



**UNIVERSITATEA NAȚIONALĂ DE ȘTIINȚĂ ȘI  
TEHNOLOGIE POLITEHNICA BUCUREȘTI**  
**FACULTATEA DE ȘTIINȚE APLICATE**



**Programul de studii: Domeniul Științe ingineresti aplicate (master), Specializarea Teoria codării și stocării  
informației (TCSI)**

## **Sistem de înregistrare și autentificare 2FA**

**ÎNDRUMĂTOR ȘTIINȚIFIC,**

Lect. dr. Andrei-George OPRINA

**STUDENT,**

Alexandru-George GÂJU

**București**

**2025**

## **Introducerea sistemului**

Această lucrare propune un sistem de înregistrare și autentificare care se bazează pe clasică conectare cu o parolă alfanumerică îmbinată cu o parolă grafică, desenată chiar de utilizatorul însuși. Astfel, se vor utiliza elemente ale sistemelor cu doi factori de autentificare și a sistemelor care folosesc schema “Draw A Secret” (DAS). Pentru a înțelege conceptele care au fost implementate, în subcapitolele de mai jos vor fi prezentate pe scurt cele două sisteme amintite.

În esență, au fost realizate două interfețe, atât de înregistrare, cât și de autentificare, unde utilizatorul trebuie să-și introducă datele pentru a putea crea un cont, ulterior acestea fiind salvate într-o bază de date. Pentru a avea acces, se vor introduce credențialele impuse. Un factor în plus, față de alte sisteme, este acela ca utilizatorului îi este impus să deseneze o parolă grafică, care de asemenea, va fi stocată în baza de date. Aplicația a fost gândită în urma analizei nevoilor utilizatorilor și a specificațiilor tehnice, care au fost transformate într-un plan detaliat de dezvoltare. Pentru a putea realiza sistemul propus, au fost utilizate anumite instrumente software. Limbajul folosit este PYTHON în mediul de dezvoltare Visual Studio Code (VS Code), iar baza de date în care au fost stocate informațiile personale ale utilizatorului se numește MongoDB.

## **Caracteristici ale sistemului dezvoltat**

Mai departe, va fi evidențiată o schemă care sugerează cum funcționează sistemul implementat, adică conexiunea dintre faza de înregistrare, stocarea în baza de date și faza de autentificare.

În figura evidențiată mai jos, este relatată foarte bine funcționarea aplicației propusă în această lucrare. Se pot remarca trei etape esențiale care duc la funcționarea corespunzătoare a aplicației. Prin parcurgerea acestor pași se dorește implementarea unei scheme fiabile, logice și cu o precizie ridicată astfel încât scopul propus în acest proiect să fie atins cu succes.

În primul rând s-a realizat etapa de înregistrare în care un utilizator va trebui să-și introducă credențiale pentru a putea trece în următorul stadiu.

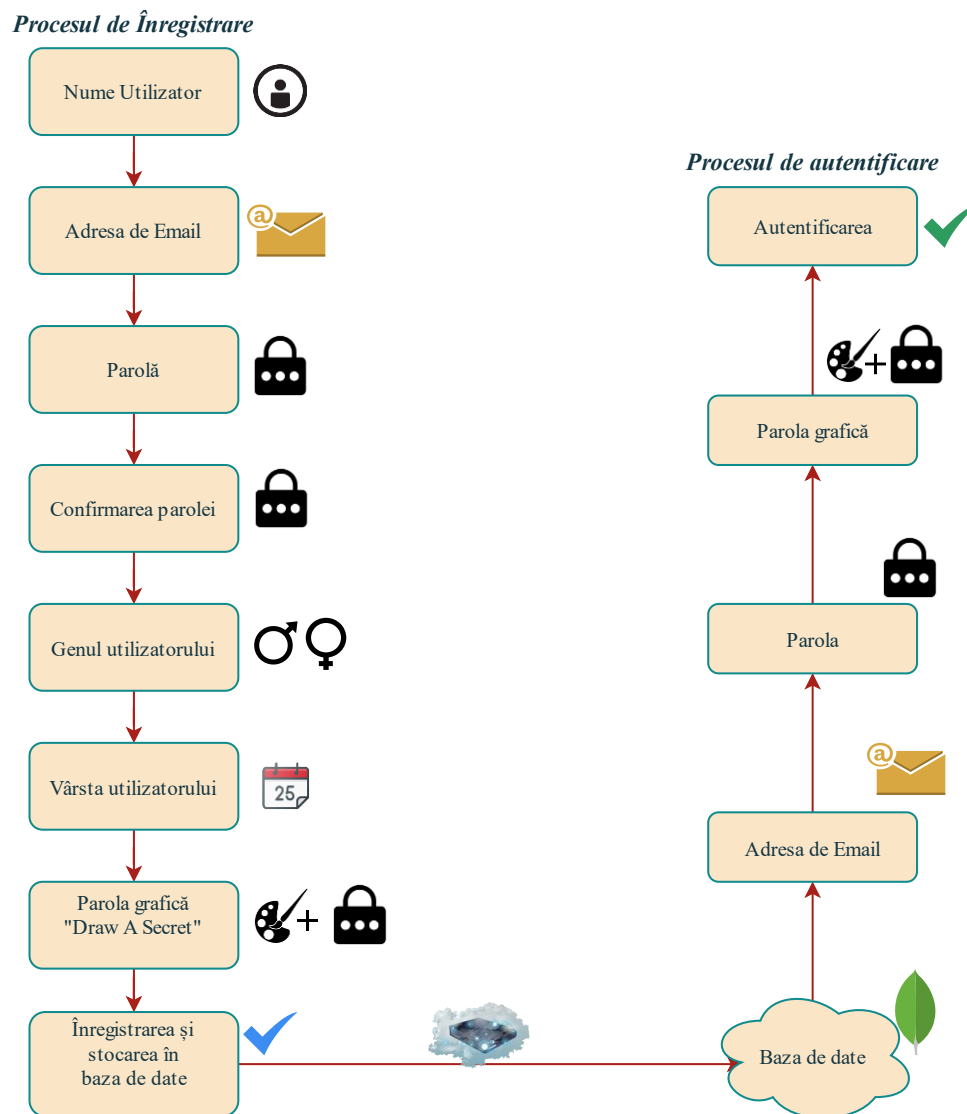


Figura 1. Schema bloc a sistemului implementat

Așadar, la înregistrare, persoana care urmează să își facă un cont va furniza următoarele informații:

- *Un nume de utilizator*, esențial pentru a se deosebi de alți membrii;
- *Adresa de email*, pentru a verifica autenticitatea persoanei sau pentru a primi diverse informații sau știri;
- *Parola*, care are scopul de a proteja contul;
- *Confirmarea parolei*, pentru a asigura utilizatorul că știe cu exactitate ce parolă a introdus la câmpul anterior;
- *Genul utilizatorului*, masculin sau feminin;
- *Vârsta*;

- *Parola grafică*, un factor aditiv pentru a spori securitatea.

După completarea acestor informații precum și datele utilizatorului vor fi stocate în baza de date.

Rolul bazei de date este de a salva credențialele fiecărei persoane care dorește să-și facă cont. Baza de date este liantul dintre etapa de înregistrare și etapa de autentificare, deoarece, fără informațiile stocate în aceasta, sistemul nu va ști cine are sau cine nu are cont de utilizator pe platformă.

În ultima etapă esențială, cea de conectare, utilizatorul trebuie să-și introducă câteva dintre datele pe care le-a introdus mai devreme, în etapa de înregistrare și anume:

- *Adresa de email*, fiind un reper de căutare pe care-l face sistemul în baza de date;
- *Parola*, pentru a verifica autenticitatea utilizatorului;
- *Parola grafică*, un factor în plus pentru a se asigura că securitatea contului nu a fost compromisă;

Dacă totul a fost respectat cu minuțiozitate, atunci autentificarea va avea loc cu succes, iar utilizatorul va putea naviga liniștit pe platformă.

Sistemul conceput în acest proiect întrunește sistemul de autentificare cu doi factori și sistemul de autentificare “DAS”, sau cel puțin preia anumiți factori pozitivi care ar putea aduce beneficii considerabile. Așadar, câteva dintre avantajele pe care le conferă această aplicație pot fi reprezentate de următoarele:

- Parola clasică și parola grafică sunt două modalități de autentificare, astfel sistemul este mult mai sigur față de unul care are doar un factor;
- Parola grafică poate fi ușor de memorat și mai greu de spart, deoarece este desenată de către utilizator, iar respectivul o personalizează în așa fel încât doar el să o poată reproduce;
- Protecție sporită împotriva atacurilor cibernetice și a furtului de identitate;
- Poate fi ușor de utilizat;
- Este o metodă interactivă care poate atrage atenția potențialilor utilizatori într-un mod plăcut și pozitiv;

Un astfel de sistem poate aduce un plus tehnologiei din zilele noastre, deoarece se poate trage concluzia că acesta se conformează cerințelor, atât ale

persoanelor care doresc o metodă de autentificare mai puțin plictisitoare și mai mult sigură, cât și a corporațiilor sau companiilor care doresc o securitate ridicată și o gestionare a resurselor mult mai adecvată pentru a face la rândul lor clienții mulțumiți.

## Parola grafică

Funcția “DAS()” a fost atribuită unui buton denumit “Draw Your Secret” care are rolul de a deschide o fereastră de tip cv2. Au fost declarate câteva variabile pentru prelucrarea ferestrei, de exemplu dimensiunile acesteia, dimensiunile unei celule, parametrii folosiți pentru a putea desena, parametrii gridului, condiția care ne spune dacă utilizatorul apasă sau nu pe mouse și încă o condiție care verifică dacă a fost sau nu ceva desenat în fereastră.

```
canvas_size = (400, 400)
cell_size = 100

drawing_color = (255, 255, 255, 255)
drawing_thickness = 3

grid_color = (255, 255, 255, 127)
grid_thickness = 1
drawing = False
pt1_x , pt1_y = None , None

canvas = np.zeros((canvas_size[0], canvas_size[1], 3), dtype=np.uint8)
valid_drawing = False
```

Figura 2. Parametrii ferestrei

În funcția “DAS()”, a fost generată o altă funcție care permite gestionarea evenimentelor mouseului. Cu ajutorul acesteia utilizatorului i se va permite să-și deseneze parola grafică cu care se va înregistra și, ulterior, cu care se va loga.

```

global pt1_x,pt1_y,drawing, canvas, valid_drawing

def draw_line(event, x, y, flags, params):
    global pt1_x,pt1_y,drawing, canvas, valid_drawing
    if event==cv2.EVENT_LBUTTONDOWN:
        drawing=True
        valid_drawing = True
        pt1_x,pt1_y=x,y
    elif event==cv2.EVENT_MOUSEMOVE:
        if drawing==True:
            cv2.line(canvas,(pt1_x,pt1_y),(x,y),color=(255,255,255),thickness=3)
            pt1_x,pt1_y=x,y
            valid_drawing = True
    elif event==cv2.EVENT_LBUTTONUP:
        drawing=False
        cv2.line(canvas,(pt1_x,pt1_y),(x,y),color=(255,255,255),thickness=3)

cv2.namedWindow("Canvas")
cv2.setMouseCallback("Canvas", draw_line)

```

Figura 3. Funcția care gestionează evenimentele mouse-ului

Pe lângă acestea, au fost realizate gridul ferestrei și o buclă principală care actualizează suprafața pe care se desenează. În final imaginile stocate vor avea eliminat gridul pentru a evita creșterea nejustificată a similitudinii atunci când imaginile vor fi comparate. Totodată, în buclaaceasta au fost programate tastele “s” și “r”, prima pentru a salva imaginea, iar ultima pentru a reseta zona de lucru.

Astfel a fost creat un mediu unde utilizatorul își poate desena parola grafică, utilizând modulul cv2 din cadrul bibliotecii OpenCV.

## Prelucrarea imaginilor

În acest capitol vor fi prezentate ultimele două funcții care au un impact esențial pentru funcționarea întregului sistem de înregistrare și autentificare.

```
def check_grid(draw1,draw2):  
  
    draw_grid1 = [[0 for i in range(4)]for i in range(4)]  
    draw_grid2 = [[0 for i in range(4)]for i in range(4)]  
  
    for i in range(4):  
        for j in range(4):  
            if 255 in draw1[i*100:(i+1)*100-1,j*100:(j+1)*100-1]:  
                draw_grid1[i][j] = 1  
  
    for i in range(4):  
        for j in range(4):  
            if 255 in draw2[i*100:(i+1)*100-1,j*100:(j+1)*100-1]:  
                draw_grid2[i][j] = 1  
  
    return draw_grid1 == draw_grid2
```

Figura 4. Funcția care verifică spațiul pe care îl ocupă desenul

Conținutul din figura 5 a fost realizat în cadrul interfeței de login, iar scopul acesteia este să marcheze fiecare grilă vizitată de desen, adică atunci când utilizatorul va desena parola sa grafică, sistemul va reține căsuțele pe care le parcurge desenul acestuia. Astfel, imaginile create atât la înregistrare, cât și la autentificare să poată fi comparate. Dacă cele două desene nu parcurg aceleași căsuțe și nu au aceeași formă, atunci utilizatorul nu va putea accesa contul, iar în caz contrar utilizatorului I se va permite accesul. Practic această funcție verifică coordonatele prin care trec cele două desene, doar că la un nivel mai general, reperul fiind căsuțele delimitate de liniile gridului. De asemenea, pe lângă suprafața acoperită, trebuie ținut cont și de similaritatea dintre cele două desene, fiind un factor important care contribuie la securitatea sistemului.

Pentru a verifica similitudinea dintre desenul stocat în baza de date cu desenul realizat în etapa de conectare, a fost implementată o altă funcție.

```
def drawing_is_similar(draw1, draw2):
    if not check_grid(draw1, draw2):
        return False

    img_canvas1=draw1.reshape(160000,3)
    img_canvas2=draw2.reshape(160000,3)

    mse = similaritymeasures.mse(img_canvas1, img_canvas2)
    print('mse', mse)

    return mse < 0.035
```

Figura 5. Funcția care verifică similaritatea dintre imagini

Cu ajutorul funcției “reshape”, imaginile vor fi aduse la aceleași dimensiuni, iar cu ajutorul erorii medii pătratice (mse) se va putea compara similitudinea dintre cele două desene.

MSE este o măsură de distanță sau de asemănare între două imagini bazată pe diferența ridicată la pătrat dintre valorile pixelilor.

Cu cât MSE este mai mic, cu atât imaginile sunt mai similare. Așadar, după efectuarea mai multor teste s-a constatat că un prag bun este 0.035, astfel a fost setat  $mse < 0.035$ .

## Baza de date

O bază de date semnifică o colecție organizată de colectare, stocare, gestionare și accesare a informațiilor. Aceasta poate fi compusă din fișiere, tabele sau documente și poate fi folosită pentru organizarea și gestionarea unei cantități mari de date. În cazul acesta, baza de date este utilizată pentru a salva informații despre utilizatori pentru a putea fi accesate rapid și eficient la nevoie.

Datele utilizatorului care au fost introduse în momentul înregistrării sunt salvate într-o bază de date. Pentru realizarea acestei aplicații a fost aleasă baza de date MongoDB.

MongoDB este un sistem popular de gestionare a bazelor de date NoSQL, open source, care utilizează un model de date orientat pe documente, fiind conceput pentru a oferi scalabilitate, disponibilitate și performanțe la un nivel înalt.



MongoDB este utilizată destul de des pentru analizarea și procesarea datelor în timp real, oferind caracteristici precum căutarea full-text, indexare, geospațialitate și suport pentru diverse limbaje de programare, în acest caz pentru python. Pe lângă cele prezentate, MongoDB se mai poate folosi la aplicațiile web.

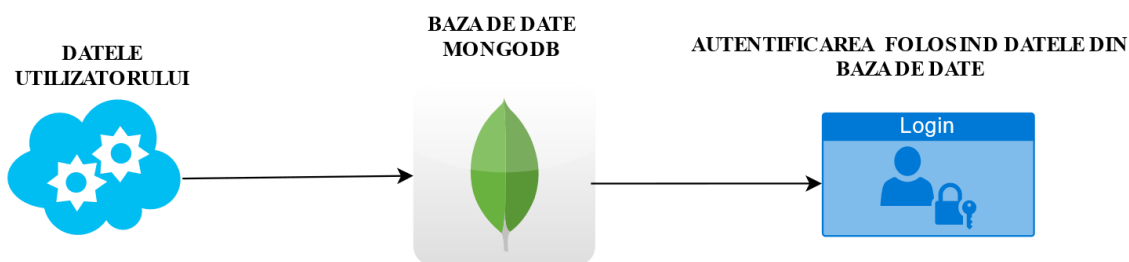


Figura 6. Corelația sistemului cu baza de date

## Concluzii

Implementarea unui sistem de înregistrare și autentificare cu doi factori (2FA) aduce un nivel suplimentar de securitate în procesul de protecție a datelor și conturilor utilizatorilor. Spre deosebire de metodele tradiționale bazate doar pe nume de utilizator și parolă, autentificarea 2FA adaugă o barieră suplimentară împotriva accesului neautorizat, folosind un al doilea factor de verificare.

Datorită domeniului larg de aplicabilitate există foarte multe soluții pentru orice problemă posibilă, lucrarea punând accent pe un sistem funcțional care prezintă câteva cerințe minimale, dar totuși complexe.

Majoritatea aplicațiilor de acest fel necesită atât prelucrarea de date, cât și prelucrarea și clasificarea conținutului unei imagini. Pentru realizarea acestora a fost înfăptuită o prelucrare generală a interfețelor de înregistrare și de autentificare, acestea fiind stilizate, îndeplinind anumite funcții și având un rol important în realizarea sistemului. Pe lângă acest lucru, aplicația presupune prelucrarea datelor utilizatorilor cu ajutorul unei baze de date, dar și prin procesarea de imagini. Prin analiza de imagini se înțelege, în această lucrare, că au fost obținute anumite date din desene, iar ulterior, acestea au fost comparate.

În concluzie, îmbinarea armonioasă a aspectelor prezentate mai sus au dus la înfaptuirea unui sistem de înregistrare și autentificare. Progresul și perfecționarea continuă a tehnicilor prezentate în acest proiect vor deschide calea ca sistemele de înregistrare și autentificare 2FA să fie și mai sigure și mai fiabile în viitor.