|  |  |
| --- | --- |
| **logo moi** | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**BÁO CÁO LẬP TRÌNH MẠNG MÁY TÍNH**

**PHẦN MỀM MÔ PHỎNG HỆ THỐNG EMAIL BẰNG JAVA SOCKET**

Ngành: **CÔNG NGHỆ THÔNG TIN**

Nhóm Sinh viên thực hiện:

Tên Lớp MSSV

Nguyễn Thanh Siêu 21DTHE4 2180609116

Trần Lê Uyên Phương 21DTHE4 2180608925

Giảng viên hướng dẫn: **ThS. Đàm Minh Lịnh**

TP. Hồ Chí Minh, 2023

# LỜI CẢM ƠN

Lời đầu tiên, chúng em xin chân thành cảm ơn Thầy Đàm Minh Lịnh hướng dẫn đồ án môn học cho chúng em.

Sau thời gian học tập và và tìm hiểu dưới sự hướng dẫn của thầy, chúng em đã rút ra được rất nhiều kinh nghiệm về cách thức phân tích và thiết kế cũng như kỹ năng làm việc mà không chỉ đơn giản là đọc trong sách vở có thể có được và một lần nữa, chúng em xin gửi lời cảm ơn sâu sắc nhất đến Thầy đã dạy bảo và hướng dẫn những kiến thức chuyên môn cần có để chúng em áp dụng tốt nhất những gì đã được học suốt những thời gian qua.

Trong quá trình thực hiện và làm báo cáo, do còn thiếu nhiều kinh nghiệm thực tế nên không tránh khỏi những sai sót. Chúng em rất mong nhận được những ý kiến đóng góp của Thầy để giúp chúng em trong lĩnh vực này được hoàn thiện hơn. Đó là hành trang quý giá giúp chúng em hoàn thiện kiến thức của mình sau này.

Chúng em xin chân thành cảm ơn và trân trọng kính chào!

# NHẬN XÉT & ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN

Qua quá trình học tập:

Giáo viên hướng dẫn có một số nhận xét, đánh giá như sau:

1/- Quá trình học tập

2/- Thực hiện báo cáo

Đánh giá chung

Kết quả đạt được:

Điểm đánh giá việc thực hiện báo cáo………/10

*TP. HCM, Ngày 03 tháng 11 năm 2023*

**GIẢNG VIÊN HƯỚNG DẪN**

**ThS.**

# MỤC LỤC

[**LỜI CẢM ƠN 1**](#_1fob9te)

[**NHẬN XÉT & ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN 1**](#_3znysh7)

[**MỤC LỤC 3**](#_tyjcwt)

[**DANH MỤC HÌNH ẢNH 5**](#_3dy6vkm)

[**1 CHƯƠNG 1: TỔNG QUAN VÀ CƠ SỞ LÝ THUYẾT 2**](#_4d34og8)

[1.1 Tổng quan 2](#_2s8eyo1)

[1.2 Đồ án đã thực thi được những yêu cầu 2](#_17dp8vu)

[1.2.1 Yêu cầu về chức năng phía client (phải có GUI): 2](#_3rdcrjn)

[1.2.2 Yêu cầu về chức năng phía server (không cần GUI): 2](#_ic3ge6r1upun)

[1.2.3 Yêu cầu chung: 2](#_hot5ps93d4tv)

[1.3 Xu hướng công nghệ 3](#_1ksv4uv)

[1.3.1 Khái niệm ngôn ngữ lập trình Java 3](#_44sinio)

[1.3.2 Tìm hiểu mô hình client-server 4](#_2jxsxqh)

[**2 CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ 6**](#_1y810tw)

[3.1 Quan hệ giữa các bảng 6](#_4i7ojhp)

[3.2 Thiết kế các bảng 6](#_1ci93xb)

[3.2.1 Bảng lưu thông tin người dùng 6](#_3whwml4)

[3.2.2 Bảng lưu thông tin giỏ hàng 6](#_qsh70q)

[3.2.3 Bảng lưu thông tin sản phẩm 7](#_1pxezwc)

[3.2.4 Bảng lưu danh mục sản phẩm 7](#_2p2csry)

[3.2.5 Bảng lưu danh sách hoá đơn đã được lập 7](#_3o7alnk)

[3.2.6 Bảng lưu chi tiết hoá đơn đã được lập 7](#_ihv636)

[3.3 Các mô hình dữ liệu 8](#_1hmsyys)

[3.3.1 Mô hình UseCase tổng quát 8](#_41mghml)

[3.3.2 Mô hình UseCase quản lý 9](#_vx1227)

[3.3.3 Mô hình UseCase mua hàng 9](#_1v1yuxt)

[3.4 Sơ đồ ClassDiagram 10](#_2u6wntf)

[3.5 Sơ đồ ERD 10](#_3tbugp1)

[3.6 Thiết kế giao diện website 11](#_nmf14n)

[3.6.1 Giao diện trang chủ 11](#_37m2jsg)

[3.6.2 Giao diện trang đăng nhập 11](#_46r0co2)

[3.6.3 Giao diện trang đăng ký tài khoản 12](#_111kx3o)

[3.6.4 Giao diện trang danh sách sản phẩm 12](#_206ipza)

[3.6.5 Giao diện trang thông tin tài khoản 13](#_2zbgiuw)

[3.6.6 Giao diện trang lịch sử mua hàng 13](#_3ygebqi)

[3.6.7 Giao diện trang giỏ hàng 14](#_sqyw64)

[3.6.8 Giao diện trang đổi mật khẩu 14](#_1rvwp1q)

[3.6.9 Giao diện trang quản lý danh sách đơn hàng 15](#_2r0uhxc)

[3.6.10 Giao diện trang quản lý người dùng 15](#_3q5sasy)

[3.6.11 Giao diện trang quản lý sản phẩm 16](#_kgcv8k)

[3.6.12 Giao diện trang thêm sản phẩm 16](#_1jlao46)

[3.6.13 Giao diện trang sửa thông tin sản phẩm 17](#_2iq8gzs)

[**4 CHƯƠNG 3: KẾT LUẬN 18**](#_3hv69ve)

[4.1 Ưu điểm 18](#_1x0gk37)

[4.2 Nhược điểm 18](#_4h042r0)

[4.3 Hướng phát triển toàn diện 18](#_2w5ecyt)

# DANH MỤC HÌNH ẢNH

Hình 2.1: Sơ đồ tổng quát mô hình 6

Hình 2.2: Mô hình ERD 6

Hình 2.3: Bảng lưu thông tin tài khoản 7

Hình 2.4: Bảng lưu thông tin tin dung lượng hộp thư 7

Hình 2.5: Bảng lưu thông tin tệp đính kèm 7

Hình 2.6 Bảng lưu thông tin chi tiết người dùng 7

Hình 2.7 Bảng lưu thông tin chi tiết tệp đính kèm 8

Hình 2.8 Bảng lưu thông tin thư 8

Hình 2.9 Bảng lưu thông tin định dạng của thư 8

Hình 2.10 Bảng lưu thời gian đăng nhập/đăng xuất 9

Hình 2.11 Bảng lưu thông tin người nhận thư 9

Hình 2.12 Bảng lưu thông tin chi tiết định dạng 9

Hình 2.13 Bảng lưu chi tiết phân loại thư 10

Hình 2.14 Procedure xóa thư 10

Hình 2.15 Procedure thêm thư mới 10

Hình 2.16 Procedure gửi thư về cho client 11

Hình 3.1 Giao diện trang chủ 12

Hình 3.2 Giao diện trang đăng nhập 13

Hình 3.3 Giao diện đăng ký tài khoản 14

Hình 3.4 Giao diện thực hiện chức năng 14

Hình 3.5 Giao diện soạn thư 15

Hình 3.6 Giao diện hộp thư đến 15

Hình 3.7 Giao diện thư đã gửi 16

Hình 3.8 Giao diện thư rác 16

Hình 3.9 Giao diện thư đã xóa 17

Hình 3.10 Giao diện trang quản lý của server 17

# CHƯƠNG 1: TỔNG QUAN VÀ CƠ SỞ LÝ THUYẾT

* 1. **Tổng quan**

Báo cáo này tập trung vào đồ án môn học "Mô phỏng hệ thống Email bằng Java Socket." Đề tài này nhằm phát triển một ứng dụng Email hoàn chỉnh, chạy trên nền tảng Java Socket. Đây là một hệ thống phân tán với phía Client và Server. Mục tiêu chính của dự án là xây dựng một ứng dụng Email đáp ứng nhu cầu của người dùng và đảm bảo tính bảo mật thông tin.

* 1. **Đồ án đã thực thi được những yêu cầu**
     1. ***Yêu cầu về chức năng phía client (phải có GUI):***
* Chức năng đăng ký/đăng nhập. Thông tin tối thiểu cần cung cấp: username (sẽ chính là địa chỉ email được cấp); họ tên; password.
* Xây dựng các chức năng gửi nhận thư tương tự một mail client đơn giản:
* Send (bao gồm cả CC và BCC) tới 1 hoặc nhiều địa chỉ email: nội dung email có thể nhúng hình ảnh, file đính kèm.
* Receive
* Reply
* Mark spam email
* Delete
* Mô tả cách phân loại email vào các nhóm: Inbox (dành cho email mới đến và chưa được đọc); Read (email đã được đọc); Sent (dành cho email đã gửi đi); Spam (dành cho email bị đánh dấu spam); Bin (dành cho email bị xóa).
  + 1. ***Yêu cầu về chức năng phía server(không cần GUI):***
* Server ghi log về quá trình login/logout của client.
* Server có thể đặt và thay đổi dung lượng hộp thư của user. Mặc định là 100 MB /user.
* Server có thể khóa hoặc gửi email thông báo tới tất cả user.
  + 1. ***Yêu cầu chung:***
* Mã hóa nội dung tin nhắn giữa client – server. Phải sử dụng key khác nhau cho các client.
* Các client phải chạy trên các máy tính khác nhau.
  1. **Xu hướng công nghệ**
     1. ***Khái niệm ngôn ngữ lập trình Java***

Java là một ngôn ngữ lập trình phổ biến và mạnh mẽ được phát triển bởi Sun Microsystems (hiện nay thuộc sở hữu của Oracle Corporation). Nó ra đời vào năm 1995 và đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới.

Java được thiết kế để làm việc trên nền tảng đa nền tảng, có nghĩa là mã nguồn Java có thể được chạy trên nhiều hệ điều hành khác nhau như Windows, macOS, Linux và các thiết bị di động như điện thoại thông minh và máy tính bảng.



**Hình 1.1: Logo ngôn ngữ lập trình Java**

* **Những ưu điểm của Java**

**Đa nền tảng:** Java được thiết kế để làm việc trên nhiều hệ điều hành khác nhau, từ máy tính để bàn đến thiết bị di động. Mã nguồn Java được biên dịch thành bytecode, có thể chạy trên môi trường Java Virtual Machine (JVM). Điều này cho phép viết một lần và chạy ở nhiều nền tảng khác nhau mà không cần thay đổi mã nguồn.

**Hướng đối tượng:** Java là một ngôn ngữ hướng đối tượng, với khả năng kế thừa, đa hình, và đóng gói. Điều này giúp tăng tính sáng tạo, khả năng tái sử dụng mã và dễ dàng bảo trì.

**Bảo mật:** Java có mô hình bảo mật mạnh mẽ. Mã bytecode Java chạy trên JVM được kiểm tra an toàn trước khi thực thi. Java cung cấp một loạt các cơ chế bảo mật như quản lý bộ nhớ, kiểm soát quyền truy cập và xử lý ngoại lệ, giúp ngăn chặn các lỗ hổng bảo mật và tấn công từ phía người dùng.

**Thư viện phong phú:** Java đi kèm với một loạt các thư viện chuẩn (Standard Library) và thư viện mở rộng (Third-party Library) phong phú. Những thư viện này cung cấp các công cụ và chức năng sẵn có để giảm thiểu công sức phát triển và tăng hiệu suất lập trình. Ví dụ: thư viện Java Collections, JDBC (Java Database Connectivity), Swing, Spring Framework, Hibernate, và nhiều thư viện khác.

**Hiệu suất cao:** Mã nguồn Java được biên dịch thành bytecode trước khi thực thi, điều này giúp cải thiện hiệu suất so với các ngôn ngữ dịch ngay lập tức (interpreted languages). Ngoài ra, JVM có khả năng tối ưu hóa mã nguồn và quản lý bộ nhớ hiệu quả, tạo ra hiệu suất cao và ổn định.

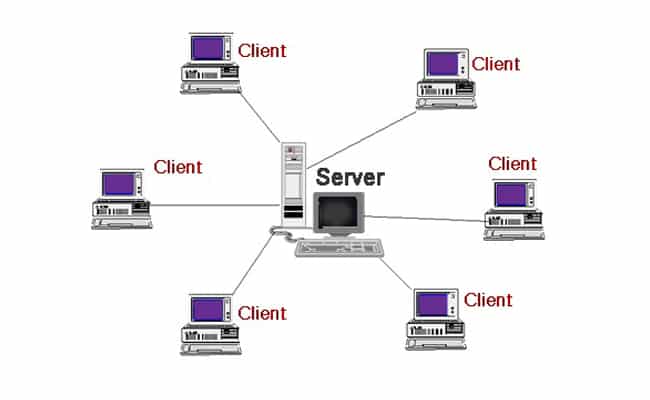
**Cộng đồng lớn:** Java có một cộng đồng phát triển lớn và tích cực. Cộng đồng này cung cấp các tài liệu, tài liệu học tập, hỗ trợ và công cụ phát triển. Điều này giúp người dùng Java dễ dàng tìm kiếm giải pháp cho các vấn đề kỹ thuật và tiếp cận kiến thức mới nhất.

**Dễ học và sử dụng:** Java có cú pháp tương tự như ngôn ngữ C++, nhưng loại bỏ đi một số tính năng phức tạp. Điều này làm cho Java dễ học và sử dụng cho người mới bắt đầu. Ngoài ra, Java có một công cụ phát triển tích hợp (IDE) mạnh mẽ, giúp tăng năng suất và dễ dàng phát triển ứng dụng Java.

.

* + 1. ***Tìm hiểu mô hình client-server***

Mô hình client-server là một kiến trúc phân tán trong phát triển phần mềm, trong đó các máy tính hoặc thiết bị được chia thành hai vai trò chính: máy khách (client) và máy chủ (server). Mô hình này cho phép phân chia công việc và trách nhiệm giữa các thành phần trong hệ thống



**Hình 1.2: Mô hình client-server**

* **Đặc điểm nổi bật**

**Phân chia công việc:** Mô hình client-server cho phép phân chia công việc giữa máy khách và máy chủ. Máy khách chịu trách nhiệm gửi yêu cầu, xử lý giao diện người dùng và hiển thị kết quả, trong khi máy chủ xử lý logic nghiệp vụ và cung cấp dữ liệu hoặc dịch vụ.

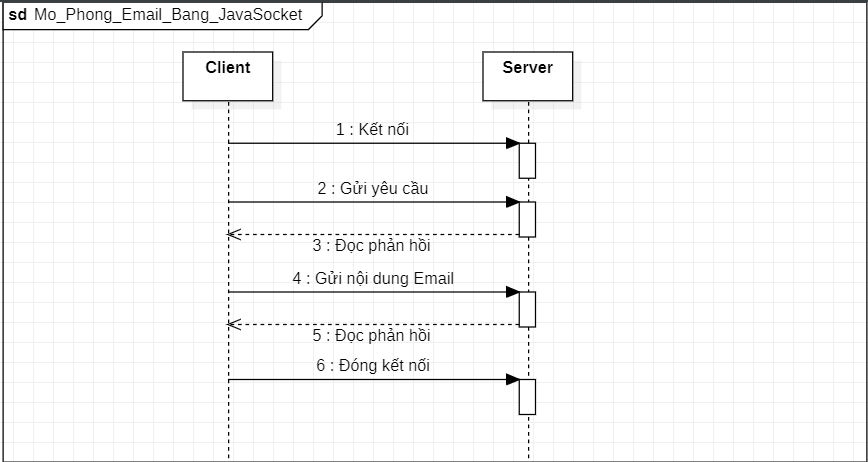
**Mở rộng linh hoạt:** Mô hình client-server cho phép mở rộng hệ thống bằng cách thêm máy khách hoặc máy chủ mới. Khi lưu lượng tăng, có thể thêm nhiều máy chủ để phân chia công việc và tăng khả năng chịu tải của hệ thống.

**Tính nhất quán và bảo mật:** Máy chủ có thể kiểm soát quyền truy cập và áp dụng các biện pháp bảo mật để bảo vệ dữ liệu và nguồn tài nguyên. Điều này giúp đảm bảo tính nhất quán và an toàn trong quá trình trao đổi thông tin.

**Tính tương thích:** Mô hình client-server cho phép các máy khách khác nhau chạy trên các nền tảng và hệ điều hành khác nhau, miễn là chúng tuân thủ các giao thức và quy tắc được xác định.

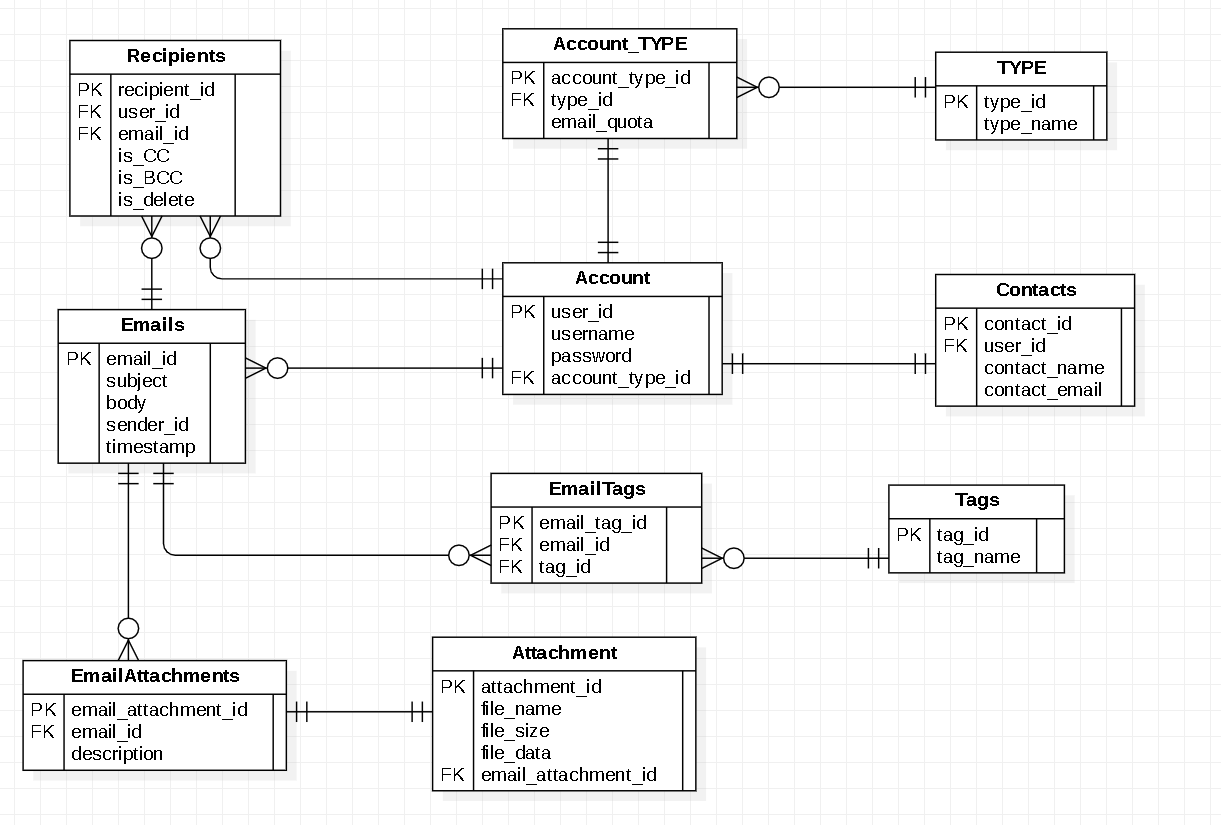
# CHƯƠNG 2: MÔ HÌNH ĐỀ XUẤT

* 1. **Sơ đồ tổng quát mô hình**



**Hình 2.1: Sơ đồ tổng quát mô hình**

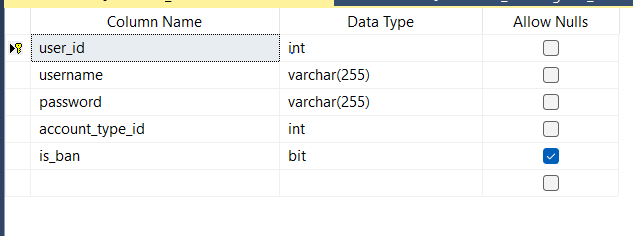
* 1. **Mô hình ERD**



**Hình 2.2: Mô hình ERD**

* 1. **Thiết kế các bảng**
     1. ***Bảng lưu thông tin tài khoản***

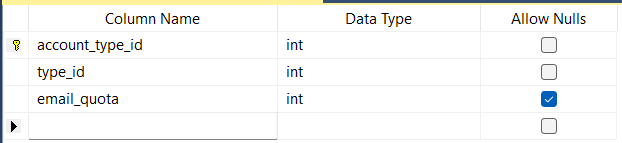
Sử dụng user\_id để làm khoá



**Hình 2.3: Bảng lưu thông tin tài khoản**

* + 1. ***Bảng lưu thông*** ***tin dung lượng hộp thư***

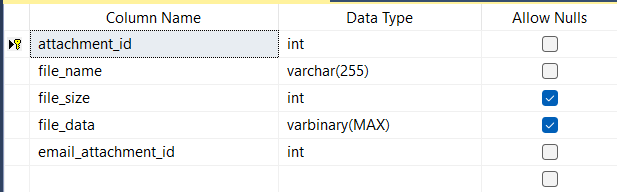
Chọn kiểu tài khoản được cấp dung lượng hộp thư cho người dùng



**Hình 2.4: Bảng lưu thông tin** **dung lượng hộp thư**

* + 1. ***Bảng lưu thông*** ***tin tệp đính kèm***

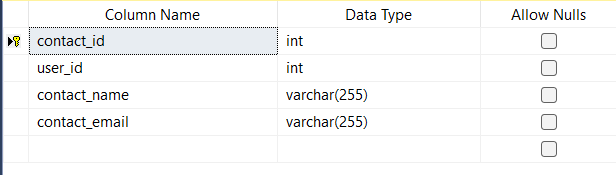
Lưu trữ thông tin file đính kèm được chọn



**Hình 2.5: Bảng lưu thông tin tệp đính kèm**

* + 1. ***Bảng lưu thông tin chi tiết người dùng***

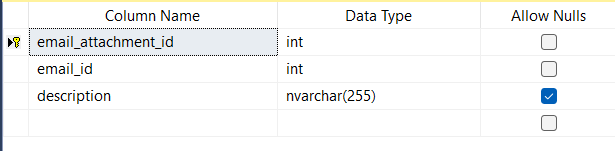
Lưu thông tin chi tiết của người dùng



**Hình 2.6: Bảng lưu thông tin chi tiết người dùng**

* + 1. ***Bảng lưu thông tin chi*** ***tiết tệp đính kèm***

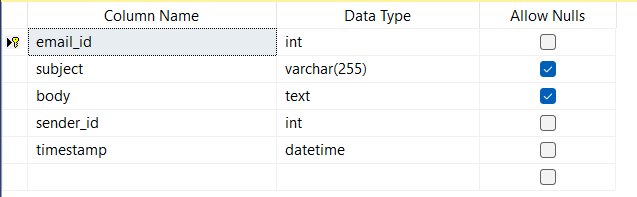
Lưu trữ thông tin chi tiết tệp đính kèm, dùng để nối thông tin giữa thư gửi và nơi lưu trữ tệp đính kèm



**Hình 2.7: Bảng lưu thông tin chi tiết tệp đính kèm**

* + 1. ***Bảng lưu thông tin*** ***thư***

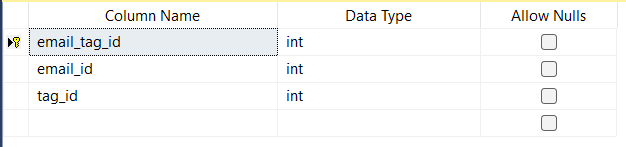
Lưu thông tin thư



**Hình 2.8: Bảng lưu thông tin thư**

* + 1. ***Bảng lưu thông tin định dạng bức thư***

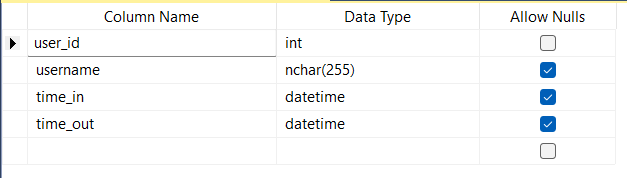
Ứng với mỗi email\_tag\_id sẽ có mỗi kiểu định dạng bức thư là đã đọc, thư gửi, thư rác, thư bị xóa



**Hình 2.9: Bảng lưu thông tin định dạng bức thư**

* + 1. ***Bảng lưu thông tin thời gian đăng nhập đăng xuất của người dùng***

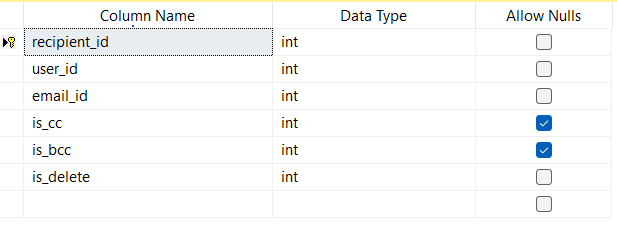
Lưu thông tin thời gian đăng nhập và đăng xuất của mỗi tài khoản



**Hình 2.10: Bảng lưu thông tin thời gian đăng nhập đăng xuất của người dùng**

* + 1. ***Bảng lưu thông tin người nhận thư***

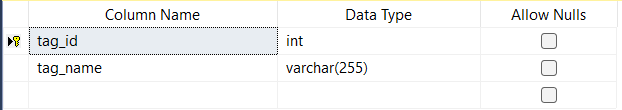
Ứng với mỗi bức thư có thể có nhiều người nhận thư



**Hình 2.11: Bảng lưu thông tin người nhận thư**

* + 1. ***Bảng lưu thông tin chi tiết định dạng bức thư***

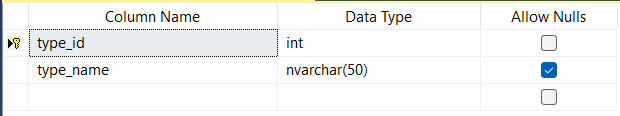
Lưu trữ chỉ định thông tin định dạng



**Hình 2.12: Bảng lưu thông tin chi tiết định dạng bức thư**

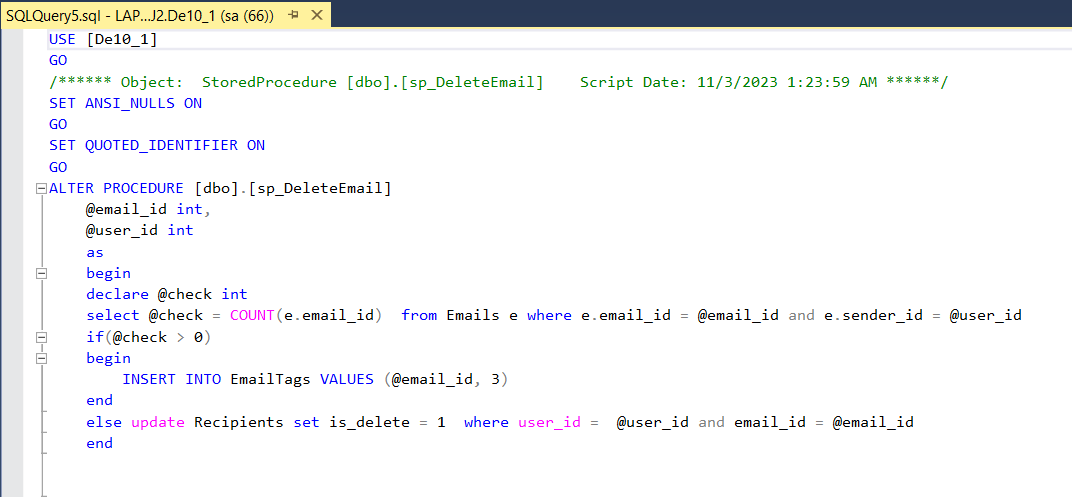
* + 1. ***Bảng lưu thông tin chi tiết phân loại thư***

Ứng với mỗi tài khoản sẽ có cách phân loại thư thuộc loại nào để nhận được dung lượng tương ứng



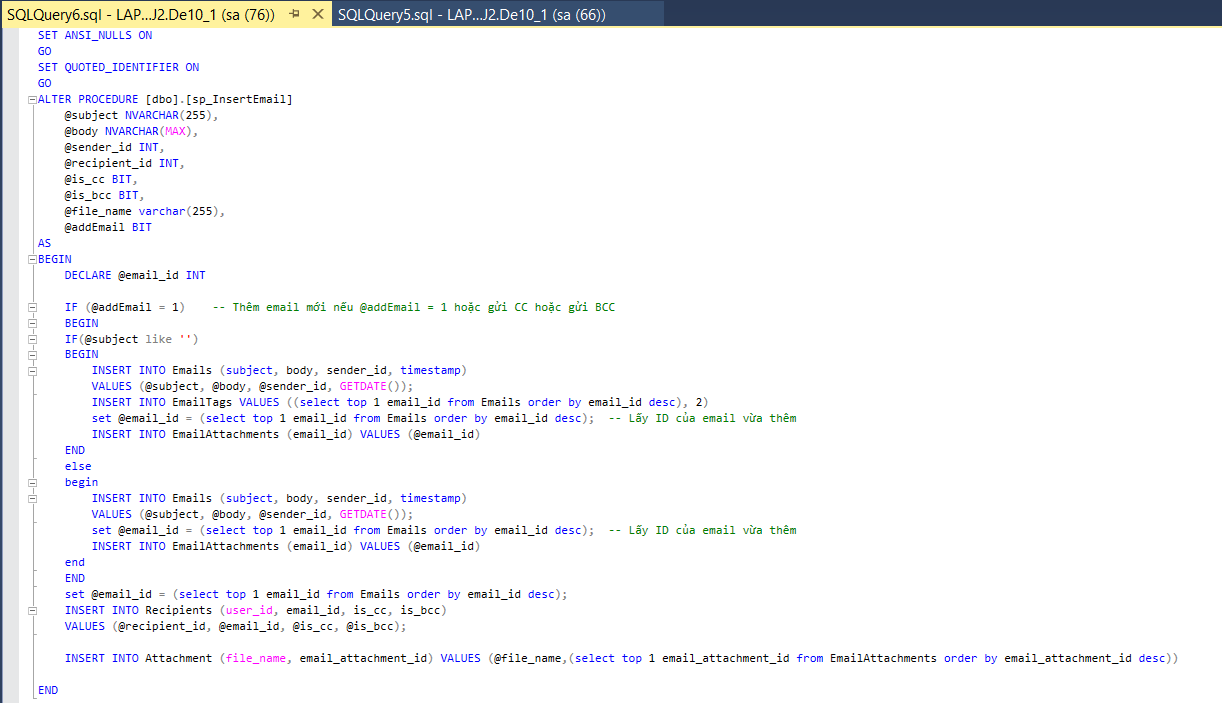
**Hình 2.13: Bảng lưu thông tin chi tiết phân loại thư**

* + 1. ***Procedure xóa thư***



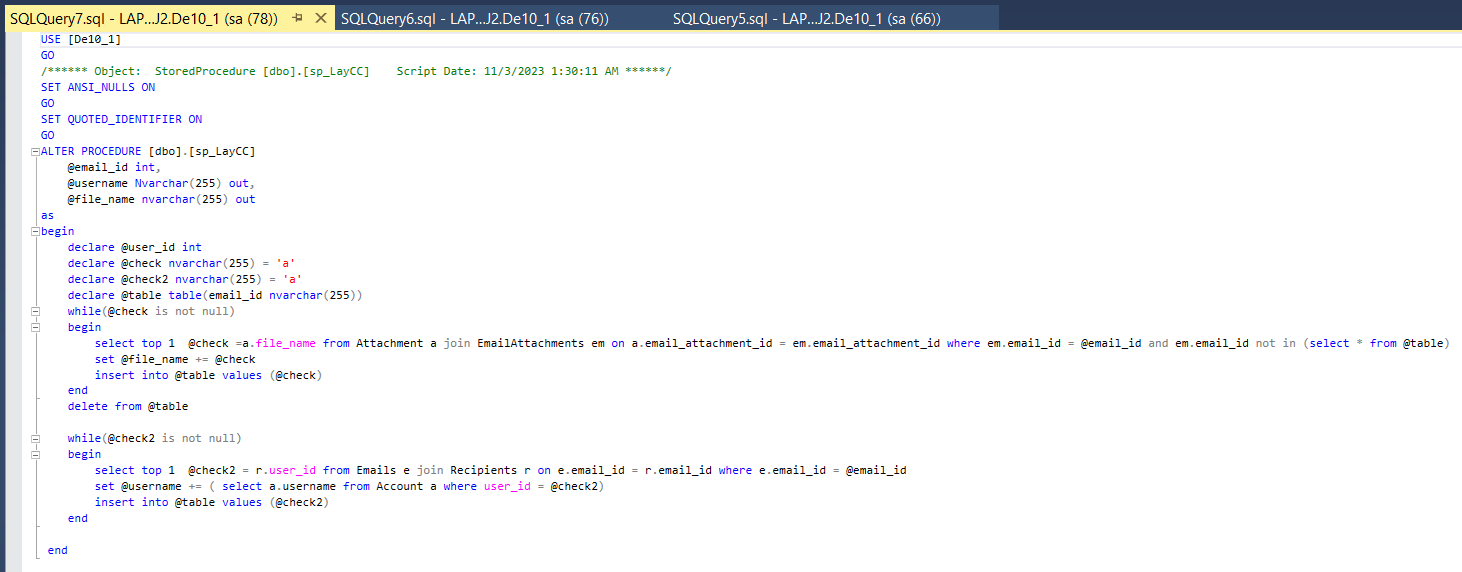
**Hình 2.14: Procedure xóa thư**

* + 1. ***Procedure thêm thư mới***



**Hình 2.15: Procedure thêm thư mới**

* + 1. ***Procedure gửi thư về cho client***



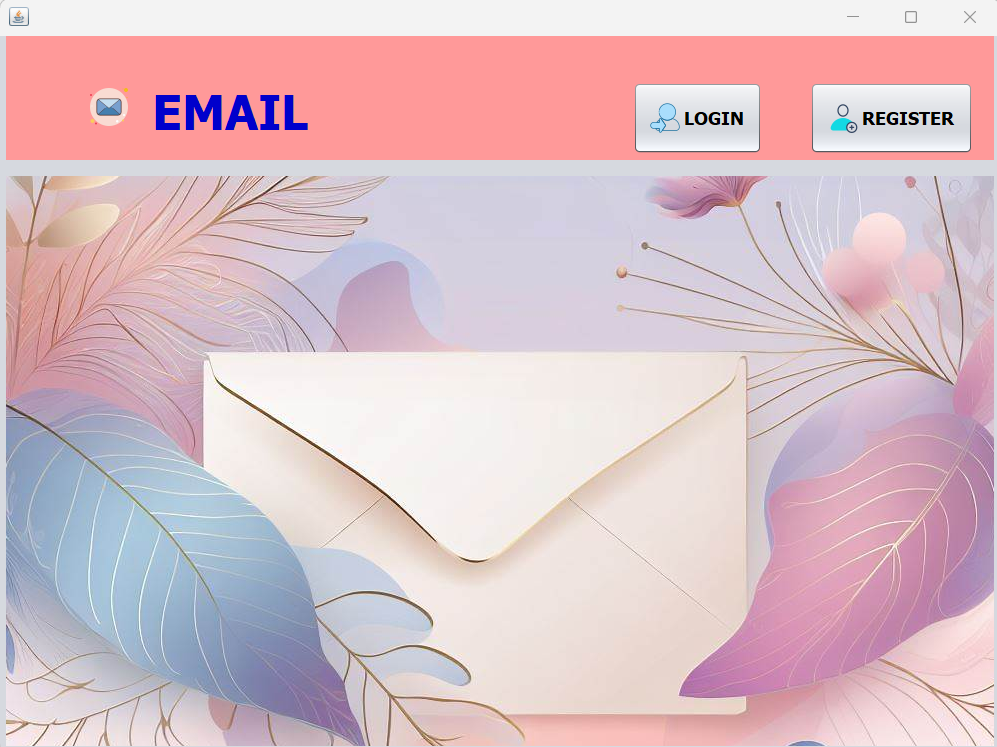
**Hình 2.16: Procedure gửi thư về cho client**

# CHƯƠNG 3: TRIỂN KHAI ỨNG DỤNG

## Ứng dụng phía Clients

### 3.1.1 Giao diện trang chủ

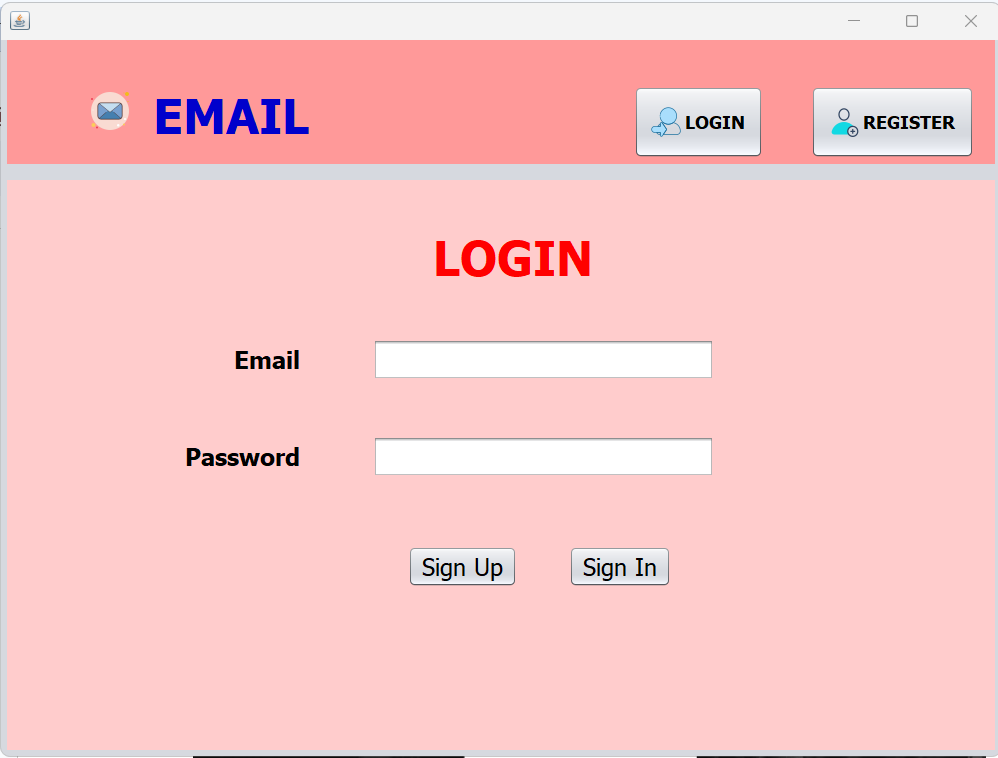
Hiển thị giao diện trang chủ, đồng thời hiện thị nút Đăng nhập (Login) và Đăng ký (Register) để người dùng tương tác



**Hình 3.1 Giao diện trang chủ**

### 3.1.2 Giao diện đăng nhập

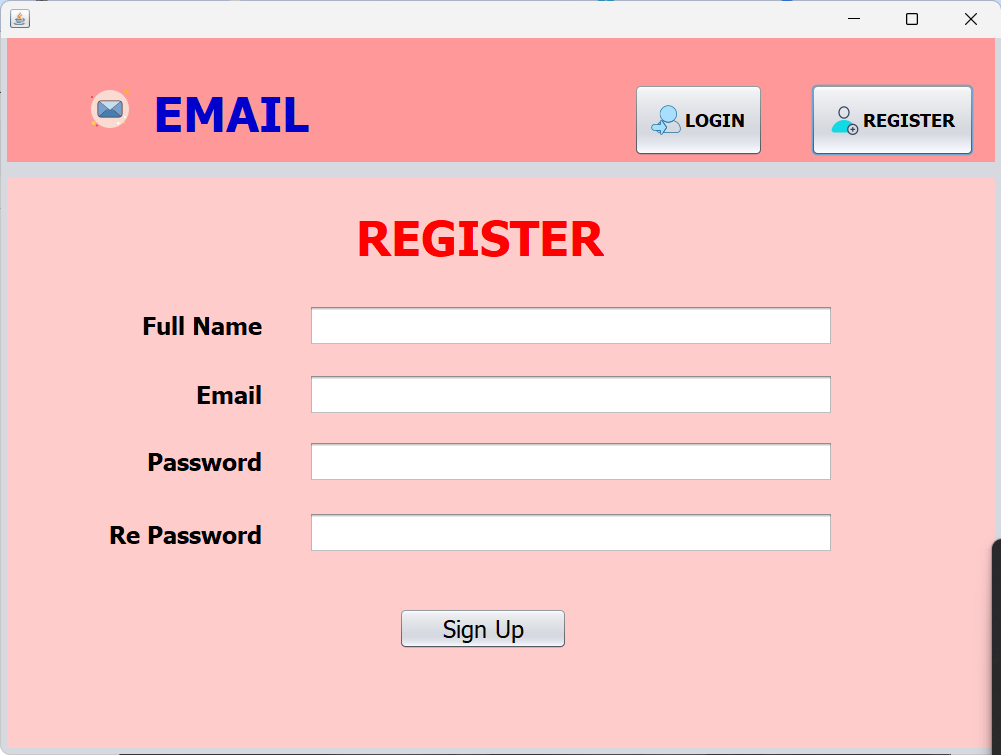
Dùng để đăng nhập vào tài khoản để mở giao diện các chức năng



**Hình 3.2 Giao diện trang đăng nhập**

### 3.1.3 Giao diện đăng ký tài khoản

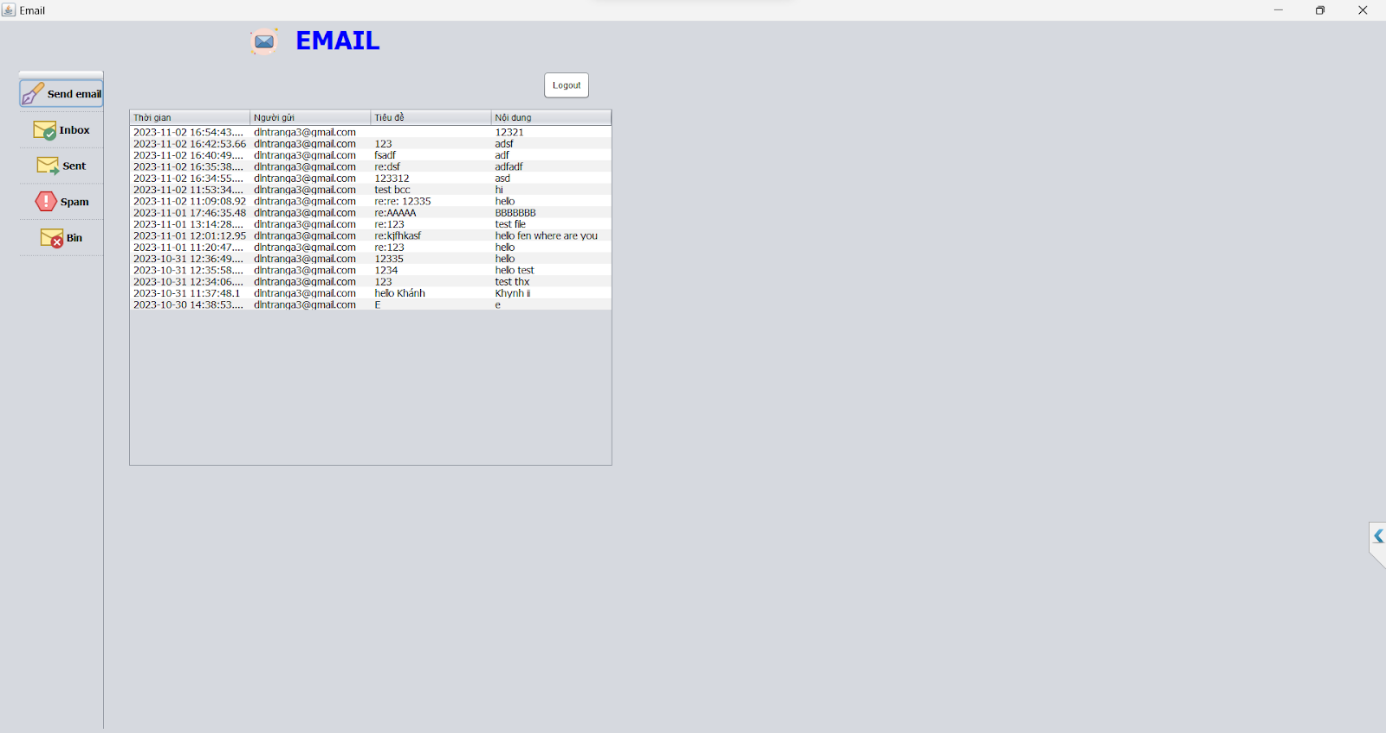
Dùng để tạo tài khoản mới



**Hình 3.3 Giao diện đăng ký tài khoản**

### 3.1.4 Giao diện thực hiện chức năng

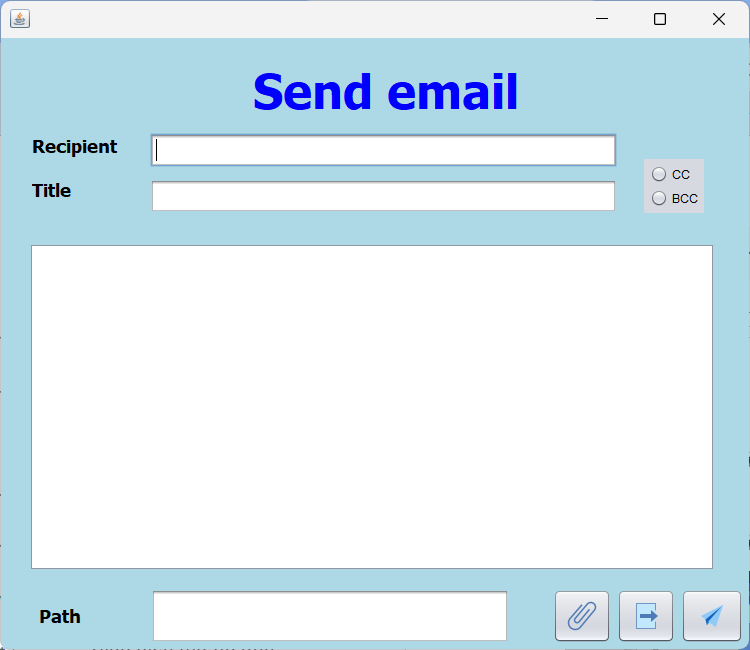
Giao diện hiển thị các chức năng cho người dùng có thể thực hiện



**Hình 3.4 Giao diện thực hiện chức năng**

### 3.1.5 Giao diện soạn thư

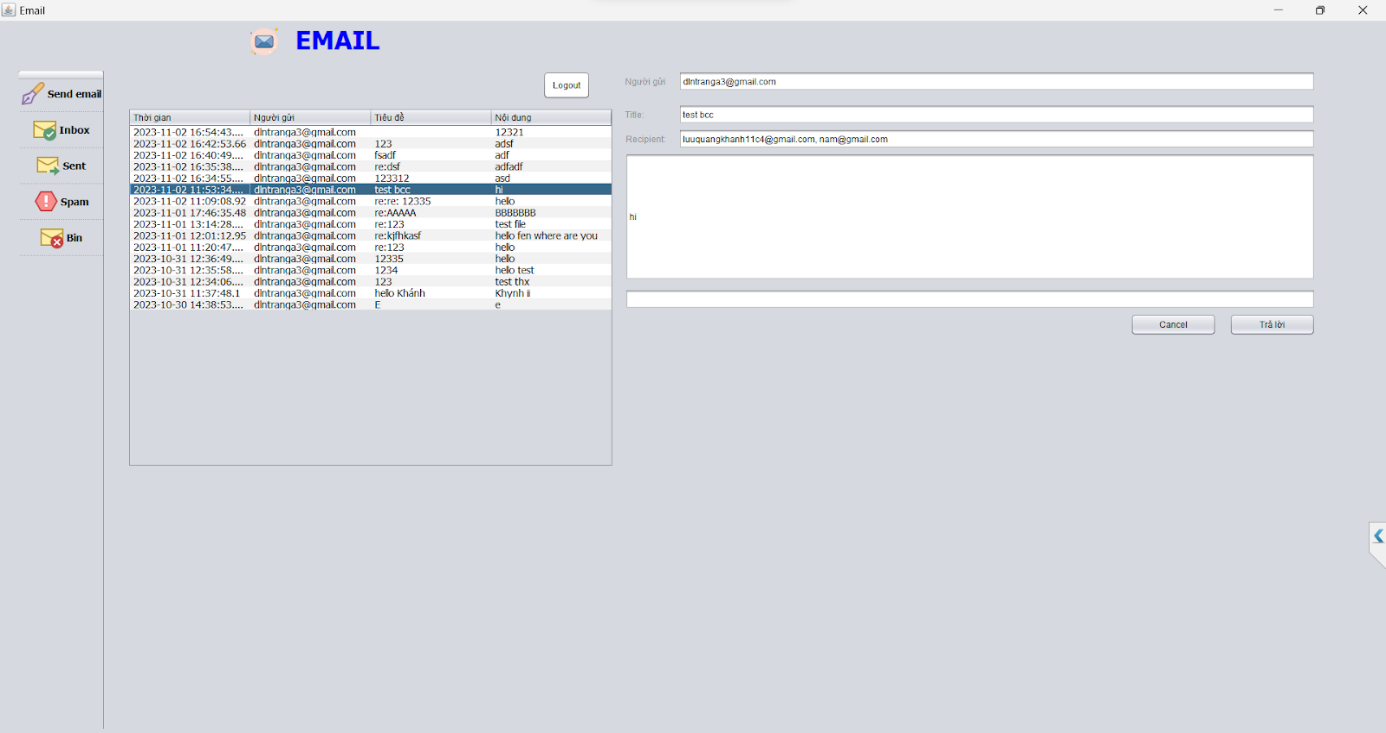
Giao diện soạn thư mới, đồng thời cũng là giao diện để trả lời thư



**Hình 3.5 Giao diện soạn thư**

***3.1.6 Giao diện hộp thư đến***

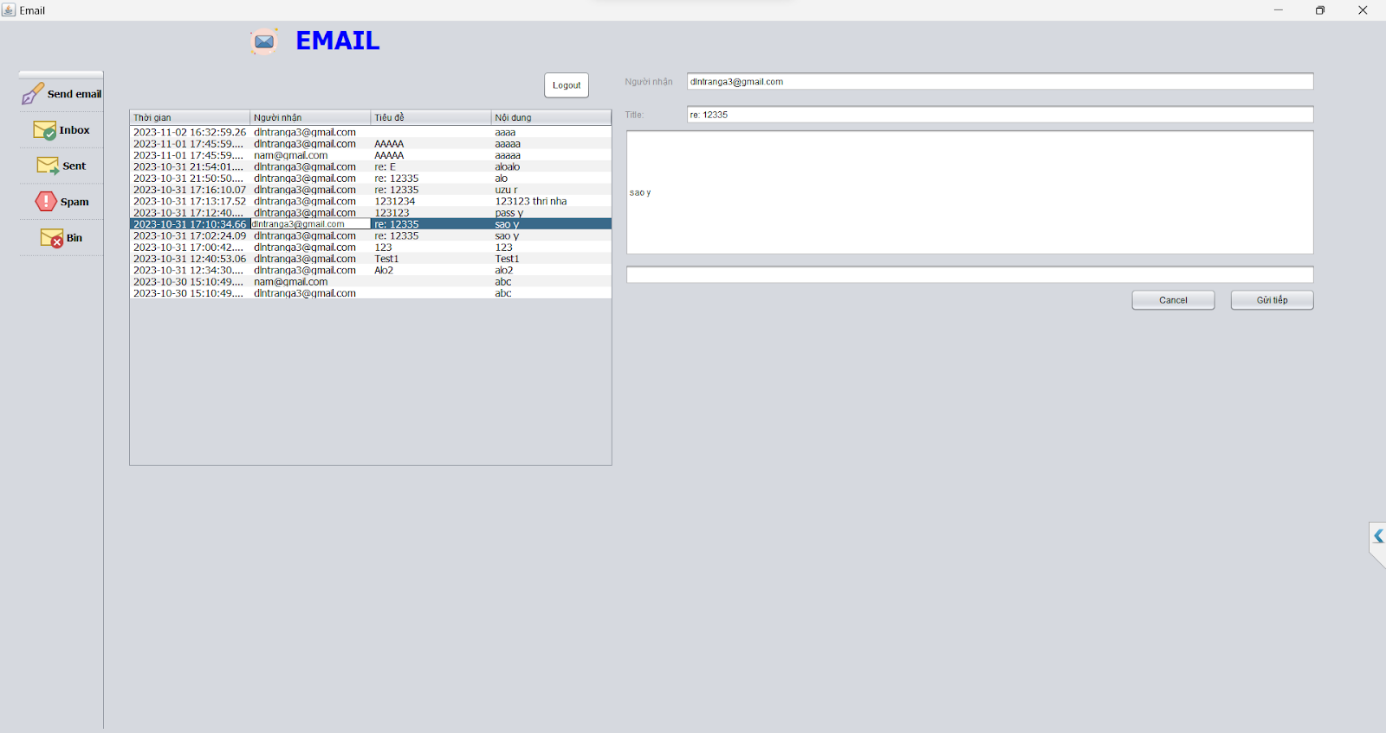
Giao diện thực hiện chức năng hiển thị hộp thư nhận được từ các tài khoản khác

******

**Hình 3.6 Giao diện hộp thư đến**

### 3.1.7 Giao diện thư đã gửi

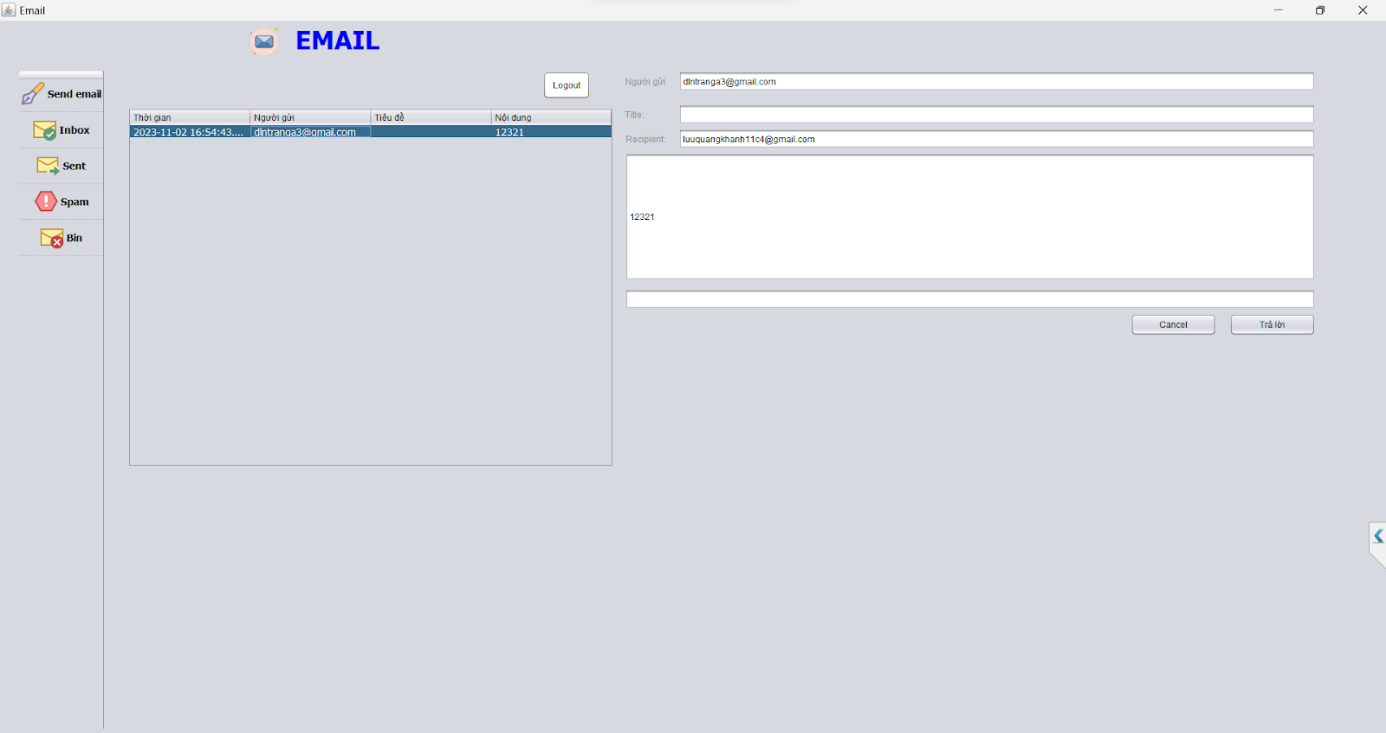
Giao diện hiển thị các thư đã gửi cho những tài khoản email khác



**Hình 3.7 Giao diện thư đã gửi**

### 3.1.8 Giao diện thư rác

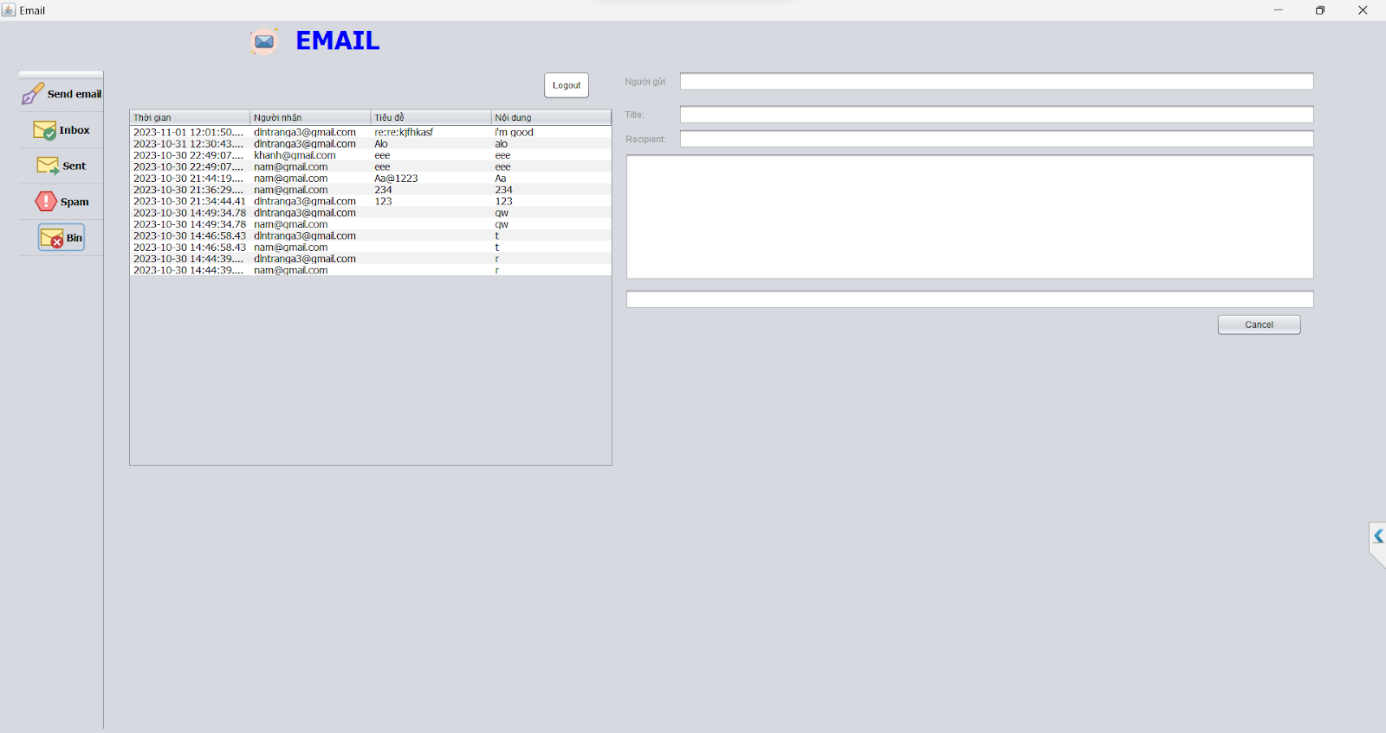
Giao diện chứa hộp thư rác, thư không có tiêu đề



**Hình 3.8 Giao diện thư rác**

### 3.1.9 Giao diện thư đã xóa

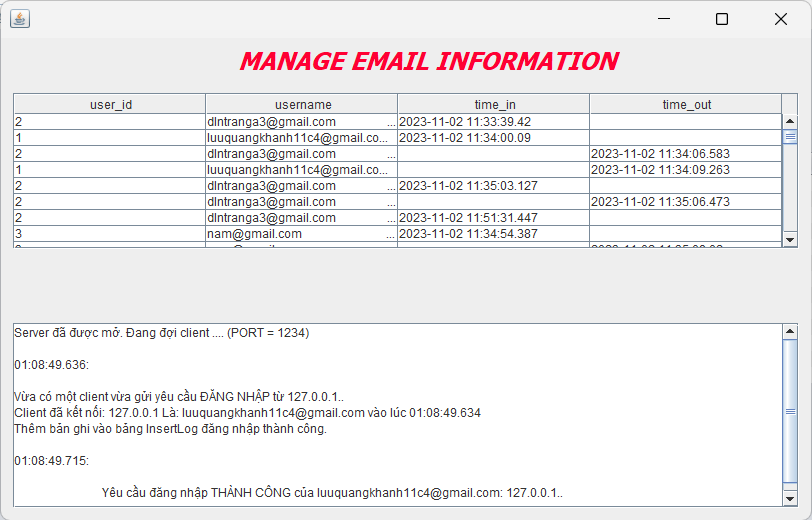
Giao diện chứa hộp thư đã xóa bởi người dùng và tự động xóa sau 30 ngày



**Hình 3.9 Giao diện thư đã xóa**

## 3.2 Ứng dụng phía Server

Giao diện hiển thị dành cho server dùng để quản lý thời gian đăng nhập, đăng xuất và các tác vụ mà người dùng đã thực hiện và có thể có quyền khóa tài khoản của người dùng



**Hình 3.10  Giao diện trang quản lý của server**

# CHƯƠNG 4: KẾT LUẬN, HƯỚNG PHÁT TRIỂN

## 4.1 Kết quả đã thực hiện

* Xây dựng được một phần mềm mô phỏng Email bằng java socket đáp ứng cơ bản quy trình nhận gửi email.
* Dữ liệu đưa vào được kiểm tra và chuẩn hóa đảm bảo sự đúng đắn và chặt chẽ
* Cung cấp đầy đủ, chính xác các thông tin về email: email người gửi/người nhận, nội dung, tiêu đề, tệp đính kèm,...
* Hệ thống đảm bảo bảo mật cho người sử dụng thông qua mã hóa, đảm bảo người dùng thực hiện đúng phạm vi chức năng

## 4.2 Nhược điểm

Phần mềm còn rất nhiều hạn chế do quá trình làm chúng em gặp nhiều khó khăn trong việc tìm kiếm và chuẩn bị về mặt tài liệu và kiến thức

Trong quá trình xây dựng và hoàn thiện chúng em cảm thấy chương trình có nhiều thiếu sót như:

* Thiếu nhiều chức năng như: replyAll, ScheduleSend,...
* Bảo mật thông tin còn kém, chưa đảm bảo mã hóa toàn vẹn thông tin
* Phương pháp thực hiện còn nhiều sai sót

## 4.3 Hướng phát triển toàn diện

* Bổ sung thêm và kiểm tra thực thi các chức năng cho phần mềm được đảm bảo và  hoàn thiện hơn.
* Tối ưu hóa bảo mật thông tin.

1. **TÀI LIỆU THAM KHẢO**

# References

|  |  |
| --- | --- |
| [1] | T. V. T. Hoàng, Lập Trình Mạng, Đại học Công nghệ Tp.HCM. |
| [2] | N. T. Hải, Mạng máy tính và các hệ thống mở, Nhà xuất bản Giáo dục. |
| [3] | Nguyễn Phương Lan, Hoàng Đức Hải, Programming the Internet with Java, Addison-Wesley. |
| [4] | Đ. V. Ban, Lập trình hướng đối tượng với Java, Nhà xuất bản Khoa học và Kỹ thuật. |
| [5] | "CodeLearn," [Online]. Available: https://codelearn.io/. |
| [6] | "VIBLO," [Online]. Available: https://viblo.asia. |
| [7] | TS. Văn Thiên Hoàng, KS. Nguyễn Trọng Minh Hoàng Phước, Thực hành lập trình mạng, Đại học Công nghệ Tp.HCM. |
| [8] | "Tạo và quản lý luồng trong Java - Freetuts," [Online]. Available: https://freetuts.net/tao-va-quan-ly-luong-trong-java-1163.html. |
| [9] | "Tutorialspoint," [Online]. Available: https://www.tutorialspoint.com/. [Accessed 3 10 2023]. |
| [10] | "stackoverflow," [Online]. Available: https://stackoverflow.com/. [Accessed 19 10 2023]. |
| [11] | "VietTuts," [Online]. Available: https://viettuts.vn/. [Accessed 3 10 2023]. |
| [12] | "vietJack.com," [Online]. Available: https://vietjack.com/. [Accessed 4 10 2023]. |