

Error Handling

Server not running

Khi Client kết nối tới Server, nếu Server không hoạt động, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu khởi động lại Server.

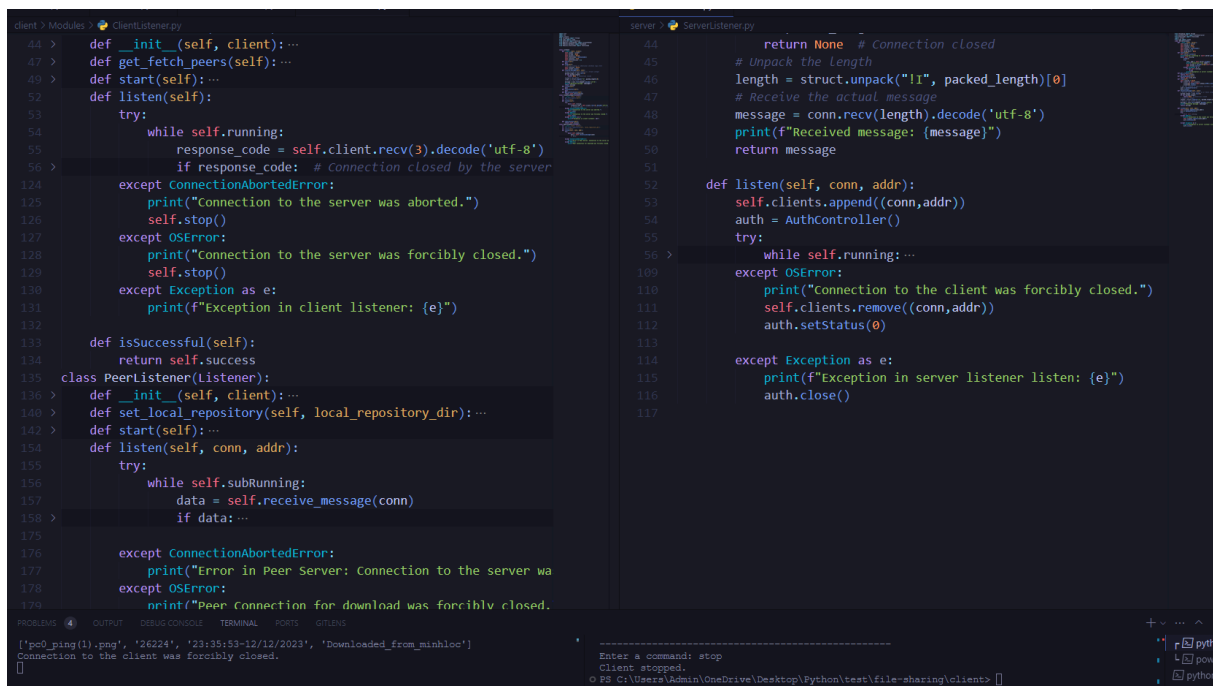


```
def start(self):
    server_host = input("Enter server host (auto for automatic): ")
    if server_host != "auto":
        self.server_host = server_host
    try:
        self.client.connect((self.server_host, self.server_port))
        print(self.client)
        self.listen_server = Thread(target=self.client_listener.listen)
        self.listen_server.start()
        print(f"Client connected to {self.server_host}:{self.server_port}")
        self.create_repository()
        self.create_file_system()
        self.login()
    except ConnectionRefusedError:
        print("Server is not running. Please start the server first.")
        return
    except OSError:
        print("Server is not running. Please start the server first.")
        return
```

Figure 1: Server not running

Connection error

Khi một trong hai kết nối giữa Client và Server bị lỗi, hệ thống sẽ hiển thị thông báo lỗi ở mỗi phía. Nếu là lỗi ảnh hưởng đến hệ thống, hệ thống sẽ tự động tắt. Nếu là lỗi không ảnh hưởng đến hệ thống, hệ thống sẽ hiển thị thông báo lỗi và tiếp tục hoạt động.



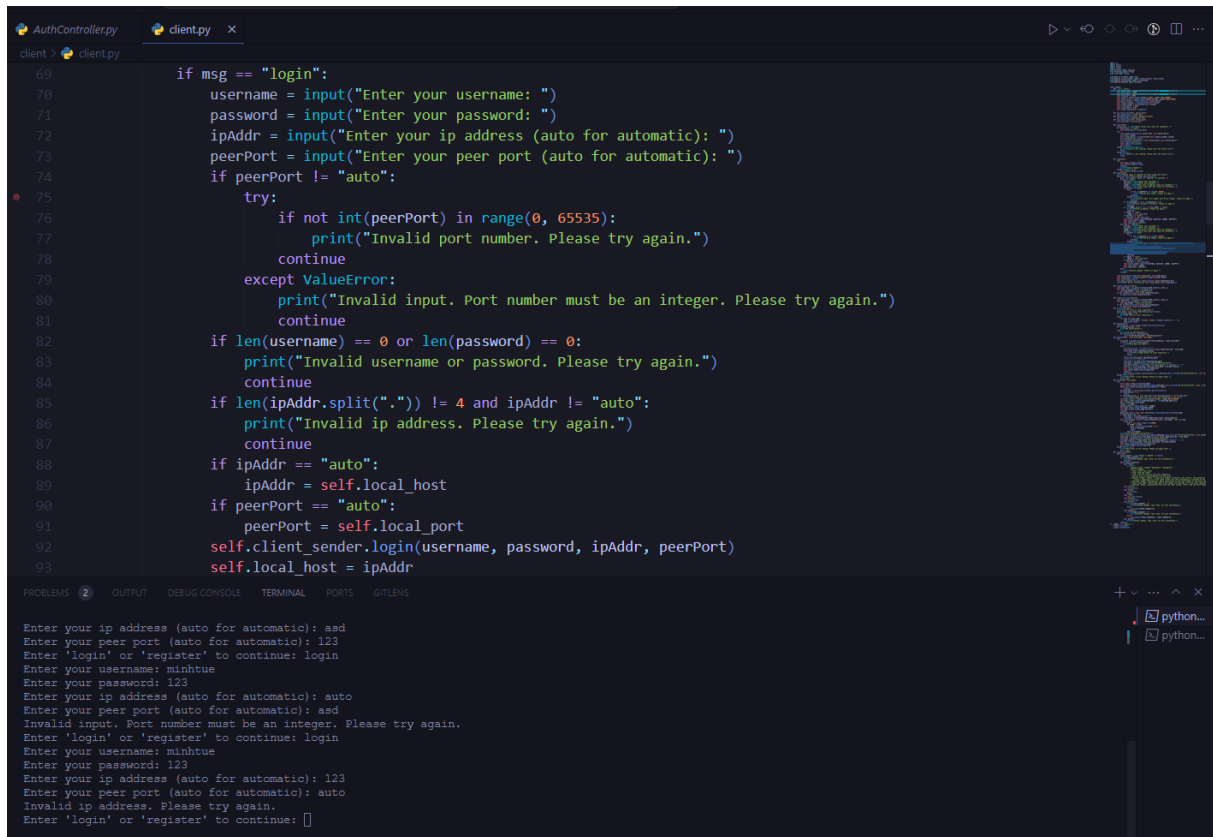
```
def __init__(self, client):
    def get_fetch_peers(self):
    def start(self):
    def listen(self):
        try:
            while self.running:
                response_code = self.client.recv(3).decode('utf-8')
                if response_code == 'Connection closed by the server':
                    except ConnectionAbortedError:
                        print("Connection to the server was aborted.")
                        self.stop()
                    except OSError:
                        print("Connection to the server was forcibly closed.")
                        self.stop()
                    except Exception as e:
                        print(f"Exception in client listener: {e}")
        def isSuccessful(self):
            return self.success
        class PeerListener(Listener):
            def __init__(self, client):
            def set_local_repository(self, local_repository_dir):
            def start(self):
            def listen(self, conn, addr):
                try:
                    while self.subRunning:
                        data = self.receive_message(conn)
                        if data:
                            except ConnectionAbortedError:
                                print("Error in Peer Server: Connection to the server wa")
                            except OSError:
                                print("Peer connection for download was forcibly closed.")
```

Figure 2: Connection error

Login & Register: Wrong input format

Hệ thống sẽ kiểm tra các trường nhập vào có đúng định dạng hay không.

Nếu không đúng định dạng, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.



```
69         if msg == "login":
70             username = input("Enter your username: ")
71             password = input("Enter your password: ")
72             ipAddr = input("Enter your ip address (auto for automatic): ")
73             peerPort = input("Enter your peer port (auto for automatic): ")
74             if peerPort != "auto":
75                 try:
76                     if not int(peerPort) in range(0, 65535):
77                         print("Invalid port number. Please try again.")
78                         continue
79                 except ValueError:
80                     print("Invalid input. Port number must be an integer. Please try again.")
81                     continue
82             if len(username) == 0 or len(password) == 0:
83                 print("Invalid username or password. Please try again.")
84                 continue
85             if len(ipAddr.split(".")) != 4 and ipAddr != "auto":
86                 print("Invalid ip address. Please try again.")
87                 continue
88             if ipAddr == "auto":
89                 ipAddr = self.local_host
90             if peerPort == "auto":
91                 peerPort = self.local_port
92             self.client_sender.login(username, password, ipAddr, peerPort)
93             self.local_host = ipAddr
```

client > client.py

```
Enter your ip address (auto for automatic): asd
Enter your peer port (auto for automatic): 123
Enter 'login' or 'register' to continue: login
Enter your username: minhtue
Enter your password: 123
Enter your ip address (auto for automatic): auto
Enter your peer port (auto for automatic): asd
Invalid input. Port number must be an integer. Please try again.
Enter 'login' or 'register' to continue: login
Enter your username: minhtue
Enter your password: 123
Enter your ip address (auto for automatic): 123
Enter your peer port (auto for automatic): auto
Invalid ip address. Please try again.
Enter 'login' or 'register' to continue: 
```

Figure 3: Wrong input format

Register: User already exist

Hệ thống sẽ kiểm tra xem username đã tồn tại hay chưa.

Nếu đã tồn tại, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.

```
9
10 def login(self, hostname, password, ipAddress, peerPort):
11     print(f"Login request: {hostname} {password} {ipAddress} {peerPort}")
12     self.user = self.database.get_user(hostname)
13     print(self.user)
14     if (not self.user):
15         return self.response_code.USER_NOT_FOUND()
16     elif (self.user[2] != password):
17         return self.response_code.WRONG_PASSWORD()
18     elif (self.user[5] == 1):
19         return self.response_code.USER_ALREADY_ONLINE()
20     else:
21         self.database.update_user(hostname, ipAddress, peerPort)
22         return self.response_code.LOGIN_SUCCESS()
23
24 def register(self, hostname, password, ipAddress, peerPort):
25     print(f"Register request: {hostname} {password} {ipAddress} {peerPort}")
26     if (self.database.get_user(hostname) != None):
27         return self.response_code.USER_ALREADY_EXISTS()
28     else:
```

Enter 'login' or 'register' to continue: register
Enter your username: minhtue
Enter your password: 123
Enter your ip address (auto for automatic): auto
Enter your peer port (auto for automatic): auto
Sending message: register minhtue 123 192.168.1.8 5001
Received response code: 201
Registration successful.
Enter 'login' or 'register' to continue: register
Enter your username: minhtue
Enter your password: 123
Enter your ip address (auto for automatic): auto
Enter your peer port (auto for automatic): auto
Sending message: register minhtue 123 192.168.1.8 5001
Received response code: 404
User already exists.
Enter 'login' or 'register' to continue:

Figure 4: User already exist

Login: User not found

Hệ thống sẽ kiểm tra xem username đã tồn tại hay chưa.

Nếu chưa tồn tại, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.

```
9
10 def login(self, hostname, password, ipAddress, peerPort):
11     print(f"Login request: {hostname} {password} {ipAddress} {peerPort}")
12     self.user = self.database.get_user(hostname)
13     print(self.user)
14     if (not self.user):
15         return self.response_code.USER_NOT_FOUND()
16     elif (self.user[2] != password):
17         return self.response_code.WRONG_PASSWORD()
18     elif (self.user[5] == 1):
19         return self.response_code.USER_ALREADY_ONLINE()
20     else:
21         self.database.update_user(hostname, ipAddress, peerPort)
22         return self.response_code.LOGIN_SUCCESS()
23
```

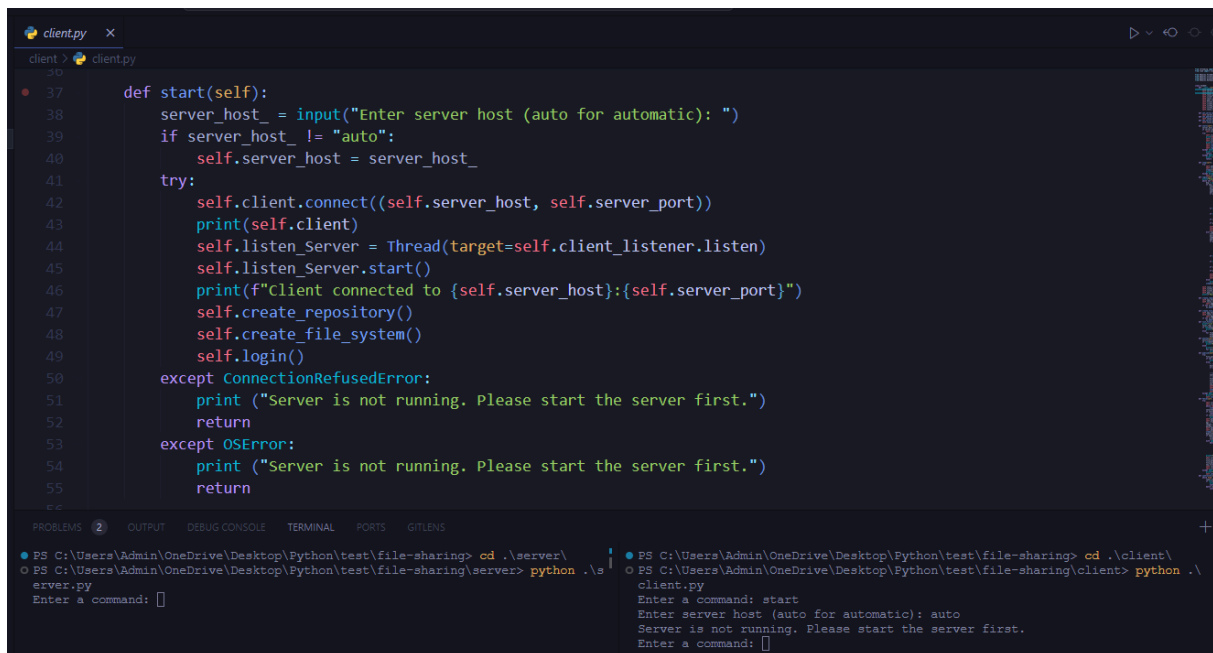
PS C:\Users\Admin\OneDrive\Desktop\Python\test\file-sharing> cd .\client\
PS C:\Users\Admin\OneDrive\Desktop\Python\test\file-sharing\client> python .\client.py
Enter a command: start
Enter server host (auto for automatic): auto
<socket.socket fd=544, family=2, type=1, proto=0, laddr=('192.168.1.8', 63000), raddr=('192.168.1.8', 5000)>
Client connected to 192.168.1.8:5000
Please login or register to start using the system.
Enter 'login' or 'register' to continue: login
Enter your username: minhtue
Enter your password: 123
Enter your ip address (auto for automatic): auto
Enter your peer port (auto for automatic): auto
Sending message: login minhtue 123 192.168.1.8 5001
Received response code: 401
User not found.
Enter 'login' or 'register' to continue: █

Figure 5: User not found

Login: Wrong password

Hệ thống sẽ kiểm tra xem password có đúng hay không.

Nếu không đúng, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.



```

client.py
client > client.py
37 def start(self):
38     server_host_ = input("Enter server host (auto for automatic): ")
39     if server_host_ != "auto":
40         self.server_host = server_host_
41     try:
42         self.client.connect((self.server_host, self.server_port))
43         print(self.client)
44         self.listen_Server = Thread(target=self.client_listener.listen)
45         self.listen_Server.start()
46         print(f"Client connected to {self.server_host}:{self.server_port}")
47         self.create_repository()
48         self.create_file_system()
49         self.login()
50     except ConnectionRefusedError:
51         print("Server is not running. Please start the server first.")
52     return
53 except OSError:
54     print("Server is not running. Please start the server first.")
55     return

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

```

PS C:\Users\Admin\OneDrive\Desktop\Python\test\file-sharing> cd .\server\
PS C:\Users\Admin\OneDrive\Desktop\Python\test\file-sharing\server> python .\server.py
Enter a command:
PS C:\Users\Admin\OneDrive\Desktop\Python\test\file-sharing> cd .\client\
PS C:\Users\Admin\OneDrive\Desktop\Python\test\file-sharing\client> python .\client.py
Enter a command: start
Enter server host (auto for automatic): auto
Server is not running. Please start the server first.
Enter a command:

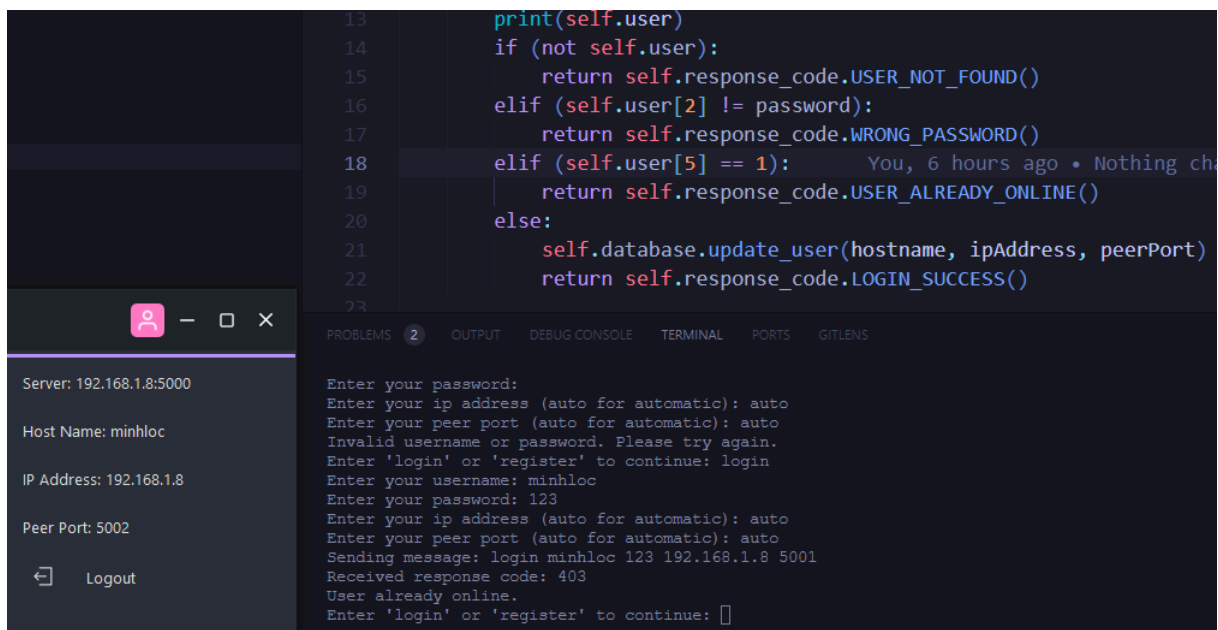
```

Figure 6: Wrong password

Login: User already online

Hệ thống sẽ kiểm tra xem username đã đăng nhập hay chưa.

Nếu đã đăng nhập, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.



```

13 print(self.user)
14 if (not self.user):
15     return self.response_code.USER_NOT_FOUND()
16 elif (self.user[2] != password):
17     return self.response_code.WRONG_PASSWORD()
18 elif (self.user[5] == 1):
19     return self.response_code.USER_ALREADY_ONLINE()
20 else:
21     self.database.update_user(hostname, ipAddress, peerPort)
22     return self.response_code.LOGIN_SUCCESS()
23

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

```

Enter your password:
Enter your ip address (auto for automatic): auto
Enter your peer port (auto for automatic): auto
Invalid username or password. Please try again.
Enter 'login' or 'register' to continue: login
Enter your username: minhloc
Enter your password: 123
Enter your ip address (auto for automatic): auto
Enter your peer port (auto for automatic): auto
Sending message: login minhloc 123 192.168.1.8 5001
Received response code: 403
User already online.
Enter 'login' or 'register' to continue:

```

Server: 192.168.1.8:5000

Host Name: minhloc

IP Address: 192.168.1.8

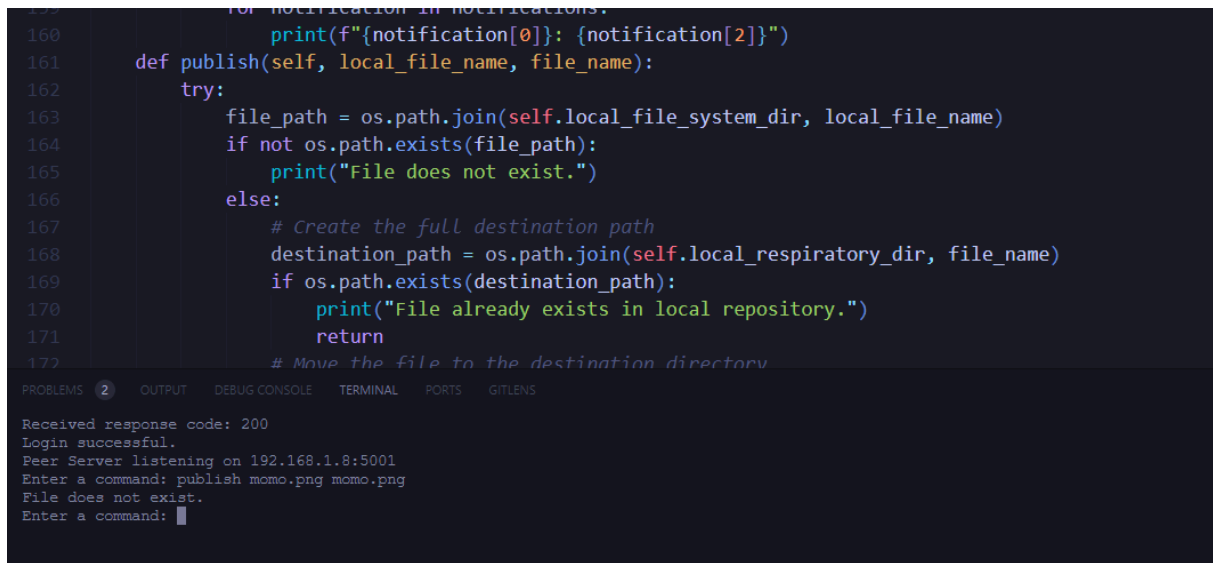
Peer Port: 5002

Logout

Figure 7: User already online

Publish: File not found

Hệ thống sẽ kiểm tra xem file có tồn tại hay không. Nếu không tồn tại, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.



```
160         print(f"{notification[0]}: {notification[2]}")
161     def publish(self, local_file_name, file_name):
162         try:
163             file_path = os.path.join(self.local_file_system_dir, local_file_name)
164             if not os.path.exists(file_path):
165                 print("File does not exist.")
166             else:
167                 # Create the full destination path
168                 destination_path = os.path.join(self.local_respiratory_dir, file_name)
169                 if os.path.exists(destination_path):
170                     print("File already exists in local repository.")
171                 return
172             # Move the file to the destination directory
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

```
Received response code: 200
Login successful.
Peer Server listening on 192.168.1.8:5001
Enter a command: publish momo.png momo.png
File does not exist.
Enter a command: 
```

Figure 8: File not found

Publish: File already published

Hệ thống sẽ kiểm tra xem file đã được Publish hay chưa. Nếu đã được Publish, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.

```
165         print( File does not exist. )
166     else:
167         # Create the full destination path
168         destination_path = os.path.join(self.local_respiratory_dir, file_name)
169         if os.path.exists(destination_path):
170             print("File already exists in local repository.")
171             return
172         # Move the file to the destination directory
173         shutil.move(file_path, destination_path)
174         # Update local_respiratory
175         file_size = os.path.getsize(destination_path)
176         file_date = datetime.now().strftime("%H:%M:%S-%d/%m/%Y")
177         file_description = input("Enter file description: ").replace(" ", "_")
178         new_file = File(file_name, file_size, file_date, file_description)
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS GIT LENS

Received response code: 200
Login successful.
Peer Server listening on 192.168.1.8:5001
Enter a command: publish momo.png momo.png
File does not exist.
Enter a command: publish result.csv result.csv
Invalid command. Type 'help' for more information.
Enter a command: publish result.csv result.csv
Enter file description: xstk
Sending message: publish result.csv 968 23:24:53-12/12/2023 xstk
Received response code: 202
File uploaded successfully.
Enter a command: list
List of files in local repository:
File name: result.csv
File size: 968B
File date: 23:24:53-12/12/2023
File description: xstk

Enter a command: publish result.csvvv result.csv
File does not exist.
Enter a command: publish result.csv result.csv
File does not exist.
Enter a command: publish result.csv result.csv
File already exists in local repository.
Enter a command:

Figure 9: File already published

Fetch: File not found

Hệ thống sẽ kiểm tra xem file có tồn tại hay không. Nếu không tồn tại, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.

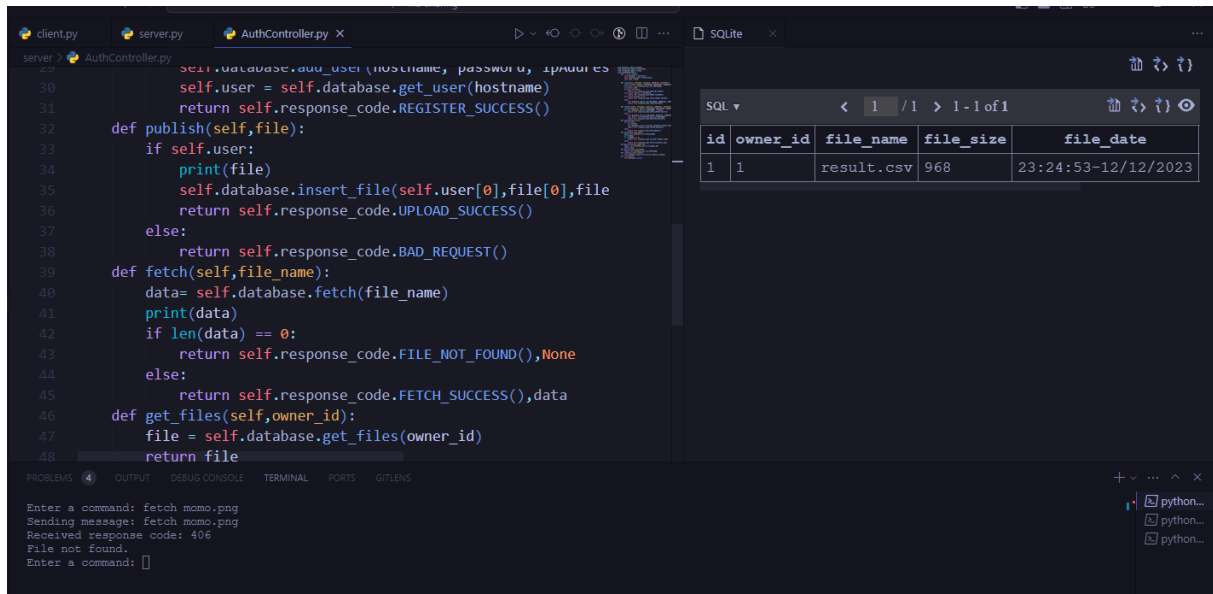


Figure 10: File not found

Fetch: File already fetched

Hệ thống sẽ kiểm tra xem file đã được Fetch hay chưa. Nếu đã được fetch, hệ thống sẽ tự động duplicate file cho người dùng.

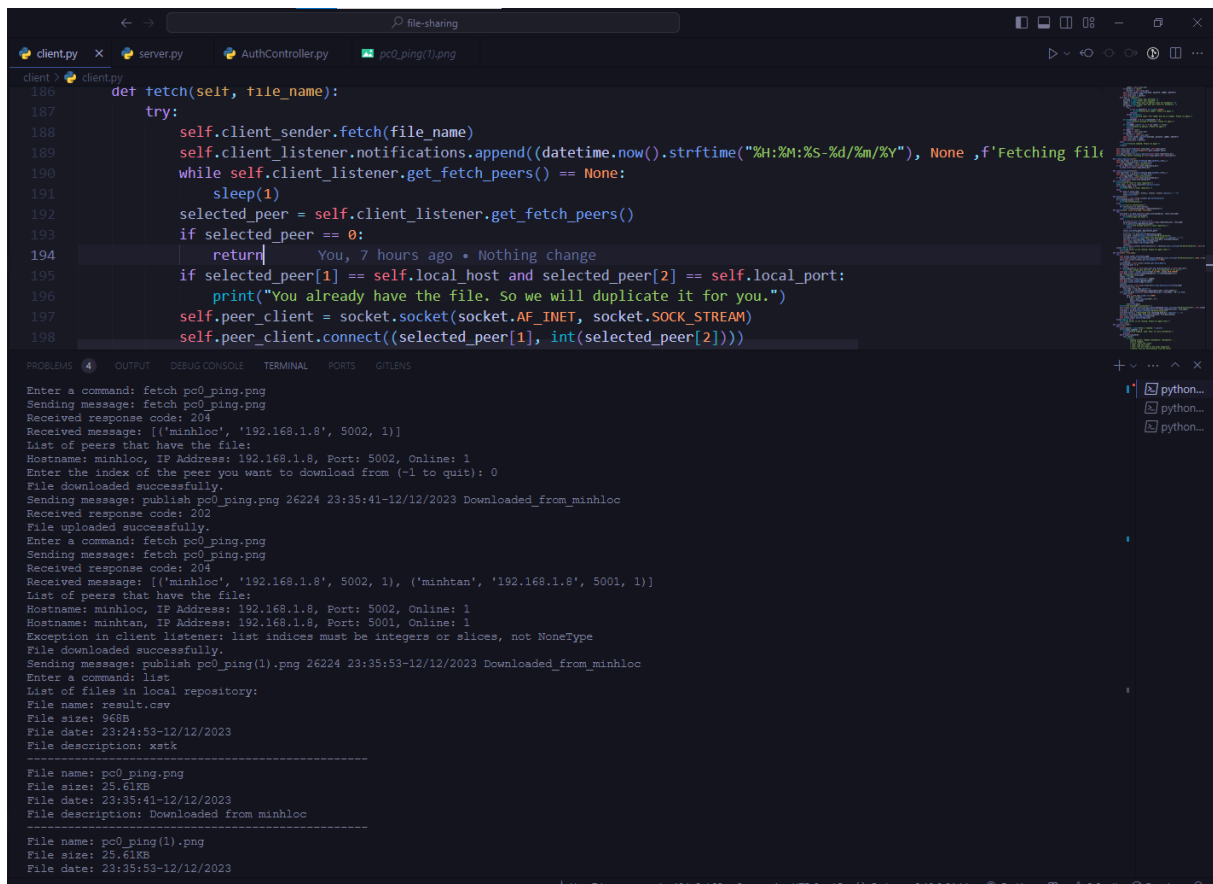


Figure 11: File already fetched

