

Proiect DAW

*Studenti : Cocan Alexandru
Țoghe Răzvan Constantin*

Semigrupa: 223/2

Sesiunea : Iunie 2020

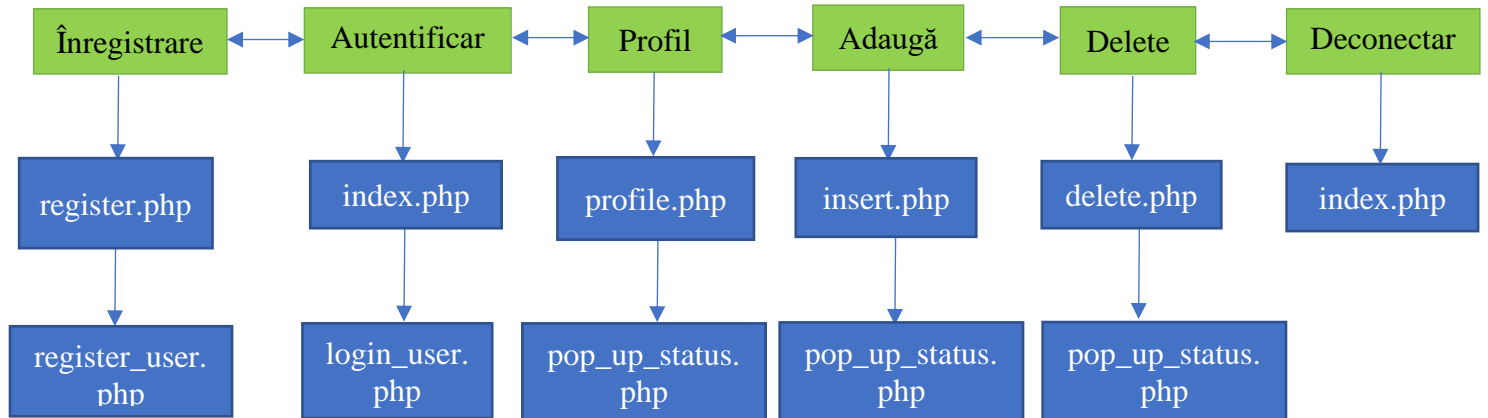
Cuprins

1. Obiective.....	3
2. Structura aplicației.....	4
3. Structura bazei de date	5
4. Descriere module și exemple de cod.....	6

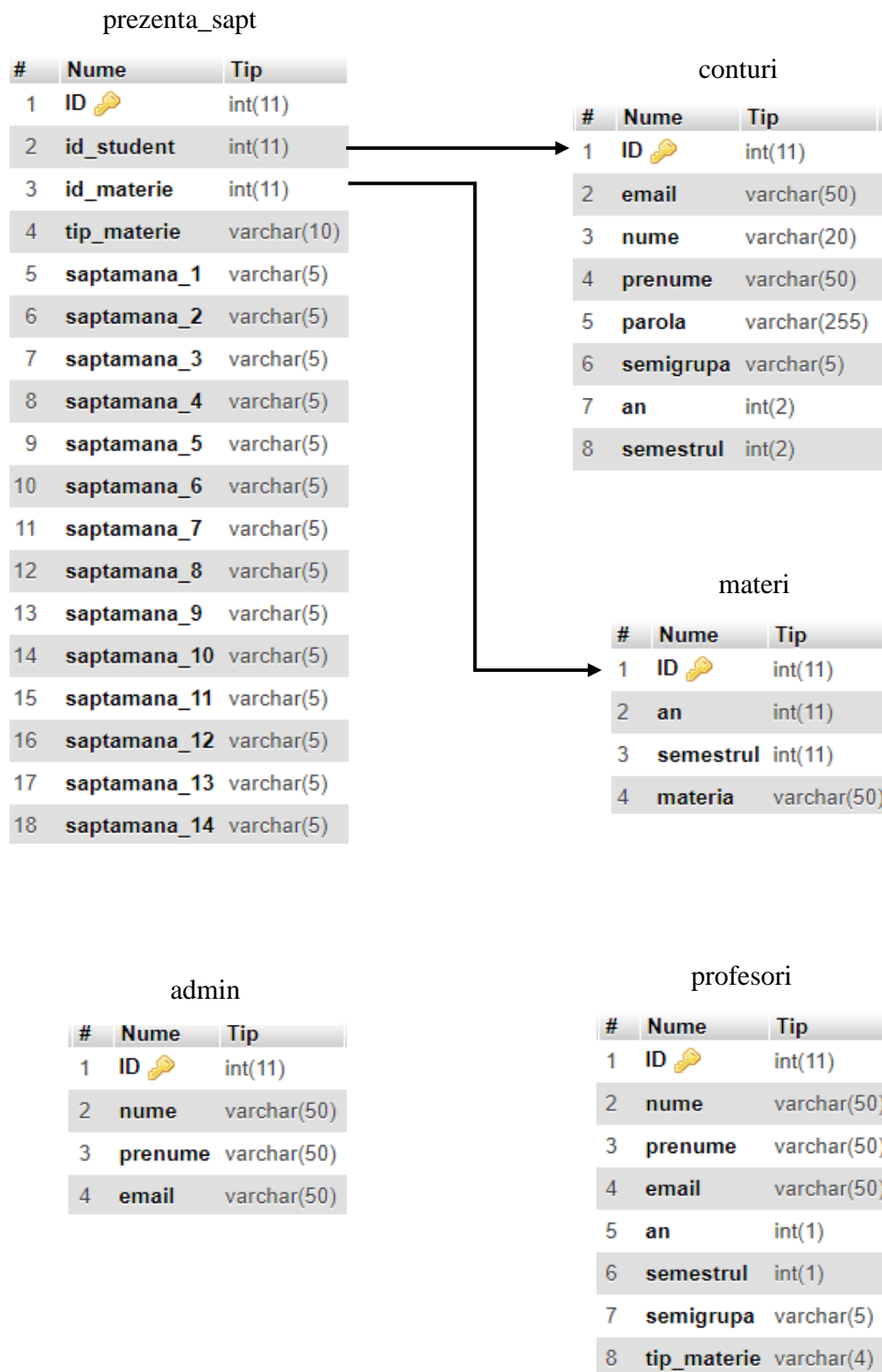
1.Objective

- ✓ Crearea unei baze de date cu mai mult de 2 tabele.
 - ✓ Crearea unei pagini de înregistrare utilizator.
 - ✓ Identificarea și blocarea creării unor conturi multiple.
 - ✓ Utilizatorii se pot înregistra doar cu cont de ULBS.
 - ✓ Criptarea parolei la înregistrarea unui utilizator în baza de date.
 - ✓ Protejarea datelor împotriva SQL injection.
 - ✓ Crearea unei pagini de autentificare utilizator.
 - ✓ Crearea unui template pentru meniu dinamic.
 - ✓ Afișarea unui tabel cu prezențe în pagina principală.
 - ✓ Posibilitatea ca profesorul să adauge prezențe studenților.
 - ✓ Posibilitatea ca adminul să șteargă un student din baza de date.
 - ✓ Folosirea JavaScript pentru afișarea unui panou cu date pentru student, profesor și admin.
 - ✓ Folosirea de JavaScript pentru a afișa semigrupele și materiile din baza de date și pentru a memora selecția utilizatorului
 - ✓ Generare de Clean URLs.
-
- Protejarea codului împotriva XSS.
 - Optimizarea pentru motoarele de căutare

2.Structura aplicației



3.Structura bazei de date



4.Descriere module și exemple de cod

Modulul register_user.php

Se pornește o sesiune apoi se verifică dacă a fost trimis formularul de înregistrare, se protejează datele împotriva SQL injection ca în exepul următor:

```
$email = mysqli_real_escape_string($con, $_POST['email']);
```

Se verifică dacă datele au fost completate corespunzător, în caz contrar se adaugă într-un array un mesaj de eroare care v-a fi ulterior afișat pe pagina de înregistrare:

```
if (strpos($email, '@ulbsibiu.ro') == false) {  
    array_push($erori, "Adresa de email nu este de ulbs");  
}
```

Dacă datele au fost completate corespunzător, se criptează parola conform:

```
$parola = password_hash($parola, PASSWORD_DEFAULT);
```

Se introduce apoi utilizatorul în baza de date, mai exact în tabela *conturi* și în tabela *prezenta_sapt* fără prezențe apoi se redirecționează utilizatorul către pagina de login.

Modulul login_user.php

Se pornește o sesiune apoi se verifică dacă a fost trimis formularul de autentificare, se protejează datele împotriva SQL injection și se verifică dacă datele au fost completate corespunzător, în caz contrar se adaugă într-un array un mesaj de eroare care v-a fi ulterior afișat pe pagina de autentificare.

Se face o interogare în baza de date pentru a verifica dacă există adresa de email introdusă, dacă există se decriptează parola și se redirecționează către pagina principală de profil:

```
$parola = mysqli_real_escape_string($con, $_POST['parola']);  
$users = mysqli_query($con, "select * from conturi where email = '$email'");  
$row = mysqli_fetch_array($users);  
$parola = password_verify($parola, $row['parola']);
```

Dacă nu există adresa de email în baza de date, se adaugă într-un array un mesaj de eroare care v-a fi afișat pe pagina de autentificare

Modulul meniu.php

Meniul este unul dinamic, cu link-uri separate pentru student, profesor și admin.

Studentul are acces doar la pagina de *home(profile.php)*, unde poate vedea prezențele tuturor studenților și la un buton aflat în partea dreapta sus, cu numele utilizatorului scris pe buton, care deschide la click o fereastră cu prezențele studentului la cursuri și laboratoare și link pentru a se deconecta.

Profesorul are acces la pagina de *home(profile.php)* și *insert(insert.php)* unde poate modifica prezențele studentului, dar și la un buton aflat în partea dreapta sus, cu numele utilizatorului, care deschide la click o fereastră cu top 5 absenți la cursuri și laboratoare și link pentru a se deconecta.

Adminul are acces la pagina de *home(profile.php)*, *insert(insert.php)*, *delete(delete.php)*, unde poate să șteargă din baza de date un cont, *adaugă profesor (adauga_profesor)* dar și la un buton aflat în partea dreapta sus, cu numele utilizatorului, care deschide la click o fereastră și link pentru a se deconecta.

Modulul profile.php

Este un modul care poate fi accesat de către toți utilizatorii (studenți, profesori, admini). În acest modul este afișat un tabel cu prezențele studenților la cursuri, tabel ce necesită interogarea a 3 tabele simultan din baza de date:

```
$sql_prezenta_sapt =  
'SELECT  
    C.*,  
    M.materia,  
    M.ID,  
    P.*  
FROM  
    conturi C,  
    materi M,  
    prezenta_sapt P  
WHERE  
    C.ID = P.id_student and  
    M.ID = P.id_materie';
```

Dacă în tabela *prezenta_sapt* apare în coloanele cu saptamani 'P', se v-a afișa un pătrat verde, în caz contrar se v-a afișa un pătrat roșu.

Modulul insert.php

Este un modul care poate fi accesat de către profesori și admini. În acest modul utilizatorul trebuie să aleagă anul, semestrul, semigrupa, tipul materiei și materia la care vrea să adauge prezențe, iar la apăsarea butonului Caută, v-a apărea un tabel corespunzător datelor selectate.

Datele introduse rămân selectate cu ajutorul scripturilor PHP și JavaScript, semigrupa și materiile sunt completate automat cu ajutorul JavaScript, care la modificarea unei valori din câmpurile an și semestru(transmise din pagina *insert.php* în script), se face o interogare în baza de date cu acele valori, se salvează într-un array, se transmit în format json către script de unde sunt ulterior preluate și completate la select că opțiuni.

După ce tabelul cerut a fost afișat pe ecran, utilizatorul are la dispoziție o serie de checkbox-uri care au culoarea verde pentru prezență și respectiv culoarea roșie pentru absență. Aceste valori sunt preluate automat din tabela *prezenta_sapt* la afișare, iar profesorul poate modifica acea prezență, schimbând culoarea checkbox-ului într-o săptămână anume.

După ce profesorul a terminat de modificat prezențele, trebuie să apese pe butonul formularului din dreapta jos a paginii numit Adaugă și datele vor fi adăugate automat în baza de date.

La începutul paginii *insert.php* a fost inclus un fișier *update.php* care conține scriptul de adăugare prezențe (caracterul 'P' în săptămâna corespunzătoare) în tabela *prezenta_sapt*, dar și codul pentru ștergerea unui student din tabelele *conturi* și *prezenta_sapt* de către admin.

Modulul delete.php

Este un modul care poate fi accesat de către admini. În acest modul utilizatorul trebuie să aleagă anul, semestrul și semigrupa pentru a afișa tabela conturilor corespunzătoare, apoi are la dispoziție o coloană cu un link de ștergere a studentului din baza de date. Și acest modul se folosește de fișierul *update.php* la fel ca modulul insert.

Modulul adauga_profesor

Este un modul care poate fi accesat de către admini. În acest modul utilizatorul trebuie să introducă datele numele ,prenumele, adresa de email ale profesorului, anul, semestrul semigrupa, tipul materiei și materia la care este repartizat, apoi apasă pe butonul de adaugă pentru a introduce datele în tabela profesori.

Modulul log_out.php

Acest modul poate fi accesat de către toți utilizatorii și are rolul de a distruge sesiunea curentă și de a redirecționa utilizatorii către pagina de login.

Parte publică

Modulele *register*, *index*, *meniu*, *log_out* și *profile*.

Parte privată

Modulele *insert*, *meniu*, *delete*, și *adaugă profesor*.

Bibliografie

Design butoane:

<https://youtu.be/ex7jGbyFgpA>

Design înregistrare/autentificare:

<https://youtu.be/BKhoo6RSEDU>

https://youtu.be/t-EMinSz_Tk

SQL injection:

<https://youtu.be/lGYixKGiY7Y?t=460>

JavaScript:

<https://youtu.be/TNPxG2yrPlM>

<https://youtu.be/KX0NR5HtQ1Q>

.htaccess:

<https://youtu.be/-6LyG9I-FPc>