



MYDISCORD





Kamelia



Mounir



Maysa

Outils utilisés



GitHub



Trello
Application ::

Visual Studio Code

Logiciel ::



Base de Données

```
mysql> SELECT DATABASE();
```

DATABASE()
mounir-merzoud_myDiscord

```
mysql> SELECT * FROM user;
```

	id	username	prenom	email
	1	merzoud	mounir	mounir.merzoud@laplateforme.io
	2	mohamdi	kamelia	kamelia.mohamdi@laplateforme.io
	3	bik	maysa	bik.maysa@laplateforme.io
	18	NULL	mmk	mkm@gamil.com
	19	NULL	kmm123	kmm@gmail.com

```
mysql> SHOW TABLES;
```

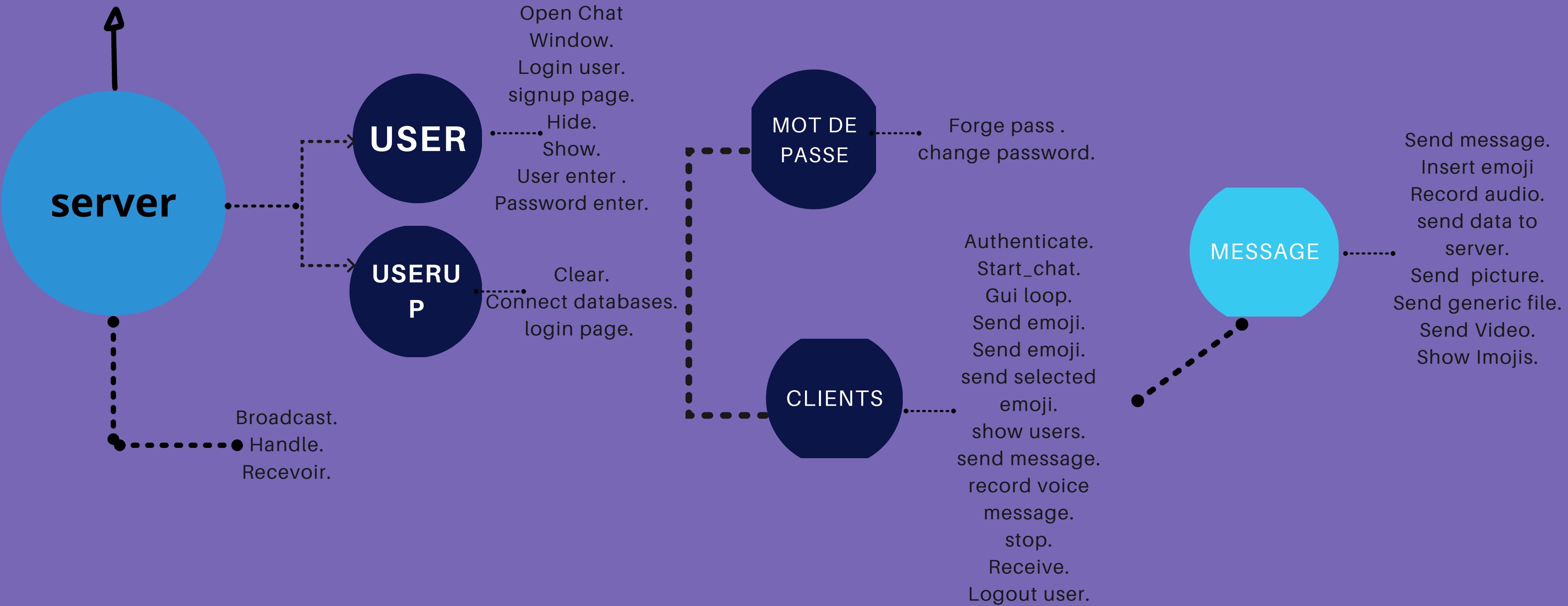
Tables_in_mounir-merzoud_myDiscord
salon
user

```
mysql> SELECT * FROM salon;
```

	id	nom	chemin_fichier
	1	Sport	myDiscord/Sport.txt
	2	Cuisine	myDiscord/Cuisine.txt
	3	Actualités	myDiscord/Actualités.
	6	drifa	drifa.txt
	8	manger	manger.txt

DIAGRAMME DES CLASSES

Création d'un serveur de chat
qui écoute sur une adresse IP
et un port spécifiés



Server

```
serveur > server.py > recevoir
1 import socket
2 import threading
3
4
5 HOST ="10.10.104.142"
6 port =9090
7 server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8 server.bind((HOST, port)) # Utilisez un tuple pour spécifier l'adresse et le port
9 server.listen()
10 clients = []
11 surnoms = []
12
13 > def broadcast(message): ...
14
15 > def handle(client): ...
16
17 > def recevoir(): ...
18
19 print("attente de connexion...")
20
21 recevoir()
```

1. Configuration du serveur .
2. Gestion des clients .
3. Fonction de diffusion (broadcast) .
4. Gestion des threads .
5. Boucle d'attente de connexion .

```
PS C:\Users\kamil\OneDrive\Bureau\myDiscord> & C:/Users/kamil/AppData/Local/Programs/Python/Python312/python.exe c:/Users/kamil/OneDrive/Bureau/myDiscord/serveur/server.py
attente de connexion...
[]
```

User

```
serveur > User.py > open_chat_window  
7   from tkinter import PhotoImage  
8   from PIL import Image, ImageTk  
9   from client import Client  
10  from MotDePass import forget_pass  
11  
12 # Définissez votre classe Client ici  
13  
14 > def open_chat_window(): ...  
22  
23  
24 > def login_user(): ...  
48  
49 > def signup_page(): ...  
52  
53 > def hide(): ...  
57  
58 > def show(): ...  
62  
63 > def user_enter(event): ...  
66  
67 > def password_enter(event): ...  
70  
71
```



1. Interface graphique .
2. Authentification .
3. Actions supplémentaires :
4. Option pour réinitialiser le mot de passe .
5. Lien pour créer un nouveau compte.

Login Page



Utilisateur

Nom ou Email

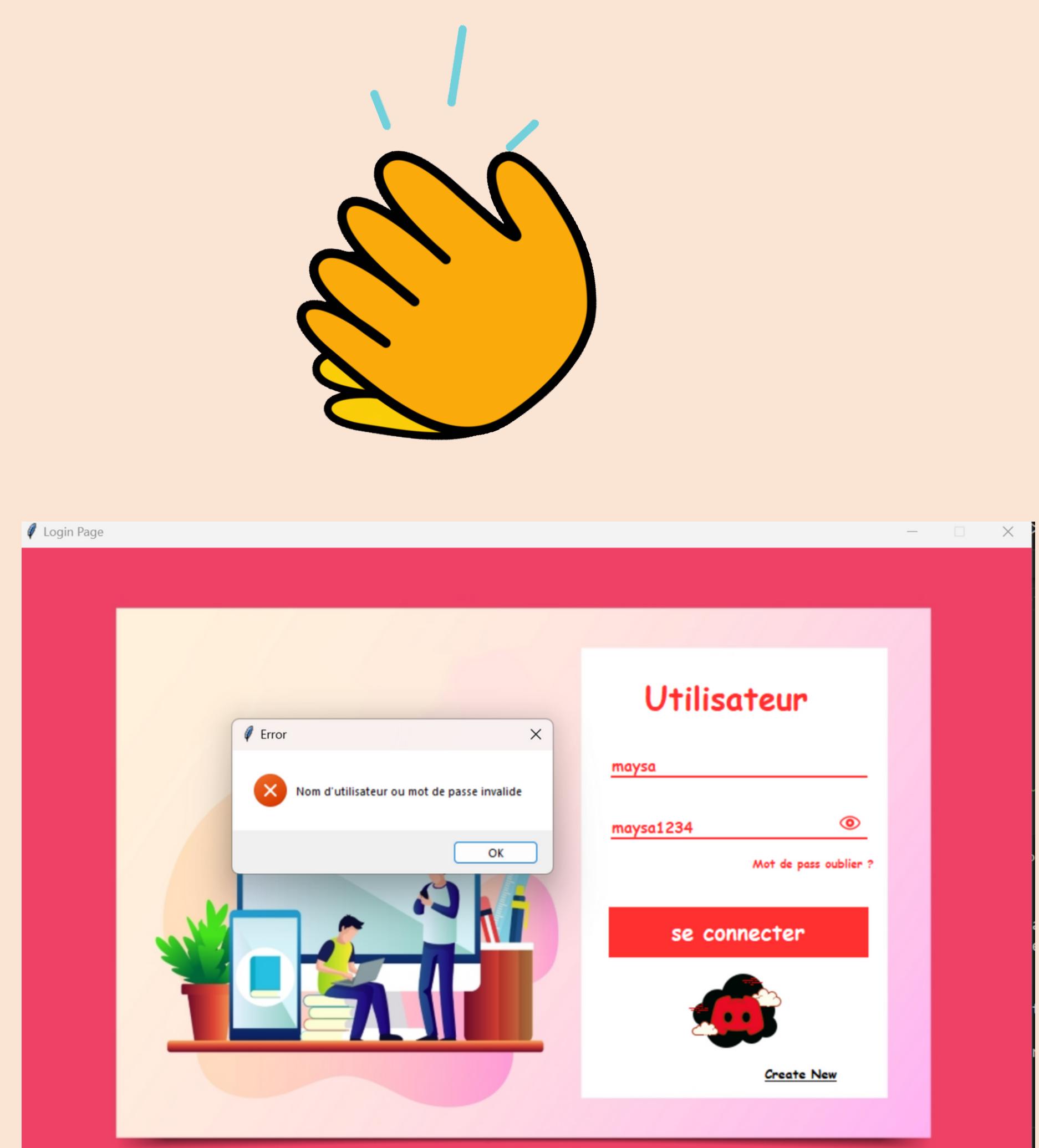
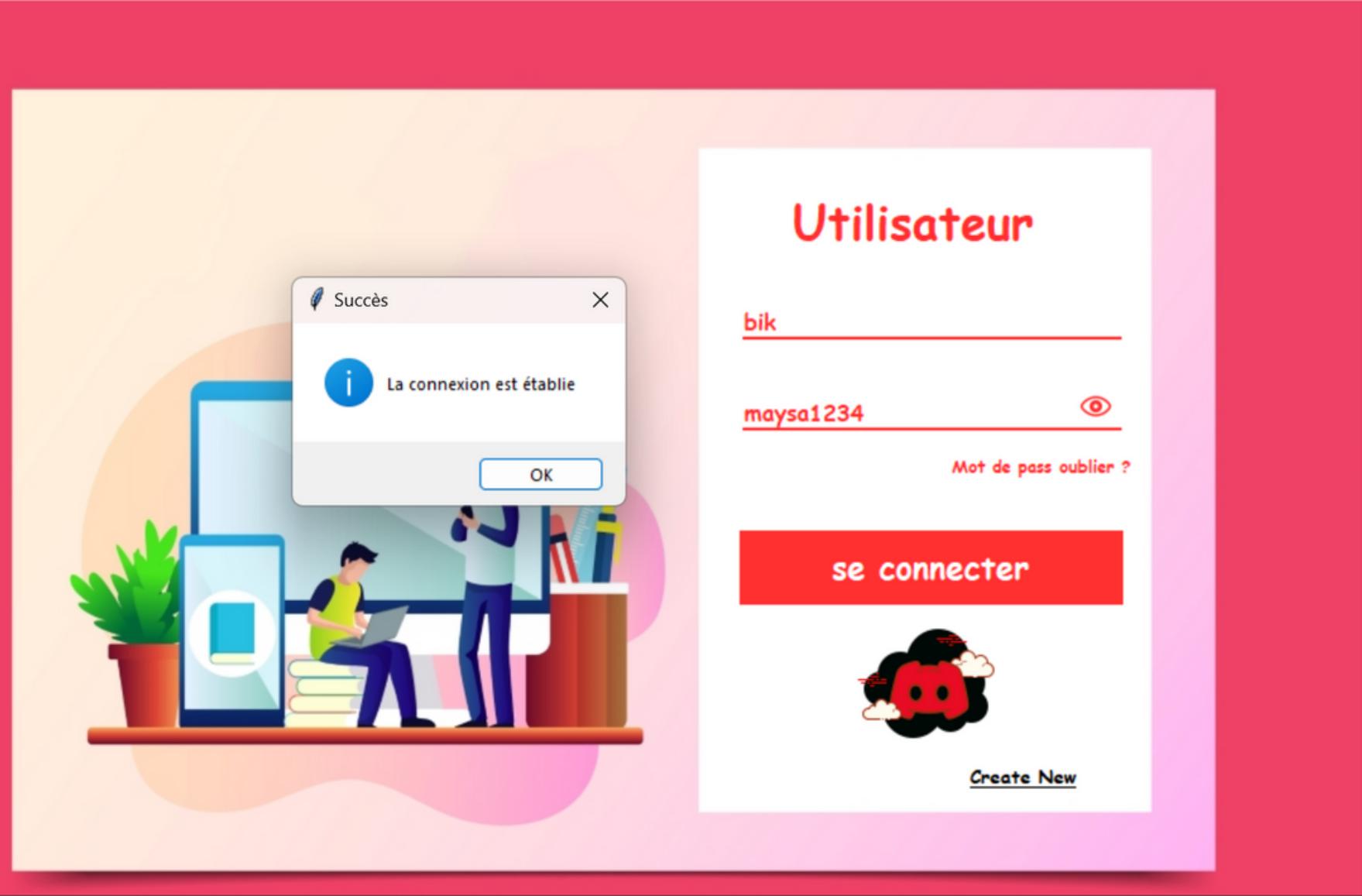
Mot de pass 👁

[Mot de pass oublier ?](#)

[se connecter](#)



[Ne pas de compte](#) [Create New](#)



UserUp



1. Interface graphique .
2. Validation des données .
3. Connexion à la base de données .
4. Navigation entre les pages .

```
serveur > userup.py > ...
1  from tkinter import *
2  from PIL import ImageTk
3  from tkinter import messagebox
4  import mariadb
5
6
7  class UserUp:
8  >     def __init__(self):...
61 >     def clear(self):...
67
68 >     def connect_database(self):...
100 >    def login_page(self):...
103
104 if __name__ == "__main__":
105     #root = Tk()
106     user_up = UserUp()
```





Create un compte

Email

Nom et Prénom

Mot de passe

Confirmer mot de passe

I agree to the Terms & conditions

Signup

[Y a pas un compt ?](#) [Log in](#)

Create un compte

Email
mkm@gmail.com

Nom et Prénom
mkm

Mot de passe
1234

Confirmer mot de passe
1234

I agree to the Terms & conditions

Signup

Connexion

Log in

Error
Veuillez accepter les termes et conditions

OK

A red error dialog box is displayed, containing the message "Veuillez accepter les termes et conditions" (Please accept the terms and conditions) and an "OK" button. The main form has fields for Email, Name, Password, and Confirm Password. A checkbox for accepting terms and conditions is present, which is not checked. The "Signup" button is red.

Userup Page

Succès
Inscription réussie

OK

Create un compte

Email
mkm@gmail.com

Nom et Prénom
mkm123

Mot de passe
12345

Confirmer mot de passe
12345

I agree to the Terms & conditions

Signup

Connexion

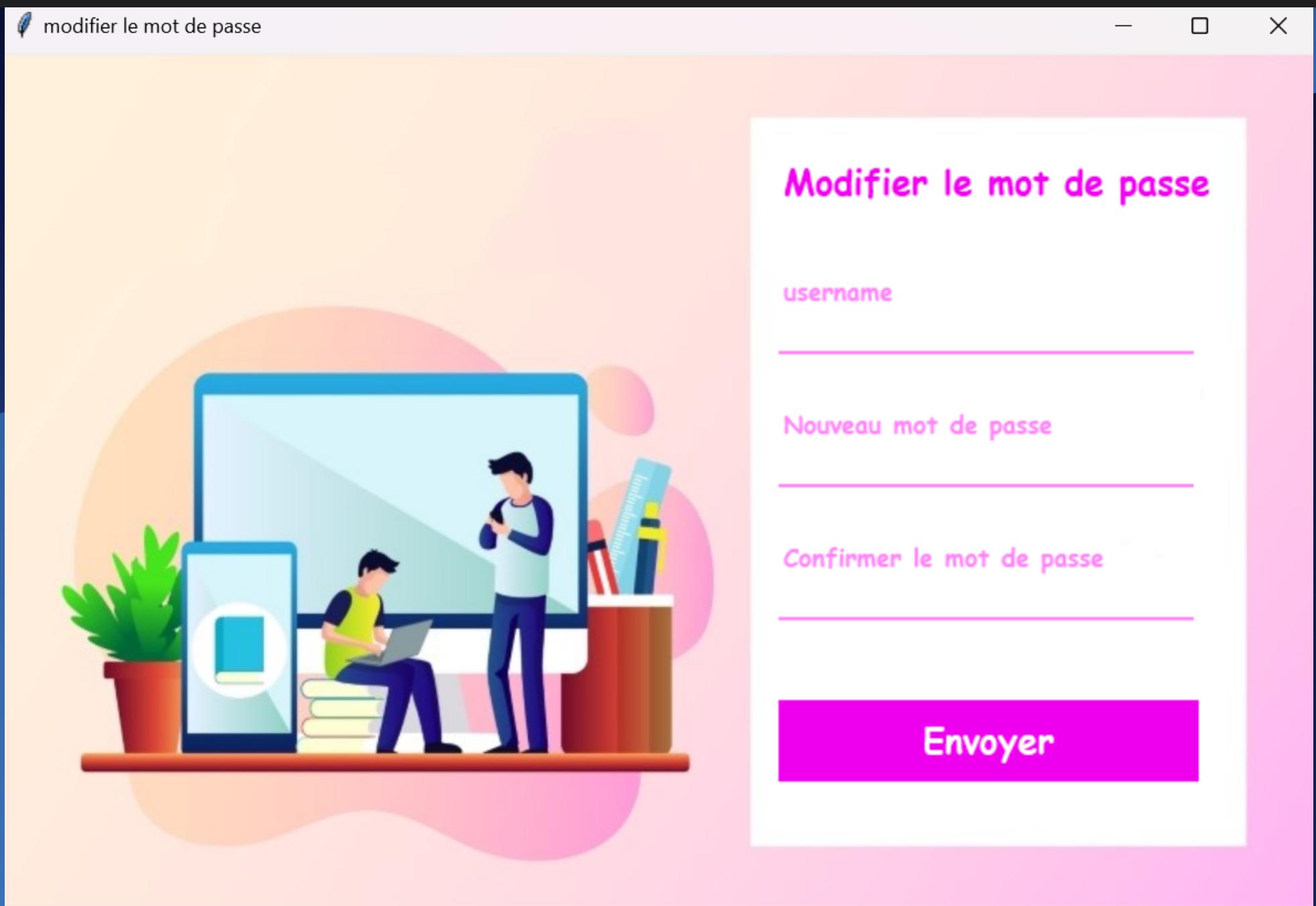
Log in

A green success dialog box is displayed, containing the message "Inscription réussie" (Registration successful) and an "OK" button. The main form has fields for Email, Name, Password, and Confirm Password. A checkbox for accepting terms and conditions is checked. The "Signup" button is red.

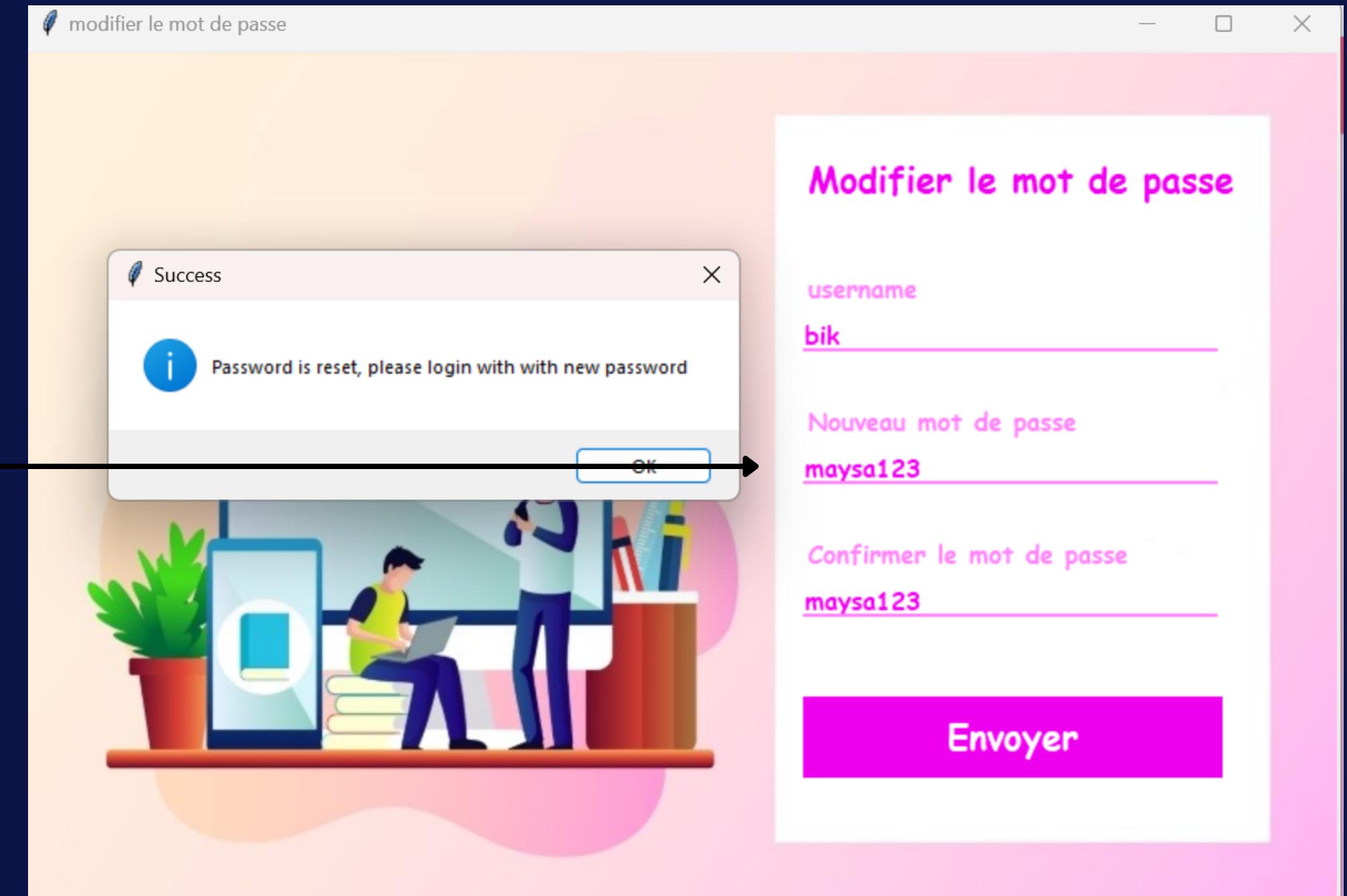
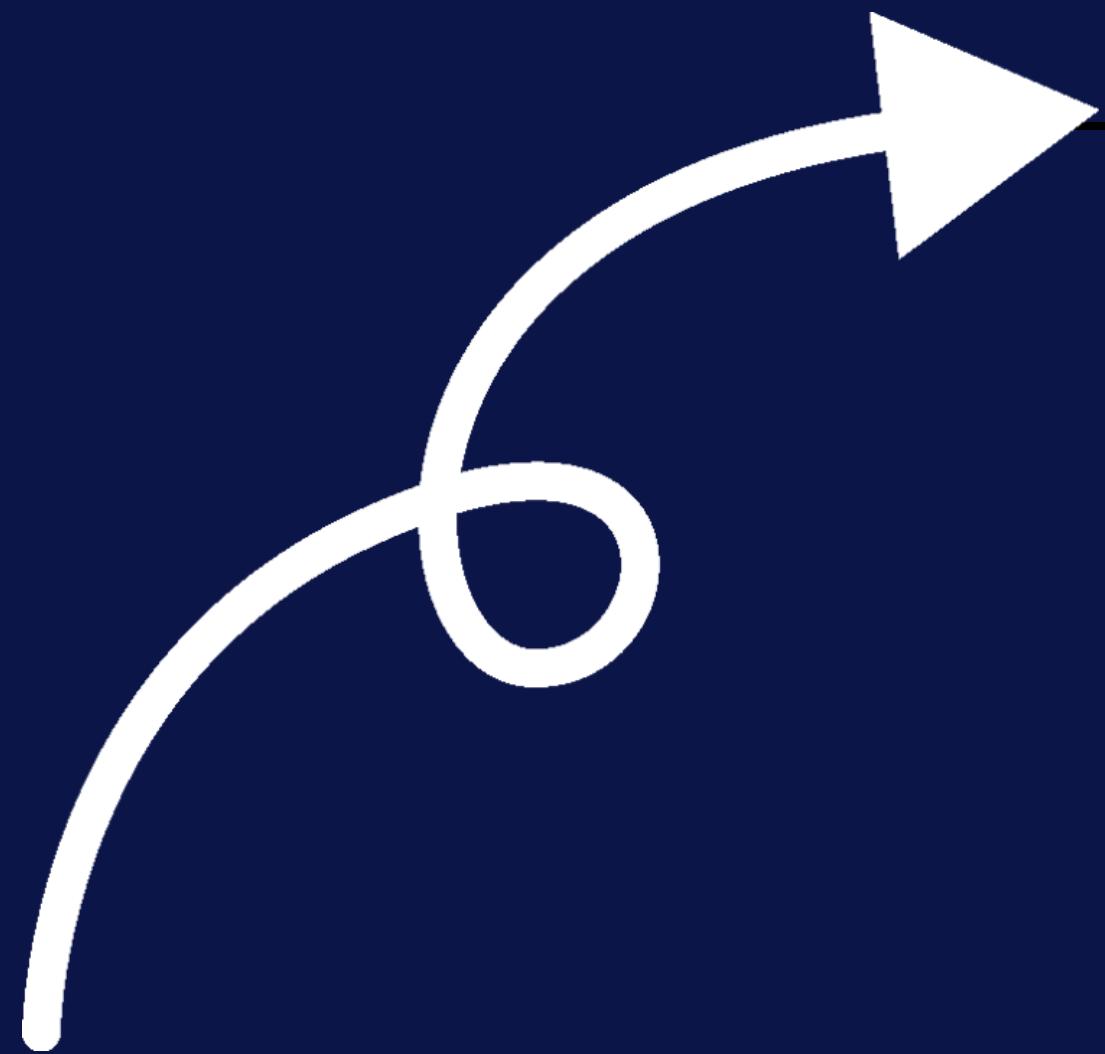


Mot De Passe

```
serveur > ⌂ MotDePass.py > ⌂ forget_pass
1  from tkinter import *
2  from tkinter import messagebox
3  from PIL import ImageTk
4  from tkinter import Toplevel
5  import mariadb
6
7
8 > def forget_pass(): ...
```



1. Interface graphique .
2. Validation des données .
3. Connexion à la base de données .
4. Affichage de messages .



Client

```
# Fonction pour enregistrer les messages dans un fichier
> def enregistrer_message(message): ...

# Définition de la classe Client
You, 51 minutes ago | 1 author (You)
class Client:
>     def __init__(self, host, port, username, password): ...

>     def authenticate(self): ...

>     def start_chat(self): ...

    # Méthode pour boucler l'interface utilisateur
>     def gui_loop(self): ...

>     def send_emoji(self): ...

>     def send_selected_emoji(self, emoji): ...

>     def show_users(self): ...

>     def send_message(self): ...
```

Ce code permet à l'utilisateur de s'authentifier, d'envoyer des messages textuels, des emojis, et d'enregistrer des messages vocaux. Les messages sont échangés avec un serveur distant via des sockets, et l'historique des messages est enregistré dans un fichier local.



```
def record_voice_message(self): ...

def stop(self): ...

def receive(self): ...

def logout_user(self): ...

if __name__ == "__main__": ...
```

Clients

1. Imports de modules .
2. Paramètres de couleur .
3. Définition de la classe Client .
4. Méthode authenticate .
5. Méthode start_chat .
6. Méthode gui_loop .
7. Méthodes pour envoyer des messages et des emojis
8. Méthode record_voice_message .
9. Point d'entrée principal .



Résultat



Chat App

Utilisateurs

Option

Informations

Déconnexion

merzoud
mohamdi
bik

SALON

```
<<<<< HEAD
bik : hi
bik : hi
bik : hello
=====
mohamdi : salut
mohamdi : salut
merzoud : salut les filles
merzoud : couco
bik : hi
>>>> 721d4eda705a6470b365b095d96d95b5c52e5edd
bik : hello
mohamdi : vous allez bien ?
mohamdi : vous allez bien ?
```

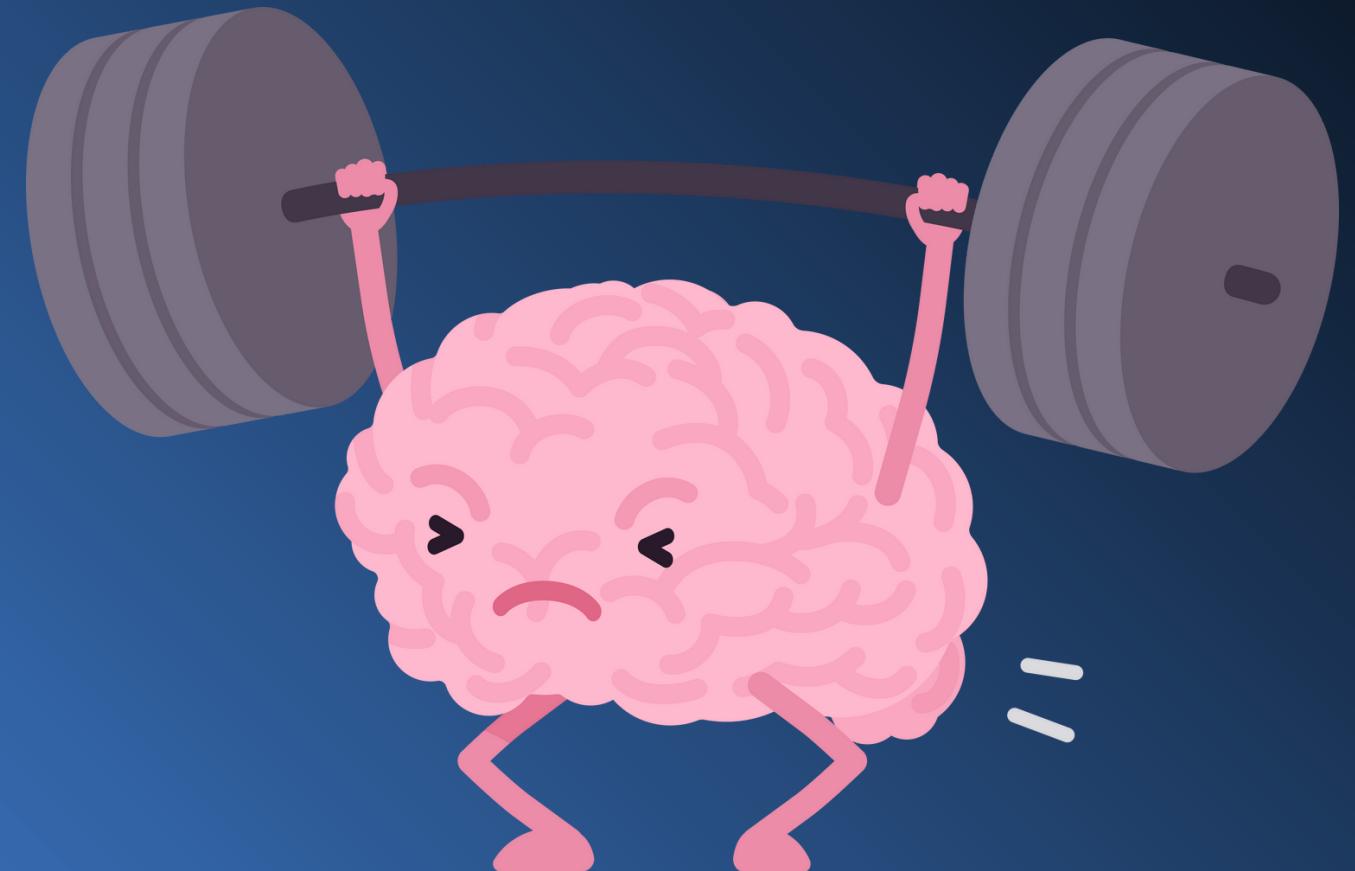
Message hi 😊😊

Difficultés rencontrées

1. Association du code message avec le client.
2. Messages vocaux publics et privés.
3. Exécution des liens web sur les canaux.
4. Bibliothèque tkinter

Points forts:

- La manipulation des serveurs et des clients avec des sockets
- Amélioration du côté technique .
- L'utilisation et à la gestion des données stockées dans une base de données.



**That's all and
thank you**

for listening to my talk

