**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN ĐHQG THÀNH PHỐ HỒ CHÍ MINH**

**Khoa Công Nghệ Thông Tin**

**\*\*\***



**MÔN HỌC: MẠNG MÁY TÍNH**

**BÁO CÁO ĐỒ ÁN SOCKET**

## Giảng viên hướng dẫn: TS. Chung Thùy Linh Lớp: 22CTT4

**Các thành viên nhóm:** Trần Đức Minh – 22120212 Phạm Tấn Nghĩa– 22120230

Hoàng Thanh Thảo Nguyên – 22120235

# LỜI CẢM ƠN

Lời đầu tiên, xin trân trọng cảm ơn Giảng viên Chung Thùy Linh vì đã tận tình hướng dẫn chúng em trong quá trình học tập cũng như trong việc hoàn thành báo cáo này để hỗ trợ cho đồ án.

Xin chân thành cảm ơn các Thầy, Cô thuộc khoa Công nghệ Thông tin nói riêng và toàn bộ Thầy, Cô ở các khoa nói chung trường Đại Học Khoa học Tự nhiên, Đại học Quốc gia Thành phố Hồ Chí Minh đã tận tình giảng dạy cho chúng em trong suốt thời gian học tập.

Do giới hạn kiến thức và khả năng lý luận của bản thân còn nhiều thiếu sót và hạn chế, kính mong sự chỉ dẫn và đóng góp của các Thầy, Cô để bản báo cáo và sản phẩm đồ án của chúng em được hoàn thiện hơn. Xin chân thành cảm ơn!

Thành phố Hồ Chí Minh, ngày 16 tháng 12 năm 2023 Đại diện nhóm

Phạm Tấn Nghĩa

**MỤC LỤC**

[LỜI CẢM ƠN 2](#_bookmark0)

1. [GIỚI THIỆU 4](#_bookmark1)
2. [NỘI DUNG 5](#_bookmark2)
   1. [Mức độ hoàn thành 5](#_bookmark3)
      1. [Giao thức trao đổi giữa client và server 6](#_bookmark4)
3. [Môi trường lập trình và thư viện 7](#_bookmark5)
   1. [Môi trường lập trình 7](#_bookmark6)
   2. [Thư viện 8](#_bookmark7)
4. [Giới thiệu chức năng 9](#_bookmark8)
5. Hướng dẫn sử dụng Error! Bookmark not defined.
   1. Trang chủ…………………………………………………………………………10
   2. New Email…………………………………………………………………………………………………………11
   3. Tải file từ Email…………………………………………………………………..11
   4. Các bộ lọc file…………………………………………………………………….12
6. [TÀI LIỆU THAM KHẢO 12](#_bookmark9)

## GIỚI THIỆU

* 1. **Thành viên nhóm**

|  |  |  |
| --- | --- | --- |
| MSSV | Họ và tên | Email |
| 22120212 | Trần Đức Minh | [22120212@student.hcmus.edu.vn](mailto:22120212@student.hcmus.edu.vn) |
| 22120230 | Phạm Tấn Nghĩa | [22120230@student.hcmus.edu.vn](mailto:22120230@student.hcmus.edu.vn) |
| 22120235 | Hoàng Thanh Thảo Nguyên | [22120235@student.hcmus.edu.vn](mailto:22120235@student.hcmus.edu.vn) |

* 1. **Kế hoạch phân chia**

|  |  |  |
| --- | --- | --- |
| MSSV | Họ và tên | Nhiệm vụ |
| 22120212 | Trần Đức Minh | * Tải nội dung có đính kèm file và xem trước không cần tải xuống. * Quản lí trạng thái email (đã/ chưa đọc). * Xử lí tự động tải email từ Mailbox theo thời gian cấu hình trong file config. * Tải các email về mail client từ Mailbox của Server. * Thiết kế, lập trình giao diện. |
| 22120230 | Phạm Tấn Nghĩa | * Xử lí lọc email theo: address, subject, content của email. * Gửi được Attach File (<3MB). * Nghiên cứu các File config dạng File có cấu trúc (.json). * Gửi email với Bcc. |
| 22120235 | Hoàng Thanh Thảo Nguyên | * Viết báo cáo * Gửi email với To, Cc. * Hỗ trợ thiết kế giao diện. |

## NỘI DUNG

Chương trình Mail Client sử dụng Socket là một ứng dụng đa nhiệm và linh hoạt, được thiết kế để cung cấp giao diện đơn giản và hiệu quả cho việc gửi và nhận Mail thông qua các giao thức truyền thông.

Với sự tiện lợi của lập trình Socket, chương trình giúp người dùng tương tác một cách thuận tiện với máy chủ Mail của họ để quản lý thư đi và thư đến.

### Mức độ hoàn thành

|  |  |  |
| --- | --- | --- |
| Stt | Nội dung | Mức độ |
| 1 | * Gửi email đến một hoặc nhiều địa chỉ người nhận (To) * Gửi email đến một hoặc nhiều địa chỉ người nhận (CC) | 100% |
| 2 | * Gửi được Email với BCC | 100% |
| 3 | * Gửi được các định dạng file như: \*.docx, \*.pdf, \*.jpg, \*.zip, \*.txt, … * Gửi được nhiều file đính kèm trong cùng 1 email * Có giới hạn dung lượng file gửi | 100% |
| 4 | * Tải nội dung email có kèm file, có lưu file xuống máy cục bộ của client | 100% |
| 5 | * Quản lý trạng thái email (đã đọc/ chưa đọc) | 70% |
| 6 | * Lọc email dựa trên địa chỉ người gửi * Lọc email dựa trên subject * Lọc email dựa trên nội dung * Di chuyển email vào thư mục cụ thể/ Spam | 100% |
| 7 | * Xử lý tự động tải email từ Mailbox theo thời gian cấu hình trong config file | 100% |
| 8 | * Có sử dụng file config dạng file có cấu trúc (\*.json.) | 100% |
| 9 | * Báo cáo | 100% |
|  | **Tổng** | 100% |

(√ : làm được , ○: chưa làm được)

### Kịch bản giao tiếp của chương trình

### Giao thức trao đổi giữa Client và Server

#### Giao thức SMTP

Giao thức **SMTP** (**Simple Mail Transfer Protocol**) là một tiêu chuẩn truyền thông được sử dụng rộng rãi để gửi email trên Internet. **SMTP** chịu trách nhiệm cho việc truyền tải thông điệp email từ máy gửi đến máy chủ email của người nhận. Quá trình này bao gồm mở kết nối, xác thực người gửi, và chuyển thông điệp email.

Giao thức này đóng vai trò quan trọng trong quá trình gửi email, và nó thường được sử dụng trong các ứng dụng và dịch vụ email trực tuyến. **SMTP** cũng có thể sử dụng các tùy chọn bảo mật như TLS/SSL để đảm bảo an toàn trong quá trình truyền dữ liệu.

#### Giao thức POP3

Giao thức **POP3** (**Post Office Protocol 3**) là một tiêu chuẩn quan trọng trong việc nhận thư điện tử. Người dùng sử dụng **POP3** để kết nối đến máy chủ email và tải về thư từ hộp thư của mình. Quá trình này bao gồm xác thực, truy cập hộp thư trên máy chủ, tải về thư, và có thể xóa thư khỏi máy chủ sau khi tải về. **POP3** cung cấp khả năng quản lý thư điện tử trực tuyến một cách tiện lợi và đơn giản cho người dùng.

### Cấu trúc thông điệp

#### Cấu trúc việc gửi mail:

##### + Giao thức SMTP:

* + - * + SMTP là một giao thức được sử dụng để truyền tải email qua mạng. Tạo và thiết lập một kết nối TCP thông qua Socket để gửi email tới một máy chủ SMTP (host).

##### + Cấu trúc Thông điệp Email:

* + - * + Thông điệp email được xây dựng bằng cách gửi các lệnh và dữ liệu theo định dạng SMTP.
        + Mỗi lệnh được gửi thông qua OutputStream và đọc bằng BufferedReader từ InputStream của Socket.

##### + Các Lệnh SMTP Quan trọng:

* + - * + **EHLO** example.com: Mở đầu với một lời chào từ client tới server SMTP.
        + **MAIL FROM**: <sender-email>: Xác định email người gửi.
        + **RCPT TO**: <recipient-email>: Xác định email người nhận.
        + **DATA**: Bắt đầu quá trình gửi dữ liệu của email.
        + **QUIT**: Kết thúc phiên làm việc với máy chủ SMTP.

##### + Cấu trúc Email và Đính kèm Tệp:

* + - * + Thông điệp email được xây dựng với các trường như Subject, From, To, Cc, Bcc, và Date.
        + Nếu có đính kèm tệp (checkSendFile là true), mã nguồn thêm vào phần đính kèm với các phần như Content-Type, Content-Disposition, và Content- Transfer-Encoding.

##### + Xử lý Tệp Đính kèm:

* + - * + Đối với mỗi tệp đính kèm, mã nguồn thực hiện việc gửi nội dung của tệp dưới dạng mã hóa Base64 để đảm bảo việc truyền tải thông qua giao thức SMTP.

##### + Cơ sở Dữ liệu:

* + - * + Trong mã nguồn này, không có sử dụng cơ sở dữ liệu (database). Thông tin như host, port\_s, username, recipientEmails, ccEmails, và bccEmails được truyền trực tiếp từ mã nguồn.

#### Cấu trúc việc nhận mail:

##### + Giao thức POP3:

* + - * + **POP3** là một giao thức được sử dụng để lấy thư từ một máy chủ email. Sử dụng Socket để tạo một kết nối TCP với máy chủ **POP3** để truy cập hộp thư.

##### + Các Lệnh POP3 Quan trọng:

* + - * + **USER** <username>: Xác định tên người dùng để đăng nhập vào hộp thư.
        + **PASS** <password>: Xác định mật khẩu để đăng nhập vào hộp thư
        + **STAT**: Truy vấn thông tin về hộp thư, bao gồm số lượng email và tổng kích thước của chúng.
        + **LIST**: Liệt kê thông tin về các email trong hộp thư.
        + **QUIT**: Kết thúc phiên làm việc với máy chủ POP3.

#### + Lấy danh sách Email:

Sau khi đăng nhập thành công (USER và PASS), sử dụng lệnh STAT để lấy thông tin tổng quan về hộp thư, sử dụng một hàm getEmailList để xử lý phản hồi và trả về danh sách email.

#### + Cơ sở Dữ liệu:

Trong đoạn mã này, không có sử dụng cơ sở dữ liệu (database). Thông tin như host, port\_r, username, và password được truyền trực tiếp từ mã nguồn.

### Môi trường lập trình và thư viện

### Môi trường lập trình

#### Phần mềm phát triển: Microsoft Visual Studio Code Ngôn ngữ lập trình: Java

**Tổ chức cơ sở dữ liệu: File .json**

Json (JavaScript Object Notation) là một định dạng dữ liệu đơn giản và nhẹ được sử dụng để truyền tải và trao đổi dữ liệu giữa máy tính. JSON thường được sử dụng trong

các ứng dụng web để truyền dữ liệu giữa máy khách (Client) và máy chủ (Server), nhưng cũng được sử dụng trong nhiều ngữ cảnh khác nhau.

File cấu trúc dùng để quản lý dữ liệu người dùng và các bộ lọc được sử dụng trong đồ án: ***“config.json”***

***File config.json*** mà chương trình sử dụng được tổ chức thành 2 thành phần chính:

**General** (chung) và **Filter** (bộ lọc).

* **General** bao gồm các thành phần:
  + **Username**: Đây là tên đăng nhập (username) được sử dụng để kết nối đến máy chủ Mail-Server.
  + **Password**: Mật khẩu liên quan đến tên đăng nhập.
  + **Mailserver**: Địa chỉ IP thực hiện việc kết nối đến Mail-Server, các user có cùng địa chỉ IP sẽ có thể giao tiếp với nhau thông qua Mail-Server.
  + **SMTP**: số cổng SMTP được sử dụng cho việc truyền dữ liệu đến Mail-Server.
  + **POP3**: Số cổng POP3 được sử dụng cho việc nhận dữ liệu đến Mail-Server.
  + **AutoLoad**: thời gian tự động tải Mail trong MailBox về, tùy theo thời gian cài đặt trong file ***config.json*** này.
* **Filter**:
  + **Project**: Lọc Mail tải về từ MailBox vào trong mục Folder Project, các Mail được đưa vào Folder Project thường là từ những người gửi được đánh dấu sẵn. VD: ***[“***[***ahihi@example.com***](mailto:ahihi@example.com)***”, “***[***ahuu@example.com***](mailto:ahuu@example.com)***”, …].***
  + **Important**: Lọc Mail tải về từ MailBox vào trong mục Folder Important, các Mail được đưa vào Folder Important thường có Subject hoặc Content chứa các từ khóa mang ***ý nghĩa quan trọng*** như ***[“ASAP”, “urgent”, …].***
  + **Work**: Lọc Mail tải về từ MailBox vào trong mục Folder Work, các Mail được đưa vào Folder Work thường có Subject hoặc Content chứa các từ khóa liên quan đến ***công việc*** như ***[“report”, “meeting”, …]***
  + **Spam**: Lọc Mail tải về từ MailBox vào trong mục Folder Spam, các Mail được đưa vào Folder Spam thường có Subject hoặc Content chứa các từ khóa mang nội dung ***độc hại*** đối với người dùng như ***[“virus”, “hack”, “crack”, …]***

### Thư viện hỗ trợ

Dưới đây là 1 số thư viện thông dụng của ngôn ngữ lập trình Java được sử dụng trong đồ án nhằm hỗ trợ việc xây chương trình:

* + - * **“Java.io” package:** Cung cấp các lớp để thao tác với đầu vào và đầu ra. Sử dụng nó để làm việc với tệp tin (File, FileInputStream, FileOutputStream, FileWriter).
      * **“Java.net” package:** Cung cấp các lớp để làm việc với mạng, bao gồm Socket. Trong mã nguồn sử dụng Socket để tạo và quản lý kết nối mạng.
      * **“Java.nio.file” package:** Cung cấp các lớp để thực hiện các thao tác trên hệ thống tệp tin. Sử dụng nó để làm việc với đường dẫn tệp tin (Paths, Files).
      * **“Java.nio.charset” package:** Là một class chứa các biến static định nghĩa các bộ ký tự tiêu chuẩn. Trong mã nguồn, sử dụng nó để xác định bộ ký tự UTF-8 khi đọc và ghi dữ liệu.
      * **“Java.util.date” package:** Được sử dụng để định dạng và làm việc với thời gian và ngày tháng. Trong mã nguồn, sử dụng nó để tạo định dạng ngày giờ cho việc ghi log.
      * **“Java.util.Base64” package:** Cung cấp các lớp và phương thức để thực hiện mã hóa và giải mã dữ liệu theo chuẩn Base64. Trong mã nguồn, sử dụng nó để mã hóa và giải mã dữ liệu Base64.
      * **“Java.util.concurrent.Executors” và**

**“Java.util.concurrent.ScheduledExcutorService” package:** Cung cấp một framework cho việc quản lý các threads và thực hiện các công việc định kỳ. Trong mã nguồn, sử dụng nó để lên lịch thực hiện một số công việc theo định kỳ.

* + - * **“org.json.simple” package:** Cung cấp các lớp để làm việc với JSON trong Java. Trong mã nguồn, sử dụng nó để parse và tạo các đối tượng JSON (JSONObject, JSONArray, JSONParser).

Trên tất cả, mã nguồn của chương trình thực hiện nhiều công việc khác nhau, bao gồm **đọc** và **ghi dữ liệu từ/đến tệp**, làm **việc với mạng thông qua Socket**, và **xử lý dữ liệu Json.**

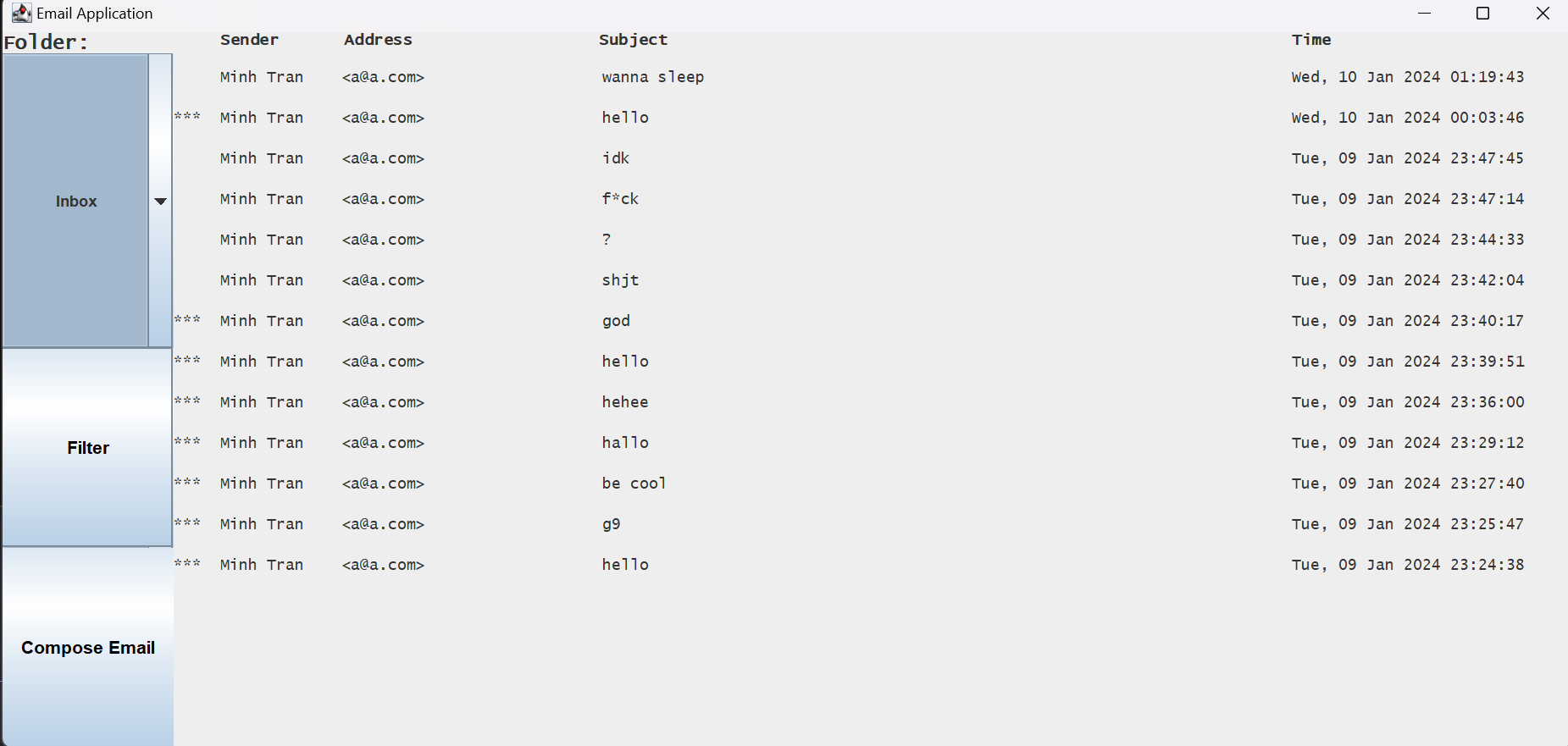
### Giới thiệu chức năng

#### Sau đây là cách sử dụng chương trình:

**-** Mặc định lúc vừa chạy, chương trình sẽ ở Folder Inbox của hộp thư, các email nằm trong Folder Inbox sẽ được hiển thị gồm các thông số (Sender, Address, Subject và Time).

- Bên trái màn hình làm việc, lần lượt từ trên xuống sẽ là: Thanh chọn Folder (“Inbox”, “Project”, “Work”, “Importan”, “Spam”, “All” và sẽ có 1 mục là “Exit” dùng để thoát chương trình.

- Nhưng thư ở trạng thái chưa đọc (unread) sẽ được đánh 3 dấu \* ở phái bên trái (\*\*\*).



* Khi nhấn chuột vào 1 thư bất kì, chi tiết thư sẽ được hiện lên trên 1 cửa sổ với các thông số như sau:
* Trong đó:

+ From: địa chỉ người gửi.

+ To: địa chỉ nhận.

+ At: thời gian gửi.

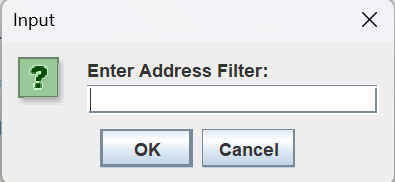
+ Subject: tiêu đề thư.

+ Content: nội dung thư.

+ Attach File: các file đính kèm, có thể xem trước (preview) hoặc tải về (download).



* Khi nhấn vào Lọc thư (Filter), sẽ hiện lên 1 cửa sổ tương ứng với trường muốn lọc, nhập thông tin cần lọc và nhấn OK để nhận được danh sách thư mong muốn.



* Màn hình gửi thư sẽ có các trường tương ứng nhập vào để gửi thư, trong đó:

+ To: người nhận thư.

+ Cc: gửi thư cho nhiều người (có tiết lộ địa chỉ thư).

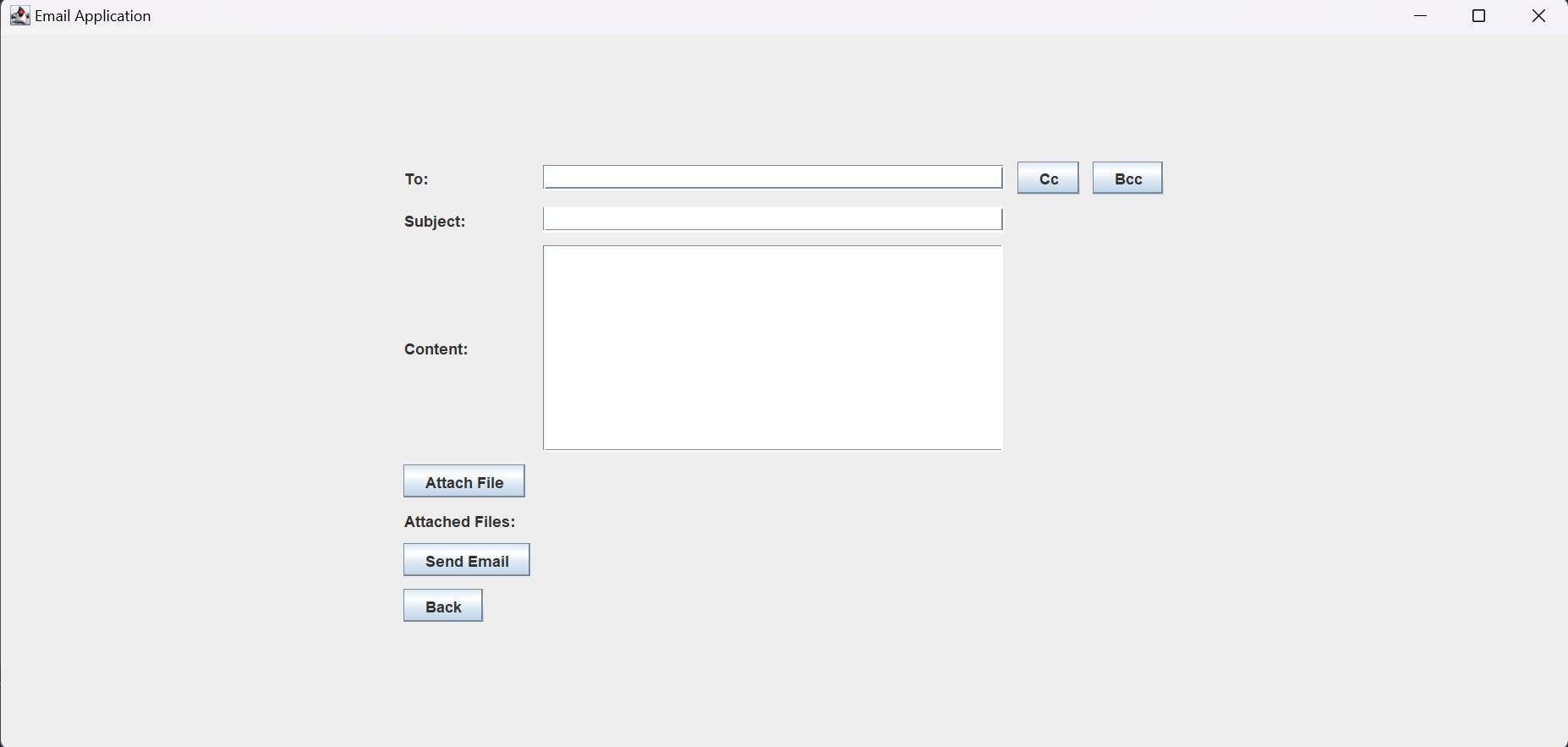
+ Bcc: gửi thư cho nhiều người (không tiết lộ địa chỉ thư).

+ Content: nội dung thư.

+ Attach File: các tập tin đính kèm trong thư.

+ Send mail: nhấn để bắt đầu quá trình gửi thư.

+ Back: trở về giao diện đọc thư.



## TÀI LIỆU THAM KHẢO

1. Giáo trình Mạng máy tính
2. Google Schoolar
3. OpenAI