndemână o serie de pârghii de acțiune și reglare, dintre care:

- Elaborarea unei politici de funcționare și dezvoltare a activității în funcție de cerințele pieței;
- Elaborarea de strategii în concordanță cu dinamica economiei de piată națională și cu reglementările existente în domeniu.

### 1.3 Indicatorii economici :

Caracterizarea evoluției și activității societății în cei 3 ani de la înființare se prezintă astfel :

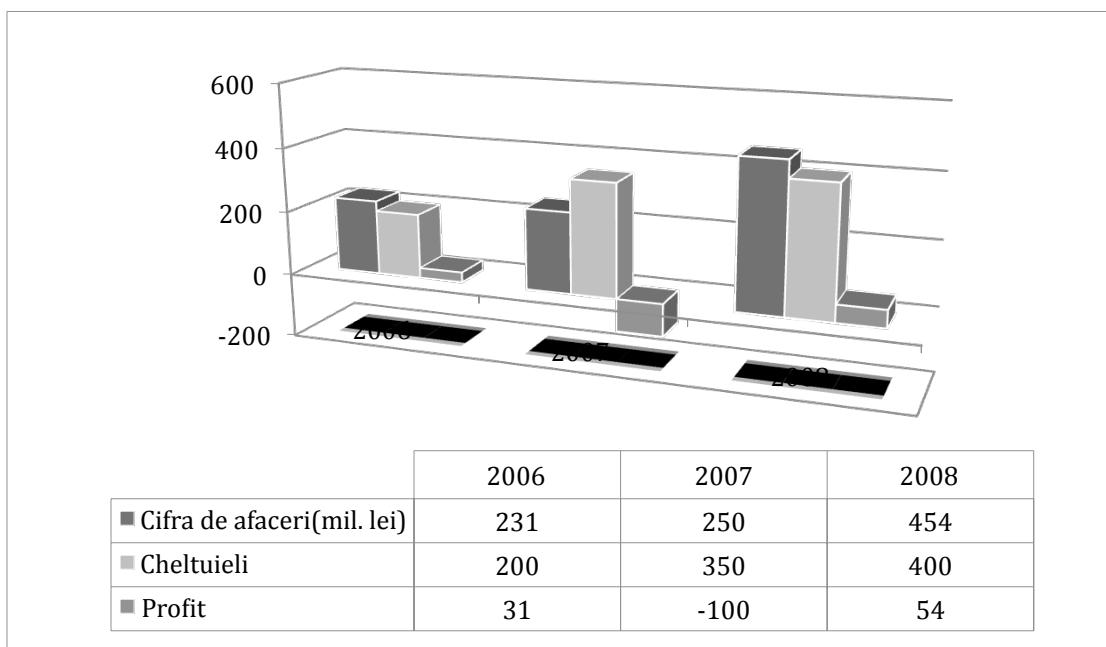


Fig. 1 Indicatori economici în perioada 2006-2008

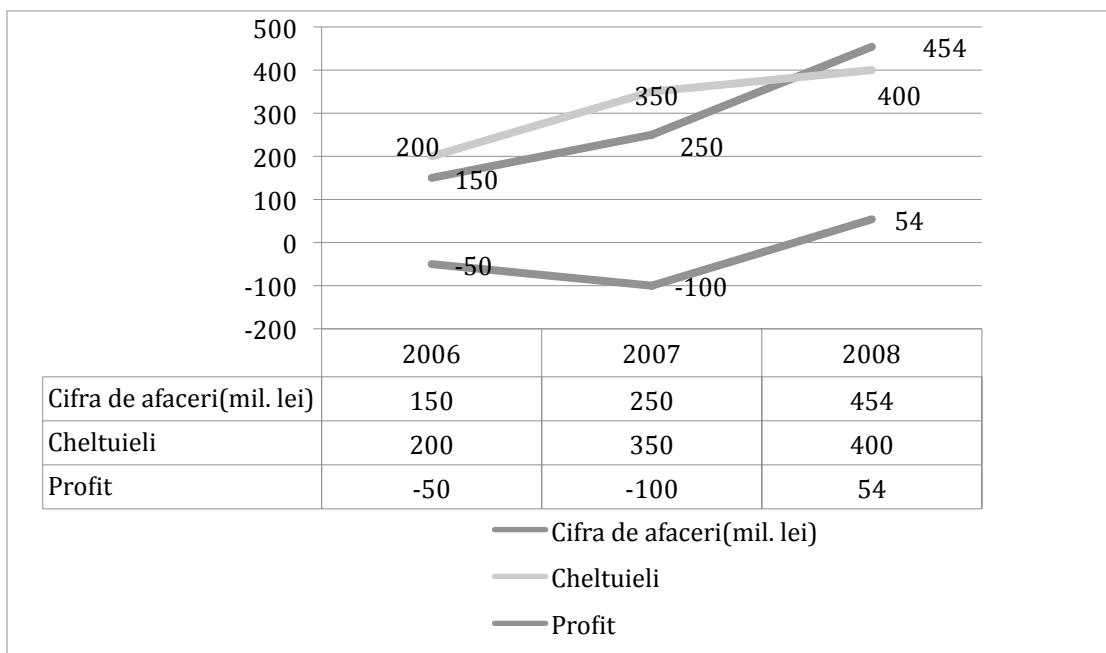


Fig. 2 Indicatori economici în perioada 2006-2008

Având în vedere faptul că anul 2006 era primul an, aceea reprezentând intrarea pe piață, cifra de afaceri (totalul vânzărilor realizate/facturate pe parcursul unui exercițiu fiscal, fără T.V.A.-ul facturat) este una în minus. Cheltuielile reprezintă salariile angajatilor, utilitățile.

### **Locul în sfera economică :**

Putem vorbi, în această categorie, despre Furnizori, Beneficiari și chiar de Concurenți.

- Furnizorii sunt autori de curs;
- Beneficiarii materialelor sunt useri înregistrați.
- Concurența este reprezentată din companiile din aceeași sferă de activitate, TIMSOFT, E-LEARNING ROMANIA, E-SCOALA ... etc.

### **1.4 Organograma societății**

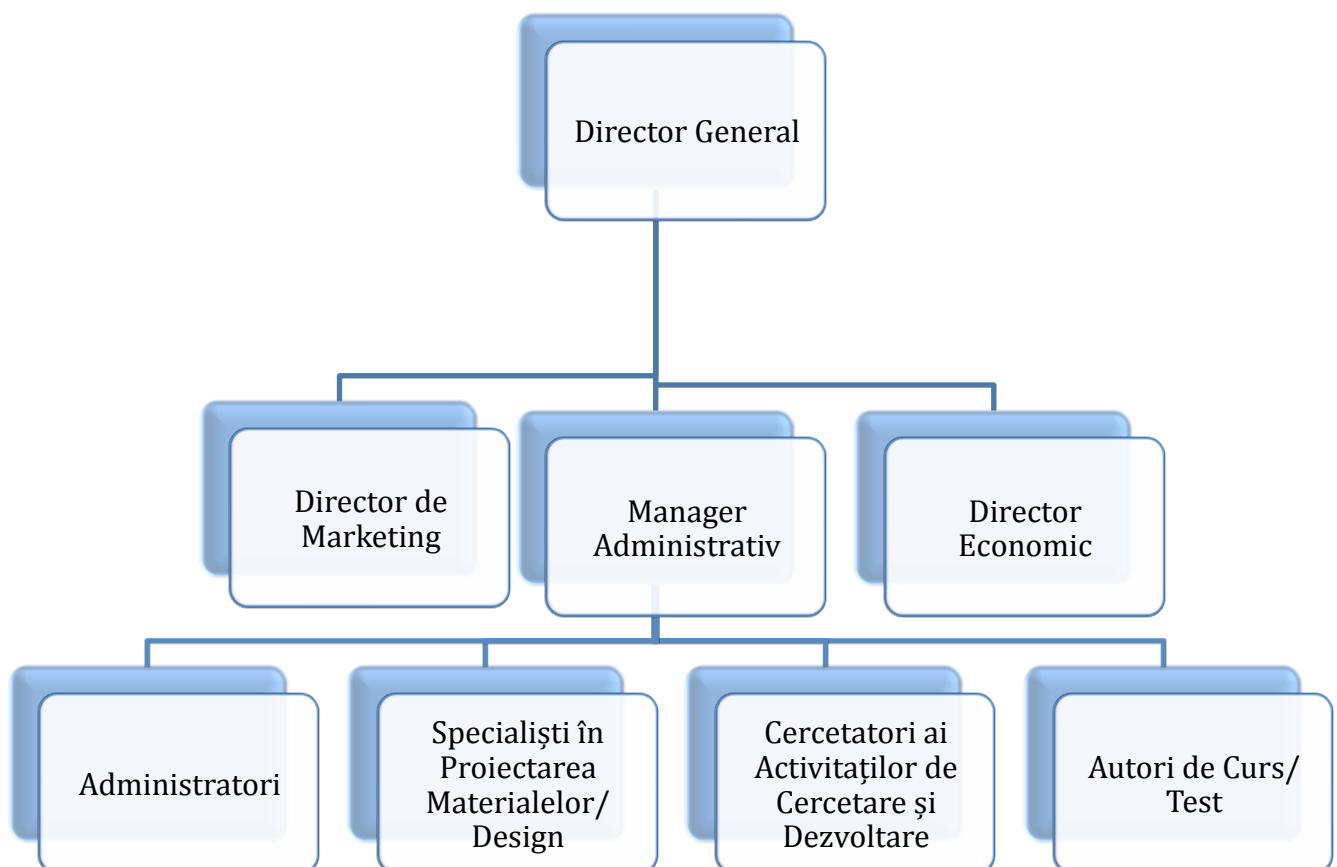


Fig. 3 Organograma societății

## **1.5 Studiul sistemului actual**

### **Directorul general :**

- Stabileste activitatea firmei în plan general și stabileste direcțiile de dezvoltare;
- Stabilăște sarcinile fiecarui serviciu;
- Asigură buna funcționare a firmei;
- Aprobă și asigură resursele necesareș
- Ia măsuri pentru îmbunătățirea continuă a sistemuluiș
- Primește rapoarte zilnice;
- Ia decizii privind angajarea/demiterea salariaților;

### **Director de marketing:**

- În lipsa directorului general îndeplinește funcția de director general adjunct cu toate atribuțiile ce decurg;
- Coordonează activitatea în unitate;
- Răspunde de punerea în aplicare a decizilor Directorului general;
- Raportează directorului general situația și evoluția sistemului informatic;
- Stabilește indicatorii de performanță pentru proiect și urmărește îndeplinirea lor;
- Este responsabil pentru procesele de relații cu clienții, proiectare și dezvoltare, furnizare de servicii;
- Răspunde de partea administrativă și economică a firmei;

### **Director economic:**

- Răspunde de organizarea și coordonarea activităților în domeniul finanțier-contabil, desfășurate în cadrul societății ;
- Este responsabil pentru procesul de aprovizionare;
- Este responsabil de controlul folosirii resurselor și cu planificarea finanțieră a societății;
- Este responsabil cu realizarea, avizarea și urmărirea contractelor încheiate între societate și terți;
- Este permanent informat cu privire la dezvoltarea și evoluția mediului finanțier;
- Întocmeste Registrele de Inventar, Notele Contabile și statele de Salarii ;
- Prezintă actele în timpul controlului fiscal.

### **Director Administrativ:**

- Coordonează activitatea firmei în plan general și stabilește direcțiile de dezvoltare;
- Primește rapoarte săptămânale privind evoluția sistemului;
- Este responsabil pentru procesele de relație cu clienții, proiectare și dezvoltare, furnizare de servicii;

- Selectează și numește coordonatorii și monitorizează performanțele acestora;
- Reprezintă societatea în relație cu partenerii, colaboratorii;0
- Administrează datele primite de la departamentele aflate în subordine;
- Răspunde de buna funcționare, folosire și stocare a sistemului.

**Coordonatori ai Activităților de Cercetare și Dezvoltare (2 angajați):**

- Propun obiective privitoare la direcțile de dezvoltare;
- Primește rapoarte zilnice privind evoluția sistemului;
- Coordonează și dezvoltă planuri de formare;
- Responsabil cu supervizarea procesului de formare.
- Responsabil cu realizarea ofertelor în domeniu formării
- Întocmesc rapoarte săptămânale privind evoluția sistemului;
- Administrează datele primite de la celelalte departamente;

**Administratori (2 angajați):**

- Asigură buna funcționare a sistemului informatic;
- Raportează directorului și coordonatorilor cu realizările și nevoile curente;
- Răspunde de punerea în aplicare a tuturor soluțiilor informaticе stabilite de conducere;
- În caz de necesitate achiziționează echipamente necesare;

**Autori de curs/test (3 angajați) :**

- Întocmesc materialele necesare sistemului;
- Raportează coordonatorilor cu realizările și nevoile curente;

**Specialiști în Proiectarea Materialelor/Desing (2 angajați) :**

- Primesc, analizează și procesează materialele pentru sistem;
- Raportează coordonatorilor cu realizările și nevoile curente;

## **1.6 Descrierea Proiectului Evolution platforma e-learning**

Proiectul Evolution platforma e-learnig va pune la dispoziția studenților și celorlalte categorii profesionale din sistemul de învățământ module de curs prin intermediul Internetului, și va fi răspunsul cel mai eficient, mai accesibil, mai flexibil și mai atractiv la necesitățile de dezvoltare profesională.

Conținutul cursurilor disponibile, bazat pe experiență profesorilor români. Cursurile se vor finaliza cu teste prin Internet. Astfel, utilizatori vor avea acces la informație de actualitate, o formă accesibilă și interactivă de educație.

În acest sens se va construi o platformă online care să întrunească atât nevoile furnizorului cât și pe cele ale cadrelor didactice care vor constitui viitorii cursanți și utilizatori ai site-ului.

## 1.7 Dotare Tehnică

**Descrierea tipului de servere ce aparțin firmei care hosteaza site-ul (Firma hostx ):**

<b>Suport tehnic online</b>		
Suport 24x7x365		Timp de răspuns redus
Secțiune "Întrebări frecvente"		Sistem HELPDESK cu acces facil
<b>Caracteristici cont</b>		
Poate fi găzduit orice tip de domeniu		Acces cu/fără www
CPanel 11		Conturi FTP
.htaccess		Statistică detaliată (Awstats, Webalizer)
Backup zilnic		File Manager
SSI		Protecție Hotlink
Pagini 404		Parcare domenii
<b>Caracteristici email</b>		
Acces POP3		Server IMAP / SMTP / POP3
Autorespondere		Email forwarders nelimitat
Filtre SPAM		PGP Mail (email criptat)
Sendmail		Catch All Email
Webmail (Horde, SquirrelMail, RoundCube)		
<b>Caracteristici dezvoltare web</b>		
PHP 5.2.x		Zend Optimizer™
Perl (5.8.x)		Module Perl preinstalate
Propriul CGI-BIN		Instalare module Perl la cerere
MySQL 5.x		PostgreSQL 8.3.x
Real Video Streaming (via http)		Real Audio Streaming (via http)
PEAR		Apache handler
<b>Caracteristici server</b>		
minim 2 x Dual Core Xeon		CentOS
4 GB RAM		RAID, SCSI/SATA, 7200/10k/15k rpm
On/Off server backup		
<b>Scripturi preinstalate</b>		
PHPBB Forum		PHPMyAdmin
Countdown		Simple CGI Wrapper
Entropy Search		Entropy Banner
Counter		Random HTML Generator
PHP MY Chat Room		Guest Book
PHPPGAdmin		Invision Board
Agora Shopping Cart		OS Commerce

<b>Garanții clienți</b>		
99,5% uptime		Politică strictă anti SPAM
Backup zilnic		
<b>Centre de date și rețea</b>		
<b>Colorado, SUA</b>		Supraveghere 24x7
5 conexiuni OC-48 și una OC-12		Conexiuni redundante
Sisteme de stingere a incendiilor		Backup generator
<b>Bucuresti, Romania</b>		Supraveghere 24x7
2x155 Mbps extern, 2 cai de acces		1 Gbps RONIX (metropolitan)
Sisteme de stingere a incendiilor		Backup generator

## **1.8 Analiza critică a sistemului existent și directivele de perfecționare**

Principalele deficiențe identificate în cadrul societății sunt urmatoarele:

1. autori redactează cursurile manual și procesarea lor necesită o durată mai indelungată de timp;
2. crearea listelor rapoartelor durează din cauza conținutului mare a rapoartelor;
3. interfața sistemului informatic are nevoie de îmbunătățiri;
4. evidența rapoartelor și a listelor de cereri se pierde din cauza nefolosirii unui program pt evidența lor.
5. Nu există o rețea intranet care să permită transferul on-line de informații;
6. Echipamentele din societate necesită un upgrade.

Pentru a remedia deficiențele se impun realizarea urmatoarelor activități:

1. Crearea unei rețele intranet pentru transferul on-line de date;
2. Îmbunătățirea echipamentelor în societate;
3. Angajarea de specialiști pentru îmbunătățirea interfeței;
4. Crearea unor programe noi pentru evidența și crearea rapoartelor și listelor de cereri;
5. Echiparea biroului autorilor cu echipamente necesare redactării mai rapide a materialelor.

## 1.9 Fluxul informational

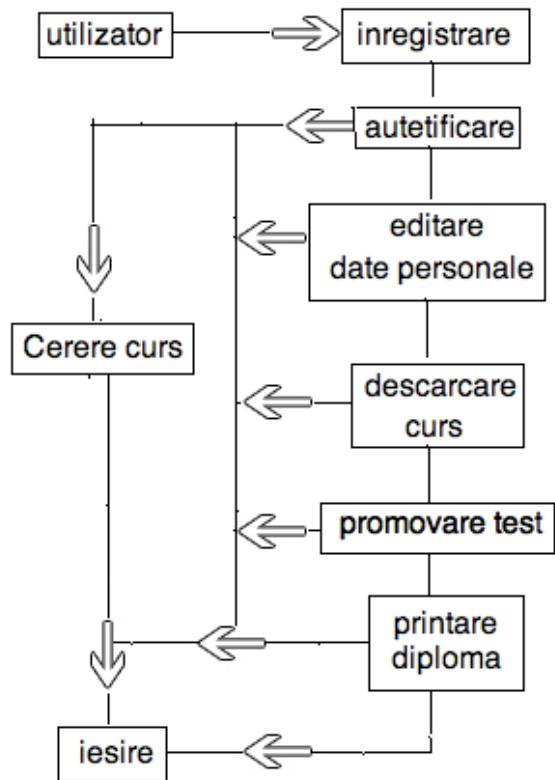


Fig. 4 Flux activitate utilizator

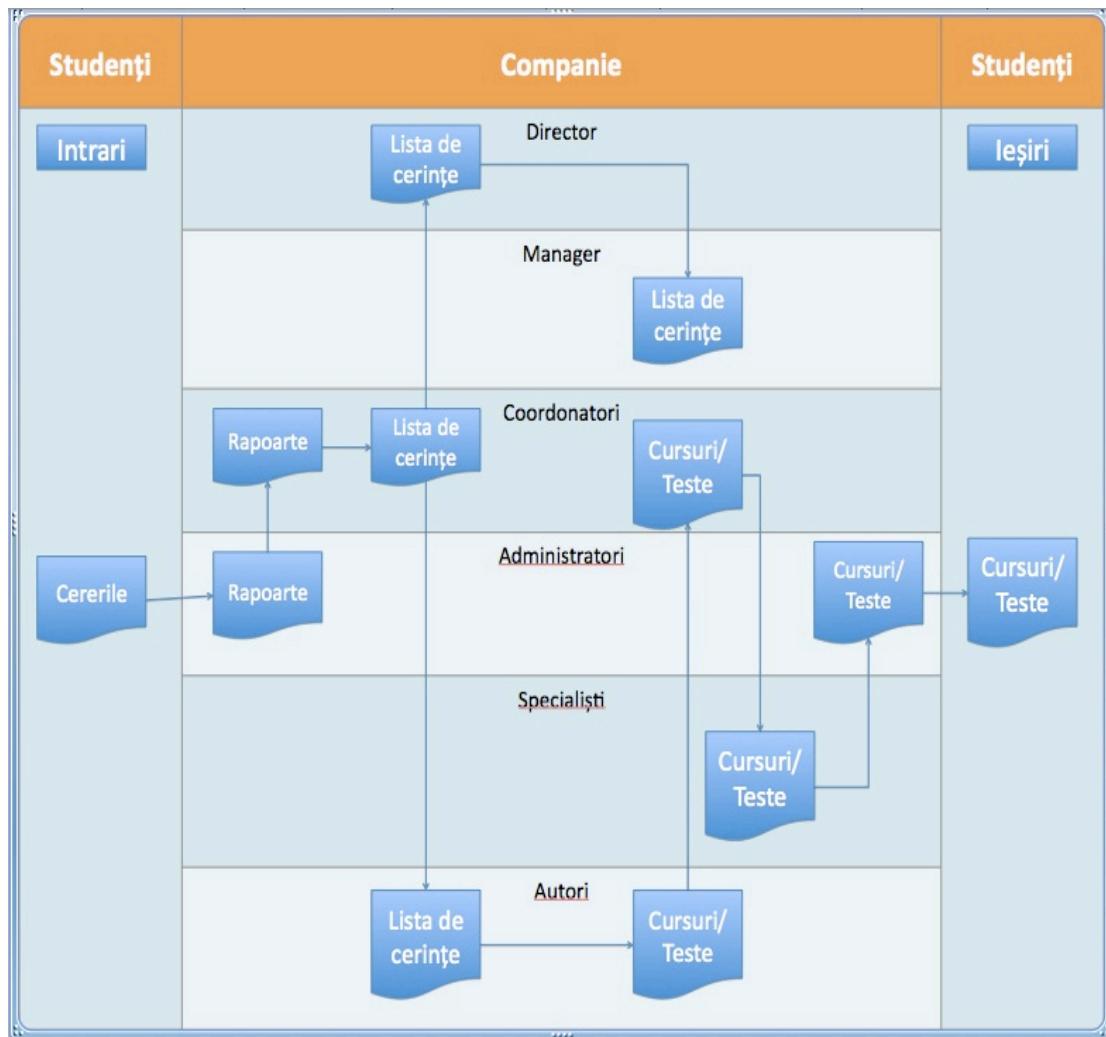


Fig. 4 Fluxul documentelor în cadrul societății

În fluxul documentelor se identifică urmatoarele proceduri:

1. Cererile studenților sunt înregistrate de administratori sistemului;
2. Cererile sunt raportate coordonatorilor;
3. Coordonatori realizază o listă de cerințe înmanate autorilor de curs/test și directorului general;
4. Autori întocmesc cursurile/testele;
5. Cursurile/testele revin coordonatorilor pentru a fi aprobate sau dezaprobată;
6. Cursurile/testele aprobate sunt analizate și procesate de specialiști;
7. Upload-area cursurilor/testelor se face de administratori.

## **1.10 Prezentarea documentelor**

**Fişa de înscriere** – este necesară pentru înscrierea utilizatorilor pe site.

**Cursuri** – cuprind materialele cerute de studenți/user-i, redacatate de autori.

**Diplomă** – devine disponibilă pentru vizualizare și tipărire pentru utilizatori care promovează teste.

**Rapoarte** - este documentul întocmit de administratori sistemului pentru a fi înmanat coordonatorilor și conține informații cu privire la cererile studenților;

**Cerere de curs** – sunt cererile studenților cu privire la cursuri.

**Raport cu activitatea utilizatorului** – fișiere cu informații legate de activitatea studenților;

**Raportul cursurilor** – fișiere cu informații despre cursuri;

**Raportul testelor** – fișiere cu informații despre teste;

**Raportul notelor** - fișiere cu informații despre note;

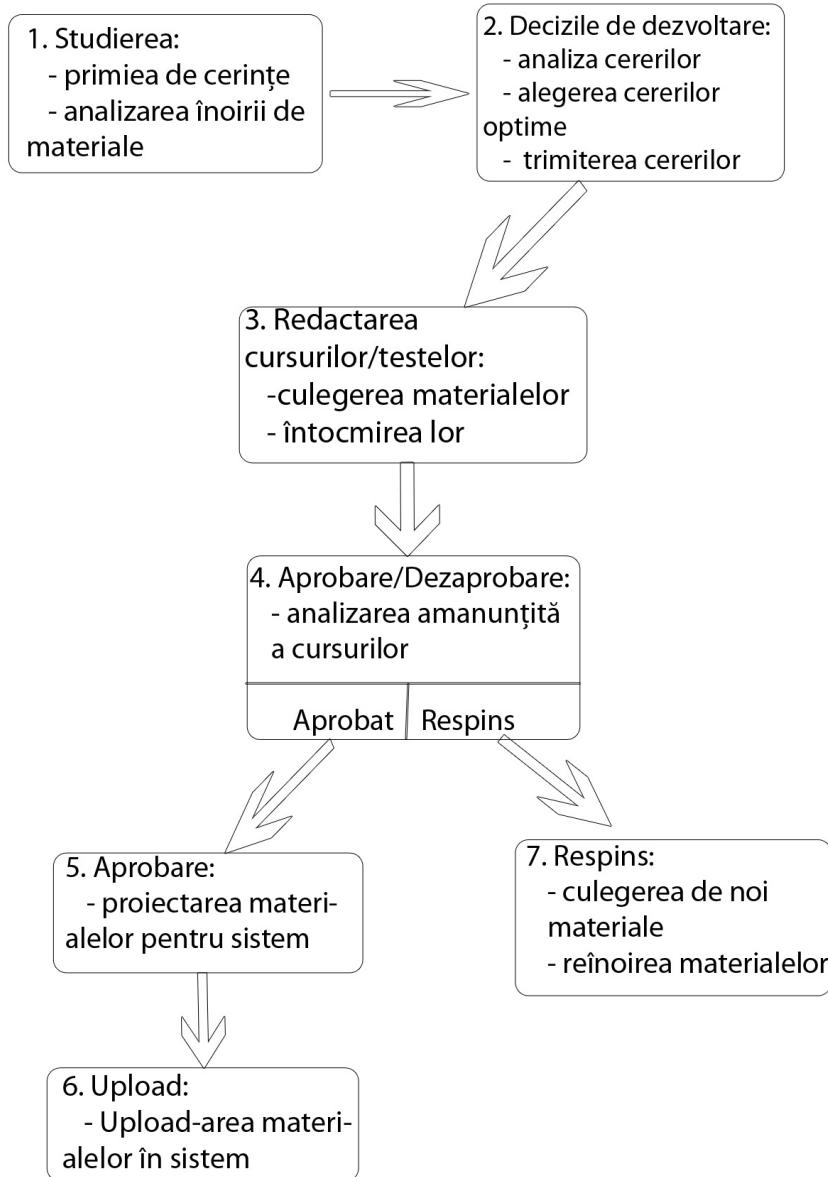


Fig. 5 Modelul prelucrărilor de cursuri/teste

#### **OPERATIUA 1 (Studierea) :**

- administratori raportează cererile primite de la useri și se analizează materialele care trebuie înnoite.

#### **OPERATIUA 2 (Decizia de dezvoltare) :**

- coordonatorii analizează cererile de la useri, se aleg cele optime în funcție de criteriul de dezvoltare ;

**OPERATIA 3** (Redactarea curs/test):

- autori strâng materialele necesare pentru redactarea cursurilor/testelor după care sunt înaintate coordonatorilor pentru analizarea lor;

**OPERATIA 4** (Aprobarea/Dezaprobaarea) :

- după analizarea amănunțită a cursurilor ele sunt aprobate sau respinse, cursurile aprobate sunt trimise specialistilor și cele respinse sunt returnate autorilor;

**OPERATIA 5** (Redactarea) :

- cursurile aprobate sunt redactate în format electronic după care sunt trimise administratorilor;

**OPERATIA 6** (Upload) :

- upload-area cursurilor pe site de administratori sau profesori;

**OPERATIA 7** (Respins):

- cursurile respinse sunt trimise autorilor pentru a fi reîntocmite.

## 1.1 Declararea și inițializarea pointerilor

Fie **TIP** un tip de date oarecare în limbajul C (inclusiv **void**). Declararea **TIP\*** *nume*; este o declarație de pointer. **TIP\*** este un nou tip de date denumit *pointer spre TIP*, iar *nume* este o variabilă de tipul *pointer spre TIP*.

**Exemple:**

- `int* n;` ⇒ *n* este o variabilă de tip *pointer spre întreg*;
- `struct complex {a,b:real;}* x;` ⇒ *x* este o variabilă de tip *pointer spre o structură de tipul complex*;
- `void* p;` ⇒ *p* este o variabilă de tip *pointer spre void*; *p* poate primi ca valoare adresa unei zone de memorie de orice tip.

Dacă **TIP** este un tip oarecare (mai puțin **void**) atunci tipul **TIP\*** este adresa unei zone de memorie de un tip cunoscut. Operațiile care se pot efectua asupra zonei respective de memorie sunt definite de tipul acesta. Dacă **TIP** este **void**, atunci **TIP\*** este adresa unei zone de memorie de tip necunoscut. Deoarece nu se cunoaște tipul zonei de memorie, nu sunt definite operațiile care se pot efectua asupra ei.

Pentru pointerii din exemplele anterioare se rezervă în memoria principală (în segmentul de date) câte o zonă de 4B în care se va memora o adresă (sub forma *segment:offset*).

Când variabila *nume* nu este inițializată prin declarare, ea primește implicit valoarea **NULL**. La execuție, poate primi ca valoare adresa unei variabile numai de tipul **TIP**. Dacă **TIP** este **void**, atunci *nume* poate primi adresa oricărei variabile, de orice tip.

**Exemple:**

```
int* nume; int a; float b;
nume = &a;    => este o atribuire corectă; nume are ca valoare adresa variabilei a.
nume = &b;    => este o atribuire incorectă; nume poate primi ca valoare doar
                  adresa unei variabile întregi.

void* nume;
int a; float b;
nume = &a;
nume = &b;    => ambele atribuirii sunt corecte; nume poate primi ca valoare
                  adresa oricărei variabile, de orice tip.
```

Inițializarea pointerilor se poate realiza ca în exemplul precedent sau, ca și pentru celelalte variabile, la declarare, astfel:

```
int a; int* nume=&a;
```

Se observă folosirea operatorului de referințiere & pentru a crea o referință către variabila *a*. La alocarea dinamică a memoriei se folosește o altă metodă pentru inițializarea unui pointer. Operatorul de dereferințiere se utilizează atât

Fig. 4 Model de curs

Curs	User
Javascript	Demo
Cookies	Gabby

Fig.5 Model de lista de cereri

# CAPITOLUL II :

## Proiectarea De Ansamblu

## A Noului Sistem

## **2.1 Definirea obiectivelor :**

Obiectivele sistemului informatic care urmează a fi implementat în cadrul societății vor fi stabilite în urma unei investigații realiste a sistemului existent deja în societate. Acest sistem informatic va trebui să țină seama (și o buna comunicare) de factorii de decizie ale tuturor nivelelor ierarhice ale societății.

In urma acestei investigații complexe, se vor identifica urmatoarele lucruri :

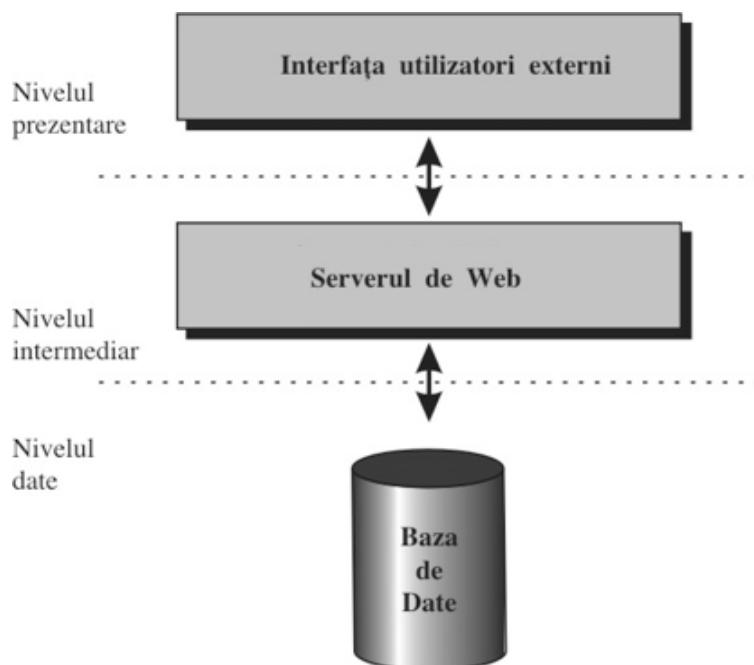
- Punctele slabe ale societății (în general, punctele slabe ale sistemului existent) ;
- Punctele forte ale societății (sau ale sistemului deja existent) ;
- Rezervele interne ;
- Eventuale necorelari (sau contradictii) care apar și care ar putea fi diminuate (sau eliminate în totalitate) prin implementarea noului sistem informatic ;

Principalul obiectiv al societății este acela de a proiecta și realiza un sistem informatic prin intermediul cărora materialele să fie mai usor procesate de utilizatori și să fie capabil să permită automatizarea și eficientizarea următoarelor activități :

- O buna evidență a userilor sistemului, prin crearea unei baze de date care să retină fiecare user, precum și informațiile legate de acesta ;
- O buna evidență a cererilor facute de studenți prin intermediul forum-ului;
- O buna evidență a materialelor;
- Alte informații solicitate (optionale).

## **2.2 Arhitectura sistemului**

Arhitectura sistemului este pe trei nivele după cum se vede în figura următoare:



### Nivelul 1 (prezentare)

- reprezintă nivelul interfețelor utilizator.

### Nivelul 2 (intermediar)

- reprezintă nivelul serverelor de aplicații. Aplicațiile realizate în medii diferite utilizând tehnologii diferite vor fi executate de către servere diferite.
- face legătura între interfața utilizator și baza de date. Cererile clientului sunt interpretate și trimise către server-ul de baze de date de la care preia apoi răspunsurile.

### Nivelul 3 (date)

- reprezintă baza de date.

## 2.3 Estimarea necesarului de resurse

Categoriile de resurse utilizate în cadrul sistemului informatic sunt:

*Resurse umane:*

- personalul de specialitate necesar pentru realizarea și perfectionarea sistemului informatic.

*Resurse hardware:*

- constă în achiziționarea de echipamente noi care să satisfacă performanțele dorite de societarea beneficiară.

*Resurse software:*

- se recomandă îmbunătățirea sistemului de operare utilizat de firmă care este învechit și nu mai poate îndeplini sarcinile sistemului.

*Resurse financiare:*

- firma va avea nevoie de fonduri suplimentare pentru implementarea noului sistem informatic;
- estimativ, valoarea investiției va ajunge la circa 10.000 de Euro.

## 2.4 Alegerea tehnologiilor de prelucrare :

În funcție de resursele alocate și a modului în care aceste resurse alocate sunt utilizate pentru satisfacerea cerințelor sistemului informatic, ținând cont de performanțele tehnice și funktionale ale sistemului, s-a optat pentru o tehnologie în timp real, care să ofere informații corecte și aproape instantaneu.

## 2.5 Planificarea realizării sistemului

La baza realizării acestei etape, din cadrul proiectării de ansamblu a sistemului informatic, stă principiul proiectării și implementării eșalonate.

Prin esalonare, se înțelege ordinea în care componentele sistemului sunt abordate, începând de la proiectarea de detaliu și terminând cu exploatarea propriu-zisă a sistemului.

Ordinea de realizare și abordare a componentelor sistemului informatic este următoarea :

- Realizarea modulului administratori (care va permite gestiunea activității userilor, întocmirea rapoarte, vizualizarea datelor în diferite formate, actualizarea datelor etc) -> aprox. 10 zile ;
- Realizarea modulului useri (care va conține vizualizarea acestora în diferite formate, modificarea, ștergerea sau cautarea datelor userilor etc) -> aprox. 10 zile ;
- Realizarea modulelor cursurilor și testelor -> aprox. 5 zile ;
- Proiectarea și realizarea situațiilor aplicațiilor -> aprox 8 zile ;
- Timp rezervat altor situații sau obiective optionale -> aprox 5 zile.

## 2.6 Definirea sistemului de codificare

Sistemul de codificare, în cazul de față, va fi bazat pe urmatoarele tehnologii / limbaje de programare :

- PHP ;
- HTML ;
- MySQL.

PHP :

PHP este un limbaj de programare "HTML-embedded". Majoritatea programului (sintaxe / instrucțiuni / codificare) este "împrumutat" din programe precum C, Java și Perl, împreună cu unele trasaturi specifice PHP-ului. Scopul limbajului, este acela de a permite web-designer-ilor să codeze pagini dinamice, într-un mod rapid. Numele limbajului de programare vine de la **PHP: Hypertext Preprocessor**.

HTML :

HTML este prescurtarea de la **HyperText Markup Language**. El este limbajul predominant din paginile web de astăzi. El ajută la descrierea unei informații bazate pe text într-un document (denotând anumite texte drept link-uri, paragrafe, antet, liste ... etc). HTML este un format text proiectat pentru a putea fi citit și editat de oameni utilizând un editor de text simplu. Totuși scrierea și modificarea paginilor în acest fel solicită cunoștințe solide de HTML și este consumatoare de timp.

MySQL :

MySQL este un sistem de gestiune a bazelor de date relațional, produs de compania suedeză MySQL AB și distribuit sub Licenta Publică Generală (GNU). El este cel mai popular S.G.B.D. (Sistem de Gestie a Bazelor de Date) open-

source la ora actuală, fiind o componentă cheie a stivei LAMP (Linux, Apache, MySQL, PHP). Deși este folosit foarte des împreună cu limbajul de programare PHP, cu MySQL se pot construi aplicații în orice limbaj major. Există multe scheme API disponibile pentru MySQL ce permit scrierea aplicațiilor în numeroase limbaje de programare pentru accesarea bazelor de date MySQL, cum ar fi: C, C++, C#, Borland Delphi, Java, Perl, PHP, Python, FreeBasic, etc.

## 2.7 Proiectarea ieșirilor

Prin „iesirile” unui sistem înțelegem totalitatea informațiilor furnizate de acestea.

- fișa de înscriere;
- raport privind cursuri;
- raport privind activitatea utilizatorilor;
- raport privind notele;
- raport privind teste;
- diplome pentru utilizatori;
- raport privind cererile.

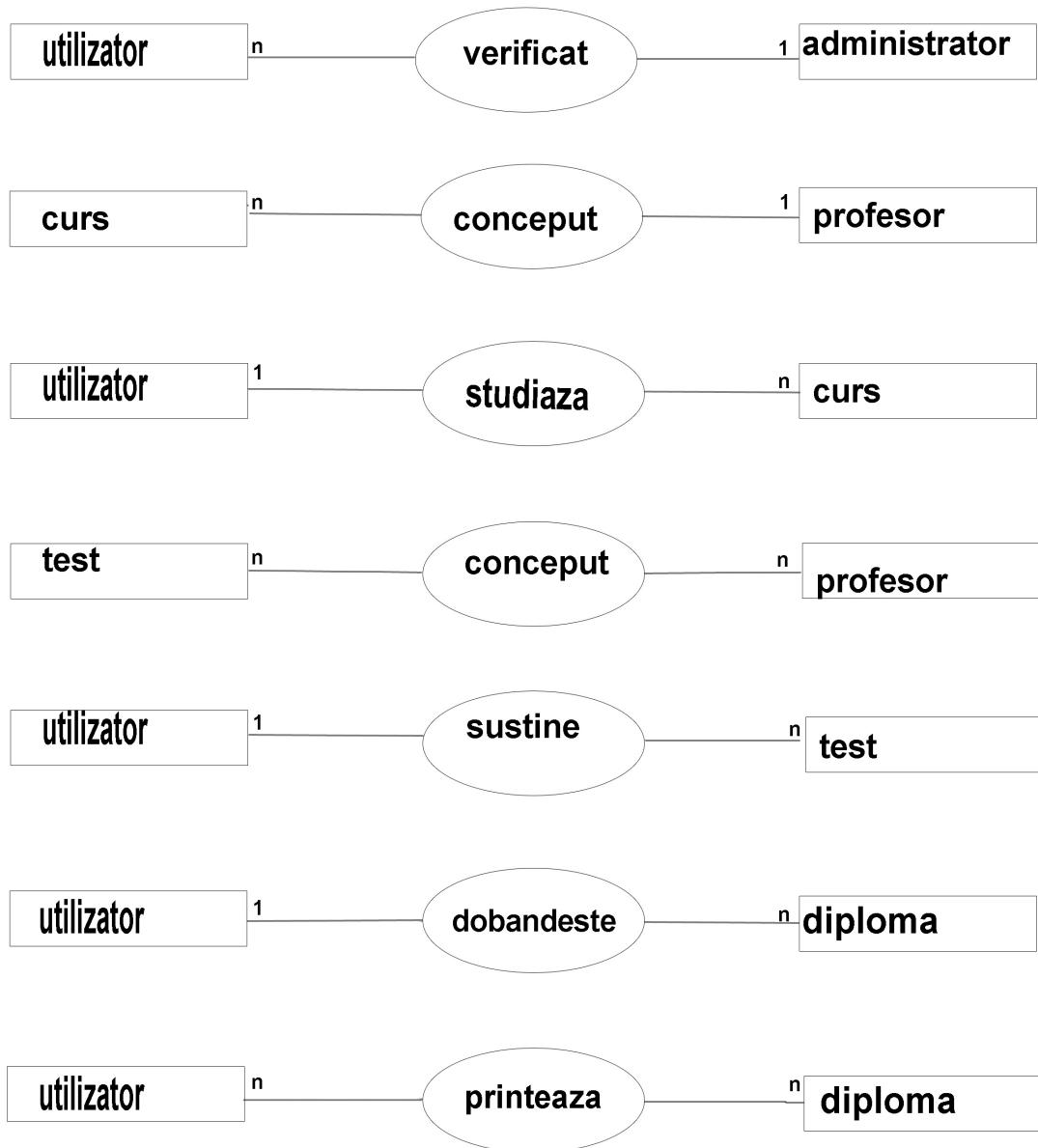
## 2.8 Proiectarea intrărilor

Intrările sistemului sunt în principal formularele de introducere a datelor precum:

- fișa de înscriere;
- cererile studentilor;
- teste;
- alte surse.

## 2.9 Diagrama Entitate – Asociere

Modelul entitate-asociere (MAE) nu este un simplu model teoreti de reprezentare a datelor, ci reprezintă un model neformalizat pentru reprezentarea unor fenomene, a unui sistem din lumea reală. El devine astfel un instrument de lucru și de comunicare între proiectanții unei aplicații informative. Scopul lui este de a stabili entitățile de date și realitățile care există între acestea. Pentru reprezentarea grafică a modelului entitate-asociere se utilizează adeseori Diagrama entitate-asocere(DEA):



**Fig. 7. Diagrama entitate-asociere**

# CAPITOLUL III:

## Proiectarea de detaliu a noului sistem

### **3.1 Definirea aplicației informatic**

Aplicația reprezintă un mod de testare online bazată pe o serie de date în care se înregistrează informațiile referitoare la utilizatori, teste, profesori, administratori, etc. Utilizatorul se înregistrează pentru a putea susține teste și acumularea de puncte necesare absolvirii cursului.

### **3.2 Proiectarea logică și fizică a intrărilor**

Prin intrările unui sistem informatic se înțelege totalitatea datelor primare necesare obținerii informațiilor de intrare ale sistemului. Datele primare reflectă starea și dinamica fenomenelor și procesele economice din unitatea economică. Datele primare sunt necesare pentru crearea și actualizarea bazelor de date, pentru obținerea de situații de ieșire și pentru realizarea de rapoarte cu ieșiri.

- documentul principal de intrare este reprezentat de **formularul de înscrisere** a utilizatorilor;
- alte documente de intrare pot fi considerate **formularul de adaugare cursuri**, **formularul de adaugare de teste** și **formularul de cerere a cursurilor**.

### **3.3 Proiectarea logică și fizică a ieșirilor**

Prin ieșirile unui subsistem informatic se va înțelege totalitatea informațiilor furnizate de acesta beneficiarilor interni și externi, respectiv rapoarte, note de informare-rapoarte, etc.

Principalele ieșiri pe care le va furniza produsul sunt următoarele:

- raport de activitate al studenților;
- raport a cursurilor active.
- formular de căutare a studenților în bază de date(după numele de familie, numele mic sau username);
- diplomă de absolvire pentru cursanții care promovează testele;

### **3.4 Proiectarea fizică și logică a bazei de date**

Pentru proiectarea logică și fizică a bazei de date s-au avut în vedere următorii pași:

- Alegerea sistemului de gestiune a bazei de date – conform subcapitolului 2.6 privind alegerea tehnologiei de prelucrare, referitor la sistemul de gestiune al bazei de date, vom alege sistemul de baze de date MySQL care implică o implementare ușoară și optimă pentru această aplicație, dar nu în ultimul rând ținând și cont de următoarele aspecte cum ar fi timpul de răspuns, confidențialitatea datelor și securitatea datelor pentru acest tip de baza de date

- Proiectarea schemei conceptuale – această schemă conceptuală impune realizarea următoarelor activități:

- Stabilirea colecțiilor de date și definirea detaliată a conținutului acestora;

- Determinarea legăturilor între colecțiile de date și a modului de

reprezentare a acestora în cadrul schemei conceptuale;

- Testarea schemei conceptuale și îmbunătățirea acesteia, dacă este cazul;

- Descrierea schemei conceptuale în limbajul de descriere a datelor MySQL și încărcarea acestor descrieri în baza de date;

· Încărcarea datelor în baza de date – conținutul acestei etape este relativ simplu, dar volumul de muncă este foarte mare și dependent de mărimea colecțiilor de date, în baza de date trebuie încărcate numai date corecte și un număr minim de parcurgeri ale ciclului validare-corectare pentru asigurarea funcționării aplicației

· Exploatarea bazei de date de către utilizatorii finali – are scopul satisfacerii cerințelor informaționale ale acestora. Sistemul de gestionare a bazei de date oferă utilizatorilor instrumente facile de exploatare cum ar fi limbajele de manipulare a datelor în vederea corectării eventualelor erori ce pot apărea pe parcurs.

Colecția de date au fost stabilită pentru a satisfacerea cerințele sistemului. Principalele entități din baza de date sunt:

UTILIZATORI (id, iuser, pass, nume, prenume, email, dn\_an, dn\_luna, dn\_zi, sex, adresă, țară, județ, localitate, cod\_poștal, tel, date1, status)

PROF (id, iuser, pass, nume, prenume, email, dn\_an, dn\_luna, dn\_zi, sex, adresă, țară, județ, localitate, cod\_poștal, tel, date1, status)

DN\_AN (id, val)

DN\_LUNA (id, val)

DN\_ZI (id, val)

JUDEȚE (id, val)

CURSURI (id, id\_user, iuser, denumire, descriere, ore\_alocate, data\_test, data1, link)

ORASE (id, val)

TARĂ (iso, name, printable\_name, iso3, numecode)

INTREBARITEST(id, q, intrebari, raspuns1, raspuns2, raspuns3, raspuns 4, rascor)

NOTE (id, iuser, test, nota)

SHOUTBOX (id, name, email, massage, date, ip)

TEST (idtest, test, descriere, date, iuser)

USERCURS (idusercurs, iuser, idcurs, dateinscriere)

USERCURSTEST (id, idusercurs,data\_test)

La nivelul logic entitățile sunt reprezentate astfel:

Tabela UTILIZATORI:

COD	TIP	LUNGIME	DESCRIERE
Id	CHAR	50	Cod student
Iuser	CHAR	50	Userul
Pass	CHAR	50	Parola
Nume	CHAR	50	Nume
Prenume	CHAR	50	Prenume
Email	CHAR	50	Email
dn_an	CHAR	50	Anul
dn_luna	CHAR	50	Luna
dn_zi	CHAR	50	Ziua
sex	CHAR	50	Sexul
adresă	CHAR	50	Adresa
țară	CHAR	50	Tara
județ	CHAR	50	Judetul
localitate	CHAR	50	Localitatea
cod_poștal	CHAR	50	Codul postal
tel	CHAR	50	Telefon
date1	CHAR	11	Data inregistrari
status	CHAR	50	-

Tabela PROF:

COD	TIP	LUNGIME	DESCRIERE
Id	CHAR	50	Cod profesor
Iuser	CHAR	50	Userul
Pass	CHAR	50	Parola
Nume	CHAR	50	Nume
Prenume	CHAR	50	Prenume
Email	CHAR	50	Email
dn_an	CHAR	50	Anul
dn_luna	CHAR	50	Luna
dn_zi	CHAR	50	Ziua
sex	CHAR	50	Sexul
adresă	CHAR	50	Adresa
țară	CHAR	50	Tara
județ	CHAR	50	Judetul
localitate	CHAR	50	Localitatea
cod_poștal	CHAR	50	Codul postal
tel	CHAR	50	Telefon
date1	CHAR	11	Data inregistrari
status	CHAR	50	-

Tabela DN\_AN:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	Id
val	Char	50	Valoare

Tabela DN\_LUNA:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	Id
val	Char	50	Valoare

Tabela DN\_ZI:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	Id
val	Char	50	Valoare

Tabela JUDEȚE:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	Id
val	Char	50	Valoare

Tabela CURSURI:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	Id
Id_user	Char	50	Id-ul userului
Iuser	Char	50	Userul
Denumire	Char	50	Denumire
Descriere	Char	50	Descriere
Ore_alocate	Char	50	Ore alocate
Data_test	Char	50	Data test
Data1	Char	11	Data introduceri

Tabela ORASE:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	Id
val	Char	50	Valoare

Tabela TARA:

COD	TIP	LUNGIME	DESCRIERE
Iso	Char	50	Prefix scurt
Name	Char	50	Nume
Printable_name	Char	50	Nume
Iso3	Char	50	Prefix lung
Nume_cod	char	50	Cod

Tabela INTREBARITEST:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	id
Q	Int	11	id test
Intrebari	Char	255	intrebarea
Raspuns1	Char	255	raspunsul1

Raspuns2	char	255	raspunsul2
Raspuns3	Char	255	raspunsul3
Raspuns4	Char	255	raspunsul4
rascor	Char	255	raspunsul corect

Tabela NOTE:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	id
Iuser	Char	255	userul
Test	char	255	numarul testului
nota	Char	255	nota

Tabela SHOUTBOX:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	id
Name	Char	255	userul
Email	char	255	email
Massage	Char	255	mesajul
Date	Date	255	data
Ip	char	255	ip-ul

Tabela TEST :

COD	TIP	LUNGIME	DESCRIERE
Idtest	Int	11	id
Test	Char	255	numarul testului
Descriere	char	255	descrierea
Data	Char	255	data test
iuser	Char	255	utilizator

Tabela USERCURS:

COD	TIP	LUNGIME	DESCRIERE
Idusercurs	Int	11	id
Iuser	Char	255	userul
idtest	char	255	numarul testului
datainscriere	Char	255	data inscriere

Tabela USERCURSTEST:

COD	TIP	LUNGIME	DESCRIERE
Id	Int	11	id
Idusercurs	Char	255	id-ul userul
Data_test	char	255	data test

### 3.5 Schema de sistem a bazei de date

În cazul aplicației de față, schema de sistem a aplicației constă în lista bazelor de date ce conțin colecțiile de date necesare pentru funcționarea sistemului on-line. Aceste colecții de date sunt:

Nr.	Nume table	Cheie primara
1	Utilizatori	Id
2	Prof	Id
3	Cursuri	Id
4	Intrebaritest	Id
5	Note	Id

### 3.6 Schema fizica a bazei de date

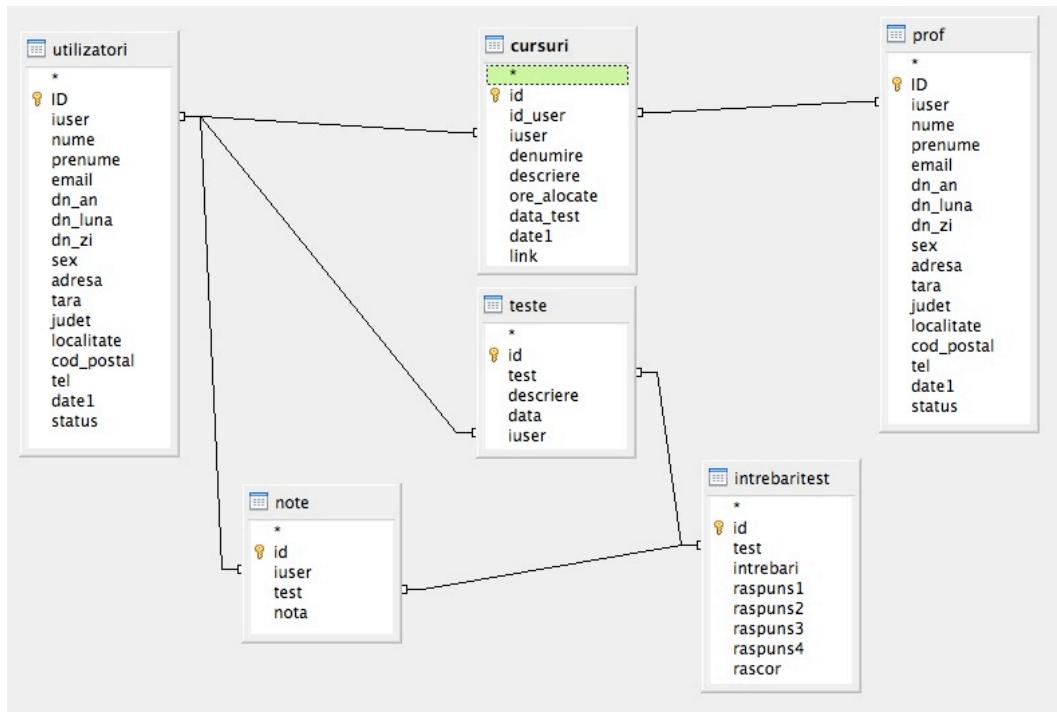


Fig. 8 Schema fizică a bazei de date

### 3.7 Schema conceptuala a bazei de date

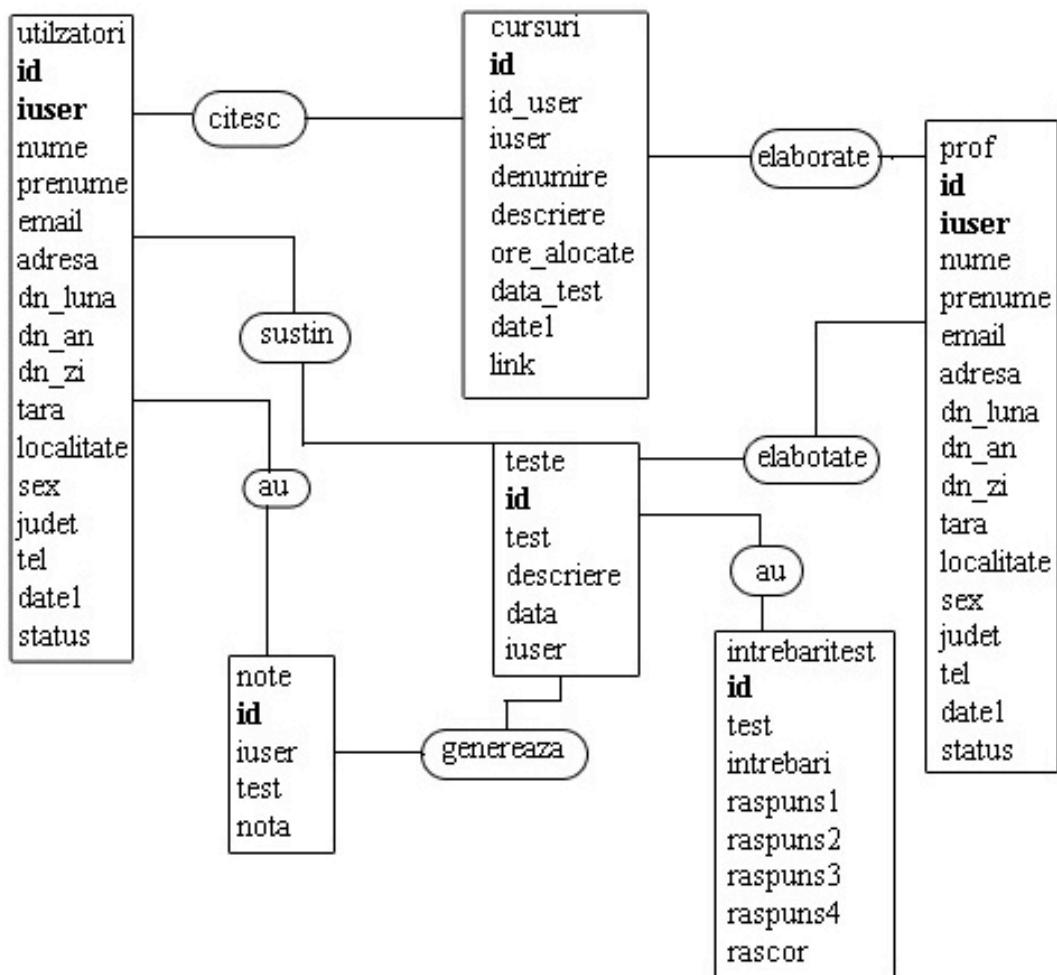
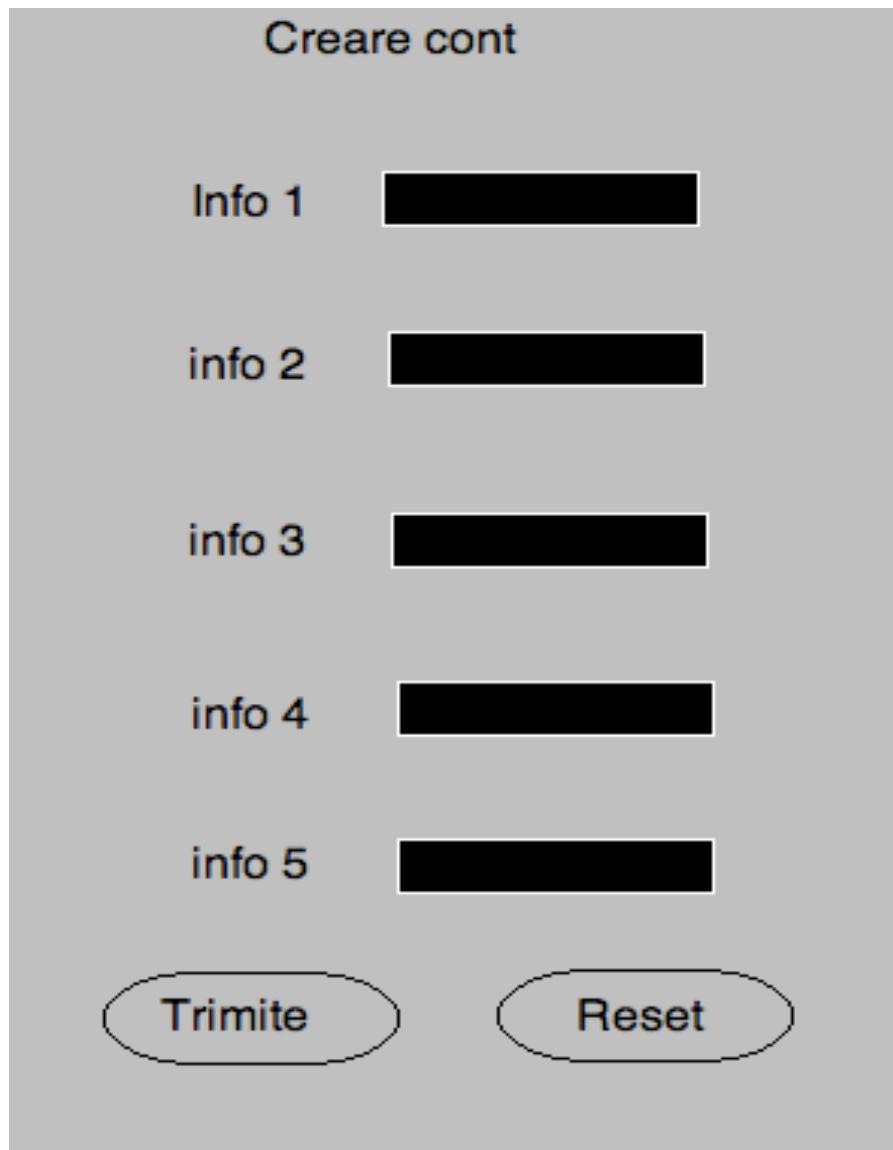


Fig. 9 Schema conceptuala

### **3.8 Proiectarea machetelor videoformat**

#### **3.8.1 Machete de intrare**

Fișă de inscriere a utilizatorului:



The image shows a wireframe or mockup of a user registration form titled "Creare cont". The form consists of five input fields labeled "Info 1", "Info 2", "Info 3", "Info 4", and "Info 5", each followed by a blacked-out rectangular placeholder. At the bottom, there are two rounded rectangular buttons labeled "Trimite" and "Reset".

Fig. 10 Macheta creare cont

Formularul pentru introducerea cursurilor în baza de date:

**Introducere curs**

Info 1

info 2

info 3

info 4

**Trimite** **Reset**

The form consists of a light gray rectangular background. At the top center, the text "Introducere curs" is displayed in bold. Below it, there are four horizontal rows. Each row contains a label on the left ("Info 1", "Info 2", "Info 3", "Info 4") followed by a black rectangular input field on the right. At the bottom center, there are two oval-shaped buttons with the text "Trimite" and "Reset" respectively.

Fig. 11 Macheta introducere curs

Formularul de cerere de curs:

**Creare curs**

Info 1

info 2

info 3

**Trimite** **Reset**

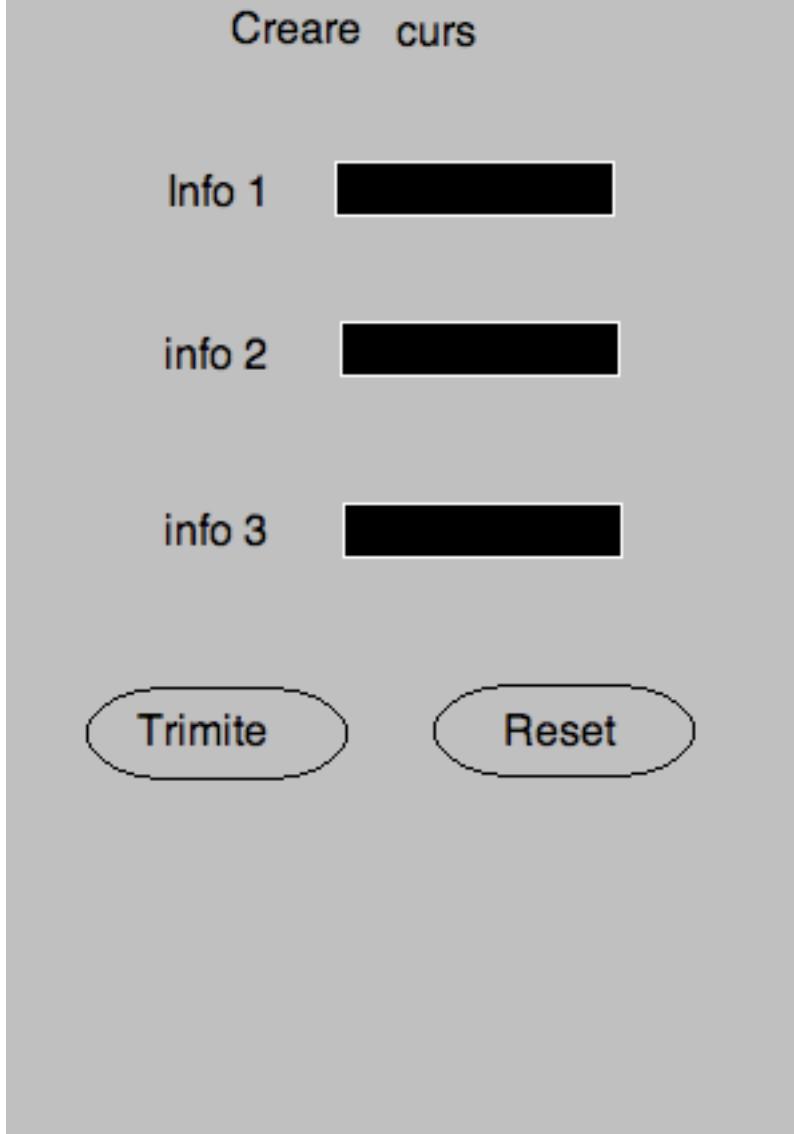


Fig. 12 Macheta cerere de curs

Formular pentru introducere de întrebări pentru teste:

**introducere intrebare**

Intrebare [REDACTAT]

raspuns 1 [REDACTAT]

raspuns 2 [REDACTAT]

raspuns 3 [REDACTAT]

raspuns 4 [REDACTAT]

raspuns corect [REDACTAT]

...

**Trimite**      **Reset**

Fig. 13 Macheta pentru introducere

Formular de cautare a studentilor, vizualizarea anumitor informații despre utilizatori :

**Cautare utilizator**

Cautare [REDACTAT]  nume  
 prenume  
 User

**Trimite**      **Reset**

### 3.8.2 Machete de ieșire

Diploma student:

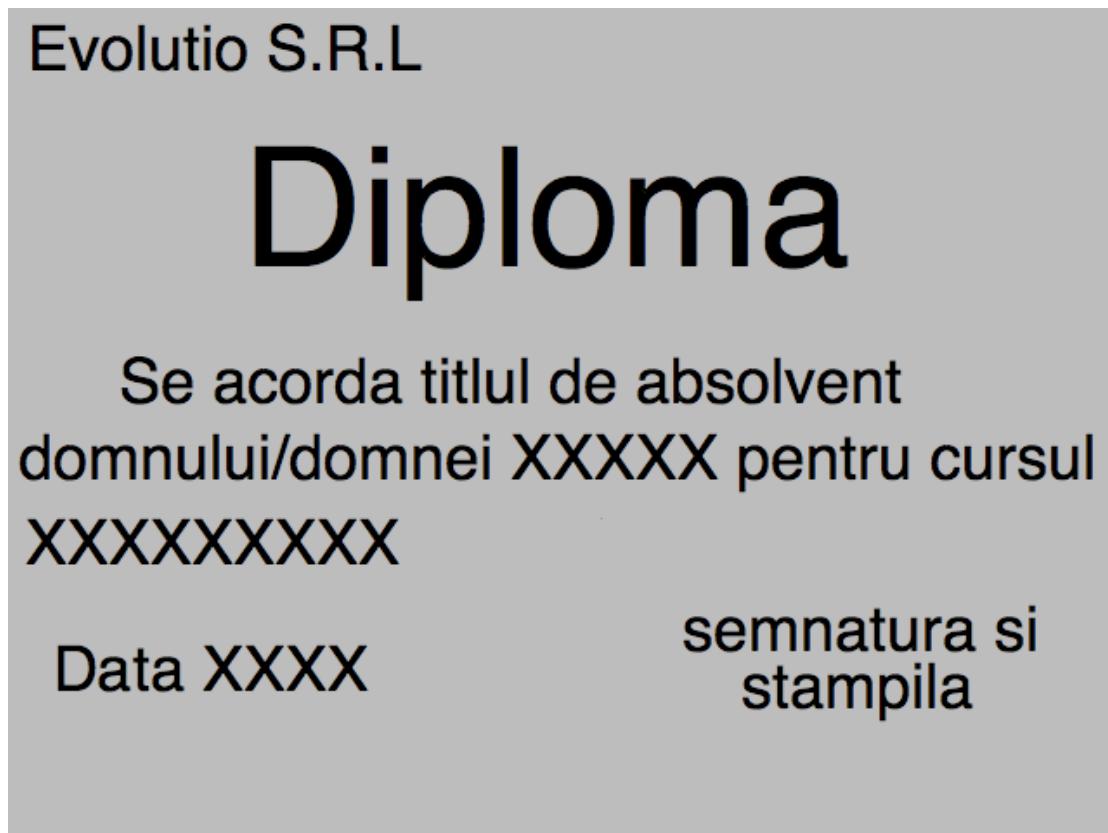


Fig. 14 Macheta Diploma absolvent

Catalogul cu note, vizualizarea notelor studenților:

Note utilizator		
user	test	nota
xxx	test x	x
xxx	test x	x

Fig. 15 Macheta cu notele utilizatorului

Raport cursuri active:

Cursuri disponibile			
Nr.	Curs	Ore alocate	Data
1	XXXXXX	3	13/08/09
2	XXXXXX	6	13/08/09
3	XXXXXX	4	14/08/09
4	XXXXXX	4	14/08/09
5	XXXXXX	4	14/08/09

Fig. 16 Macheta cursuri

Informații cont, editarea informațiilor despre utilizator:

**Informatii cont**

Info 1 [REDACTED]

Info 2 [REDACTED]

Info 3 [REDACTED]

Info 4 [REDACTED]

Info 5 [REDACTED]

**Trimite**      **Reset**

Fig. 17 Macheta Informații cont

# CAPITOLUL IV: PREZENTAREA PRODUSULUI SOFTWARE

## 4.1 Prezentarea site-ului Evolution

Site-ul conține o serie de informații despre firmă, despre activitatea ei precum și despre serviciile oferite. Toți clienții (potențiali sau deja existenți) vor fi înregistrați într-o baza de date aderentă site-ului Evolution. Pagina principală a site-ului permite accesul utilizatorilor la informațiile generale despre societate și la serviciile oferite.

Această pagina principală a site-ului reprezintă 'ghidul' oricărui utilizator, în funcție de informația dorită de acesta. Site-ul se poate accesa la adresa <http://www.evolution.ro> și se prezintă printr-o pagină principală, cu opțiuni pentru toate funcțiile aplicatiei, și anume:

- Despre noi
- Cursuri
- Solutii
- Parteneri
- Contact
- User login

Cu un click pe oricare opțiune se ajunge la o nouă pagină, în care este dezvoltată opțiunea respectivă.

## 4.2 Prezentarea aplicației

Sistemul este bazat pe cea mai completă arhitectură de baze de date pentru membrii . Ea a fost creată pentru realizarea unei pagini web interactive de care avea nevoie firma.

Baza de date a fost realizată cu ajutorul scriptului php : phpmyadmin

Table	Action	Records	Type	Collation	Size	Overhead
admin		1	MyISAM	latin1_swedish_ci	2.8 Kib	-
cursuri		18	MyISAM	latin1_swedish_ci	33.5 Kib	-
dn_an		60	MyISAM	latin1_swedish_ci	5.2 Kib	-
dn_luna		12	MyISAM	latin1_swedish_ci	2.6 Kib	-
dn_zi		31	MyISAM	latin1_swedish_ci	2.6 Kib	-
judete		42	MyISAM	latin1_general_ci	2.8 Kib	-
orase		111	MyISAM	latin1_swedish_ci	32.2 Kib	-
tara		239	MyISAM	latin1_general_ci	12.6 Kib	-
utilizatori		2	MyISAM	latin1_swedish_ci	3.6 Kib	-

Fig. 18 Secțiunea phpMyAdmin

Baza de date cu numele Evolu are 13 tabele. Aici a fost creată structura tablelor cu atributurile lor. Au fost setate câmpurile, tipurile de date, lungimea etc. Aici au fost testate comenzi SQL , folosite în PHP , modificare sau ștergere de date.

În realizarea acestui site s-au folosit CSS-uri(Cascade Style Sheets). În header au fost definite clase care prin urmare sunt folosite de site. Ele se definesc în <head><style> și se apelează pe parcurs cu class.

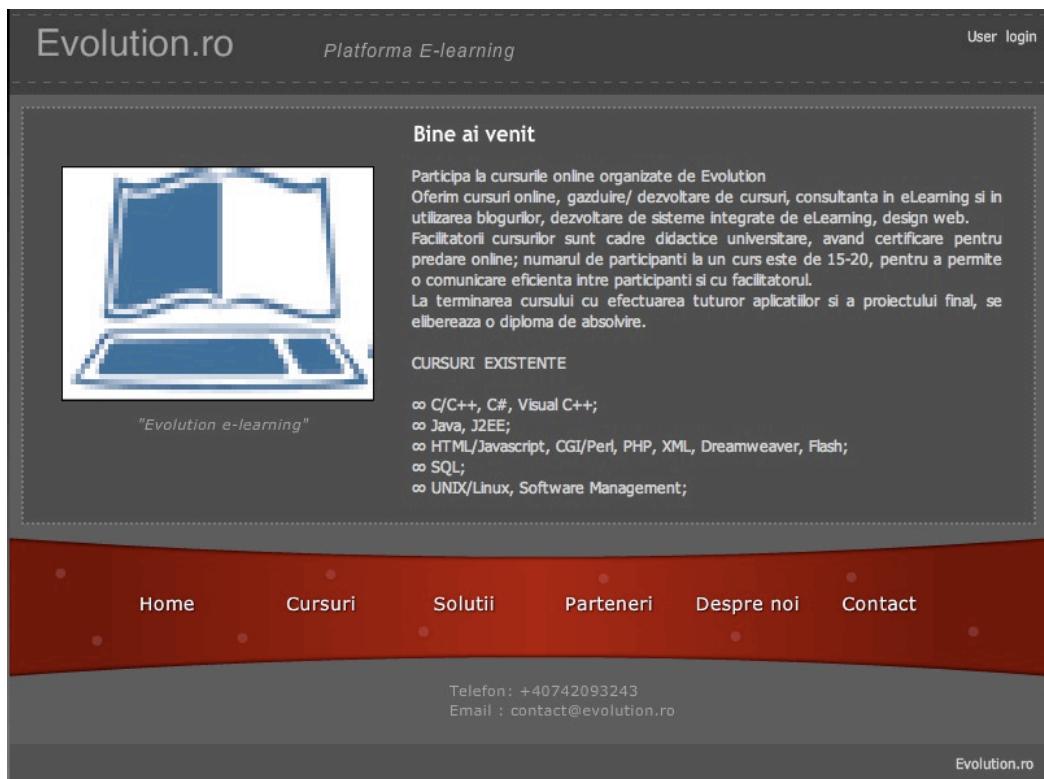


Fig. 19 Pagina principală a aplicației

Pagina principală a aplicației este prezentat ceea ce ofera firma spre utilizatori. Meniul situat în partea inferioară pagini oferă o varietate de alegeri printre care: Cursuri ( prezentarea cursurilor), Solutii (oferte de soluți practice și folositoare utilizatorului), Parteneri (prezentarea partenerilor), despre noi(informați legate despre echipa evolution), contact. În partea superioară a pagini există opțiunea User login, unde se va face logarea și participarea la cursuri, teste, discuți, etc.

Pe această pagină se gasesc toate opțiunile pe care le are la dispoziție utilizatorul pentru a accesa funcțiile aplicației.

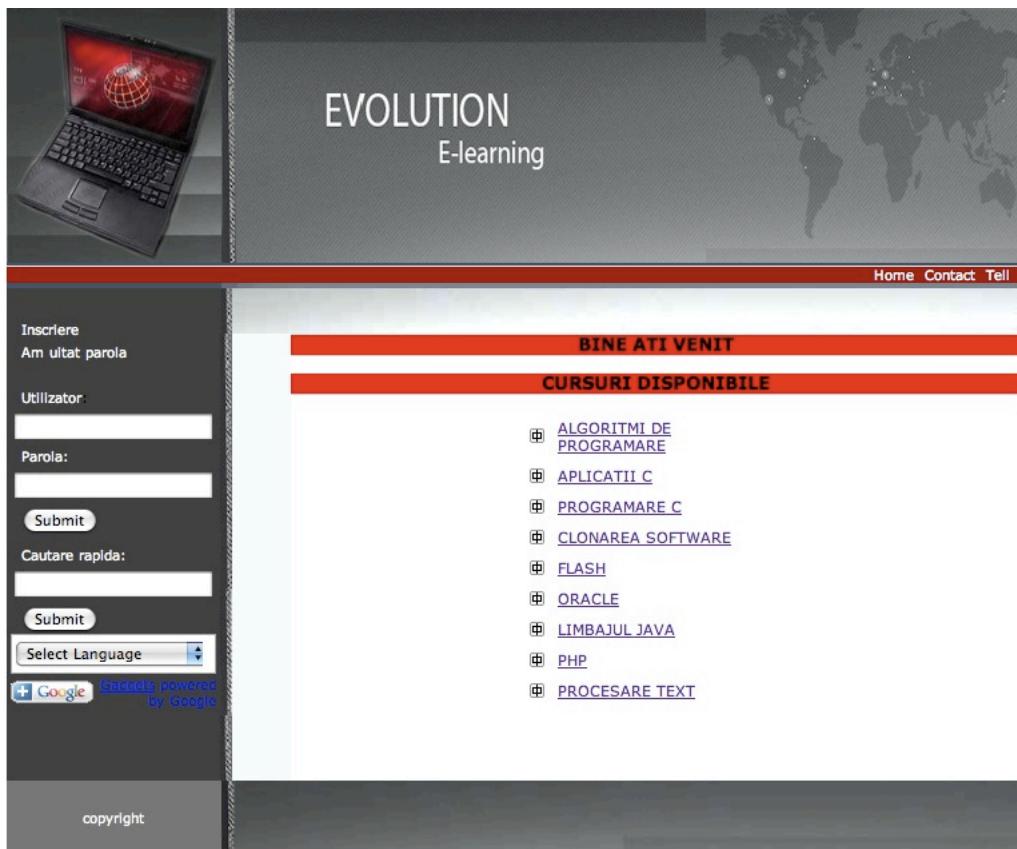


Fig. 20 user login

Informațiile de pe prima user login (HOME) sunt afișate ultimile 9 cursuri înregistrate. Fiecare curs are un link către informații detaliate și documentație.

Pagina este împărțită în header meniu main footer(vedeți anexa). În header este partea de sus, butoanele(Home, Contacte și Tell). În menu partea din stânga în care se află meniul (și submeniurile). În Main se află continutul paginii . În footer se subsolul paginii.

Toate celelalte informații sunt deschise în mijlocul paginii, header-ul și footer-ul rămânând la fel . În realizarea fiecărei pagini s-a folosit funcția include('header.php') și include('footer.php') care preia informațiile din celelalte două pagini.

Există multă lume care se ferește să-și ofere adresa de e-mail (ori să trimă mesaje către adrese necunoscute) din cauza SPAM-ului (mesaje de e-mail comerciale, nesolicitante). Pentru aceasta, pe lângă adresa de e-mail și forum, vizitatorii au la dispoziție un formular de contact, într-o pagină specială, care să nu le solicite decât completarea unor rubrici și apăsarea unu buton, fără a le mai deschide programul de e-mail. Buton Forum deschide o pagina care oferă utilizatorului posibilitatea de a lăsa un mesaj cu privire la problemele lui personale legate de aplicatie/cursuri. Sub formularul de înregistrare a forum-ului se găsesc toate mesajele celorlați utilizatori. În mesajele din Forum apare numele, data și ora înscriserii și mesajul propriu-zis. Datele de contact ale acestuia nu sunt afișate, ele pot fi accesate numai de către administrator.

**CONTACT :**

Sediul central:  
Jud. Satu Mare, Loc. Satu Mare.  
str. Sagetatorului, bl. L3/9  
Tel: +40742093243  
FAX: +40361819209

Ne puteti contacta pe [forum](#) sau [e-mail](#)

Toate campurile sunt obligatorii !!! Va rugam sa completati formularul in intregime!

Nume: \_\_\_\_\_ Prenume: \_\_\_\_\_

Email: \_\_\_\_\_ Telefon: \_\_\_\_\_

Mesajul dumneavoastra:

Anuleaza Trimite

Meniul aplicației este diferit de la utilizator, profesor, admin sau simplu vizitator, în urmatoarele rânduri vom descrie meniul.

Un simplu vizitator va avea acces limitat la funcțiile aplicației, el se va putea înregistra, trimite mail sau folosirea forumului și va avea acces la prezentarea cursurilor.

**CREARE CONT**

\* - INFORMATII OBLIGATORII

\*UTILIZATOR: \_\_\_\_\_

\*PAROLA: \_\_\_\_\_

\*CONFIRMARE PAROLA: \_\_\_\_\_

\*NUME: \_\_\_\_\_

\*PRENUME: \_\_\_\_\_

\*EMAIL: \_\_\_\_\_

\*DATA NASTERII: Luna: \_\_\_\_\_ Ziua: \_\_\_\_\_ Anul: \_\_\_\_\_

\*SEX:  BARBAT  FEMEIE

\*ADRESA: \_\_\_\_\_

\*TARA: \_\_\_\_\_

\*JUDET: \_\_\_\_\_

\*LOCALITATE: \_\_\_\_\_

COD POSTAL: \_\_\_\_\_

TELEFON (FIX): \_\_\_\_\_

Trimite Reset

Fig. 21 Crearea unu nou cont

Apăsând click pe Înregistrare se va deschide un subdomeniu cu Înscriere student sau Înscriere profesor. În urma alegerii uneia dintre ele se deschide un formular care are notate câmpurile cu \*. ceea ce reprezintă câmpurile care trebuie completeate obligatoriu. Datele sunt înregistrate în baza de date .În cazul în care nu s-a introdus un câmp (obligatoriu) apare o pagina cu un mesaj care arată ce câmp nu a fost completat, datele care au fost introduse sunt păstrate datorita tipului de variabilă \$\_SESSION.

Pentru înscrierea datelor în baza de date se apasă butonul trimite, iar pentru ștergerea datelor din formular se apasă butonul Reset. După înregistrare utilizatorul se va putea autentifica la sistem unde va avea acces la funcțiile aplicației.

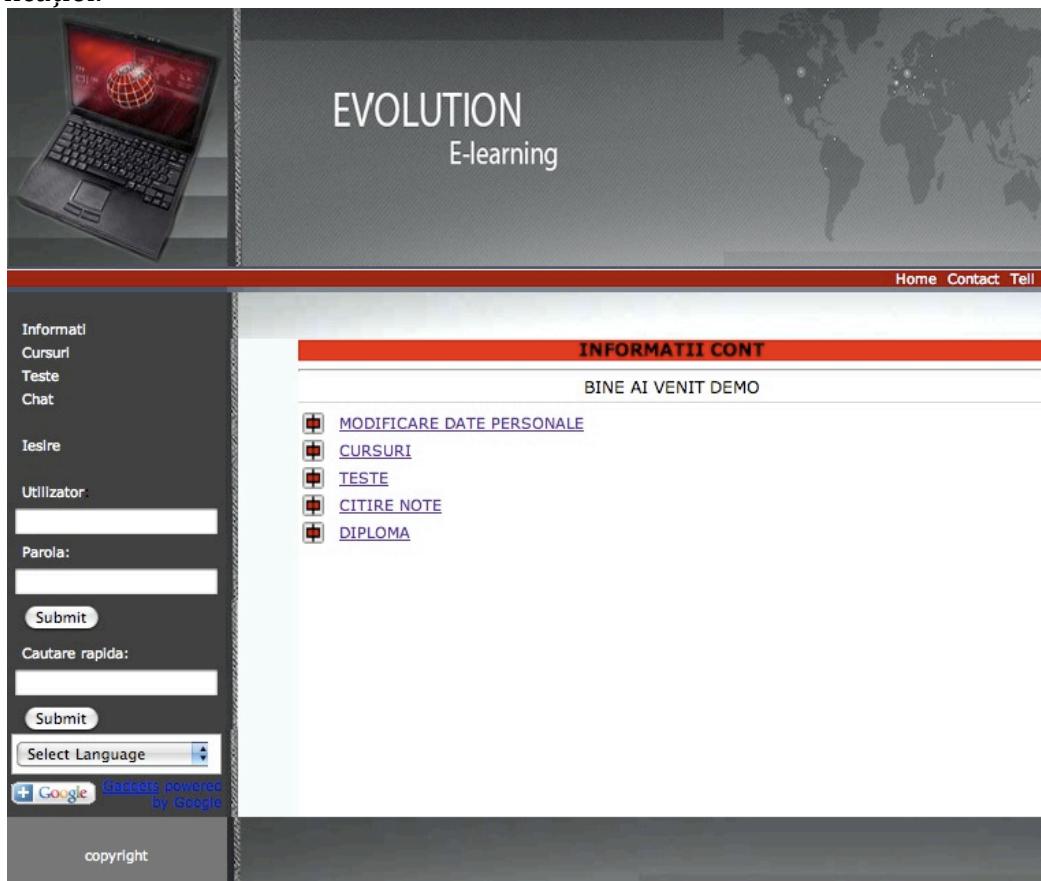


Fig. 22 Home page pentru student

Meniul pentru student conține următoarele secțiuni:

- Infomati – îl va reîntoare pe student la meniul principal;
- Cursuri – se va deschide prezentarea cursurilor;
- Modificare date personale – corectarea anumitor greșeli/ schimbări;
- Teste – se va deschide submeniu rezervat testelor;
- Chat – se va deschide un chat;
- Vizualizare note – prezentarea catre student a notelor;
- Diploma – diplomele dobândite;
- Iesire – delogare din sistem.

În următoarele pagini vom descrie fiecare secțiune în detaliu:

Sectiunea Cursuri. Cursurile sunt prezentate sub formă de document electronic (pdf/doc) unele fiind însoțite de aplicații, exemple. Accesarea documentației cursurilor este doar posibilă prin logarea la sistem.

NR.	DENUMIRE	ORE ALOCATE	DATA TEST	DATA INTRODUCERI
1	<a href="#">ALGORITMI DE PROGRAMARE</a>	10	2009-06-01	2009-05-12 11:06:07
2	<a href="#">APLICATII C</a>	10	2009-06-01	2009-05-12 11:06:07
3	<a href="#">PROGRAMARE C</a>	10	2009-06-01	2009-05-12 11:06:07
4	<a href="#">CLONAREA SOFTWARE</a>	10	2009-06-01	2009-05-12 11:06:07
5	<a href="#">FLASH</a>	10	2009-06-01	2009-05-12 11:06:07
6	<a href="#">ORACLE</a>	10	2009-06-01	2009-05-12 11:06:07
7	<a href="#">LIMBAJUL JAVA</a>	10	2009-06-01	2009-05-12 11:06:07
8	<a href="#">PHP</a>	10	2009-06-01	2009-05-12 11:06:07
9	<a href="#">PROCESARE TEXT</a>	10	2009-06-01	2009-05-12 11:06:07
10	<a href="#">PROGRAMARE VISUAL C++</a>	10	2009-06-01	2009-05-12 11:06:07
11	<a href="#">RETELE</a>	10	2009-06-01	2009-05-12 11:06:07
12	<a href="#">SISTEME DE OPERARE</a>	10	2009-06-01	2009-05-12 11:06:07
13	<a href="#">UNIX</a>	10	2009-06-01	2009-05-12 11:06:07
14	<a href="#">PYTHON</a>	10	2009-06-01	2009-05-12 11:06:07

Fig. 23 Afisarea cursurilor

Denumirea fiecarui curs deține un link către informații detaliate și documentație.

Name	Last modified	Size	Description
Parent Directory		-	
<a href="#">algoritmi de programare..&gt;</a>	22-Jun-2009 23:00	100K	
<a href="#">cap1.pdf</a>	28-May-2009 19:59	216K	
<a href="#">cap2.pdf</a>	28-May-2009 19:59	214K	
<a href="#">cap3.pdf</a>	28-May-2009 19:59	276K	
<a href="#">cap4.pdf</a>	28-May-2009 19:58	198K	
<a href="#">cap5.pdf</a>	28-May-2009 19:58	306K	
<a href="#">cap6.pdf</a>	28-May-2009 19:58	1.8M	
<a href="#">cap7.pdf</a>	28-May-2009 19:58	181K	
<a href="#">cap8.pdf</a>	28-May-2009 19:58	402K	

PREV 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 NEXT

<b>DENUMIRE</b>	ALGORITMI DE PROGRAMARE
<b>ORE ALOCATE</b>	10
<b>DATA TEST</b>	2009-06-01
<b>DATA ADAUGARII</b>	2009-05-12 11:06:07
<b>DESCRIEIRE</b>	ALGORITMI DE PROGRAMARE

[Documentatie](#)

Fig. 24 Documentație pentru fiecare curs in parte

Secțiunea teste servește la verificarea cunoștințelor dobândite de către student. Aceasta poate să fie accesată în momentul în care studentul este convins ca stăpânește cunoștințele necesare.

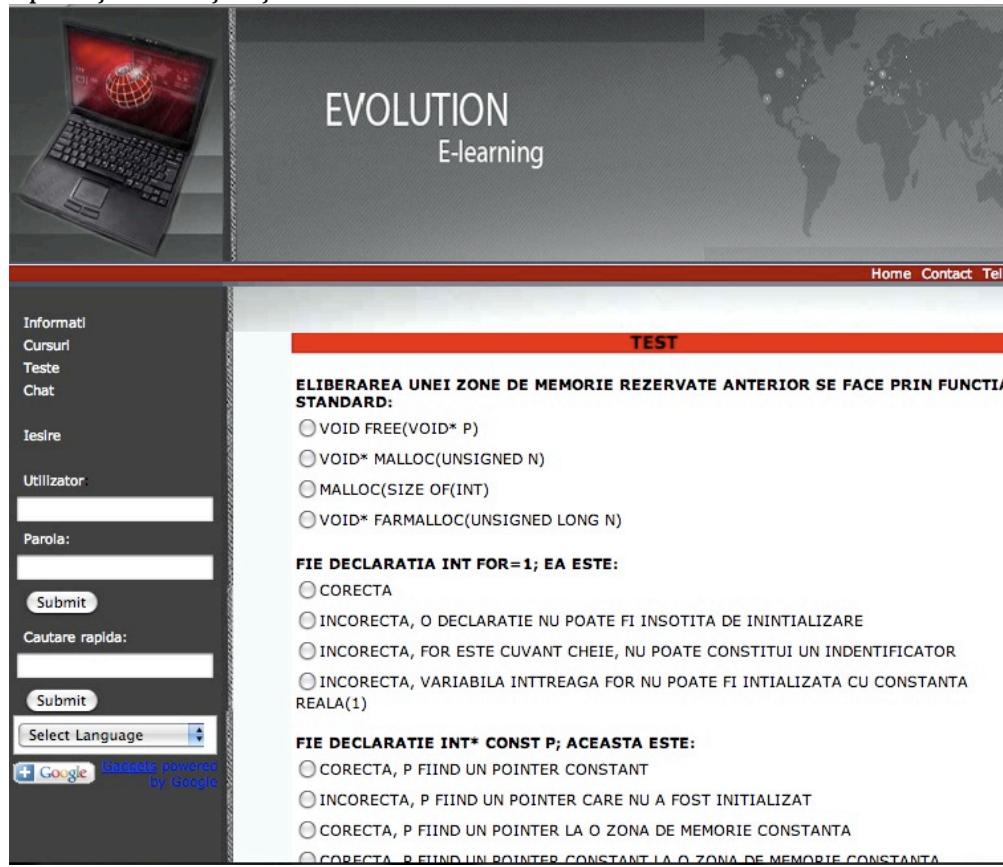


Fig. 25 Captură de ecran test

După rezolvarea testului utilizatorul va trebui să apese butonul "Sa vedem ce ai făcut" pentru ca testul să fie verificat și introducerea notei în baza de date.

Secțiunea Vizualizare note va deschide un subdomeniu unde se va afisa teste si notele dobândite de student.

AFISARE NOTE			
NR.	USER	TEST	NOTA
1	DEMO	TEST1	3
2	DEMO	TEST2	7
3	DEMO	TEST3	6
4	DEMO	TEST4	4
5	DEMO	TEST5	9

Fig. 26 Afisarea notelor

Secțiunea Diploma este vizibilă studentului doar după promovarea unui test/examen.



Fig. 27 Captură de ecran Diplomă

Secțiunea Chat deservește utilizatori și profesori pentru o comunicare rapidă, fără a se loga sau a trimite e-mail. Este foarte usor de folosit prin introducerea numelui, e-mail și mesajul dorit de transmis.

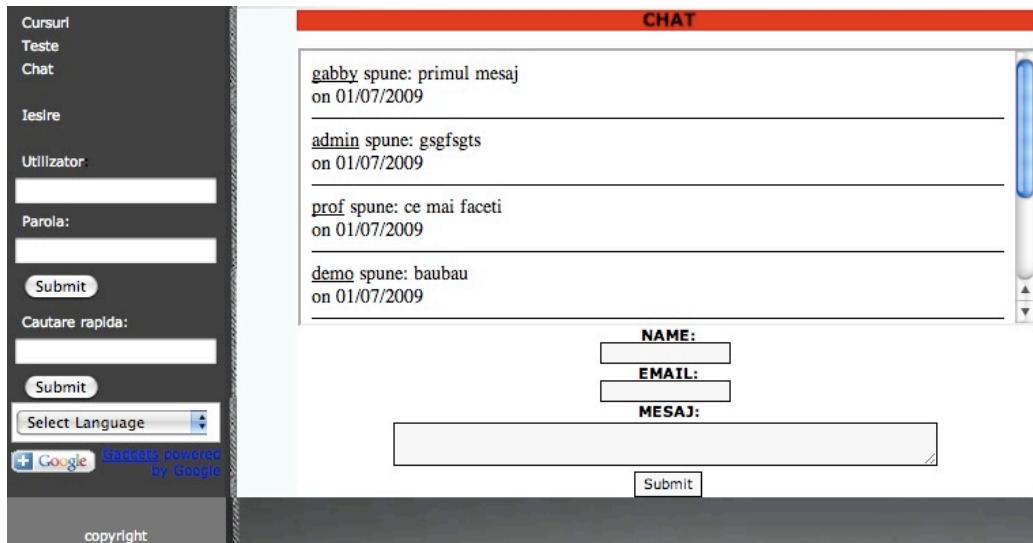


Fig. 28 Captură de ecran Chat

Meniul Profesor conține aceleași secțiuni ca și meniul studentului dar cu câteva secțiuni noi, neaccesibile cursanților:

- Modificare date personale;
- Adaugare cursuri ;
- Modificare teste;
- cautare student;
- Note studenti;

Descrierea acestora:

Fig. 29 Captura de ecran Modificare date personale

Secțiunea Modificare date personale este identică ca la student și ajută pe cei care au greșit datele personale sau trebuie să modifice una din câmpurile de mai sus.

Secțiunea Adaugare cursuri permite profesorilor de a adauga rapid și usor cursurile, fără a le trimite administratorului. Adăugarea de curs este formată din două formulare primul cel de informații, în care profesorul completează numele cursului, o descriere, orele alocate cursului după care se înaintează la cel de al doilea formular uploadarea propriu-zisă a documentației.

**ADAUGA CURSURI**

DENUMIRE:

Descriere:

ORE :

DATA TEST (YYYY-MM-DD):

Fig. 30 Formular de adaugare curs

Secțiunea Modificare teste este dedicată pentru sterge, adauga și modificare întrebări.

**EDITARE TEST**

[STERGE INTREBARE](#)

[ADAUGARE INTREBARE](#)

[LISTA DE INTREBARI](#)

Fig. 31 Captura de ecran meniu teste

Secțiunea Cautare student, cautarea se face după un criteriu selecționat de profesor, după nume, prenume sau user.

Fig. 32 Captura de ecran cautare student

Secțiune note studenți prezintă situația studenților din timpul anului.

NR.	USER	TEST	NOTA
1	GABBY	TEST1	9
2	GABBY	TEST2	9
3	GABBY	TEST3	10
4	GABBY	TEST4	4
5	GABBY	TEST5	7
6	DEMO	TEST1	3
7	DEMO	TEST2	7
8	DEMO	TEST3	6
9	DEMO	TEST4	4
10	DEMO	TEST5	9

Fig. 33 Captura de ecran note

#### **4.3 Descrierea funcțiilor utilizate în aplicație**

Isset() – funcție care verifică dacă a fost setată o valoare a unei variabileș

Strlen() – funcție ce returnează lungimea unui sir de caractere

Require() – funcție care include în scriptul curent conținutul unui fișier cu verificarea de a fi introdus o singură data (spre deosebire de funcția include() – care permite includerea fișierului de mai multe ori )

Printf() – funcție care afișează un sir de caractere formatat

Crypt(\$pass,'BB') – funcție care cripteză un sir de caractere (\$pass) după o cheie de criptare(BB)

Mail(\$to,\$subject,\$message,\$headers) – funcție folosită pentru a trimite un email (de la \$to, cu subiectul \$subject, având ca mesaj \$message cu headerele adiționale \$header);

strcmp() – funcție care compară două siruri de caractere (asemănător funcției standard în limbajul C)

Trim() – funcție care elimină spațiile goale de la începutul și sfârșitul unui sir de caractere specificat ca parametru (asemănător funcției standard în C)

Mysql\_query() – trimit o comanda MySQL bazei de date active de pe server

Mysql\_fetch\_array() – returnează o linie din rezultatul comenzii

Mysql\_error() – returnează mesajul de eroare sub formă de sir de caractere generat de baza de date mysql daca este cazul

Mysql\_num\_rows() – returnează numărul de linii din rezultatul unei cereri mysql

Mysql\_connect("localhost",\$user,\$pass) deschide o conexiune către un server mysql

Mysql\_select\_db(\$dbname) – alege o bază de date mysql după conectarea la un server mysql

Session\_start() inițializează o secțiune de date bazată pe id-ul de sesiune trimis printr-o cerere GET sau POST

Session\_destroy – distrugă toate datele asociate cu sesiunea curentă

Ob\_start() – oprește afișarea bufferului până la terminarea execuției scriptului

# CAPITOLUL V: EFICIENȚA APLICAȚIEI

## Eficiența unui site

Un website bine făcut este cel care se încarcă rapid și oferă vizitatorilor o funcționalitate impecabilă și un conținut complet. Arhitectura informației este simplă și clară, asigurând o navigare intuitivă pentru acces rapid la toate informațiile.

Criteriile necesare pentru eficiența unui site sunt :

- Încărcare rapidă
- Conținut relevant
- Navigare ușoară
- Prezentare consecventă
- Aspect placut
- Optimizarea pentru motoarele de căutare

Un design de calitate înseanmă și un aspect plăcut, care pune în valoare funcționalitatea site-ului și informația oferită. Animățile și imaginile măresc timpul de încărcare al paginii. Pe Internet, însă rapiditatea face legea. Satisticile spun că un vizitator așteaptă în medie 12 secunde ca pagina să se încarce, după care pleacă. În această aplicație s-a ținut cont și de acest lucru.

Poate una dintre cele mai importante realizări este aceea de a oferi o alternativă fiabilă la modelul tradițional de învățământ la distanță. În acest fel s-a răspuns nevoilor persoanelor ce reușesc astfel să eliminate costuri mari de participare la cursuri între care amintim costul transportului și cazării precum și deranjul călătoriei.

Aplicația este una deschisă la care se pot face modificări și se pot introduce cursuri și teste noi de-a lungul timpului. Introducerea de noi cursuri va asigura menținerea unui nivel ridicat al interesului studenților.

Exploatarea aplicației se va face pe o perioadă îndelungată de timp, neexistând termen limită de funcționare.

Toate modificările se vor face doar de către personalul firmei care a furnizat aplicația, pentru asigurarea securității datelor. Versiunea curentă a site-ului va suporta adăugări cu caracter de update-uri sau patch-uri pentru rezolvarea unor eventuale greșeli, erori ce vor fi descoperite implementării sau cu caracter de înnoire a procedurilor și design-ului. Aplicația a creată pentru a fi adaptată ușor la toate modificări și fără suplimentare majore.

Aplicația realizată în cadrul societății Evolution va fi destinată să servească managementul activității de distribuție a serviciilor societății.

Această aplicație informatică are ca obiectiv crearea unui mijloc modern de promovarea a învățamantului de la distanță.

## Bibliografie

“Proiectarea Sistemelor Informatice” - Metode de realizare de V.Chichernea ,C Botezatu,Editura Sylvi Bucuresti 2006

“Dezvoltarea Aplicațiilor WEB cu PHP și MySQL” de Luke Welling,Laura Thomson,Editura TEORA 2008

“Build Your Own Database Driven Website Using PHP & MySQL” –By Kevin Yank, Editura Sitepoint

“Teach Yourself PHP, MySQL and Apache All in One, 2nd Edition” - By Julie C. Meloni.Published by Sams.

“Dezvoltarea aplicatiilor WEB folosind XHTML, PHP si MySQL” de Traian Anghel , Editura Polirom 2005

“HTML pentru World Wide Web cu XHTML si CSS - Ghid de invatare rapida prin imagini” de Elizabeth Castro .Editura Corint

“PHP pentru World Wide Web, in imagini” - de Larry Ulman Editura TEORA 2004

“Proiectarea Sistemelor Informatice” - Metode de realizare de V.Chichernea ,C Botezatu,Editura Sylvi Bucuresti 2002

<http://www.w3schools.com/php/default.asp>

<http://www.phpromania.net>

<http://www.rophp.net/>

## Anexa

### Index.php

```
<? session_start();  
$ip = $_SERVER['REMOTE_ADDR'];  
$ban = array('');  
$count = count($ban);  
for ($i=0; $i<$count; $i++) {  
if($ip == $ban[$i]) { die("I'm sorry, you've been banned. $ip"); }  
}  
include_once("functions/config.php");  
include_once("functions/functions.php");  
if($_REQUEST['pag']=='iesire')  
{  
    iesire();  
}  
include_once("header.php");  
include_once("menu.php");  
include_once("index_main.php");  
include_once("footer.php");  
?>
```

### index\_main.php

```
<TR>  
    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=1  
HEIGHT=335 ALT="">  
        </TD>  
        <TD COLSPAN=5 align="center" valign="top"  
class="smallheader">  
            <h2>Bine ati venit</h2>  
            <br />  
            <h2>Cursuri disponibile</h2>
```

```

<br />
<table class='mtable' width="200" cellpadding="4" cellspacing="2"
>
<?
$query = "SELECT * FROM cursuri ORDER by date1 LIMIT 0,9";
$result=mysql_query($query);
while($row=mysql_fetch_array($result))
{
print"<tr >";
print "<td ><a href='cursuri.php'><img
src='images/menup.gif' border='0'></a></td>";
print "<td ><a
href='cursuri.php'>".$row['denumire']."'</a></td>";
print"</tr>";
}
?>
</table>

<br />
</TD>
</TR>

```

### **footer.php**

```

<TR>
<TD COLSPAN=2 bgcolor="#666666" align="center"
valign="middle" class="small">copyright
</TD>
<TD COLSPAN=6 >
<IMG SRC="images/207_16.gif" WIDTH=599 HEIGHT=56
ALT=""></TD>
</TR>
<TR>

```

```

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=160 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=1 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=52 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=53 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=115 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=110 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=145 HEIGHT=1
ALT=""></TD>

<TD>
    <IMG SRC="images/spacer.gif" WIDTH=124 HEIGHT=1
ALT=""></TD>

</TR>
</TABLE>
<!-- End ImageReady Slices --&gt;
&lt;/BODY&gt;
&lt;/HTML&gt;

<b>header.php

<HTML>
<HEAD>

```

```

<meta http-equiv="Content-Language" content="en-us">
<TITLE>Evolution </TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=windows-1252">
<link href="css/main_stylesnew.css" rel="stylesheet" type="text/css">
<style type="text/css">
<!--
.style1 {color: #CC0033}

-->
</style>
</HEAD>
<BODY BGCOLOR="#FFFFFF LEFTMARGIN=0 TOPMARGIN=0 MARGINWIDTH=0 MARGINHEIGHT=0>
<!-- ImageReady Slices (207.psd) -->
<TABLE WIDTH=760 BORDER=0 CELLPADDING=0 CELLSPACING=0 align=center>
<TR>
    <TD COLSPAN=2>
        <a href="index.php"><IMG SRC="images/207_01.gif" WIDTH=161 HEIGHT=191 ALT="" border="0"></a></TD>
    <TD COLSPAN=2>
        <IMG SRC="images/207_02.gif" WIDTH=105 HEIGHT=191 ALT=""></TD>
    <TD>
        <IMG SRC="images/207_03.gif" WIDTH=115 HEIGHT=191 ALT=""></TD>
    <TD>
        <IMG SRC="images/207_04.gif" WIDTH=110 HEIGHT=191 ALT=""></TD>
    <TD>
        <IMG SRC="images/207_05.gif" WIDTH=145 HEIGHT=191 ALT=""></TD>

```

```

<TD>
    <IMG SRC="images/207_06.gif" WIDTH=124 HEIGHT=191
ALT=""></TD>

</TR>

<TR>

    <TD COLSPAN=8>
        <IMG SRC="images/207_07.gif" WIDTH=760 HEIGHT=2 ALT=""></TD>

    </TR>

    <TR>

        <TD COLSPAN=8 align="right" background="images/207_08.gif"
bgcolor="#2c2c2f">
            <a href="index.php" class="printf" style="text-decoration: none; color: inherit; font-size: 1em; font-weight: bold; margin-right: 10px;">Home&ampnbsp <a href="contact.php" class="printf" style="text-decoration: none; color: inherit; font-size: 1em; font-weight: bold; margin-right: 10px;">Contact&ampnbsp&ampnbsp<a href="tell1.php" class="printf" style="text-decoration: none; color: inherit; font-size: 1em; font-weight: bold; margin-right: 10px;">Tell&ampnbsp&ampnbsp&ampnbsp</TD>
    </TR>

```

### **menu.php**

```

<TR>

    <TD COLSPAN=8>
        <IMG SRC="images/207_09.gif" WIDTH=760 HEIGHT=4
ALT=""></TD>

    </TR>

    <TR>

        <TD ROWSPAN=2 bgcolor="#333333" valign="top"><table
width="100%" border="0">
            <tr>
                <td >&nbsp;&nbsp;<a href="" class="printf" style="text-decoration: none; color: inherit; font-size: 1em; font-weight: bold; margin-right: 10px;"></a> </td>
            </tr>
            <?
                if(isset($_SESSION['iuser']))
                {
                    ?>

```

```

<tr>
<td>&nbsp;&nbsp;<a href="cont.php" class="printf">Informati</a></td>
</tr>

<tr>
<td>&nbsp;&nbsp;<a href="cursuri_disp.php" class="printf">Cursuri
</a></td>
</tr>

<tr>
<td>&nbsp;&nbsp;<a href="quiz1.php" class="printf">Teste</a> </td>
</tr>

<tr>
<td>&nbsp;&nbsp;<a href="forum.php" class="printf">Chat</a> </td>
</tr>

<tr>
<td>&nbsp;&nbsp;</td>
</tr>

<tr>
<td>&nbsp;&nbsp;<a href="index.php?pag=iesire"
class="printf">Iesire</a></td>
</tr>

<?
}

else

{
?>

<tr>
<td></td>
</tr>
<tr>

```

```

<td>&nbsp;&nbsp;<a href="inscriere.php"
class="printf">Inscriere</a></td>
</tr>

<tr>

<td>&nbsp;&nbsp;<a href="recuperare.php" class="printf">Am uitat
parola</a></td>
</tr>

<?
}

?>

<tr>
<td>&nbsp;</td>
</tr>

<tr>
<td>&nbsp;&nbsp;<span class="printf">Utilizator</span>:</td>
</tr>

<tr>
<td align="center">
<form name="form1" method="post" action="cont.php">
<input type="text" name="user">
</td>
</tr>
<tr>
<td>&nbsp;&nbsp;<span class="printf">Parola:</span></td>
</tr>
<tr>
<td align="center"><input type="password" name="parola"></td>
</tr>
<tr>

```

```

<td>&nbsp;&nbsp;<input type="submit" name="Submit"
value="Submit"></form></td>

</tr>

<tr>

<td></td>

</tr>

<tr>

<td>&nbsp;&nbsp;<span class="printf">Cautare rapida:</span></td>

</tr>

<tr>

<td align="center" ><input type="text" name="search" ></td>

</tr>

<tr>

<td >&nbsp;&nbsp;<input type="submit" name="Submit2"
value="Submit"></form>

<script
src="http://www.gmodules.com/ig/ifr?url=http://cowburn.info/wp-
content/themes/cowburn/translate.xml&synd=open&w=140&h=
22&title=&border=%23ffff%7C3px%2C1px+solid+%23999999&
;output=js"></script>

</td>

</tr>

</table>           </TD>

<TD>

<A HREF="#">

<IMG SRC="images/207_11.gif" WIDTH=1
HEIGHT=34 BORDER=0 ALT=""></A></TD>

<TD COLSPAN=6 background="images/207_12.gif">

</TD>

</TR>

```

## **autentificare.php**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
</head>
<body>
</body>
</html>
```

## **cont.php**

```
<? session_start();
ob_start();
include_once("functions/config.php");
include_once("functions/functions.php");
include_once("header.php");
include_once("menu.php");
if($_GET['pag']=='modifica')
{
if($_SESSION['tip_utilizatori']=='utilizatori')//admin
{
    include_once("modificare_utilizatori_main.php");
}
```

```

if($_SESSION['tip_utilizatori']=='prof')//utilizatori
{
    include_once("modificare_prof_main.php");
}
}

elseif($_GET['pag']=='modifica_prof')
{
    include_once("modificare_date_prof.php");
}

elseif($_GET['pag']=='adauga_cursuri')
{
    include_once("adauga_cursuri.php");
}

elseif($_GET['pag']=='afisare_cursuri')
{
    include_once("cursuri_disp.php");
}

elseif(isset($_GET['page']))
{
    include_once("cursuri_disp.php");
}

elseif($_GET['pag']=='citire')
{
    include_once("vizualizare_note.php");
}

elseif($_GET['pag']=='modifica_test')
{
    include_once("editquizlist.php");
}

```

```

elseif($_GET['pag']=='teste')
{
    quiz1.php;
}

elseif($_GET['pag']=='cauta')
{
    include_once("cautare.php");
}

elseif($_GET['pag']=='note')
{
    include_once("note.php");
}

else
{
    include("cont_main.php");
}

include_once("footer.php");
?>

```

### **cont\_main.php**

```

<link href="css/main_stylesnew.css" rel="stylesheet" type="text/css">
<TR>
    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53
HEIGHT=335 ALT="">
        </TD>
<TD COLSPAN=5 align="center" valign="top" class="smallheader">
    <h2>Informatii Cont </h2>
    <hr />

```

```

<?

$sql_utilizatori="select * from utilizatori where
iuser='".$POST['user']."' and pass='".$POST['parola']."'";

$sql_prof="select * from prof where iuser='".$POST['user']."' and
pass='".$POST['parola']."'";

$result_utilizatori=@mysql_query($sql_utilizatori);

$result_prof=@mysql_query($sql_prof);

$nr_utilizatori=@mysql_num_rows($result_utilizatori);

$nr_prof=@mysql_num_rows($result_prof);

while($row_utilizatori=@mysql_fetch_array($result_utilizatori))

{

$_SESSION['id']=$row_utilizatori['id'];

$_SESSION['iuser']=$row_utilizatori['iuser'];

$_SESSION['nume']=$row_utilizatori['nume'];

$_SESSION['tip_utilizatori']='utilizatori';

}

while($row_prof=@mysql_fetch_array($result_prof))

{

$_SESSION['id']=$row_prof['id'];

$_SESSION['iuser']=$row_prof['iuser'];

$_SESSION['nume']=$row_prof['nume'];

$_SESSION['tip_utilizatori']='prof';

}

if(($nr_utilizatori!='1')&&($nr_prof!='1')&&(empty($_SESSION['id'])))

{

?>

<table width="229" border="0" align="center">

<tr>

<td width="30"></td>

```

```

<td width="189" class="redbold">Utilizator sau parola gresita</td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>
</tr>
</table>

<?
}

elseif($_SESSION['tip_utilizatori']=='utilizatori')
{
?>

<h5><? print "Bine ai venit ".$_SESSION['iuser'];?></h5>
<table width="519" border="0" align="left">

<tr>
<td width="23" bgcolor="#FFFFFF"></td>
<td width="486" background="cont.php"><a href="cont.php?pag=modifica" class="toplevel">Modificare date personale</a></td>
</tr>
<tr>
<td></td>
<td><a href="cont.php?pag=afisare_cursuri" class="toplevel">Cursuri</a></td>
</tr>
<tr>

```

```

<td></td>
<td><a href="quiz1.php" class="toplevel">Teste</a></td>
</tr>
<tr>
<td></td>
<td><a href="cont.php?pag=citire" class="toplevel">Citire note</a></td>
</tr>
<?
$q = "SELECT * FROM note ORDER BY id ";
$r = mysql_query($q);
$q1 = "SELECT * FROM test ORDER BY idtest ";
$r1 = mysql_query($q1);
if ( $r['test'] == $r1['nume'] )
{
?>
<tr>
<td></td>
<td><a href="diploma.php" class="toplevel">Diploma</a></td>
</tr>
<?
}
?>
</table>
<?
}
elseif($_SESSION['tip_utilizatori']=='prof')
{
?>
<h5><? print "Bine ai venit ".$_SESSION['iuser'];?></h5>

```

```

<table width="519" border="0" align="left">

<tr>
    <td width="23" bgcolor="#FFFFFF"></td>
    <td width="486" background="cont.php"><a href="cont.php?pag=modifica_prof" class="toplevel">Modificare date personale</a></td>
</tr>

<tr>
    <td></td>
    <td><a href="cont.php?pag=afisare_cursuri" class="toplevel">Afisare cursuri</a></td>
</tr>

<tr>
    <td></td>
    <td><a href="cont.php?pag=adauga_cursuri" class="toplevel">Adauga cursuri</a></td>
</tr>

<tr>
    <td></td>
    <td><a href="quiz1.php" class="toplevel">Teste</a></td>
</tr>

<tr>
    <td></td>
    <td><a href="cont.php?pag=modifica_test" class="toplevel">Modifica Teste</a></td>
</tr>

<tr>
    <td></td>
    <td><a href="cont.php?pag=cauta" class="toplevel">Cautare student</a></td>
</tr>

```

```

<tr>
    <td></td>
    <td><a href="cont.php?pag=note" class="toplevel">Note
studenti</a></td>
</tr>
</table>
<?
}
?>
</TD>
</TR>

```

### **main\_stylenew.css**

```

P {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}
TABLE {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}
TD {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}
TH {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}
TR {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}
OL {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}
UL {
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,
Arial, Helvetica, sans-serif
}

```

```
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,  
    Arial, Helvetica, sans-serif  
}  
LI {  
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,  
    Arial, Helvetica, sans-serif  
}  
DIV {  
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,  
    Arial, Helvetica, sans-serif  
}  
BODY {  
    FONT-SIZE: 11px; COLOR: #000000; FONT-FAMILY: Verdana,  
    Arial, Helvetica, sans-serif  
}  
BODY {  
    BACKGROUND-COLOR: white  
}  
TH {  
    TEXT-ALIGN: left  
}  
H2 {  
    PADDING-RIGHT: 4px; BORDER-TOP: #eed 1px solid; PADDING-  
    LEFT: 4px; FONT-SIZE: 13px; BACKGROUND: #ff0000; PADDING-  
    BOTTOM: 2px; MARGIN: 0px; COLOR: #000000; PADDING-TOP: 1px;  
    BORDER-BOTTOM: #ccc 1px solid  
}  
H2 A {  
    COLOR: #336; TEXT-DECORATION: none  
}  
H2 A:hover {  
    COLOR: #336; TEXT-DECORATION: none  
}  
H2 A:visited {  
    COLOR: #336; TEXT-DECORATION: none  
}  
  
H3 {  
    FONT-WEIGHT: bold; FONT-SIZE: 14px; MARGIN-BOTTOM: 0px;  
    COLOR: #000000; LINE-HEIGHT: 15px; FONT-FAMILY: Verdana,  
    Arial, Helvetica, sans-serif  
}  
H4 {  
    FONT-WEIGHT: normal; FONT-SIZE: 18px; MARGIN-BOTTOM: 0px;  
    COLOR: #997b35; LINE-HEIGHT: 20px; FONT-FAMILY: Arial,  
    Verdana, Helvetica, sans-serif  
}
```

```

H5 {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; FONT-WEIGHT: normal; FONT-SIZE: 12px; PADDING-BOTTOM: 4px; MARGIN: 0px 0px 4px; COLOR: #000000; LINE-HEIGHT: 13px; PADDING-TOP: 0px; BORDER-BOTTOM: #cccccc 1px solid; FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif
}
.b {
    BORDER-RIGHT: #ccc 1px solid; PADDING-RIGHT: 0px; BORDER-TOP: #eed 1px solid; PADDING-LEFT: 0px; BACKGROUND: #fff; PADDING-BOTTOM: 0px; MARGIN: 1px; BORDER-LEFT: #eed 1px solid; PADDING-TOP: 0px; BORDER-BOTTOM: #ccc 1px solid; WHITE-SPACE: nowrap
}
.smallheader {
    FONT-WEIGHT: bold; FONT-SIZE: 10px; MARGIN-BOTTOM: 0px; PADDING-BOTTOM: 0px; TEXT-TRANSFORM: uppercase; COLOR: #475F78; LINE-HEIGHT: 120%
}
#footercontainer {
    BORDER-TOP: #e0dccc 1px solid; MARGIN-TOP: 15px; MARGIN-BOTTOM: 10px; WIDTH: 751px; PADDING-TOP: 10px
}
.footer {
    FONT-SIZE: 10px; WORD-SPACING: normal; LINE-HEIGHT: 160%; FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; LETTER-SPACING: normal
}
.left {
    FLOAT: left
}
.right {
    FLOAT: right
}
.headline {
    FONT-WEIGHT: bold; FONT-SIZE: 14px; MARGIN-BOTTOM: 0px; COLOR: #000000; LINE-HEIGHT: 15px; FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif
}
.headline2 {
    FONT-WEIGHT: bold; FONT-SIZE: 13px; MARGIN-BOTTOM: 0px; COLOR: #000000; LINE-HEIGHT: 15px; FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif
}
.featureheadline {
    FONT-WEIGHT: normal; FONT-SIZE: 18px; MARGIN-BOTTOM: 0px; COLOR: #997b35; LINE-HEIGHT: 20px; FONT-FAMILY: Arial,

```

```

    Verdana, Helvetica, sans-serif
}
#newsfeatures {
    MARGIN-TOP: 10px; FONT-SIZE: 11px; MARGIN-BOTTOM: 25px;
    LINE-HEIGHT: 13px; FONT-FAMILY: Verdana, Arial, Helvetica,
    sans-serif
}
.news {
    FONT-SIZE: 11px; LINE-HEIGHT: 13px; FONT-FAMILY: Verdana,
    Arial, Helvetica, sans-serif
}
.news STRONG {
    FONT-WEIGHT: bold; FONT-SIZE: 11px; LINE-HEIGHT: 13px;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif
}
.news I {
    FONT-SIZE: 11px; LINE-HEIGHT: 18px; FONT-STYLE: italic
}
.news EM {
    FONT-SIZE: 11px; LINE-HEIGHT: 18px; FONT-STYLE: italic
}
.more {
    FONT-SIZE: 11px; FONT-STYLE: italic; FONT-FAMILY: Verdana,
    Arial, Helvetica, sans-serif
}
.captionright {
    FONT-SIZE: 10px; FONT-STYLE: italic; FONT-FAMILY: Verdana,
    Arial, Helvetica, sans-serif; TEXT-ALIGN: right
}
.captionleft {
    MARGIN-TOP: 5px; FONT-SIZE: 10px; FONT-STYLE: italic;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    ALIGN: left
}
.err {
    MARGIN-TOP: 3px; FONT-SIZE: 8px; FONT-STYLE: italic; FONT-
    FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-ALIGN:
    left
}
.introparagraph {
    FONT-SIZE: 13px; LINE-HEIGHT: 16px; FONT-FAMILY: Verdana,
    Arial, Helvetica, sans-serif
}
.small {
    FONT-SIZE: 10px; LINE-HEIGHT: 11px; FONT-FAMILY: Verdana,
    Arial, Helvetica, sans-serif;
    color: #FFFFFF
}

```

```

}

.smaller {
    FONT-SIZE: 9px; LINE-HEIGHT: 10px; FONT-FAMILY: Arial,
    Verdana, Helvetica, sans-serif
}
.redbold {
    FONT-WEIGHT: bold; COLOR: #930000
}
#newsandeventscontainer {
    BORDER-RIGHT: #ebebdb 1px solid; PADDING-RIGHT: 0px;
    BORDER-TOP: #ebebdb 1px solid; PADDING-LEFT: 0px; BACKGROUND:
    #ebebdb; PADDING-BOTTOM: 0px; BORDER-LEFT: #ebebdb 1px solid;
    WIDTH: 201px; PADDING-TOP: 0px; BORDER-BOTTOM: #ebebdb 1px
    solid
}
#newsandevents {
    DISPLAY: block; FONT-SIZE: 11px; BACKGROUND: #ebebdb;
    MARGIN: 0px 10px 30px; WIDTH: 181px
}
#newsandevents P.newstitle {
    FONT-WEIGHT: bold; FONT-SIZE: 11px; PADDING-BOTTOM: 10px;
    MARGIN: 8px 0px 0px; BORDER-BOTTOM: #dfdfdf 1px solid
}
#newsandevents P {
    FONT-WEIGHT: normal; FONT-SIZE: 11px
}
#newsandevents A {
    FONT-WEIGHT: normal; FONT-SIZE: 11px; LINE-HEIGHT: 13px
}
#specialfeatures {
    PADDING-RIGHT: 0px; BORDER-TOP: #ececec 1px solid; MARGIN-
    TOP: 13px; DISPLAY: block; PADDING-LEFT: 0px; FONT-SIZE: 11px;
    PADDING-BOTTOM: 0px; WIDTH: 201px; PADDING-TOP: 10px
}
#specialfeatures .title {
    FONT-WEIGHT: bold; FONT-SIZE: 11px; MARGIN: 0px 0px 5px
}
#specialfeatures P {
    FONT-WEIGHT: normal; FONT-SIZE: 11px; MARGIN: 0px 0px 5px
}
#specialfeatures A {
    FONT-WEIGHT: normal; FONT-SIZE: 11px; LINE-HEIGHT: 13px
}
#contactheader {
    FONT-WEIGHT: bold; FONT-SIZE: 13px; MARGIN: 20px 0px 0px
}
#contactbox {

```

```

        PADDING-RIGHT: 10px; MARGIN-TOP: 5px; PADDING-LEFT: 12px;
PADDING-BOTTOM: 15px; WIDTH: 530px; PADDING-TOP: 10px;
BACKGROUND-COLOR: #e9e3d7
}

#utilitynavcontainer {
    BORDER-RIGHT: #003311 1px solid; BORDER-TOP: #255134 1px
solid; DISPLAY: block; BACKGROUND: #003311; BORDER-LEFT:
#003311 1px solid; WIDTH: auto; BORDER-BOTTOM: #003311 1px
solid; HEIGHT: auto
}
#utilitynav {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; PADDING-LEFT:
0px; BORDER-LEFT-WIDTH: 0px; FLOAT: left; BORDER-BOTTOM-WIDTH:
0px; PADDING-BOTTOM: 0px; MARGIN: 0px 0px 0px 10px; WIDTH:
700px; PADDING-TOP: 0px; LIST-STYLE-TYPE: none; BORDER-RIGHT-
WIDTH: 0px
}
#utilitynav UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; PADDING-LEFT:
0px; BORDER-LEFT-WIDTH: 0px; FLOAT: left; BORDER-BOTTOM-WIDTH:
0px; PADDING-BOTTOM: 0px; MARGIN: 0px 0px 0px 10px; WIDTH:
700px; PADDING-TOP: 0px; LIST-STYLE-TYPE: none; BORDER-RIGHT-
WIDTH: 0px
}
#utilitynav A {
    PADDING-RIGHT: 0px; DISPLAY: block; PADDING-LEFT: 0px;
FONT-WEIGHT: normal; FONT-SIZE: 10px; PADDING-BOTTOM: 0px;
COLOR: #f0e9ce; PADDING-TOP: 0px; TEXT-DECORATION: none
}
#utilitynav A:hover {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM:
0px; PADDING-TOP: 0px; TEXT-DECORATION: underline
}
#utilitynav LI {
    PADDING-RIGHT: 11px; BACKGROUND-POSITION: 0px 50%; MARGIN-
TOP: 5px; PADDING-LEFT: 11px; FLOAT: left; BACKGROUND-IMAGE:
url(..../images/navbullet.gif); PADDING-BOTTOM: 0px; PADDING-
TOP: 0px; BACKGROUND-REPEAT: no-repeat
}
#navcontainer {
    BORDER-RIGHT: black 1px solid; BORDER-TOP: black 1px
solid; DISPLAY: block; Z-INDEX: 5; BACKGROUND:
url(..../images/nav_bg.gif) #006633; BORDER-LEFT: black 1px
solid; WIDTH: auto; BORDER-BOTTOM: black 1px solid; HEIGHT:
20px
}

```

```

#nav {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; PADDING-LEFT: 0px;
    BORDER-LEFT-WIDTH: 0px; Z-INDEX: 5; FLOAT: left; BORDER-
    BOTTOM-WIDTH: 0px; PADDING-BOTTOM: 0px; MARGIN: 0px 0px 0px
    10px; WIDTH: 725px; PADDING-TOP: 0px; LIST-STYLE-TYPE: none;
    HEIGHT: 29px; BORDER-RIGHT-WIDTH: 0px
}
#nav UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; PADDING-LEFT: 0px;
    BORDER-LEFT-WIDTH: 0px; Z-INDEX: 5; FLOAT: left; BORDER-
    BOTTOM-WIDTH: 0px; PADDING-BOTTOM: 0px; MARGIN: 0px 0px 0px
    10px; WIDTH: 725px; PADDING-TOP: 0px; LIST-STYLE-TYPE: none;
    HEIGHT: 29px; BORDER-RIGHT-WIDTH: 0px
}
#nav A {
    PADDING-RIGHT: 0px; DISPLAY: block; PADDING-LEFT: 10px;
    FONT-WEIGHT: normal; FONT-SIZE: 11px; Z-INDEX: 5; BACKGROUND:
    #006633; PADDING-BOTTOM: 5px; COLOR: #ffffff; PADDING-TOP:
    3px; FONT-FAMILY: Arial, Verdana, Helvetica, sans-serif; TEXT-
    DECORATION: none
}
#nav A:hover {
    Z-INDEX: 5; COLOR: #ffcc66; BACKGROUND-COLOR: #003311;
    TEXT-DECORATION: underline
}
#nav A.toplevel {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; PADDING-LEFT: 0px;
    FONT-WEIGHT: normal; BORDER-LEFT-WIDTH: 0px; FONT-SIZE:
    12px; BACKGROUND: #006633; BORDER-BOTTOM-WIDTH: 0px; PADDING-
    BOTTOM: 7px; COLOR: #ffffff; PADDING-TOP: 7px; BORDER-RIGHT-
    WIDTH: 0px
}
#nav A.toplevel:hover {
    COLOR: #ffcc66; BACKGROUND-COLOR: #006633; TEXT-
    DECORATION: underline
}
#nav LI {
    PADDING-RIGHT: 6px; PADDING-LEFT: 6px; Z-INDEX: 5; FLOAT:
    left; PADDING-BOTTOM: 0px; PADDING-TOP: 0px
}
#nav LI UL {
    PADDING-RIGHT: 1px; PADDING-LEFT: 1px; Z-INDEX: 5; LEFT: -999em;
    PADDING-BOTTOM: 1px; MARGIN: 0px; WIDTH: 129px;
    PADDING-TOP: 0px; POSITION: absolute; HEIGHT: auto;
    BACKGROUND-COLOR: #003311
}
#nav LI LI {

```

```
PADDING-RIGHT: 0px; MARGIN-TOP: 1px; PADDING-LEFT: 0px; Z-  
INDEX: 5; PADDING-BOTTOM: 0px; WIDTH: 129px; PADDING-TOP: 0px  
}  
#nav LI UL A {  
    DISPLAY: block; Z-INDEX: 7; BACKGROUND: #006633; WIDTH:  
119px  
}  
#nav LI UL UL {  
    Z-INDEX: 7; MARGIN: -21px 0px 0px 129px  
}  
#nav LI:hover UL UL {  
    Z-INDEX: 7; LEFT: -999em  
}  
#nav LI:hover UL UL UL {  
    Z-INDEX: 7; LEFT: -999em  
}  
#nav LI.sfhover UL UL {  
    Z-INDEX: 7; LEFT: -999em  
}  
#nav LI.sfhover UL UL UL {  
    Z-INDEX: 7; LEFT: -999em  
}  
#nav LI:hover UL {  
    Z-INDEX: 7; LEFT: auto  
}  
#nav LI LI:hover UL {  
    Z-INDEX: 7; LEFT: auto  
}  
#nav LI LI LI:hover UL {  
    Z-INDEX: 7; LEFT: auto  
}  
#nav LI.sfhover UL {  
    Z-INDEX: 7; LEFT: auto  
}  
#nav LI LI.sfhover UL {  
    Z-INDEX: 7; LEFT: auto  
}  
#nav LI LI LI.sfhover UL {  
    Z-INDEX: 7; LEFT: auto  
}  
#nav LI:hover {  
    Z-INDEX: 5  
}  
#nav LI.sfhover {  
    Z-INDEX: 5  
}  
#leftnavcontainer {
```

```

Z-INDEX: 1; OVERFLOW: hidden; WIDTH: 230px; POSITION:
relative; HEIGHT: 211px; BACKGROUND-COLOR: #f0e9ce
}
#leftnav UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; DISPLAY: block;
PADDING-LEFT: 0px; BORDER-LEFT-WIDTH: 0px; LIST-STYLE-IMAGE:
none; BORDER-BOTTOM-WIDTH: 0px; PADDING-BOTTOM: 0px; MARGIN:
0px; WIDTH: 230px; PADDING-TOP: 0px; BACKGROUND-REPEAT: no-
repeat; LIST-STYLE-TYPE: none; BORDER-RIGHT-WIDTH: 0px
}
#leftnav LI {
    BORDER-TOP-WIDTH: 0px; DISPLAY: block; FONT-WEIGHT: bold;
BORDER-LEFT-WIDTH: 0px; FONT-SIZE: 12px; BORDER-BOTTOM-WIDTH:
0px; PADDING-BOTTOM: 4px; MARGIN-LEFT: 18px; WIDTH: 200px;
PADDING-TOP: 4px; FONT-FAMILY: Arial, Verdana, Helvetica,
sans-serif; BORDER-RIGHT-WIDTH: 0px
}
#leftnav A {
    COLOR: #006633; TEXT-DECORATION: none
}
#leftnav A:hover {
    COLOR: #006633; TEXT-DECORATION: underline
}
#othernav UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; MARGIN-TOP:
7px; DISPLAY: block; PADDING-LEFT: 0px; BORDER-LEFT-WIDTH:
0px; LIST-STYLE-IMAGE: none; BORDER-BOTTOM-WIDTH: 0px; MARGIN-
BOTTOM: 0px; PADDING-BOTTOM: 0px; MARGIN-LEFT: 12px; WIDTH:
200px; PADDING-TOP: 0px; BACKGROUND-REPEAT: no-repeat; LIST-
STYLE-TYPE: none; BORDER-RIGHT-WIDTH: 0px
}
#othernav LI {
    PADDING-RIGHT: 0px; BACKGROUND-POSITION: 0px 50%; DISPLAY:
block; PADDING-LEFT: 13px; FONT-WEIGHT: normal; FONT-SIZE:
12px; BACKGROUND-IMAGE: url(..../images/bullet_arrow.gif);
PADDING-BOTTOM: 2px; WIDTH: 200px; PADDING-TOP: 2px;
BACKGROUND-REPEAT: no-repeat; FONT-FAMILY: Arial, Verdana,
Helvetica, sans-serif
}
#othernav A {
    COLOR: #006633; TEXT-DECORATION: none
}
#othernav A:hover {
    COLOR: #006633; TEXT-DECORATION: underline
}
#intleftnav UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; DISPLAY: block;

```

```

PADDING-LEFT: 0px; BORDER-LEFT-WIDTH: 0px; Z-INDEX: 1; LIST-
STYLE-IMAGE: none; BORDER-BOTTOM-WIDTH: 0px; PADDING-BOTTOM:
0px; MARGIN: 10px 0px 0px; WIDTH: 160px; PADDING-TOP: 0px;
BACKGROUND-REPEAT: no-repeat; LIST-STYLE-TYPE: none; BORDER-
RIGHT-WIDTH: 0px
}
#intleftnav LI {
    PADDING-RIGHT: 0px; DISPLAY: block; PADDING-LEFT: 0px;
    FONT-WEIGHT: bold; FONT-SIZE: 12px; Z-INDEX: 1; PADDING-
    BOTTOM: 3px; MARGIN-LEFT: 3px; WIDTH: 150px; PADDING-TOP: 3px;
    FONT-FAMILY: Arial, Verdana, Helvetica, sans-serif
}
#intleftnav LI.title {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; FONT-WEIGHT: bold;
    Z-INDEX: 1; PADDING-BOTTOM: 0px; PADDING-TOP: 10px
}
#intleftnav A {
    Z-INDEX: 1; COLOR: #006633; TEXT-DECORATION: none
}
#intleftnav A:hover {
    Z-INDEX: 1; COLOR: #006633; TEXT-DECORATION: underline
}
#intleftnavplain UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; DISPLAY: block;
    PADDING-LEFT: 0px; BORDER-LEFT-WIDTH: 0px; Z-INDEX: 1; LIST-
    STYLE-IMAGE: none; BORDER-BOTTOM-WIDTH: 0px; PADDING-BOTTOM:
    0px; MARGIN: 10px 0px 0px; WIDTH: 150px; PADDING-TOP: 0px;
    BACKGROUND-REPEAT: no-repeat; LIST-STYLE-TYPE: none; BORDER-
    RIGHT-WIDTH: 0px
}
#intleftnavplain LI {
    DISPLAY: block; FONT-WEIGHT: normal; FONT-SIZE: 11px; Z-
    INDEX: 1; MARGIN-LEFT: 3px; WIDTH: 150px; LINE-HEIGHT: 12px
}
#intleftnavplain LI.title {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; FONT-WEIGHT: bold;
    Z-INDEX: 1; PADDING-BOTTOM: 0px; PADDING-TOP: 10px
}
#intleftnavplain A {
    Z-INDEX: 1; COLOR: #006633; TEXT-DECORATION: none
}
#intleftnavplain A:hover {
    Z-INDEX: 1; COLOR: #006633; TEXT-DECORATION: underline
}
.printf {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #FFFFFF;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-

```

```

DECORATION: none
}
A.printff:link {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #FFFFFF;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: none
}
A.printff:visited {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #FFFFFF;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: none
}
A.printff:hover {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #B5C6D5;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: underline
}

.printf {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #FFFFFF;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: none
}
A.printf:link {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #FFFFFF;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: none
}
A.printf:visited {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #FFFFFF;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: none
}
A.printf:hover {
    FONT-WEIGHT: normal; FONT-SIZE: 10px; COLOR: #475F78;
    FONT-FAMILY: Verdana, Arial, Helvetica, sans-serif; TEXT-
    DECORATION: underline
}

#deptlist UL {
    BACKGROUND-POSITION: left 3px; MARGIN-TOP: 10px; PADDING-
    LEFT: 0px; LIST-STYLE-POSITION: outside; LIST-STYLE-IMAGE:
    url(..../images/bullet_dotbig.gif); MARGIN-LEFT: 15px;
    BACKGROUND-REPEAT: no-repeat
}
#deptlist LI {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; FONT-WEIGHT:

```

```

normal; FONT-SIZE: 12px; PADDING-BOTTOM: 2px; MARGIN: 0px;
LINE-HEIGHT: 13px; PADDING-TOP: 2px
}
#listbullet UL {
    BORDER-TOP-WIDTH: 0px; PADDING-RIGHT: 0px; DISPLAY: block;
PADDING-LEFT: 0px; BORDER-LEFT-WIDTH: 0px; LIST-STYLE-IMAGE:
none; BORDER-BOTTOM-WIDTH: 0px; PADDING-BOTTOM: 0px; MARGIN:
0px 0px 20px; WIDTH: 175px; PADDING-TOP: 0px; BACKGROUND-
REPEAT: no-repeat; LIST-STYLE-TYPE: none; BORDER-RIGHT-WIDTH:
0px
}
#listbullet LI {
    PADDING-RIGHT: 0px; BACKGROUND-POSITION: left 2px;
DISPLAY: block; PADDING-LEFT: 12px; FONT-WEIGHT: normal; FONT-
SIZE: 12px; BACKGROUND-IMAGE: url(..../images/bullet_dot.gif);
PADDING-BOTTOM: 2px; LINE-HEIGHT: 13px; PADDING-TOP: 1px;
BACKGROUND-REPEAT: no-repeat; FONT-FAMILY: Arial, Verdana,
Helvetica, sans-serif
}
#listbullet LI.title {
    PADDING-RIGHT: 0px; DISPLAY: block; PADDING-LEFT: 0px;
FONT-WEIGHT: bold; BACKGROUND-IMAGE: none; PADDING-BOTTOM:
7px; PADDING-TOP: 2px
}
#listbullet A {
    COLOR: #006633; TEXT-DECORATION: none
}
#listbullet A:hover {
    COLOR: #006633; TEXT-DECORATION: underline
}
UL {
    BACKGROUND-POSITION: left 3px; MARGIN-TOP: 10px; PADDING-
LEFT: 0px; LIST-STYLE-POSITION: outside; LIST-STYLE-IMAGE:
url(..../images/bullet_dotbig.gif); MARGIN-LEFT: 15px;
BACKGROUND-REPEAT: no-repeat
}
LI {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM:
5px; MARGIN: 0px; PADDING-TOP: 5px
}
#greenbullet UL {
    BACKGROUND-POSITION: left 3px; MARGIN-TOP: 10px; PADDING-
LEFT: 0px; LIST-STYLE-POSITION: outside; LIST-STYLE-IMAGE:
url(..../images/bullet_dotgreen.gif); MARGIN-LEFT: 15px;
BACKGROUND-REPEAT: no-repeat
}
#greenbullet LI {

```

```

        PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM:
3px; MARGIN: 0px; PADDING-TOP: 3px
}
#listplain UL {
    BACKGROUND-POSITION: 0px 50%; MARGIN-TOP: 10px; PADDING-
LEFT: 0px; LIST-STYLE-IMAGE: none; MARGIN-LEFT: 0px;
BACKGROUND-REPEAT: no-repeat; LIST-STYLE-TYPE: none
}
#listplain LI {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM:
3px; MARGIN: 0px; PADDING-TOP: 3px
}
#listplain LI.title {
    FONT-WEIGHT: bold
}
FORM {
    MARGIN: 0px
}
INPUT {
    FONT-WEIGHT: normal; FONT-SIZE: 11px; FONT-FAMILY:
Verdana, Geneva, Arial, Helvetica, sans-serif
}
#inputfield {
    BORDER-RIGHT: #d9d9d9 1px solid; PADDING-RIGHT: 3px;
BORDER-TOP: #999 1px solid; MARGIN-TOP: 3px; PADDING-LEFT:
3px; FONT-SIZE: 11px; BACKGROUND: #ffffff; PADDING-BOTTOM:
3px; BORDER-LEFT: #999 1px solid; PADDING-TOP: 3px; BORDER-
BOTTOM: #d9d9d9 1px solid
}
TEXTAREA {
    BORDER-RIGHT: #d9d9d9 1px solid; PADDING-RIGHT: 3px;
BORDER-TOP: #999 1px solid; MARGIN-TOP: 3px; PADDING-LEFT:
3px; FONT-SIZE: 11px; BACKGROUND: #ffffff; PADDING-BOTTOM:
3px; BORDER-LEFT: #999 1px solid; PADDING-TOP: 3px; BORDER-
BOTTOM: #d9d9d9 1px solid;
    width: 400px;
    min-width: 200px
}
SELECT {
    MARGIN-TOP: 3px; FONT-WEIGHT: normal; FONT-SIZE: 11px;
FONT-FAMILY: Verdana, Geneva, Arial, Helvetica, sans-serif
}
#searchcontainer {
    BORDER-TOP: #eddeded 1px solid; MARGIN-TOP: 15px; DISPLAY:
block; MARGIN-BOTTOM: 10px; PADDING-BOTTOM: 13px; MARGIN-LEFT:
12px; WIDTH: 180px; PADDING-TOP: 6px; BORDER-BOTTOM: #eddeded
1px solid
}

```

```

}

#search {
    BORDER-RIGHT: #d9d9d9 1px solid; PADDING-RIGHT: 3px;
    BORDER-TOP: #999 1px solid; MARGIN-TOP: 3px; PADDING-LEFT:
    3px; FONT-SIZE: 11px; BACKGROUND: #ffffff; PADDING-BOTTOM:
    3px; BORDER-LEFT: #999 1px solid; PADDING-TOP: 3px; BORDER-
    BOTTOM: #d9d9d9 1px solid
}
TEXTAREA {
    BORDER-RIGHT: #d9d9d9 1px solid; PADDING-RIGHT: 3px;
    BORDER-TOP: #999 1px solid; MARGIN-TOP: 3px; PADDING-LEFT:
    3px; FONT-SIZE: 11px; BACKGROUND: #ffffff; PADDING-BOTTOM:
    3px; BORDER-LEFT: #999 1px solid; PADDING-TOP: 3px; BORDER-
    BOTTOM: #d9d9d9 1px solid
}
#submitbutton {
    VERTICAL-ALIGN: bottom
}
.searchtitle {
    FONT-WEIGHT: bold; FONT-SIZE: 9px; TEXT-TRANSFORM:
    uppercase; COLOR: #8a330d
}
#intsearchcontainer {
    PADDING-RIGHT: 0px; DISPLAY: block; PADDING-LEFT: 0px;
    PADDING-BOTTOM: 0px; MARGIN: 25px 0px 10px 3px; WIDTH: 155px;
    PADDING-TOP: 0px
}
#intsearch {
    BORDER-RIGHT: #d9d9d9 1px solid; PADDING-RIGHT: 3px;
    BORDER-TOP: #999999 1px solid; PADDING-LEFT: 3px; FONT-SIZE:
    11px; BACKGROUND: #ffffff; MARGIN-BOTTOM: 3px; PADDING-BOTTOM:
    3px; BORDER-LEFT: #999999 1px solid; WIDTH: 135px; PADDING-
    TOP: 3px; BORDER-BOTTOM: #d9d9d9 1px solid
}
TEXTAREA {
    BORDER-RIGHT: #d9d9d9 1px solid; PADDING-RIGHT: 3px;
    BORDER-TOP: #999999 1px solid; PADDING-LEFT: 3px; FONT-SIZE:
    11px; BACKGROUND: #ffffff; MARGIN-BOTTOM: 3px; PADDING-BOTTOM:
    3px; BORDER-LEFT: #999999 1px solid; WIDTH: 400px; PADDING-
    TOP: 3px; BORDER-BOTTOM: #d9d9d9 1px solid
}
#intsubmitbutton {
    VERTICAL-ALIGN: bottom
}
.intsearchtitle {
    FONT-WEIGHT: bold; FONT-SIZE: 9px; TEXT-TRANSFORM:
    uppercase; COLOR: #8a330d
}

```

```
}
```

.mtable {  
 BORDER-RIGHT: #ed7c6a 0px solid; BORDER-TOP: #ed7c6a 0px  
solid; BORDER-LEFT: #ed7c6a 0px solid; BORDER-BOTTOM: #ed7c6a  
0px solid; BACKGROUND-COLOR: #ffffff  
}

### **config.php**

```
<?  
  
$user='root';  
  
$host='localhost';  
  
$pass='root';  
  
$database='evolu';  
  
$table = "shoutbox";  
  
/////////////////////////////  
  
mysql_connect($host,$user,$pass) or die("Couldnt connect, try again. ".  
mysql_error());  
  
mysql_select_db($database) or die("Cannot select database! Please Try again.  
". mysql_error());  
  
?>
```

### **functions.php**

```
<?  
  
function  
select_($sql,$select_name,$selected_value,$initial_value,$val1,$val2,$post)  
{  
  
$result=mysql_query($sql);  
  
?>  
  
<select name="<? print $select_name;?>">  
  
<?  
  
if(!empty($post))
```

```

{
    while($row=mysql_fetch_array($result))
    {
        print'<option ';
        if($row[$val1]==$post)
        {
            print'selected="selected"';
        }
        print'<option ';
        print'value="";
        print $row[$val1];
        print">";
        print $row[$val2];
        print'</option>';
    }
}

elseif(!empty($initial_value))
{
    print'<option value="" selected="selected">'.$initial_value.'</option>';
    while($row=mysql_fetch_array($result))
    {
        print'<option ';
        print'value="";
        print $row[$val1];
        print">";
        print $row[$val2];
        print'</option>';
    }
}

```

```

else
{
    while($row=mysql_fetch_array($result))
    {
        print'<option ';
        if($row[$val1]==$selected_value)
        {
            print'selected="selected"';
        }
        print'<option ';
        print'value="";
        print $row[$val1];
        print">";
        print $row[$val2];
        print'</option>';
    }
}
print'</select>';
?>
<?
}
?>
<?

function iesire()
{
    foreach($_SESSION as $key=>$value)
    {
        unset($_SESSION[$key]);
    }
}

```

```
}
```

```
?>
```

### **recuperare.php**

```
<? session_start();  
include_once("functions/config.php");  
include_once("functions/functions.php");  
include_once("header.php");  
include_once("menu.php");  
include_once("recuperare_main.php");  
include_once("footer.php");  
?>
```

### **recuperare\_main.php**

```
<TR>  
    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53  
HEIGHT=335 ALT="">  
    </TD>  
    <TD COLSPAN=5 align="center" valign="top" class="smallheader">  
        <h2>Recuperare parola</h2>  
        <br />  
        <?  
        if($_POST['submit']=="Trimite")&&($_POST['email']=="")  
        {  
            ?>  
            <table width="376" border="0" align="center"  
            <tr>  
                <td width="30"></td>  
                <td width="336" class="redbold">Introduceti adresa de e-mail! </td>
```

```

        </tr>
    </table>
    <?
        }

        elseif ($_POST['submit']=="Trimite")
        {
            $sql_utilizatori="select * from utilizatori where
email="".$_POST['email']."'"";

            $sql_prof="select * from prof where
resp_email="".$_POST['email']."'"";

            $result_utilizatori=@mysql_query($sql_utilizatori);

            $result_prof=@mysql_query($sql_prof);

            $nr_utilizatori=@mysql_num_rows($result_utilizatori);

            $nr_prof=@mysql_num_rows($result_prof);

            if(($nr_utilizatori!='1')&&($nr_prof!='1'))
            {
                ?>

                <table width="376" border="0" align="center">

                <tr>
                    <td width="30"></td>
                    <td width="336" class="redbold">Adresa de email nu a fost gasita in baza de
date! </td>
                </tr>
            </table>
            <?
        }

        else
        {
            ?>

            <table width="459" border="0" align="center">

```

```

<tr>
  <td width="30"></td>
  <td width="419" class="smallheader" align="center">Noua parola a fost
trimisa la adresa de email </td>
</tr>
</table>

<?
}

}

?>

<form action="recuperare.php" method="POST">
<table border="0" cellpadding="0" cellspacing="5">
  <tr>
    <td width="65">*e-mail:</td>
    <td width="182"><input type="text" size="20" name="email" <? if
(!empty($_REQUEST['email'])) {print 'value="'. $_REQUEST['user']. '"';}?> />
    &nbsp;
  </td>
</tr>
<tr>
  <td colspan="2"><br />           </td>
</tr>
<tr>
  <td align="center"><input type="submit" value="Trimite" name="submit"
/></td>
  <td align="center">&nbsp;</td>
</tr>
<tr>
  <td colspan="2"></td>
</tr>

```

```
</table></form></TD>
```

```
</TR>
```

### **inregistrare.php**

```
<? session_start();  
include_once("functions/config.php");  
include_once("functions/functions.php");  
include_once("header.php");  
include_once("menu.php");  
$sql_c="insert into cursuri  
(idcurs,iuser,denumire,descriere,ore_alocate,data_test,date1)values('".$_SESSION['idcurs']."'','".$_SESSION['nume']."'','$_REQUEST['denumire']."'','$_REQUEST['descriere']."'','$_REQUEST['ore_alocate']."'','$_REQUEST['data_test']."'",now());  
$result=mysql_query($sql_c)or die('Insert error');  
header('Location: cont.php');  
include_once("aa.php");  
include_once("footer.php");  
?>
```

### **inscriere\_main.php**

```
<TR>
```

```
    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53  
HEIGHT=335 ALT="">
```

```
        </TD>
```

```
            <TD COLSPAN=5 align="center" valign="top"  
class="smallheader">
```

```
                <table width="346" border="0">
```

```
                    <tr>
```

```
                        <td align="center"><a href="inscriere_utilizatori.php"></a></td>
```

```
                        <td align="center"><a href="inscriere_prof.php"></a></td>
```

```

</tr><br><br><br><br><br><br>
<tr>
    <td width="167" align="center"><a href="inscriere_utilizatori.php">Inscriere<br> Studenti </a></td>
    <td width="169" align="center"><a href="inscriere_prof.php">Inscriere<br> Profesori </a></td>
</tr>
</table>
</TD>
</TR>

```

### **inscriere\_prof.php**

```

<?
include_once("functions/config.php");
include_once("functions/functions.php");
include_once("header.php");
include_once("menu.php");
include_once("inscriere_prof_main.php");
include_once("footer.php");
?>

```

### **inscriere\_prof\_main.php**

```

<?
$arrErrors = array();
if (!empty($_POST)) {
    $sql_user="select * from utilizatori where iuser='".$_.POST['user']."'\"";
    $result_user=@mysql_query($sql_user);
    while($row_user=@mysql_fetch_array($result_user))
    {

```

```

$user_db=$row_user['iuser'];

}

$sql_user2="select * from prof where iuser='".$POST['user']."'";

$result_user2=@mysql_query($sql_user2);

while($row_user2=@mysql_fetch_array($result_user2))

{

$user_db2=$row_user2['iuser'];

}

if ($POST['user']==$user_db)

$arrErrors['user']='Acest utilizator exista in baza de date';

if ($POST['user']==$user_db2)

$arrErrors['user']='Acest utilizator exista in baza de date';

if ($POST['user']=="") 

$arrErrors['user']='Introduceti user';

if ($POST['pass']=="") 

$arrErrors['pass']='Introduceti parola';

if ($POST['pass']!=$POST['pass2'])

$arrErrors['pass2']='Parola nu corespunde';

if ($POST['pass']!=$POST['pass2'])

$arrErrors['pass']='Parola nu corespunde';

if ($POST['pass2']=="") 

$arrErrors['pass2']='Introduceti verificare parola';

if ($POST['nume']=="") 

$arrErrors['nume']='X';

if ($POST['prenume']=="") 

$arrErrors['prenume']='X';

if ($POST['email']=="") 

$arrErrors['email']='X';

if ($POST['dn_luna']=="")

```

```

$arrErrors['dn_luna']='X';
if($_POST['dn_zi']=="")
$arrErrors['dn_zi']='X';
if($_POST['dn_an']=="")
$arrErrors['dn_an']='X';
if($_POST['adresa']=="")
$arrErrors['adresa']='X';
if($_POST['judet']=="")
$arrErrors['judet']='X';
if($_POST['localitate']=="")
$arrErrors['localitate']='X';
if(count($arrErrors) == 0) {
$iuser=$_POST['user'];
$pass=$_POST['pass'];
$nume=$_POST['nume'];
$prenume=$_POST['prenume'];
$email=$_POST['email'];
$dn_luna=$_POST['dn_luna'];
$dn_zi=$_POST['dn_zi'];
$dn_an=$_POST['dn_an'];
$sex=$_POST['sex'];
$adresa=$_POST['adresa'];
$tara=$_POST['tara'];
$judet=$_POST['judet'];
$localitate=$_POST['localitate'];
$cod_postal=$_POST['cod_postal'];
$tel1=$_POST['tel1'];
$sql="insert into
prof(iuser,pass,nume,prenume,email,dn_an,dn_luna,dn_zi,sex,adresa,tara,judet,lo
calitate,cod_postal,tel1,date1)values('".$iuser."','".$.$pass."','".$.$nume."','".$.$prenum

```

```

e."",".$email."",".$dn_an."",".$dn_luna."",".$dn_zi."",".$sex."",".$adresa."",".$tara."
,".$judet."",".$localitate."",".$cod_postal."",".$tel1.",now());
}

$result=mysql_query($sql) or die("Error INSERT x");

header('Location: inregistrare.php');

ob_end_flush();

}

else {

// The error array had something in it. There was an error.

// Start adding error text to an error string.

$strError = '<table class="be-inst" border="0" cellpadding="4"
cellspacing="0" width="100%">

<tbody><tr><td width="4"></td>

<td width="25"></td>

<td width="4"></td>

<td><table class="be-inst" border="0" cellpadding="1"
cellspacing="0"><tbody><tr>

<td colspan="2" valign="top"><div class="be-h1">Please fix the
following errors:</div></td>

</tr><tr><td colspan="2"></td>

</tr><tr><td valign="top">';

foreach ($arrErrors as $error) {

$strError .= "<li>$error</li>";

}

$strError.= '</td></tr></tbody></table></td></tr></tbody></table>';

}

}

ob_end_flush();

?>

```

```

<TR>
    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53
HEIGHT=335 ALT="">
    </TD>
    <TD COLSPAN=5 align="center" valign="top" class="smallheader"
align="center"><h2>Creare cont </h2>
    <hr />
    * - informatii obligatorii
<form action="inscriere_prof.php" method="post">
    <table cellspacing="5" cellpadding="0" border="0" >
        <tbody>
            <tr>
                <td width="155"><span class="style1">*</span>Utilizator:</td>
                <td width="472"><input size="15" name="user"<? if
(!empty($_REQUEST['user'])) {print 'value="'. $_REQUEST['user'].'"';}?> />
                    &nbsp; <?php if (!empty($arrErrors['user'])) echo '<span class="err">'.'.$arrErrors['user']. "</span>"; ?></td>
            </tr>
            <tr>
                <td><span class="style1">*</span>Parola:</td>
                <td><input type="password" size="15" value="" name="pass" />
                    &nbsp;
                    <?php if (!empty($arrErrors['pass'])) echo '<span class="err">'.'.$arrErrors['pass']. "</span>"; ?></td>
            </tr>
            <tr>
                <td><span class="style1">*</span>Confirmare parola:</td>
                <td><input type="password" size="15" value="" name="pass2" />
                    &nbsp;
                    <?php if (!empty($arrErrors['pass2'])) echo '<span class="err">'.'.$arrErrors['pass2']. "</span>"; ?></td>

```

```

</tr>
<tr>
    <td><span class="style1">*</span>Nume:</td>
    <td><input name="nume">? if (!empty($_REQUEST['nume'])) {print
'value="'. $_REQUEST['nume'].'"';}?> />
    &nbsp;
    <?php if (!empty($arrErrors['nume'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>
<tr>
    <td><span class="style1">*</span>Prenume:</td>
    <td><input name="prenume">? if (!empty($_REQUEST['prenume'])) {print
'value="'. $_REQUEST['prenume'].'"';}?> />
    &nbsp;
    <?php if (!empty($arrErrors['prenume'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>
<tr>
    <td><span class="style1">*</span>Email:</td>
    <td><input size="30" name="email" >? if (!empty($_REQUEST['email'])) {
print 'value="'. $_REQUEST['email'].'"';}?> />
    &nbsp;
    <?php if (!empty($arrErrors['email'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>
<tr>
    <td><span class="style1">*</span>Data nasterii:</td>
    <td><?
        $sql_dnl="select * from dn_luna";
        $post_dnl=$_REQUEST['dn_luna'];
        $select_name_dnl="dn_luna";

```

```

$initial_value_dnl=" Luna ";
$selected_value_dnl="";
    $val1_dnl="id";
    $val2_dnl="val";

select_($sql_dnl,$select_name_dnl,$selected_value_dnl,$initial_value_dnl,$val1_dnl,$val2_dnl,$post_dnl);

?>

 
<?php if (!empty($arrErrors['dn_luna'])) echo ''; ?>

 
<?

$sql_dnz="select * from dn_zi";
$post_dnz=$_REQUEST['dn_zi'];

$select_name_dnz="dn_zi";
$initial_value_dnz=" Ziua ";
$selected_value_dnz="";
    $val1_dnz="id";
    $val2_dnz="val";
select_($sql_dnz,$select_name_dnz,$selected_value_dnz,$initial_value_dnz,$val1_dnz,$val2_dnz,$post_dnz);

?>

 
<?php if (!empty($arrErrors['dn_zi'])) echo ''; ?>

 
<?

$sql_dna="select * from dn_an";
$post_dna=$_REQUEST['dn_an'];

$select_name_dna="dn_an";
$initial_value_dna=" Anul ";

```

```

$selected_value_dna="";
    $val1_dna="id";
    $val2_dna="val";
select_($sql_dna,$select_name_dna,$selected_value_dna,$initial_value_dna,$val1_
dna,$val2_dna,$post_dna);

?>

&nbsp;

<?php if (!empty($arrErrors['dn_an'])) echo ''; ?></td>

</tr>

<tr>

<td><span class="style1">*</span>Sex:</td>

<td><?

if($_REQUEST['sex'])

{

switch ($_REQUEST['sex'])

{

case 'f':

echo '<input type="radio" value="b" name="sex" />

Barbat

</INPUT>

<input type="radio" value="f" checked="checked" name="sex" />

Femeie

</INPUT>';

break;

case 'b':


echo '<input type="radio" checked="checked" value="b" name="sex" />

Barbat

</INPUT>

<input type="radio" value="f" name="sex" />

Femeie
/>

```

```

        </INPUT> ';
break;
default:
}
}

else
{
?>

<input type="radio" checked="checked" value="b" name="sex" />
Barbat
<input type="radio" value="f" name="sex" />
Femeie
<?
}

?>      </td>

</tr>
<tr>
<td><span class="style1">*</span>Adresa: </td>
<td><input size="50" name="adresa" <? if (!empty($_REQUEST['adresa'])) {print 'value="'. $_REQUEST['adresa']. '"';}?> />
&nbsp;
<?php if (!empty($arrErrors['adresa'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>
<tr>
<td><span class="style1">*</span>Tara:</td>
<td>&nbsp;<?
$sql="select * from tara";
$post=$_REQUEST['tara'];
$select_name="tara";

```

```

$initial_value="--Alege--";
$selected_value="RO";
    $val1="iso";
    $val2="printable_name";
select_($sql,$select_name,$selected_value,$initial_value,$val1,$val2,$post);
?>      </td>
</tr>
<tr>
<td><span class="style1">*</span>Judet:</td>
<td><input size="15" name="judet" <?
if (!empty($judet))
{
    print 'value="'. $judet .'";
}
elseif (!empty($_REQUEST['judet']))
{
    print 'value="'. $_REQUEST['judet'] .'";
}
?> /> &nbsp;<?php if (!empty($arrErrors['judet'])) echo '<br /><span class="errortext"></span>';
?>  </td>
</tr>
<tr>
<td><span class="style1">*</span>Localitate:</td>
<td><input size="15" name="localitate" <? if
(!empty($_REQUEST['localitate'])) {print 'value="'. $_REQUEST['localitate'] ."';}?>
?>
&nbsp;
<?php if (!empty($arrErrors['localitate'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>

```

```

<tr>
    <td>&nbsp;Cod postal:</td>
    <td><input size="6" name="cod_postal" <? if
(!empty($_REQUEST['cod_postal'])) {print
'value="'. $_REQUEST['cod_postal']. '"';}?> />
    &nbsp;

    <?php if (!empty($arrErrors['cod_postal'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>

<tr>
    <td>&nbsp;Telefon (fix): </td>
    <td><input size="15" name="tel1" <? if (!empty($_REQUEST['tel1'])) {print
'value="'. $_REQUEST['tel1']. '"';}?> /></td>
</tr>

<tr>
    <td colspan="2"><br />      </td>
</tr>

<tr>
    <td align="middle"><input type="submit" value="Trimite" name="submit"
/></td>
    <td align="middle"><input type="reset" value="Reset" name="reset"
/></td>
</tr>

<tr>
    <td colspan="2"><br />
        <input type="hidden" name="inscriere" value="profesori"/></td>
</tr>

</tbody>
</table>
</form>           </TD>
</TR>

```

### **inscriere\_utilizatori.php**

```
<?

include_once("functions/config.php");
include_once("functions/functions.php");
include_once("header.php");
include_once("menu.php");
include_once("inscriere_utilizatori_main.php");
include_once("footer.php");

?>
```

### **inscriere\_utilizatori\_main.php**

```
<?

$arrErrors = array();
if (!empty($_POST)) {
    $sql_user="select * from utilizatori where iuser='". $_POST['user']."' ";
    $result_user=@mysql_query($sql_user);
    while($row_user=@mysql_fetch_array($result_user))
    {
        $user_db=$row_user['iuser'];
    }
    $sql_user2="select * from prof where iuser='". $_POST['user']."' ";
    $result_user2=@mysql_query($sql_user2);
    while($row_user2=@mysql_fetch_array($result_user2))
    {
        $user_db2=$row_user2['iuser'];
    }
    if ($_POST['user']==$user_db)
```

```

$arrErrors['user']='Acest utilizator exista in baza de date';
if($_POST['user']==$user_db2)
    $arrErrors['user']='Acest utilizator exista in baza de date';
if($_POST['user']=="")
    $arrErrors['user']='Introduceti user';
if($_POST['pass']=="")
    $arrErrors['pass']='Introduceti parola';
    if($_POST['pass']!=$_POST['pass2'])
        $arrErrors['pass2']='Parola nu corespunde';
        if($_POST['pass']!=$_POST['pass2'])
            $arrErrors['pass']='Parola nu corespunde';
if($_POST['pass2']=="")
    $arrErrors['pass2']='Introduceti verificare parola';
if($_POST['nume']=="")
    $arrErrors['nume']='X';
    if($_POST['prenume']=="")
        $arrErrors['prenume']='X';
        if($_POST['email']=="")
            $arrErrors['email']='X';
            if($_POST['dn_luna']=="")
                $arrErrors['dn_luna']='X';
                if($_POST['dn_zi']=="")
                    $arrErrors['dn_zi']='X';
                    if($_POST['dn_an']=="")
                        $arrErrors['dn_an']='X';
                        if($_POST['adresa']=="")
                            $arrErrors['adresa']='X';
                            if($_POST['judet']=="")
                                $arrErrors['judet']='X';

```

```

if ($_POST['localitate']=="")
$arrErrors['localitate']='X';

if (count($arrErrors) == 0) {
$iuuser=$_POST['user'];
$pass=$_POST['pass'];
$nume=$_POST['nume'];
$prenume=$_POST['prenume'];
$email=$_POST['email'];
$dn_luna=$_POST['dn_luna'];
$dn_zi=$_POST['dn_zi'];
$dn_an=$_POST['dn_an'];
$sex=$_POST['sex'];
$adresa=$_POST['adresa'];
$tara=$_POST['tara'];
$judet=$_POST['judet'];
$localitate=$_POST['localitate'];
$cod_postal=$_POST['cod_postal'];
$tel1=$_POST['tel1'];

$sql="insert into
utilizatori(iuser,pass,nume,prenume,email,dn_an,dn_luna,dn_zi,sex,adresa,tara,ju
det,localitate,cod_postal,tel1,date1)values('".$iuuser."','".$$.pass."','".$$.nume."','".$$.pr
enume."','".$$.email."','".$$.dn_an."','".$$.dn_luna."','".$$.dn_zi."','".$$.sex."','".$$.adresa."','".$$.d
tara."','".$$.judet."','".$$.localitate."','".$$.cod_postal."','".$$.tel1."',now())";

$result=mysql_query($sql) or die("Error INSERT x");

header('Location: reusire.php');
ob_end_flush();

}

else {

// The error array had something in it. There was an error.

// Start adding error text to an error string.

```

```

$strError = '<table class="be-inst" border="0" cellpadding="4"
cellspacing="0" width="100%>          <tbody><tr><td width="4"></td>

<td width="25"></td>

<td width="4"></td>

<td><table class="be-inst" border="0" cellpadding="1"
cellspacing="0"><tbody><tr>

<td colspan="2" valign="top"><div class="be-h1">Please fix the
following errors:</div></td>

</tr><tr><td colspan="2"></td>

</tr><tr><td valign="top">';

foreach ($arrErrors as $error) {

    $strError .= "<li>$error</li>";

}

$strError.= '</td></tr></tbody></table></td></tr></tbody></table>';

}

}

ob_end_flush();

?>

<TR>

    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53
HEIGHT=335 ALT="">

    </TD>

    <TD COLSPAN=5 align="center" valign="top" class="smallheader"
align="center"><h2>Creare cont </h2>

    <hr />

* - informatii obligatorii

<form action="inscriere_utilizatori.php" method="post">

<table cellspacing="5" cellpadding="0" border="0" >

<tbody>

```

```

<tr>
    <td width="155"><span class="style1">*</span>Utilizator:</td>
    <td width="472"><input size="15" name="user">? if
        (!empty($_REQUEST['user'])) {print 'value="'. $_REQUEST['user'].'"';}?> />
        &nbsp; <?php if (!empty($arrErrors['user'])) echo '<span class="err">'.'.$arrErrors['user']. "</span>"; ?></td>
</tr>

<tr>
    <td><span class="style1">*</span>Parola:</td>
    <td><input type="password" size="15" value="" name="pass" />
        &nbsp;
        <?php if (!empty($arrErrors['pass'])) echo '<span class="err">'.'.$arrErrors['pass']. "</span>"; ?></td>
</tr>

<tr>
    <td><span class="style1">*</span>Confirmare parola:</td>
    <td><input type="password" size="15" value="" name="pass2" />
        &nbsp;
        <?php if (!empty($arrErrors['pass2'])) echo '<span class="err">'.'.$arrErrors['pass2']. "</span>"; ?></td>
</tr>

<tr>
    <td><span class="style1">*</span>Nume:</td>
    <td><input name="nume">? if (!empty($_REQUEST['nume'])) {print
        'value="'. $_REQUEST['nume'].'"';}?> />
        &nbsp;
        <?php if (!empty($arrErrors['nume'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>

<tr>
    <td><span class="style1">*</span>Prenume:</td>

```

```

<td><input name="prenume">? if (!empty($_REQUEST['prenume'])) {print
'value="'. $_REQUEST['prenume'].'"';}?> />

<?php if (!empty($arrErrors['prenume'])) echo '<br /><span class="errortext"></span>'; ?></td>

</tr>

<tr>

<td><span class="style1">*</span>Email:</td>

<td><input size="30" name="email" >? if (!empty($_REQUEST['email'])) {
print 'value="'. $_REQUEST['email'].'"';}?> />

<?php if (!empty($arrErrors['email'])) echo '<br /><span class="errortext"></span>'; ?></td>

</tr>

<tr>

<td><span class="style1">*</span>Data nasterii:</td>

<td><?

$sql_dnl="select * from dn_luna";

$post_dnl=$_REQUEST['dn_luna'];

$select_name_dnl="dn_luna";

$initial_value_dnl=" Luna ";

$selected_value_dnl="";

$val1_dnl="id";

$val2_dnl="val";

select_($sql_dnl,$select_name_dnl,$selected_value_dnl,$initial_value_dnl,$val1_dnl,$val2_dnl,$post_dnl);

?>

<?php if (!empty($arrErrors['dn_luna'])) echo ''; ?>

<?

```

```

$sql_dnz="select * from dn_zi";

$post_dnz=$_REQUEST['dn_zi'];

$select_name_dnz="dn_zi";
$initial_value_dnz=" Ziua ";
$selected_value_dnz="";
$val1_dnz="id";
$val2_dnz="val";
select_($sql_dnz,$select_name_dnz,$selected_value_dnz,$initial_value_dnz,$val1_dnz,$val2_dnz,$post_dnz);

?>

 
<?php if (!empty($arrErrors['dn_zi'])) echo ''; ?>

 
<?

 
$sql_dna="select * from dn_an";
$post_dna=$_REQUEST['dn_an'];

$select_name_dna="dn_an";
$initial_value_dna=" Anul ";
$selected_value_dna="";
$val1_dna="id";
$val2_dna="val";
select_($sql_dna,$select_name_dna,$selected_value_dna,$initial_value_dna,$val1_dna,$val2_dna,$post_dna);

?>

 
<?php if (!empty($arrErrors['dn_an'])) echo ''; ?></td>

</tr>
<tr>
<td><span class="style1">*</span>Sex:</td>

```

```

<td><?

if($_REQUEST['sex'])

{

switch ($_REQUEST['sex'])

{



case 'f':

echo '<input type="radio" value="b" name="sex" />

Barbat

</INPUT>

<input type="radio" value="f" checked="checked" name="sex" />

Femeie

</INPUT>';

break;

case 'b':

echo '<input type="radio" checked="checked" value="b" name="sex" />

Barbat

</INPUT>

<input type="radio" value="f" name="sex" />

Femeie

</INPUT>';

break;

default:

}

}

else

{



?>

<input type="radio" checked="checked" value="b" name="sex" />

```

Barbat

<input type="radio" value="f" name="sex" />

Femeie

<?

}

?> </td>

</tr>

<tr>

<td><span class="style1">\*</span>Adresa: </td>

<td><input size="50" name="adresa" <? if (!empty(\$\_REQUEST['adresa'])) {print 'value="'. \$\_REQUEST['adresa'].'"';}?> />

&ampnbsp

<?php if (!empty(\$arrErrors['adresa'])) echo '<br /><span class="errortext"></span>'; ?></td>

</tr>

<tr>

<td><span class="style1">\*</span>Tara:</td>

<td>&ampnbsp<?

```

$sql="select * from tara";
$post=$_REQUEST['tara'];
$select_name="tara";
$initial_value="--Alege--";
$selected_value="RO";
$val1="iso";
$val2="printable_name";
select_($sql,$select_name,$selected_value,$initial_value,$val1,$val2,$post);
?> </td>
</tr>
<tr>
<td><span class="style1">*</span>Judet:</td>

```

```

<td><input size="15" name="judet" <?
    if (!empty($judet))
    {
        print 'value="'. $judet .'";
    }
    elseif (!empty($_REQUEST['judet']))
    {
        print 'value="'. $_REQUEST['judet'] .'";
    }
    ?> /> &nbsp;<?php if (!empty($arrErrors['judet'])) echo '<br /><span class="errortext"></span>';
?> </td>
</tr>

<tr>
    <td><span class="style1">*</span>Localitate:</td>
    <td><input size="15" name="localitate" <? if
(!empty($_REQUEST['localitate'])) {print 'value="'. $_REQUEST['localitate'] ."';}?:>
/>
    &nbsp;
    <?php if (!empty($arrErrors['localitate'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>

<tr>
    <td>&nbsp;Cod postal:</td>
    <td><input size="6" name="cod_postal" <? if
(!empty($_REQUEST['cod_postal'])) {print
'value="'. $_REQUEST['cod_postal'] ."';}?:> />
    &nbsp;
    <?php if (!empty($arrErrors['cod_postal'])) echo '<br /><span class="errortext"></span>'; ?></td>
</tr>
<tr>

```

```

<td>&nbsp;Telefon (fix): </td>
<td><input size="15" name="tel1" <? if (!empty($_REQUEST['tel1'])) {print
'value="'. $_REQUEST['tel1'].'";}?> /></td>
</tr>

<tr>
<td colspan="2"><br /> </td>
</tr>
<tr>
<td align="middle"><input type="submit" value="Trimite" name="submit"
/></td>
<td align="middle"><input type="reset" value="Reset" name="reset"
/></td>
</tr>
<tr>
<td colspan="2"><br />
<input type="hidden" name="inscriere" value="utilizatori"/></td>
</tr>
</tbody>
</table>
</form> </TD>
</TR>

```

### Contact.php

```

<?
include_once("header.php");
include_once("menu.php");
include_once("contact_main.php");
include_once("footer.php");
?>

```

### Contact\_main.php

<TR>

<TD COLSPAN=2 background ="images/207\_13.gif" WIDTH=53  
HEIGHT=335 ALT="">

<TD COLSPAN=5 align="center" valign="top" >

<h2><strong>CONTACT :</h2>

<p>Sediul central:<br>

Jud. Satu Mare, Loc. Satu Mare.<br>

str. Sagetatorului, bl. L3/9<br>

Tel: +40742093243<br>

FAX: +40361819209<br>

</p></strong>

Ne puteti contacta pe <a href="http://localhost:8888/phpBB3/"> forum</a>  
 sau <a href="mailto:contact@evolutio.ro">e-mail</a> <br />

<br />

<span class="style1">Toate campurile sunt obligatorii !!! Va rugam sa completati formularul in intregime!<br><br> </span>

<form action="processor.php" method="post" name="form1" class="formular">

<strong>Nume:</strong>

<input name='ams[Nume]' type="text" class="textfield" />

&nbsp;<strong>Prenume</strong>:<input name='ams[Prenume]' type="text" class="textfield" /><br /><br />

<strong>Email</strong>:

<input name='ams[E-mail]' type="text" class="textfield" /><strong>Telefon</strong>:<input name='ams[Tel]' type="text" class="textfield" />

<br /><br />

<strong>Mesajul dumneavoastră:</strong><br />

<textarea name='ams[Mesajul\_dumneavostra]' id="textarea" cols="58" rows="10"></textarea>

<br />

<br />

```

<input name="Submit2" type="reset" value="Anuleaza" class="input" />
<input name="Submit" type="submit" value="Trimite" class="input" />
</center>
</form><br />
</TD>
</TD>
</TR>

```

### **Processor.php**

```

<meta http-equiv="refresh" content="3; url=contact.php">
<?
$to = "ulicigabriel@gmail.com";
$subject = "Formular contact online: {$_POST['ams']['Destinatia']} ";
$from = $_POST['ams']['E-mail'];
$valid=1;
$message ='';
foreach ($_POST['ams'] as $k=>$v){
if (trim($v)=='')$valid=0;
$k = str_replace('_', ' ', $k);
$message .="$k : $v<br>";
}
$message .= "<hr />IP: {$_SERVER['REMOTE_ADDR']}";
function sndmail($from,$to,$subject,$message){
$headers = "MIME-Version: 1.0\r\n";
$headers .= "Content-type: text/html; charset=iso-8859-1\r\n";
$headers .= "From:".$from ."\r\n";
mail($to, $subject, $message, $headers);
}

```

```

if ($valid=='1') {
    sndmail($from,$to,$subject,$message);
    print('Email trimis');
}
else Print('Email ne trimis') ;
?>

```

### **Cursuri.php**

```

<? session_start();
include_once("functions/config.php");
include_once("functions/functions.php");
include_once("header.php");
include_once("menu.php");
include_once("cursuri_main.php");
include_once("footer.php");
?>

```

### **Cursuri\_main.php**

```

<link href="css/main_stylesnew.css" rel="stylesheet" type="text/css" />
<TR>
    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53
HEIGHT=335 ALT="">
        </TD>
        <TD COLSPAN=5 align="center" valign="top"
class="smallheader">
            <h2>Afisare Cursuri </h2>
            <hr />
            <?
            if($_POST['submit'])
            {
                ?>
                <h4>Documentatie</h4><br>

```

```

<iframe height="300" width="540" scrolling=auto src="doc.php"
name="documentatie" style="font: Verdana"></iframe>

<br><br>

        <?

    }

?>

        <?

$num = $_GET['num'];

if(empty($num))

{

$num = 1;

};

$limit = 1;

$start = ($num-1)*$limit;

$start = round($start,0);//rounds the result

print $_REQUEST['search'];

if(isset($_REQUEST['search']) )

{

$query="select * from cursuri where denumire like '%".$_GET['search']."%' or
descriere like '%".$_GET['search']."%' or iuser like '%".$_GET['search']."%' order
by date1 LIMIT $start, $limit";

$totalpages = mysql_num_rows(mysql_query("SELECT * from cursuri where
denumire like '%".$_GET['search']."%' or descriere like '%".$_GET['search']."%' or
iuser like '%".$_GET['search']."%'"));

}

else

{

$query = "SELECT * FROM cursuri ORDER by date1 LIMIT $start, $limit";

$totalpages = mysql_num_rows(mysql_query("SELECT * from cursuri"));

}

$result = mysql_query($query);

```

```

$totalpages = $totalpages / $limit;
$totalpages = round($totalpages,0);
$c = 0;
echo "<br>";
if($_GET['num']!=1)
{
    $x=$_GET['num'];
    $prev=$x-1;

    print"<a href=\"$_SERVER['PHP_SELF']."?num=".$prev."&search=".$_REQUEST['search']."'>PREV</a>";
}
else
{
    print"PREV";
}
print" ";
for($i = 1; $i <= $totalpages; $i++){
    if($i == $_GET['num']){
        print"<font color='red'>".$i."</font>&ampnbsp";
    }
    else{
        print"<a href=\"$_SERVER['PHP_SELF']."?num=".$i."&search=".$_REQUEST['search']."'>$i</a>";
    }
}
print" ";
if($totalpages>$_GET['num'])
{
    $x=$_GET['num'];
    $next=$x+1;
}

```

```

    print"<a
href=\"$_SERVER['PHP_SELF']."?num=\"$next."&search=\"$_REQUEST['search']."
>NEXT</a>";
}

else

{
print"NEXT";

}

print"<br><br>";

print'<table border="1" align="center" cellpadding="3"
cellspacing="2" bordercolor="#000066" class="introparagraph">';

while($row=mysql_fetch_array($result))

{

?>

<tr>

<td class="newstitle"><b>denumire</b></td>

<td class="newstitle"><? print $row['denumire'];?></td>

</tr>

<tr>

<td class="newstitle"><b>ore alocate</b></td>

<td class="newstitle"><? print $row['ore_alocate'];?></td>

</tr>

<tr>

<td class="newstitle"><b>data test</b></td>

<td class="newstitle"><? print $row['data_test'];?></td>

</tr>

<tr>

<td><b>data adaugarii</b></td>

<td class="newstitle"><? print $row['date1'];?></td>

</tr>

```

```

<tr>
    <td><b>descriere</b></td>
    <td class="newstitle"><? print $row['descriere'];?></td>
</tr>

<?
}

?>

</table>
<br />
<?

if(($_SESSION['tip_utilizatori']=='utilizatori')&&($_SESSION['id'])) {
    ?>
        <form action="cursuri.php" method="post"
name="documentatie"><input name="submit" type="hidden" value="dd"
/><input type="submit" name="Submit" value="Documentatie"></form>
    <?
}
?>

<br />
</TD>
</TR>

```

### **Cursuri\_disp.php**

```

<? session_start();>
include_once("functions/config.php");
include_once("functions/functions.php");
include_once("header.php");
include_once("menu.php");
include_once("cursuri_disp_main.php");
include_once("footer.php");

```

?>

## Cursuri\_disp\_main.php

```
<link href="css/main_stylesnew.css" rel="stylesheet" type="text/css" />

<TR>

    <TD COLSPAN=2 background ="images/207_13.gif" WIDTH=53
HEIGHT=335 ALT="">

    </TD>

<TD COLSPAN=5 align="center" valign="top" class="smallheader">

    <h2>Afisare Cursuri</h2 >

    <hr />

    <?

    $search=$_REQUEST['search'];

    ?>

    <table width="414" border="0" cellpadding="5" cellspacing="0"
class="h2">

    <tr>

        <td width="10"><strong>Nr.</strong></td>

        <td width="77"><strong>Denumire</strong></td>

        <td width="64"><strong>Ore alocate</strong></td>

        <td width="77"><strong>Data test</strong></td>

        <td width="110"><strong>Data introduceri</strong></td>

    </tr>

    <?

    $sql="select * from cursuri where denumire like
'%".$_GET['search']."' or descriere like '%".$_GET['search']."' or iuser like
'%".$_GET['search']."'%'order by date1";

    $result=mysql_query($sql);

    $i=0;

    while($row=mysql_fetch_array($result))

        {$i++;

        if($i%2==0)
```

```

{$bgcolor="#FFFFFF";}
else
{$bgcolor="#C5CBC7;"}
print"<tr>";
print"<td bgcolor=\"$bgcolor.\"><a
href=cursuri.php?num=\"$i.\"&search=\"$search.\">$i.</a></td>";
print"<td bgcolor=\"$bgcolor.\"><a
href=cursuri.php?num=\"$i.\"&search=\"$search.\">$row['denumire'].</a></td
>";
print"<td bgcolor=\"$bgcolor.\">$row['ore_alocate'].</td>";
print"<td bgcolor=\"$bgcolor.\">$row['data_test'].</td>";
print"<td bgcolor=\"$bgcolor.\">$row['date1'].</td>";
print"</tr>";
}

?>
</table> <br><br> </TD>
</TR>

```