**sUPdo**

**Argument**

sUPdo este destinat pentru utilizarea la clasă de catre profesori, pentru a primi si trimite fișiere elevilor, aceștia fiind structurați pe clase. În comparație cu modalitatea de a trimite fișiere pe mail sau prin intermediul serverului ftp, platforma creeată de mine facilitează o metodă ușoară și sigură prin care elevul nu poate copia sau prelua surse de la alți colegi. Soft’ul nu poate fi folosit doar in scopul de a vizualiza diferite resurse, ci și pentru trimiterea unor teme sau proiecte și comunicarea dintre elev și profesor.

Platforma facilitează comunicarea elev-profesor, permițând schimbul de fișiere si mesaje între acestia: elevii pot vizualiza și trimite fișiere (lucrări, probleme, documente) și mesaje către profesor, iar profesorul poate la rândul lui să vizualizeze și să trimită fișiere către acesta.

**Arhitectura aplicației**

Înregistrare

Autentificare

Trimitere fișiere către elev

Vizualizare fișiere trimise de elevi

Modul profesor:

Modul elev:

Înregistrare

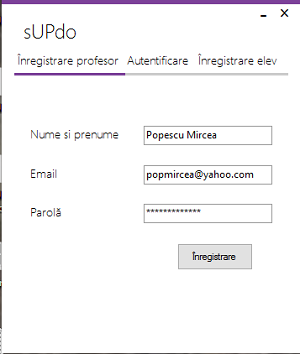
Autentificare

Trimitere fișiere către profesor

Vizualizare fișiere trimise de profesor

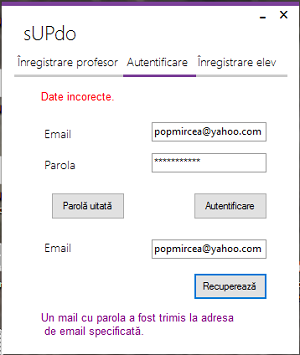
**Descrierea aplicației**

Soft-ul permite înregistrarea și logarea în 2 conturi diferite, de profesor și de elev.

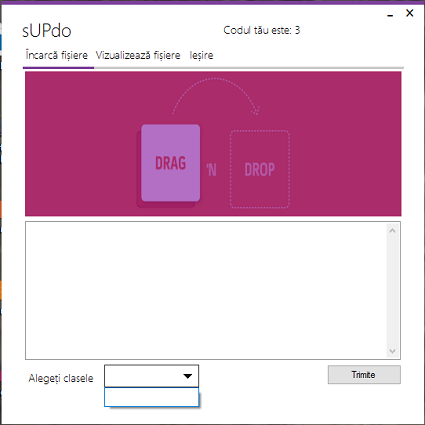
* Înregistrarea pentru profesori cuprinde date minimale despre acesta: numele, adresa de email si parola.

Înregistrarea prezintă toate erorile necesare (câmpuri necompletate, email greșit)

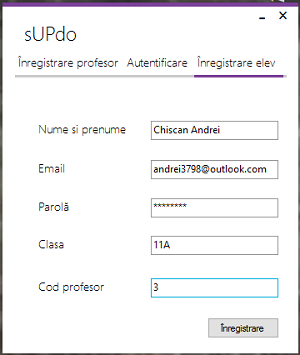
* Autentificarea solicită email-ul și parola utilizatorului.
* În cazul în care utilizatorul și-a uitat parola, va putea s-o recupereze pe baza adresei de email, acesta primind un mesaj cu parola pe mail.



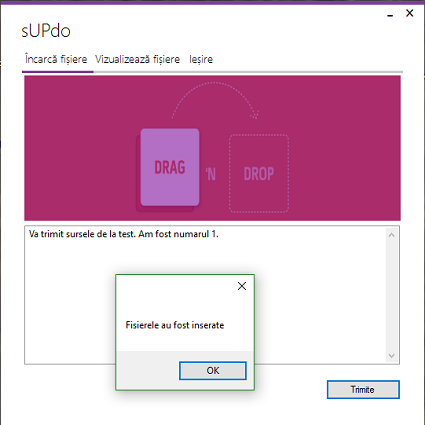
* După autentificare, profesorul va putea trimite fișiere către elevi și să vizualizeze fișierele primite de la aceștia, numai ca în cazul de fața, în clasele profesorului nu exista nici un elev, încă.



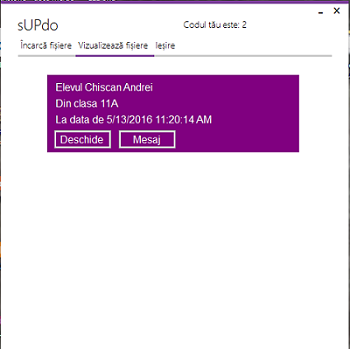
* Înregistrarea pentru elevi cuprinde numele, adresa de email, parola, clasa în care se află și codul profesorului pe care l-am văzut anterior.



* Autentificarea este aceeași ca mai devreme.
* Interfața din contul elevului este aproape aceeași, permițându-i elevului să trimită fișiere numai profesorului și să vizualizeze fișierele primite de acesta.
* Alegerea fișierelor se realizează printr-un simplu drag and drop, apoi elevul poate trimite și un anumit mesaj odată cu fișierele.



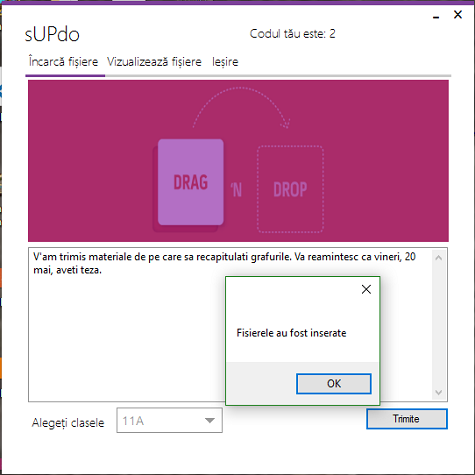
* Acuma, după ce se va autentifica profesorul, in tab-ul „Vizualizează fișierele” va vedea toate fișierele primite de la elevi, in cazul nostru cel de la Chișcan Andrei, din clasa 11A, la data de 5/13/2016.



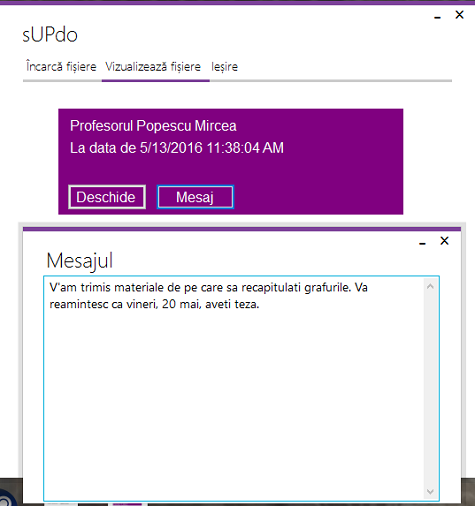
* Acum profesorul poate deschide fișierul/fișierele si sa citească mesajul/mesajele trimise de elevi.
* Odată cu vizualizarea fișierelor, se creează automat pe desktop un folder unde sunt recreate aceste fișiere, astfel încât profesorul va putea sa le deschidă direct de acolo.



* Acum, clasa „11A” este formata din elevul Chișcan Andrei, iar profesorul poate sa trimită si el la rândul lui fișiere.



* Odată logat în contul elevului, acesta va putea vedea fișierele si mesajele primite de la profesor, în cazul nostru:



* Ca și în cazul profesorului, elevului i se creează pe desktop un folder cu fișierele primite:

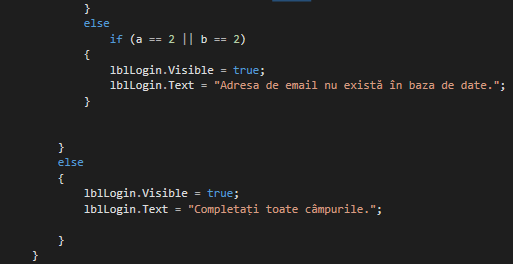
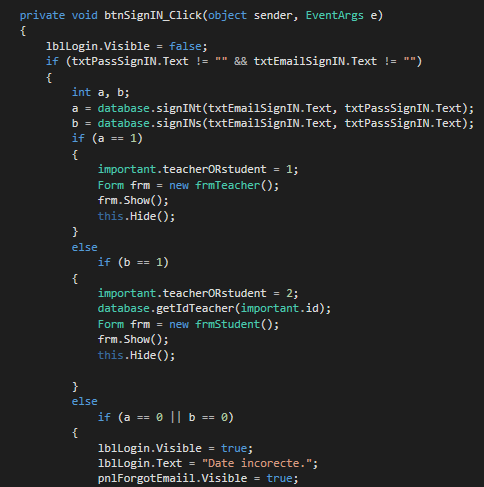


**Clasele folosite în aplicație**

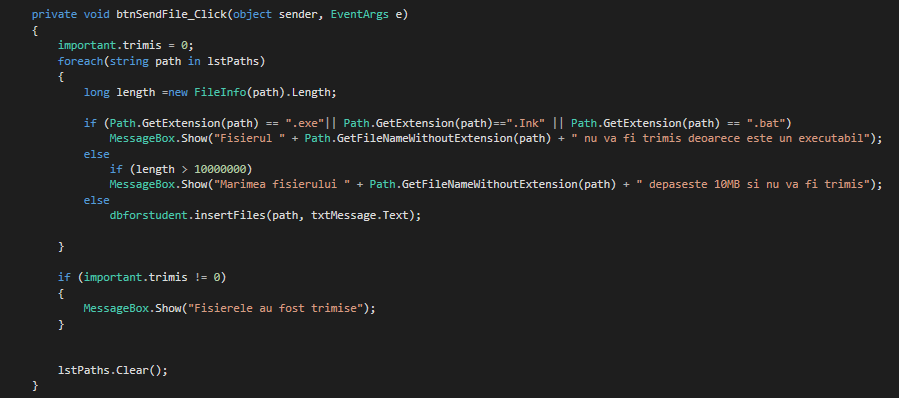
|  |  |
| --- | --- |
| Database.cs | clasă care facilitează interogările cu baza de date pentru înregistrarea profesorului/elevului si autentificarea în aplicație. |
| Dbforstudent.cs | clasă care facilitează interogările cu baza de date pentru trimiterea/primirea fișierelor de către elev. |
| Dbforteacher.cs | clasă care facilitează interogările cu baza de date pentru trimiterea/primirea fișierelor de către profesor. |
| Functions.cs | clasă care facilitează criptarea si decriptarea string’urilor precum si trimiterea unui mail cu parola în caz că aceasta a fost uitată. |
| Important.cs | clasă care facilitează structure si variabile folosite în timpul programului. |
| RegexUtilities.cs | clasă care verifică dacă adresa de email e într’un format corespunzător. |

**Exemple de scripturi**

Această funcție se execută la apăsarea butonului de autentificare. Funcția verifică dacă textbox’urile sunt completate, apoi se apelează alte funcții care verifică dacă contul în care are loc autentificarea este de profesor sau de elev. În cazul în care nici una din funcții nu



returnează valoarea 1, atunci se aruncă erorile: parolă greşită / adresă de email nu există, în funcție de caz.



Această funcție se execută la apăsarea butonului de trimitere a fișierelor. Funcția parcurge fiecare fișier adăugat, verificând dacă extensia și mărimea sunt corespunzătoare, iar în caz afirmativ, se apelează o alta funcție care le inserează în baza de date alături de mesaj.

**Concluzie**

Acest software a fost folosit la clasă de către profesorul meu de informatică, imbunătățind foarte mult ora, reducănd din timpul pierdut în timpul împărțirii temelor, documentelor, testelor și sporind securitatea la teste. Alte avantaje pe care le creează această platformă sunt acelea că organizează eficient fișierele primite de profesor/elev și că soft-ul fii folosită atăt la școală, cât și acasă.

Odată cu dezvoltarea acestui soft utilitar, mi-am îmbunătăţit cunoştinţele legate de programarea orientată pe obicte, în mediul Visual Studio, utilizănd limbajul de programare C#.