

MẠNG MÁY TÍNH

ĐỒ ÁN SOCKET EMAIL CLIENT

Nhóm <14>

<22120219> - <Mai Nhật Nam>

<22120250> - <Âu Lê Tuấn Nhật>

<22120263> - <Nguyễn Thành Phát>



Khoa Công nghệ Thông tin
Đại học Khoa học Tự nhiên TP HCM
Tháng 12/2023

MỤC LỤC

I. TỔNG QUAN THÔNG TIN NHÓM.....	4
II. KẾ HOẠCH ĐỒ ÁN VÀ PHÂN CÔNG CÔNG VIỆC.....	4
1. Kế hoạch đồ án	4
2. Bảng phân công công việc	5
III. ĐÁNH GIÁ TỔNG QUAN.....	5
IV. KỊCH BẢN GIAO TIẾP CỦA CHƯƠNG TRÌNH	5
1. Giao thức trao đổi giữa client và server	5
1.1 Socket.....	5
1.2 SMTP (Simple Mail Transfer Protocol)	6
1.3 POP3 (Post Office Protocol version 3).....	6
2. Cách tổ chức cơ sở dữ liệu.	6
3. Cấu trúc và kiểu dữ liệu của thông điệp	7
V. MÔI TRƯỜNG LẬP TRÌNH	9
1. Ngôn ngữ lập trình	9
2. Các thư viện được sử dụng	9
VI. HƯỚNG DẪN SỬ DỤNG CHƯƠNG TRÌNH	10
1. Cấu hình file Config	10
1.1 Server:.....	10
1.2 Authentication:.....	10
1.3 Filter:	10
1.4 Autoload:.....	10
2. Gửi Email đến TO, CC và BCC.....	11
3. Gửi File ở nhiều định dạng (. pdf, jpg ...).....	16
4. Tải được các email về mail client từ mailbox của Server và quản lý trạng thái đọc của Email	17
5. Xử lí lọc Email	19

5.1. Lọc email dựa trên địa chỉ người gửi.....	19
5.2. Lọc email dựa trên Subject.....	20
5.3. Lọc email dựa trên nội dung	22
5.4 Lọc email mà nó có chứa từ khóa hoặc nội dung vào thư mục cụ thể (Ví dụ: thư mục Spam).....	23
6. Xử lí Autoload.....	24
VII. TÀI LIỆU THAM KHẢO	25

I. TỔNG QUAN THÔNG TIN NHÓM

MSSV	Họ tên	Email
22120219	Mai Nhật Nam	22120219@student.hcmus.edu.vn
22120250	Âu Lê Tuấn Nhật	22120250@student.hcmus.edu.vn
22120263	Nguyễn Thành Phát	22120263@student.hcmus.edu.vn

II. KẾ HOẠCH ĐỒ ÁN VÀ PHÂN CÔNG CÔNG VIỆC

Trong đồ án này, chúng em đã viết chương trình Mail client, có thể có giao diện hoặc không có giao diện. Mail client kết nối đến Server để gửi email (SMTP) và lấy email trong mailbox (POP3). Sau đây là kế hoạch đồ án của nhóm và bảng phân công việc như sau:

1. Kế hoạch đồ án

Cột mốc	Công việc dự kiến	Ước lượng (man hour)
15/11/2023	- Demo sử dụng phần mềm ThunderBird và Telnet	10
18/11/2023	- Họp phân công công việc và lên ý tưởng cho đồ án	1
20/11/2023	- Demo Client gửi và nhận được tin nhắn	10
23/11/2023	- Demo gửi được đến TO, CC, BCC	10
28/11/2023	- Gửi được attached File - Họp kiểm tra tiến độ đồ án lần 1	20
1/12/2023	- Tải được các nội dung Email đã đọc	20
3/12/2023	- Họp kiểm tra tiến độ đồ án lần 2	1
5/12/2023	- Tổ chức File config - Quản lí được trạng thái Email đã đọc hay chưa đọc	20
7/12/2023	- Xử lí lọc Email	20
9/12/2023	- Xử lý tự động tải email từ Mailbox theo thời gian cấu hình trong config file	20
11/12/2023	- Viết báo cáo - Chỉnh sửa giao diện cho phần mềm Email Client	20
13/12/2023	- Họp góp ý chỉnh sửa đồ án lần 3	20
15/12/2023	- Nộp đồ án (hoàn thiện 80%) qua Moodle	0,1
16/12/2023 → 02/01/2024	- Chỉnh sửa đồ án hoàn thiện hơn và chuẩn bị phần vấn đáp đồ án Socket	10

2. Bảng phân công công việc

STT	Tên công việc	Người thực hiện	MSSV
1	Gửi mail đến TO, CC và BCC	Mai Nhật Nam	22120219
2	Gửi được các attached File theo nhiều định dạng khác nhau (pdf, xlsx, doc...)	Nguyễn Thành Phát	22120263
3	Tải được các email về mail client từ mailbox của Server	Âu Lê Tuấn Nhật	22120250
4	Quản lý Email ở trạng thái đã đọc hay chưa đọc	Nguyễn Thành Phát	22120263
5	Xử lý lọc Email	Âu Lê Tuấn Nhật	22120250
6	Tổ chức file config	Mai Nhật Nam	22120219
7	Xử lý tự động tải email từ Mailbox theo thời gian cấu hình trong config file	Âu Lê Tuấn Nhật	22120250
8	Thiết kế giao diện	Nguyễn Thành Phát	22120263
9	Viết báo cáo	Mai Nhật Nam	22120219
10	Kiểm thử phần mềm	Mai Nhật Nam, Âu Lê Tuấn Nhật	22120250

III. ĐÁNH GIÁ TỔNG QUAN

- Nếu đánh giá một cách khái quát nhất, các thành viên nhóm về cơ bản đã hoàn thành nhiệm vụ mình được giao. Email Client đã chạy và đáp ứng được các nhu cầu mà giảng viên đã đề ra. Để phục vụ thuận tiện nhất cho việc tiến hành đồ án, cả nhóm đã thống nhất sử dụng ngôn ngữ lập trình Python.

- Tuy nhiên, do đồ án này các thành viên tiếp xúc với một ngôn ngữ lập trình mới, nên trong quá trình làm chắc chắn không thể bỏ qua một số thiếu sót nhất định. Có thể kể đến như:

- Các thành viên lần đầu làm việc với nhau nên chưa kết hợp ăn ý trong một số khâu.
- Vì một số lý do khách quan, một số công việc đã bị trì trệ và chưa hoàn thành đúng hạn.
- Với ngôn ngữ lập trình mới, các thành viên cũng gặp đôi chút khó khăn trong việc tiếp thu và lập trình. Nhưng các thành viên vẫn đảm bảo mình phải hiểu chức năng của cú pháp trước khi sử dụng.

Đánh giá tổng thể: 80 – 90 %.

IV. KỊCH BẢN GIAO TIẾP CỦA CHƯƠNG TRÌNH

1. Giao thức trao đổi giữa client và server

1.1 Socket

- Socket là giao diện lập trình ứng dụng được dùng để truyền và nhận dữ liệu trên Internet. Giữa hai chương trình chạy trên mạng cần có một liên kết giao tiếp hai chiều để kết nối hai process trò chuyện với nhau. Điểm cuối của liên kết này được gọi là socket.
- Socket giúp các tầng TCP hoặc TCP Layer định danh ứng dụng mà dữ liệu sẽ được gửi tới thông qua sự ràng buộc với một cổng port từ đó tiến hành kết nối client - server.
- Hai ứng dụng cần truyền thông tin có thể nằm trên cùng một máy hoặc hai máy khác nhau. Nhưng để socket có thể hoạt động thì số hiệu cổng không được trùng nhau.

1.2 SMTP (Simple Mail Transfer Protocol)

- SMTP là giao thức chuẩn dùng để gửi email từ client lên server hoặc từ server này sang server khác.

- Trong đồ án của chúng em, các hàm như **send_mail** và **login** (được gọi trong phương thức **connect** của lớp **MailClient**) sử dụng SMTP để gửi email và xác thực người dùng.
- SMTP hoạt động theo mô hình yêu cầu - phản hồi, nơi client gửi yêu cầu đến server (ví dụ: gửi email, xác thực), và server trả lời với một thông điệp phản hồi.

1.3 POP3 (Post Office Protocol version 3)

- POP3 là giao thức dùng để lấy email từ server về client.

- Các hoạt động như **authenticate** và **download_mail** trong lớp **MailClient** sử dụng POP3 để xác thực người dùng và tải email từ server về máy của người dùng.
- POP3 cũng hoạt động theo mô hình yêu cầu - phản hồi. Client yêu cầu (ví dụ: lấy email), và server phản hồi với dữ liệu hoặc thông báo trạng thái.

2. Cách tổ chức cơ sở dữ liệu.

- Ở trong đồ án này, nhóm chúng em sẽ tổ chức chương trình gồm có 5 file: config.yml, mail_client.py, using.py, utils.py và file constant.py.

- Ở file config.yml, nhóm chúng em thiết kế file để cấu hình các chức năng cần thiết trong Email Client (Ví dụ như set địa chỉ IP, port, mail gửi đi, mail nhận...)
- Ở file mail_client.py và using.py, nhóm chúng em chứa thông tin các đoạn code cho các hàm chức năng cần sử dụng trong đồ án này
- Ở file utils.py, nhóm chúng em thiết kế hàm main để chạy chương trình Email Client.
- Ở file constant.py, nhóm em thiết kế từ điển từ khóa để sử dụng trong các file còn lại.

3. Cấu trúc và kiểu dữ liệu của thông điệp

- Ở trong đồ án này, nhóm chúng em sẽ tổ chức chương trình gồm có 5 file: config.yml, mail_client.py, using.py, utils.py và file constant.py.

- Ở file mail_client.py, chúng em có các hàm như:

1/ `__init__` (self, host, smtp_port, pop3_port, filter, bufferSize=1024): Khởi tạo đối tượng MailClient với thông tin cần thiết để kết nối đến máy chủ SMTP và POP3.

2/ `connect`(self): Kết nối đến cả máy chủ SMTP và POP3.

3/ `response` (self, server): Nhận phản hồi từ máy chủ.

4/ `request` (self, server, message, expect_return_msg=True, display_msg=True): Gửi yêu cầu đến máy chủ và nhận phản hồi. Có thể hiển thị hoặc ẩn thông điệp tùy thuộc vào đối số.

5/ `quit` (self): Gửi lệnh QUIT đến máy chủ SMTP để đóng kết nối.

6/ `login` (self, user: str, password: str): Thực hiện đăng nhập bằng cách mã hóa tên người dùng và mật khẩu, sau đó gửi lệnh AUTH PLAIN đến máy chủ SMTP.

7/ `send_mail` (self, sjt, msg, from_addr, to_addr, cc_addr, bcc_addr, attach_file=None): Gửi email với tiêu đề, nội dung, người gửi, người nhận, người nhận ẩn và tệp đính kèm tùy chọn.

8/ `authenticate` (self, user): Xác thực người dùng với máy chủ POP3.

9/ `current_mail_number_on_server`(self): Lấy số lượng email hiện tại trên máy chủ POP3.

10/ `save_files` (self, path, files): Lưu trữ tệp đính kèm của email vào đường dẫn được chỉ định.

11/ `download_mail` (self, user): Tải về email từ máy chủ POP3, lưu trữ chúng trong các thư mục tương ứng dựa trên bộ lọc và lưu chúng dưới dạng tệp JSON.

12/ `get_mail_boxes`(self): Lấy danh sách từ điển, mỗi từ điển đại diện cho một email, từ máy chủ POP3.

- Ở file using.py, chúng em có các hàm như:

1/ load_menu (): Chức năng này làm nhiệm vụ hiển thị giao diện menu chính và xử lý lựa chọn của người dùng.

2/ auto_download_mail (): Chức năng này chạy trong một luồng riêng biệt và thực hiện tự động tải về email từ máy chủ cứ sau mỗi khoảng thời gian.

3/ __main__:

- + Tạo một đối tượng MailClient để kết nối đến máy chủ SMTP và POP3.
- + Sử dụng hai luồng riêng biệt (t1 và t2) để thực hiện load_menu và auto_download_mail.
- + Luồng load_menu xử lý giao diện người dùng và tương tác với MailClient để gửi và tải về email.
- + Luồng auto_download_mail tự động tải về email từ máy chủ theo khoảng thời gian được đặt trước.

- Ở file utisls.py, chúng em có các hàm như:

1/ extract_mail(text): Chuyển đổi đoạn văn bản email thành từ điển Python chứa thông tin về ngày, người gửi, người nhận, chủ đề, nội dung, và các tệp đính kèm (nếu có).

2/ check_contains (list, string): Kiểm tra xem chuỗi có chứa bất kỳ phần tử nào trong danh sách không.

3/ box_filter (filter_config, dict, box): Lọc email vào các hộp thư (inbox, spam, important, project, work) dựa trên cấu hình bộ lọc.

4/ filter (dict, filter_config): Áp dụng bộ lọc để xác định xem một email thuộc vào các hộp thư nào.

5/ create_folder(user): Tạo các thư mục hộp thư cho người dùng nếu chúng không tồn tại.

6/ current_mail_number_on_local(user): Đếm số email hiện tại trong thư mục local của người dùng.

7/ read_mail_boxes(user): Đọc và trả về danh sách email trong các hộp thư của người dùng.

- Ở file using.py, chúng em có các hàm như:

1/ menu(user): Hiển thị menu chính của ứng dụng, bao gồm các tùy chọn như gửi email, tải email, mở hộp thư, và thoát.

2/ `send_mail_console()`: Giao diện điều khiển cho quá trình gửi email. Người dùng nhập thông tin như địa chỉ gửi, địa chỉ nhận, chủ đề, nội dung và tệp đính kèm (nếu có).

3/ `mail_boxes_console()`: Hiển thị menu cho việc chọn hộp thư (Inbox, Important, Work, Project, Spam) hoặc di chuyển email đến một thư mục khác.

4/ `mail_box_console(dict)`: Hiển thị danh sách email trong hộp thư, bao gồm trạng thái đọc và thông tin chủ đề và người gửi.

5/ `mail_content_console(dict)`: Hiển thị nội dung chi tiết của một email, bao gồm thông tin về ngày, người gửi, người nhận, chủ đề, nội dung và tệp đính kèm.

6/ `display_files_console(files)`: Hiển thị danh sách các tệp đính kèm trong một email.

V. MÔI TRƯỜNG LẬP TRÌNH

1. Ngôn ngữ lập trình

Như đã đề cập ở trước, Python đã được chọn để làm ngôn ngữ lập trình cho đồ án lần này vì sự thuận tiện, đa năng của nó. Chúng em đã thống nhất sử dụng Microsoft Visual Studio Code là những phần mềm chuyên dùng để code với ngôn ngữ lập trình Python để hoàn thành đồ án này.

2. Các thư viện được sử dụng

- Như đã đề cập bên trên, ở trong đồ án này, nhóm chúng em sẽ tổ chức chương trình gồm có 5 file: `config.yml`, `mail_client.py`, `using.py`, `utils.py`, `constant.py`.

- Tại file `mail_client.py`, nhóm chúng em đã sử dụng những thư viện hỗ trợ sau:

- **socket**: Thư viện **socket** được sử dụng để cung cấp cơ sở hạ tầng mạng cho việc tạo và quản lý các socket. Trong đồ án này, nó được dùng để tạo kết nối TCP/IP với máy chủ SMTP và POP3, qua đó cho phép gửi và nhận email.
- **base64**: Thư viện **base64** được sử dụng cho việc mã hóa và giải mã dữ liệu theo chuẩn Base64. Trong đồ án này, nó được dùng để mã hóa thông tin đăng nhập (tên người dùng và mật khẩu) khi gửi tới máy chủ SMTP để xác thực.
- **json**: Thư viện **json** được sử dụng để làm việc với dữ liệu dạng JSON. Trong đồ án này, nó được sử dụng để ghi (serialize) dữ liệu ra file dưới dạng JSON, có thể là thông tin email sau khi đã được xử lý và lọc.
- **datetime**: Module **datetime** được sử dụng để làm việc với ngày và giờ. Trong đồ án này, **datetime** được dùng để lấy thời gian hiện tại và định dạng nó khi gửi email.

- Tại file `utils.py`, nhóm chúng em đã sử dụng thư viện hỗ trợ sau:

- **os**: Thư viện **os** được sử dụng để tương tác với hệ điều hành. Trong đồ án này, nó được sử dụng để thực hiện các công việc liên quan đến hệ thống file như kiểm tra sự tồn tại của một thư mục (**os.path.exists**) và tạo thư mục mới (**os.makedirs**). Nó cũng được sử dụng để liệt kê các tệp và thư mục con trong một thư mục cụ thể thông qua **os.listdir**. Điều này giúp trong việc quản lý các thư mục chứa email của người dùng.

- Tại file `using.py`, nhóm chúng em đã sử dụng thư viện hỗ trợ sau:

- **yaml**: Thư viện **yaml** được sử dụng để đọc và phân tích cú pháp các tệp YAML. Trong trường hợp này, nó được sử dụng để đọc cấu hình từ tệp **config.yml**, có thể chứa thông tin như cài đặt máy chủ, thông tin đăng nhập, và các thông số khác.
- **json**: Thư viện **json** được sử dụng để làm việc với dữ liệu dạng JSON. Trong đồ án này, nó được sử dụng để ghi (serialize) dữ liệu ra file dưới dạng JSON, có thể là thông tin email sau khi đã được xử lý và lọc.

VI. HƯỚNG DẪN SỬ DỤNG CHƯƠNG TRÌNH

1. Cấu hình file Config

- Ở file config dưới đây, ta có thể thấy các chức năng trong ứng dụng email Client được cấu hình như sau và Ở đây ta có thể tự điều chỉnh lại các thông số theo ý thích mình:

1.1 Server:

- Ở Server, ta có thể thấy các thông tin được cấu hình như địa chỉ IP của MailServer, Port của SMTP và POP3 và bufferSize.

1.2 Authentication:

- Ở Authentication, thông tin được hiển thị bao gồm username và password.

1.3 Filter:

- Do trong ứng dụng Email Client này có chức năng là xử lý lọc Email theo từ khóa và nội dung nên ta có thể thấy mục Filter được tạo ra để người dùng có thể tự cấu hình lại các từ khóa, subject mà họ muốn lọc vào các folder tương ứng.

1.4 Autoload:

- Ở trong file config này còn có mục autoload dùng để điều chỉnh lại thời gian autoload file.

```
Server:
  MailServer: 127.0.0.1
  SMTP: 2500
  POP3: 1100
  bufferSize: 2046
Authentication:
  username: mainhatnam01@gmail.com
  password: xxxx
Filter:
  Important:
    From: []
    Subject: ['urgent', 'ASAP']
    Content: []
  Work:
    From: []
    Subject: []
    Content: ['report', 'meeting']
  Spam:
    From: []
    Subject: []
    Content: ['virus', 'hack', 'crack']
  Project:
    From: ['myboss@gmail.com']
    Subject: []
    Content: []
Autoload:
  Time: 5
```

2. Gửi Email đến TO, CC và BCC.

- Ở file config khi này, nhóm em đã tiến hành cấu hình lại tên user là mainhatnam01@gmail.com nên mọi Email sau khi được gửi tới địa chỉ mail này đều sẽ nằm trong mail là mainhatnam01@gmail.com

```
config.yml
1  Server:
2      MailServer: 127.0.0.1
3      SMTP: 2500
4      POP3: 1100
5      bufferSize: 2046
6  Authentication:
7      username: mainhatnam01@gmail.com
8      password: xxxx
```

- Đầu tiên, chúng ta sẽ vào chạy code file using.py và màn hình console hiện ra như sau:

```
====Welcome to mail client=====
Current user: mainhatnam01@gmail.com

1. Send mail
2. Download mail
3. Open mail boxes
4. Exit
Your choice: []
```

- Ở màn hình lúc này hiện ra cho chúng ta thông tin user và các sự lựa chọn để tiến hành thao tác với ứng dụng. Các lựa chọn như SendMail (nhập 1), Download mail (nhập 2), Open mailboxes (nhập 3) và thoát ra (nhập 4).

- Bây giờ chúng ta sẽ thử chức năng SendMail như sau:

```
Connecting to SMTP server...
Connecting to POP3 server...
====Send mail=====

From: tuannhat448@gmail.com
To: mainhatnam01@gmail.com
CC:
BCC:
Subject: TEST 02
Message: test
Attach file (Press Enter to skip):
Send mail successfully!

Press any key to continue . . .
```

+ Đầu tiên, chúng ta sẽ nhập địa chỉ mail gửi (FROM) (ví dụ như) tuannhat448@gmail.com đến địa chỉ mail nhận là mainhatnam01@gmail.com.

+ Subject là TEST 02.

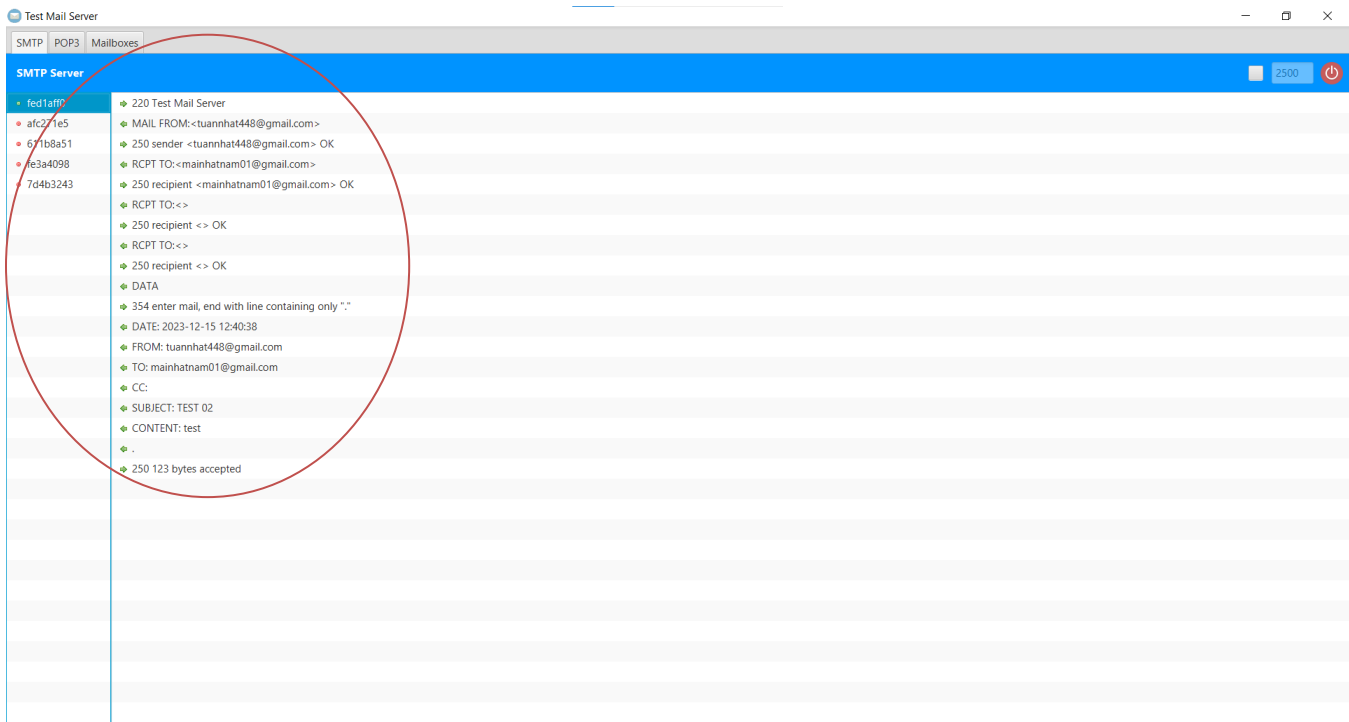
+ Message có thông điệp là "test"

+ Nếu chúng ta có gửi File thì chúng ta chỉ cần dán đường dẫn của File đó vào ứng dụng và sau đó gửi đi.

➔ Và sau khi đã hoàn tất thì console hiện thông báo Send mail Successfully báo hiệu chúng ta đã gửi mail thành công.

- Để kiểm tra chúng ta có thể xài ứng dụng Test Mail Server được cung cấp để nhìn thông tin như sau:

+ Nhìn vào hình dưới, ta đã thấy ở vùng khoanh đỏ, Test Mail Server đã hiển thị đủ và đúng các thông tin như người gửi, nhận và thông điệp mail giống như chúng ta đã setup trên giao diện console bên trên.



- Nếu như chúng ta muốn gửi mail theo dạng CC và BCC thì chúng ta chỉ cần thao tác y chang như bên trên và nhập thêm các địa chỉ mail vào vùng CC và BCC hiện trên console. Chúng ta có thể thao tác như sau:

```

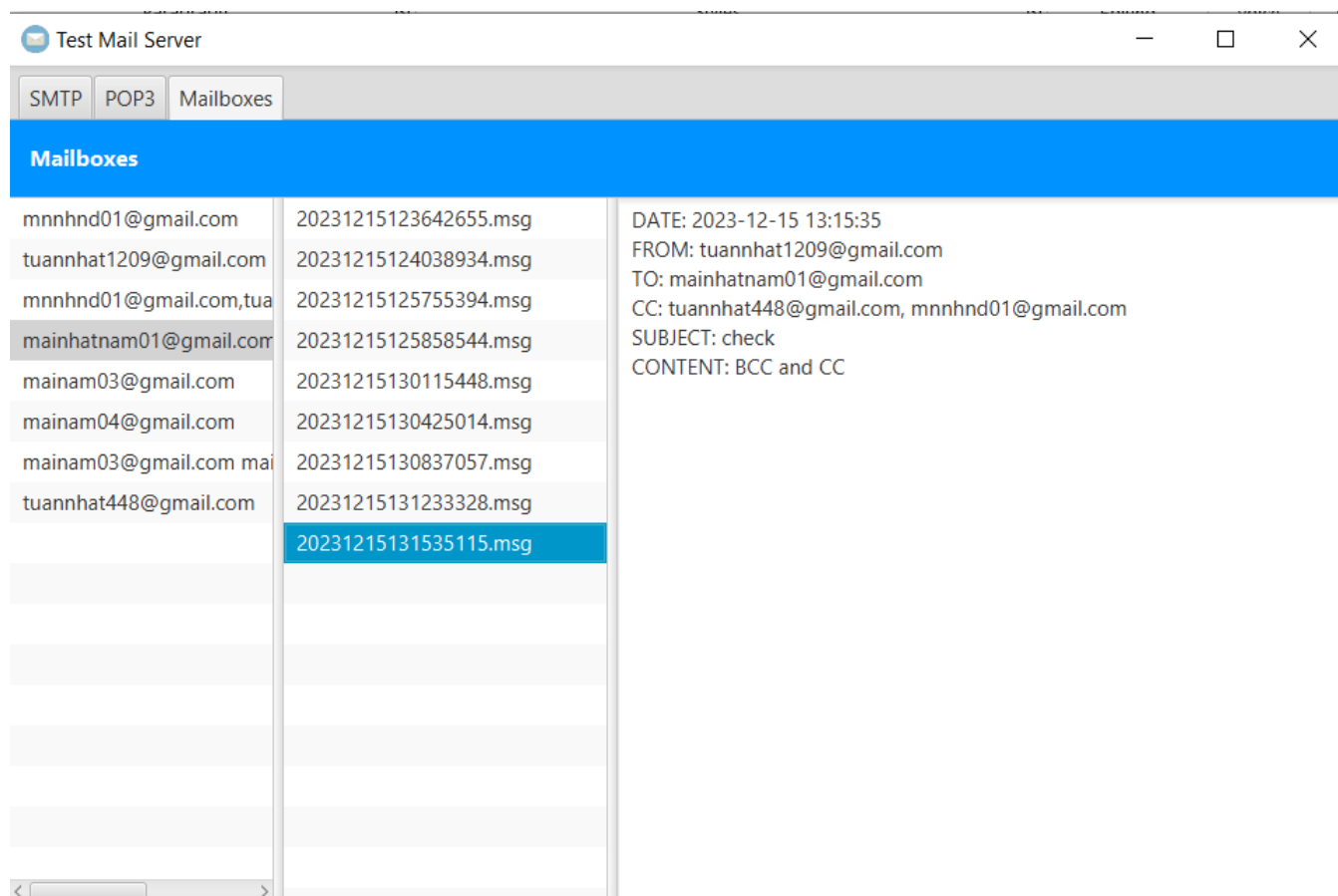
Connecting to SMTP server...
Connecting to POP3 server...
====Send mail====

From: tuannhat1209@gmail.com
To: mainhatnam01@gmail.com
CC: tuannhat448@gmail.com mnnhnd01@gmail.com
BCC: mainam03@gmail.com mainam04@gmail.com
Subject: check
Message: BCC and CC
Attach file (Press Enter to skip):
Send mail successfully!

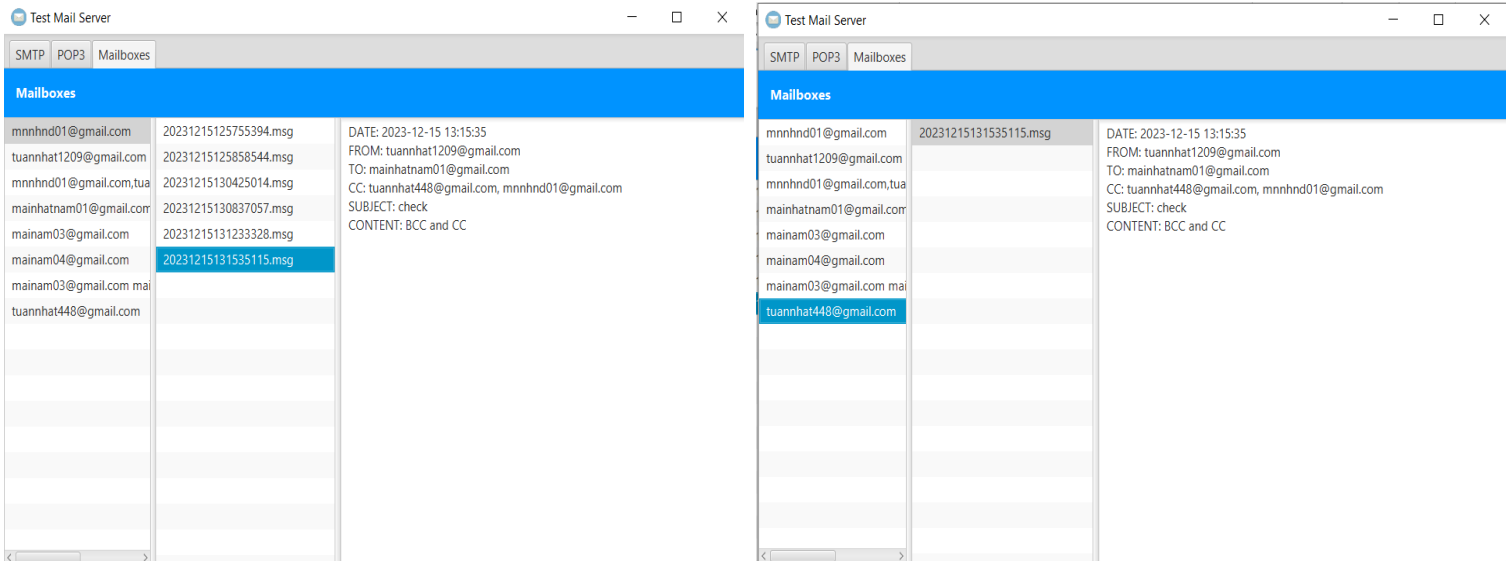
```

+ Ở đây, ta nhập địa chỉ gửi là tuannhat1209@gmail.com và gửi đến TO: mainhatnam01@gmail.com và CC tới 2 địa chỉ tuannhat448@gmail.com và mnnhnd01@gmail.com và BCC tới 2 địa chỉ mainam03@gmail.com và mainam04@gmail.com. Lưu ý, nhóm chúng em sẽ dùng khoảng trắng để tách 2 email kề nhau.

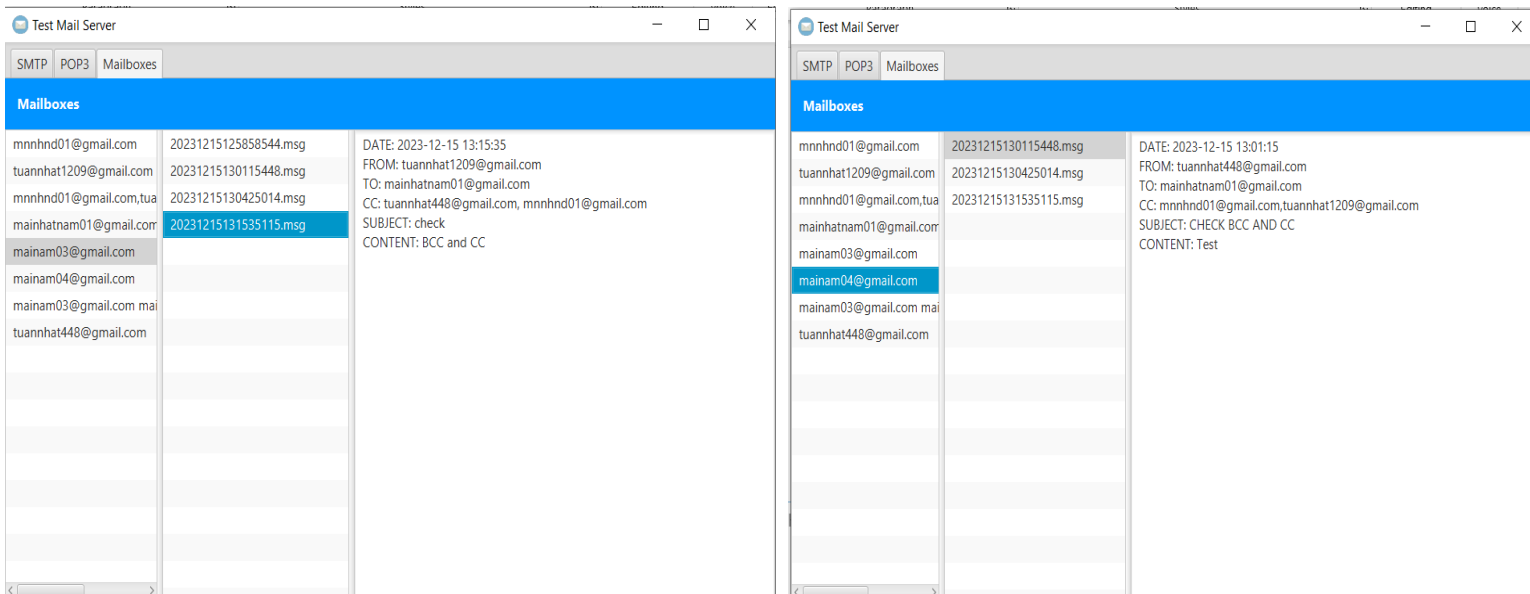
+ Ta sẽ nhìn các hình dưới đây để kiểm chứng kết quả



- Ở hình này, mail mainhatnam01@gmail.com đã nhận được đầy đủ thông tin như lúc ta nhập liệu trong console và đồng thời hiển thị được 2 email nhận dưới dạng CC là tuannhat448@gmail.com và mnnhnd01@gmail.com



- Ở 2 hình này, 2 email nhận dưới dạng CC là tuannhat448@gmail.com và mnnhnd01@gmail.com đã hiển thị rõ đúng thông tin giống với dữ liệu đã nhập ở màn hình console.



- Ở 2 hình này, 2 email nhận dưới dạng BCC là mainam03@gmail.com và mainam04@gmail.com đã hiển thị rõ đúng thông tin giống với dữ liệu đã nhập ở màn hình console tuy nhiên 2 mail này không hiển thị được là đã nhận ở dạng BCC (tức người dùng mail mainam03@gmail.com không biết người dùng mail mainam04@gmail.com)

3. Gửi File ở nhiều định dạng (. pdf, jpg ...)

```
Current user: mainhatnam01@gmail.com
```

1. Send mail
2. Download mail
3. Open mail boxes
4. Exit

```
Your choice: 1
```

- Đầu tiên, ta vẫn chọn 1 để gửi mail và nhập như thông tin console dưới đây:

```
Current user: mainhatnam01@gmail.com
```

1. Send mail

```
Connecting to SMTP server...
```

```
Connecting to POP3 server...
```

```
====Send mail====
```

```
From: mainhatnam01@gmail.com
```

```
To: mainhatnam01@gmail.com
```

```
CC:
```

```
BCC:
```

```
Subject: Test file
```

```
Message: This mail has a lot of files
```

```
Attach file (Press Enter to skip): a.pdf
```

```
Attach file (Press Enter to skip): c.pdf
```

```
File size is too large (>3Mb)!
```

```
Attach file (Press Enter to skip): b.jpg
```

```
Attach file (Press Enter to skip):
```

```
Send mail successfully!
```

```
Press any key to continue . . .
```

- Ta nhập file cần gửi bằng cách dán lại tên file vào ô attach file và chúng ta có thể gửi nhiều file trong cùng 1 email. Lưu ý, nếu chúng ta không muốn gửi file thì chúng ta cứ bấm enter để bỏ qua!

- Chú ý rằng nếu ta gửi 1 file lớn hơn 3MB thì console sẽ báo lỗi và chúng ta phải gửi lại. Trong hình trên File c.pdf có dung lượng lớn hơn 3MB nên console báo lỗi và ta phải nhập file mới.

> Desktop > Mail Client >

Name	Date modified	Type	Size
.idea	12/12/2023 11:07 AM	File folder	
__pycache__	12/15/2023 3:41 PM	File folder	
mail_boxes	12/15/2023 3:14 PM	File folder	
a.pdf	12/9/2023 9:11 PM	Microsoft Edge PD...	638 KB
b.jpg	10/11/2023 12:13 PM	JPG File	216 KB
c.pdf	12/12/2023 11:42 PM	Microsoft Edge PD...	6,928 KB

4. Tải được các email về mail client từ mailbox của Server và quản lý trạng thái đọc của Email

```
=====Welcome to mail client=====
Current user: mainhatnam01@gmail.com
1. Send mail
2. Download mail
3. Open mail boxes
4. Exit
Your choice: █
```

- Để tiến hành tải các email về máy client, ta chọn phím 2.

```
=====Download mail=====
Download mail successfully!
Press any key to continue . . . █
```

- Sau khi nhấn phím 2, console sẽ báo tải thành công. Lúc này, ta có thể vào chọn mục 3 và open mailboxes để kiểm tra tất cả các email này giờ chúng ta đã gửi tới địa chỉ. Sau khi bấm chọn mục 3, màn hình console sẽ hiển thị như sau:

```
=====Mail boxes=====
0. Inbox
1. Important
2. Work
3. Project
4. Spam
Enter. Back to menu
Your choice: █
```

- Ta thấy màn hình hiện list danh sách chọn như thế này, lưu ý rằng tất cả các Email đã gửi đều sẽ ở trong thư mục Inbox nên ta sẽ chọn 0 để tiếp tục.

```
====Mail box====

0. SUBJECT: MAIL CHECK - FROM: mainhatnam01@gmail.com

* 1. SUBJECT: Download email - FROM: mainhatnam01@gmail.com

2. SUBJECT: BCC and CC check - FROM: mainhatnam01@gmail.com

3. SUBJECT: BCC and CC check 02 - FROM: mainhatnam01@gmail.com

4. SUBJECT: Filecheck - FROM: mainam01@gmail.com

5. SUBJECT: Filecheck02 - FROM: mainam01@gmail.com

6. SUBJECT: Autoload check with file - FROM: mainam01@gmail.com

* 7. SUBJECT: File test 01 - FROM: mainhatnam01@gmail.com

* 8. SUBJECT: Test file - FROM: mainhatnam01@gmail.com

Enter. Back

Your choice: █
```

- Sau khi chọn xong, màn hình sẽ cho ra tất cả các email có trong thư mục. Những email có dấu * là những email chúng ta chưa đọc. Giả sử ta muốn đọc Email thứ 8 thì ta nhập 8 và sau đó màn hình sẽ hiển thị đầy đủ nội dung Email như sau:

```
====Mail content====

DATE: 2023-12-15 16:20:54

FROM: mainhatnam01@gmail.com

TO: mainhatnam01@gmail.com

CC:

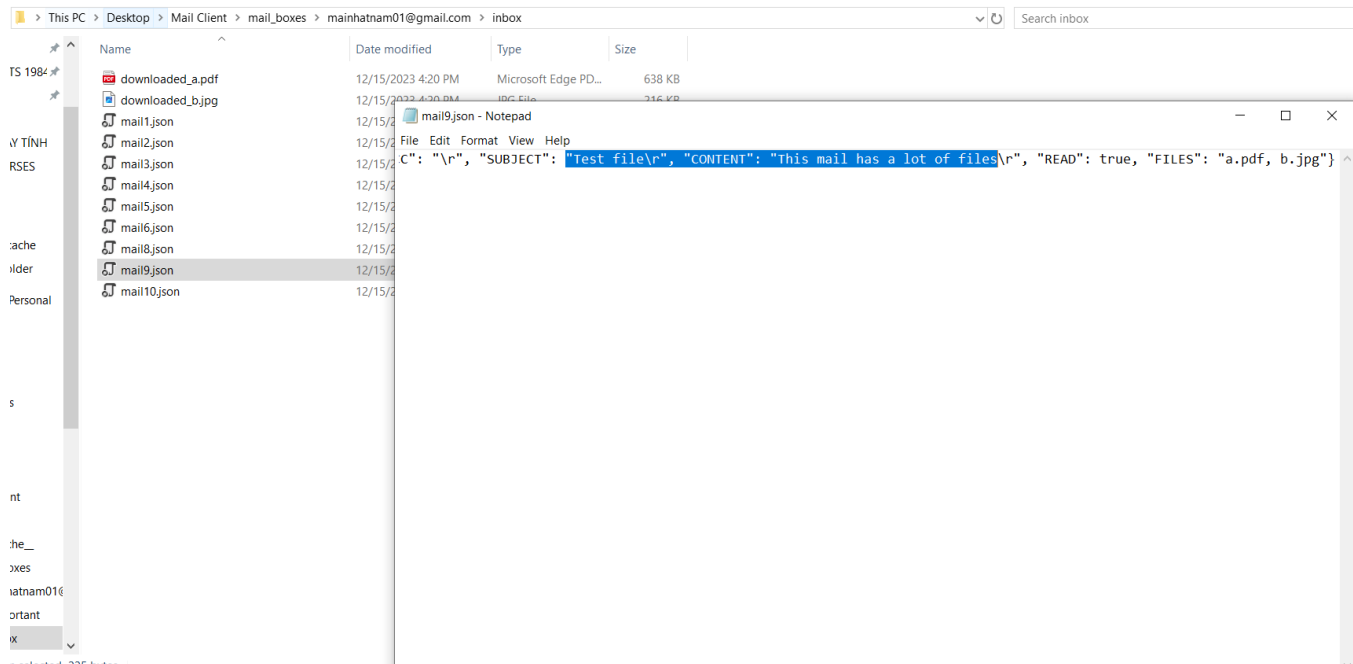
SUBJECT: Test file

CONTENT: This mail has a lot of files

FILES: a.pdf, b.jpg

Press any key to continue . . . █
```

- Màn hình hiện tại sau khi ta nhấn phím 8 đã hiển thị đầy đủ tất cả các nội dung ta đã gửi bao gồm nội dung và 2 file phương tiện. Tiếp theo, để kiểm tra file phương tiện này đã nằm ở máy cục bộ Client chưa thì ta sẽ vào thư mục mailbox tương ứng của máy cục bộ và kiểm tra.



- Như ta thấy trong ảnh trên mail09.json đang chứa nội dung của Email thứ 8 giống màn hình console bên trên và có cả luôn 2 file a.pdf và b.jpg đã được download sẵn. Như vậy, ta hoàn tất việc tải Email kèm file và lưu file xuống máy cục bộ.

5. Xử lý lọc Email

5.1. Lọc email dựa trên địa chỉ người gửi

- Giả sử ta cần di chuyển các email được gửi từ địa chỉ myboss@gmail.com vào thư mục Project. Đầu tiên, ta cũng sẽ tiến hành gửi Email tương tự như bước gửi Email đến 1 người dùng bất kì.

```
Connecting to SMTP server...
Connecting to POP3 server...
====Send mail=====

From: myboss@gmail.com
To: mainhatnam01@gmail.com
CC:
BCC:
Subject: Filter to Project
Message: Check Project
Attach file (Press Enter to skip):
Send mail successfully!

Press any key to continue . . .
```

- Sau đó, để kiểm tra email này đã vào thư mục Project chưa, ta sẽ vào lại menu và chọn 3 (Open Mail Boxes) sau đó chọn tiếp 3 để mở thư mục Project

```
====Mail boxes====  
0. Inbox  
1. Important  
2. Work  
3. Project  
4. Spam  
Enter. Back to menu  
Your choice: 3
```

- Sau khi chọn xong ta thấy ở đây có 1 Email có tựa "Filter to Project" người gửi là myboss@gmail.com chưa đọc ta có thể chọn đọc email này.

```
====Mail box====  
* 0. SUBJECT: Filter to Project - FROM: myboss@gmail.com  
Enter. Back  
Your choice:
```

- Sau khi chọn đọc email xong, màn hình console hiển thị như sau:

```
DATE: 2023-12-15 16:49:22  
FROM: myboss@gmail.com  
TO: mainhatnam01@gmail.com  
CC:  
SUBJECT: Filter to Project  
CONTENT: Check Project  
FILES: None  
Press any key to continue . . .
```

- Màn hình sẽ hiển thị đúng nội dung Email chúng ta đang cần lọc vào thư mục Project.

5.2. Lọc email dựa trên Subject

- Giả sử ta cần di chuyển các email có Subject là "urgent" hoặc "ASAP" vào thư mục Important. Đầu tiên, ta cũng sẽ tiến hành gửi Email tương tự như bước gửi Email đến 1 người dùng bất kì.

```

○ Connecting to SMTP server...
Connecting to POP3 server...
=====Send mail=====

From: mainam01@gmail.com
To: mainhatnam01@gmail.com
CC:
BCC:
Subject: urgent
Message: This is an emergency case. You need to be fast. Otherwise, Ms Linh will punish you.
Attach file (Press Enter to skip):
Send mail successfully!

Press any key to continue . . .

```

- Sau đó, để kiểm tra email này đã vào thư mục Important chưa, ta sẽ vào lại menu và chọn 3 (Open Mail Boxes) sau đó chọn tiếp 1 để mở thư mục Important.

```

○ =====Mail boxes=====

0. Inbox
1. Important
2. Work
3. Project
4. Spam
Enter. Back to menu
Your choice: 1

```

- Sau khi chọn xong ta thấy ở đây có 1 Email có Subject "urgent" chưa đọc ta có thể chọn đọc email này.

```

=====Mail box=====

* 0. SUBJECT: urgent - FROM: mainam01@gmail.com

* 1. SUBJECT: urgent - FROM: mainam01@gmail.com

2. SUBJECT: urgent - FROM: mainam01@gmail.com
Enter. Back
Your choice:

```

- Sau khi chọn đọc email xong, màn hình console hiển thị như sau:

```

DATE: 2023-12-15 18:15:04
FROM: mainam01@gmail.com
TO: mainhatnam01@gmail.com
CC:
SUBJECT: urgent
CONTENT: This is an emergency case. You need to be fast. Otherwise, Ms Linh will punish you.
FILES: None
Press any key to continue . . .

```

➔ Màn hình sẽ hiển thị đúng nội dung Email chúng ta đang cần lọc vào thư mục Important.

5.3. Lọc email dựa trên nội dung

- Giả sử ta cần di chuyển các email trong nội dung chứa các từ "report", "meeting" vào thư mục Work. Đầu tiên, ta cũng sẽ tiến hành gửi Email tương tự như bước gửi Email đến 1 người dùng bất kì.

```
Connecting to SMTP server...
Connecting to POP3 server...
=====Send mail=====

From: mainam01@gmail.com
To: mainhatnam01@gmail.com
CC:
BCC:
Subject: Classes
Message: MMT report
Attach file (Press Enter to skip):
Send mail successfully!

Press any key to continue . . .
```

- Sau đó, để kiểm tra email này đã vào thư mục Work chưa, ta sẽ vào lại menu và chọn 3 (Open Mail Boxes) sau đó chọn tiếp 2 để mở thư mục Work.

```
=====Mail boxes=====

0. Inbox
1. Important
2. Work
3. Project
4. Spam

Enter. Back to menu

Your choice: 2
```

- Sau khi chọn xong ta thấy ở đây có 1 Email có tựa "Classes" chưa đọc ta có thể chọn đọc email này.

```
=====Mail box=====

* 0. SUBJECT: Classes - FROM: mainam01@gmail.com

Enter. Back

Your choice: 
```

- Sau khi chọn đọc email xong, màn hình console hiển thị như sau:

```
DATE: 2023-12-15 18:20:33

FROM: mainam01@gmail.com
TO: mainhatnam01@gmail.com
CC:
SUBJECT: Classes
CONTENT: MMT report
FILES: None

Press any key to continue . . .
```

➔ Màn hình sẽ hiển thị đúng nội dung Email chúng ta đang cần lọc vào thư mục Work.

5.4 Lọc email mà nó có chứa từ khóa hoặc nội dung vào thư mục cụ thể (Ví dụ: thư mục Spam)

- Giả sử ta cần di chuyển các email có nội dung hoặc Subject chứa từ "virus" hay "crack", ... vào thư mục Spam. Đầu tiên, ta cũng sẽ tiến hành gửi Email tương tự như bước gửi Email đến 1 người dùng bất kì.

```
Connecting to SMTP server...
Connecting to POP3 server...
=====Send mail=====

From: mainam01@gmail.com
To: mainhatnam01@gmail.com
CC:
BCC:
Subject: virus
Message: crack
Attach file (Press Enter to skip):
Send mail successfully!

Press any key to continue . . .
```

- Sau đó, để kiểm tra email này đã vào thư mục Spam chưa, ta sẽ vào lại menu và chọn 3 (Open Mail Boxes) sau đó chọn tiếp 4 để mở thư mục Spam

```
=====Mail boxes=====

0. Inbox
1. Important
2. Work
3. Project
4. Spam

Enter. Back to menu

Your choice: 4
```

- Sau khi chọn xong ta thấy ở đây có 1 Email có tựa "virus" chưa đọc ta có thể chọn đọc email này.

```
=====Mail box=====

* 0. SUBJECT: virus - FROM: mainam01@gmail.com

Enter. Back

Your choice:
```

- Sau khi chọn đọc email xong, màn hình console hiển thị như sau:

```
DATE: 2023-12-15 18:22:38

FROM: mainam01@gmail.com

TO: mainhatnam01@gmail.com

CC:

SUBJECT: virus

CONTENT: crack

FILES: None

Press any key to continue . . .
```

→ Màn hình sẽ hiển thị đúng nội dung Email chúng ta đang cần lọc vào thư mục Spam.

6. Xử lí Autoload

```
Server:
  MailServer: 127.0.0.1
  SMTP: 2500
  POP3: 1100
  bufferSize: 2046
Authentication:
  username: mainhatnam01@gmail.com
  password: xxxx
Filter:
  Important:
    From: []
    Subject: ['urgent', 'ASAP']
    Content: []
  Work:
    From: []
    Subject: []
    Content: ['report', 'meeting']
  Spam:
    From: []
    Subject: []
    Content: ['virus', 'hack', 'crack']
  Project:
    From: ['myboss@gmail.com']
    Subject: []
    Content: []
Autoload:
  Time: 5
```

- Như ta đang thấy trên file config, thời gian để ứng dụng reload lại email mới sau khi vừa được gửi đi là 5 giây. Thời gian ở file này ta có thể cấu hình lại theo tùy ý người dùng.

VII. TÀI LIỆU THAM KHẢO

- [1]: Kurose, J. F., & Ross, K. W, *SMTP Programming Lab*. Retrieved from [Link](#)
- [2]: "Computer Networking: A Top-down Approach", Kurose & Ross, 5th edition, Addison-Wesley, 2010.
- [3]: Tài liệu hướng dẫn đồ án Socket
- [4]: Real Python, *Python Sockets Tutorial*. Retrieved from [Link](#)

