

Đại học Quốc gia Thành phố Hồ Chí Minh

Trường Đại học Khoa học Tự nhiên



BÁO CÁO ĐỒ ÁN ĐIỀU KHIỂN MÁY TÍNH BẰNG GMAIL

Nhóm sinh viên thực hiện:

20120019 - Nguyễn Đức Mạnh

20120250 - Trần Bảo Anh

20120304 - Phan Trần Khanh

Giảng viên hướng dẫn: ThS. Đỗ Hoàng Cường

Học phần Mạng máy tính

Lớp 20TN

06 - 2022

Mục Lục

1	Giới thiệu	2
2	Bảng phân công	2
3	Xử lý tác vụ trên máy điều khiển	3
3.1	Các thư viện hỗ trợ	3
3.2	Luồng hoạt động chính của chương trình	4
3.3	Các hàm và chức năng chính	5
3.3.1	send_email.py	5
3.3.2	base_test.py	5
3.3.3	gmail.py	6
4	Phần mềm nhận email yêu cầu và điều khiển máy tính	8
4.1	Các thư viện hỗ trợ	8
4.2	Luồng hoạt động chính của chương trình	9
4.3	Các hàm chức năng chính	10
5	Hướng dẫn sử dụng	16
5.1	Chương trình nhận email và điều khiển máy tính	16
5.2	Chương trình gửi email yêu cầu	19
6	Cấu trúc Email	23
7	Đánh giá mức độ hoàn thành	24
	Tài liệu tham khảo	25

1. Giới thiệu

Đồ án xây dựng phần mềm sử dụng email hỗ trợ điều khiển quản trị máy tính từ xa được thực hiện nhằm mục đích giúp người dùng có thể tự do điều khiển máy tính với một số tính năng nhất định khi đang ở khoảng cách xa thông qua việc gửi email yêu cầu đến email máy tính và nhận về kết quả trong hộp thư. Đồ án này được nhóm em thực hiện bằng việc sử dụng một dịch vụ gửi nhận email đến từ Google là Gmail.

Vì Google đã thông báo sẽ ngưng hỗ trợ tính năng "Less Secure Apps" từ ngày 30 tháng 05 năm 2022, đồng nghĩa với việc nhóm em không thể sử dụng giao thức SMTP truy cập hòm thư Gmail. Vậy nên nhóm em buộc phải dùng Gmail API.

Nhóm em quyết định cài đặt thêm giao diện phần mềm để người dùng không cần phải soạn email một cách chi tiết. Tuy vậy, chương trình vẫn sử dụng dịch vụ Gmail API [1] để thực hiện việc gửi và nhận email và người dùng sẽ được điều khiển từng máy tính khác nhau ứng với email của riêng máy tính đó.

Trong báo cáo này, nhóm em cũng sẽ cung cấp cấu trúc của email, như vậy nếu người dùng sử dụng nền tảng web gmail của Google trên tất cả các thiết bị có thể truy cập được, cũng sẽ điều khiển được máy tính từ xa khi gửi email đi. Nếu người dùng không tuân thủ cấu trúc email thì máy bị điều khiển sẽ không thể xử lý tác vụ theo yêu cầu và mặc định đó không phải là email yêu cầu thực hiện tác vụ.

2. Bảng phân công

20120019	Nguyễn Đức Mạnh	Lập trình giao diện cho phần mềm điều khiển
20120250	Trần Bảo Anh	Lập trình xử lý trên máy tính bị điều khiển và giao diện đơn giản Viết báo cáo Quay video chạy thử phần mềm
20120304	Phan Trần Khanh	Chuẩn bị mẫu Latex Lập trình xử lý trên máy tính điều khiển Viết báo cáo Thiết kế giao diện phần mềm điều khiển Lồng tiếng thuyết trình

3. Xử lý tác vụ trên máy điều khiển

3.1. Các thư viện hỗ trợ

- **google-api-python-client, google-auth-httpplib2, google-auth-oauthlib, oauth2client:**
Đây là bốn thư viện cần thiết thuộc Google client library để chương trình có thể gửi yêu cầu đến Gmail API nhằm sử dụng dịch vụ hỗ trợ gửi và nhận thư từ điện tử.[\[2\]](#)
- **base64:** Thư viện hỗ trợ mã hóa dữ liệu, phục vụ cho việc gửi thư đi và giải mã dữ liệu khi nhận thư điện tử.
- **os:** Thư viện hỗ trợ mở tập tin đính kèm và các tác vụ liên quan đến đường dẫn tập tin đính kèm.

3.2. Luồng hoạt động chính của chương trình

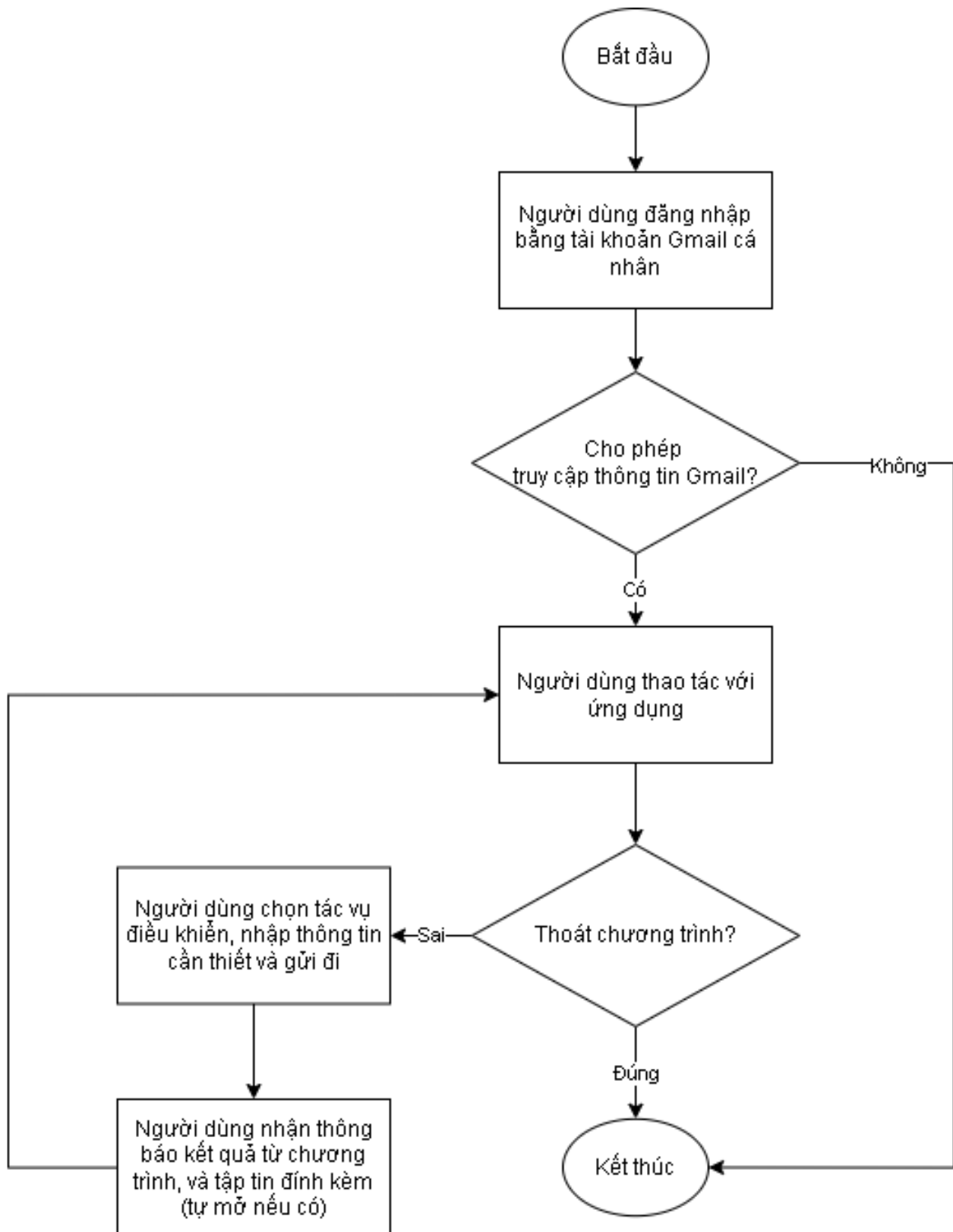


Figure 1: Luồng hoạt động chính của phần mềm điều khiển.

3.3. Các hàm và chức năng chính

3.3.1. `send_email.py`

`create_message:`

Đây là hàm tạo thư điện tử có nội dung là văn bản ký tự đơn giản.

Hàm có các tham số như sau:

- *sender*: Địa chỉ Email người gửi.
- *to*: Địa chỉ Email người nhận.
- *subject*: Tiêu đề của thư điện tử.
- *message_text*: Nội dung văn bản của thư.

Hàm trả về một đối tượng email được mã hóa bằng base64url.

`send_message:`

Đây là hàm gửi thư điện tử, có sử dụng try-except để bắt lỗi.

Hàm có các tham số như sau:

- *service*: Dịch vụ Gmail API đã được người dùng cấp phép.
- *user_id*: Địa chỉ Email của người dùng, ở đây nhóm sử dụng địa chỉ email đã cấp phép cho chương trình thực thi.
- *message*: Đối tượng Email cần được gửi đi (tạo từ hàm `create_message`).

Hàm trả về lỗi liên quan đến giao thức HTTP nếu có, nếu thành công sẽ trả về id của thư vừa gửi đi.

3.3.2. `base_test.py`

Tập tin này chứa đối tượng BaseTest giúp cho chương trình truy cập Gmail API, gửi yêu cầu đến người dùng để truy cập Gmail và thực hiện các tác vụ được người dùng cấp phép.

`create_credentials:`

Đây là hàm bao gồm việc kiểm tra xác thực chương trình, gửi yêu cầu đến Gmail API xin cấp quyền hạn từ người dùng và kiểm tra quyền hạn sử dụng dịch vụ Gmail.

Những quyền hạn cần được cấp phép của chương trình nằm trong phạm vi sau:

1. Tạo, đọc, cập nhật và xóa bản nháp. Gửi tin nhắn và thư nháp.
2. Gửi tin nhắn. Không có quyền đọc hoặc sửa đổi trên hộp thư.
3. Tạo, đọc, cập nhật và xóa nhãn thư.

4. Quản lý cài đặt thư cơ bản.
5. Quản lý cài đặt thư nhạy cảm, bao gồm các quy tắc chuyển tiếp và bí danh.
6. Toàn quyền truy cập vào hộp thư của tài khoản cấp phép, bao gồm xóa vĩnh viễn các chuỗi thư và thư.

create_service:

Đây là hàm sau khi được cấp quyền hạn từ người dùng sẽ bắt đầu tạo "liên kết" để sử dụng dịch vụ Gmail từ Gmail API.

setUpClass:

Đây là hàm chính của lớp đối tượng BaseTest nhằm tạo dịch vụ Gmail từ Gmail API (create_service) và ghi nhận thông tin địa chỉ Gmail đã cấp quyền hạn phục vụ cho quá trình gửi thư điện tử.

3.3.3. gmail.py

Đây là tập tin chính để thao tác xử lý tác vụ ở máy điều khiển.

Tập tin có bao gồm lớp đối tượng Gmail và SECRET KEY (có giá trị mặc định là '019250304') để phân biệt thư điện tử yêu cầu xử lý tác vụ.

Lớp đối tượng Gmail kế thừa từ lớp đối tượng BaseTest.

setUpRecipient:

Đây là hàm cập nhật địa chỉ Email người nhận.

Hàm có tham số như sau:

- *email_address*: Địa chỉ Email người nhận.

receive_email:

Đây là hàm đọc email từ hòm thư của người dùng. Bao gồm việc tìm kiếm thư mới nhất chưa đọc theo tiêu đề thư, giải mã dữ liệu thư điện tử, tải về tập tin đính kèm và đánh dấu thư đã đọc.

Hàm có tham số như sau:

- *subject*: Tiêu đề thư cần tìm kiếm.

Hàm trả về theo hai trường hợp:

1. Thành công: Văn bản thô, địa chỉ tập tin đính kèm.
2. Không thành công: Lỗi tìm kiếm thư điện tử hoặc lỗi liên quan đến giao thức HTTP.

send_*:

Một loạt các hàm gửi thư điện tử theo từng tiêu đề Email ứng với mỗi thao tác yêu cầu xử lý ở máy bị điều khiển. Các hàm này sẽ nhận những tham số cần thiết cho việc xử lý và sử dụng try-except để trả về lỗi liên quan đến giao thức HTTP nếu có.

- shutdown: Thời điểm tắt nguồn máy.
- restart: Thời điểm khởi động lại nguồn.
- copy_file: Đường dẫn tập tin cũ và đường dẫn tập tin mới.
- capture_screen, capture_webcam, list_processes: Không có tham số.
- record_webcam: Độ dài thời gian ghi hình.
- kill_process: Mã ID của tiến trình.
- keypress: Độ dài thời gian bắt phím nhấn.
- registry_key: Đường dẫn registry key, giá trị, kiểu giá trị.

open_file:

Đây là hàm sử dụng lệnh Command Prompt để mở trực tiếp tập tin đính kèm.

Hàm có tham số như sau:

- *locations*: Một list đường dẫn tập tin đính kèm.

4. Phần mềm nhận email yêu cầu và điều khiển máy tính

4.1. Các thư viện hỗ trợ

Để cài đặt các chức năng cần thiết của chương trình, nhóm sử dụng các thư viện sau đây:

- **email, google-api-python-client, google-auth-httpplib2, google-auth-oauthlib, pickle, mimetypes**: gửi và nhận email thông qua dịch vụ Gmail API và các thao tác liên quan.
- **tkinter, tkhtmlview, markdown, pystray**: tạo giao diện đồ hoạ.
- **beautifulSoup4, base64**: giải mã nội dung email.
- **pandas**: tạo tập tin csv để liệt kê các tiến trình.
- **threading**: điều khiển đa luồng.
- **pynput**: bắt phím nhấn từ bàn phím.
- **open-cv**: chụp và quay video từ webcam.
- **shutil**: sao chép tập tin.
- **pyautogui**: chụp ảnh màn hình.
- **psutil**: lấy danh sách các tiến trình đang chạy.
- **re, winreg**: làm việc với registry và các thao tác liên quan.

4.2. Luồng hoạt động chính của chương trình

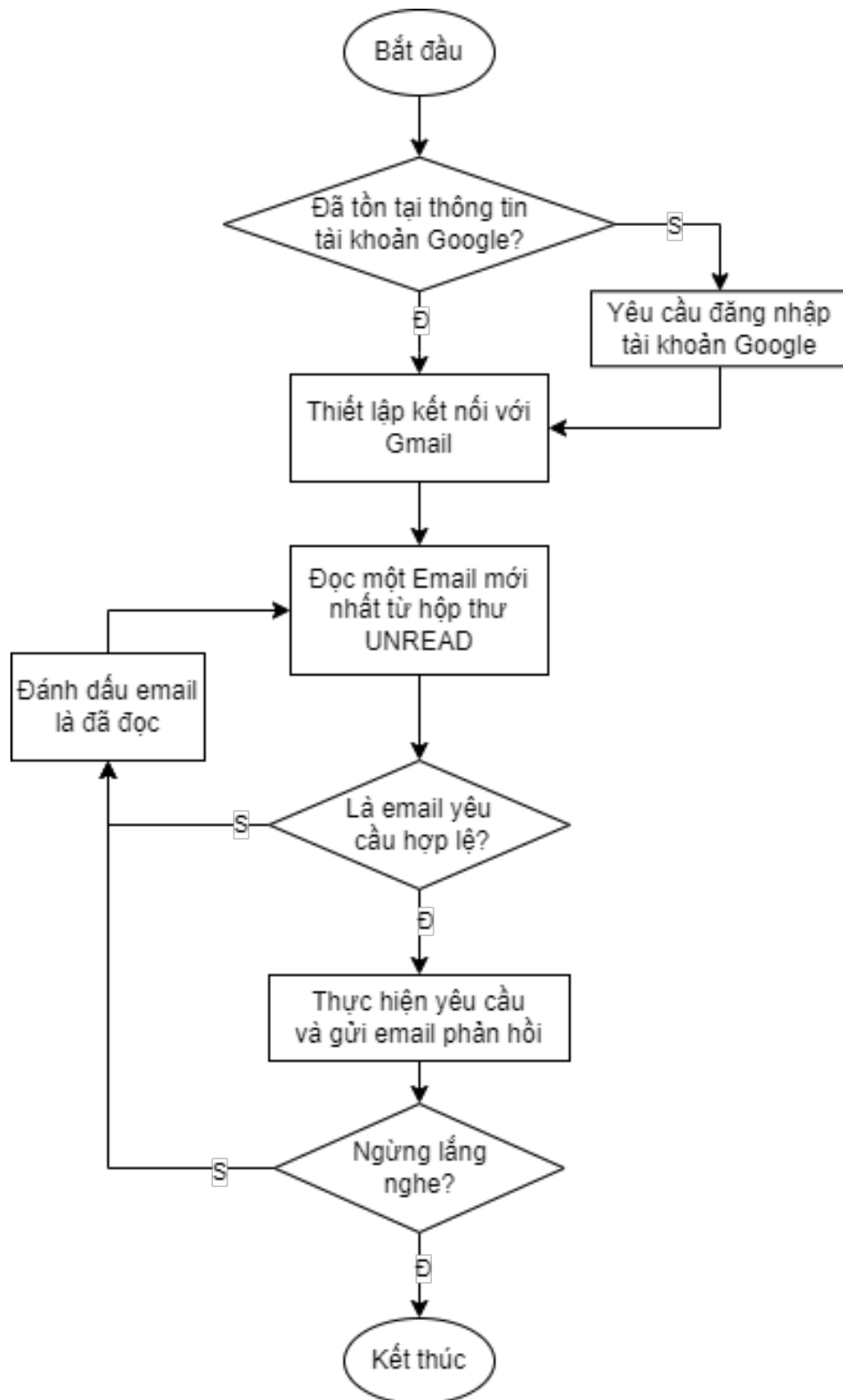


Figure 2: Luồng hoạt động chính của chương trình

4.3. Các hàm chức năng chính

Hàm `read_email()`

- Nhiệm vụ chính: đọc một email mới nhất từ hộp thư UNREAD và đánh dấu email là đã đọc, kiểm tra xem email có phải là email yêu cầu hợp lệ hay không. Một email là email yêu cầu hợp lệ nếu tiêu đề email nằm trong tập các tiêu đề xác định trước trong mục 6 và nội dung email chứa mã nhận diện đúng.
- Đầu vào: không có.
- Đầu ra:
 - Nếu email đọc được là một email yêu cầu hợp lệ, trả về đối tượng email, kiểu dictionary gồm 3 khoá: subject, sender, context.
 - Nếu email đọc được không phải là một email yêu cầu hợp lệ, hoặc xảy ra lỗi trong quá trình đọc email, trả về None.

Hàm `build_file_part(file)`

- Nhiệm vụ chính: tạo thành phần tập tin đính kèm cho nội dung email.
- Đầu vào:
 - file: kiểu string, là đường dẫn đến tập tin đính kèm.
- Đầu ra:
 - Nếu không có lỗi xảy ra, trả về một đối tượng kiểu MIMEText, MIMEImage, MIMEAudio hoặc MIMEBase tùy thuộc vào tập tin đính kèm.
 - Nếu có lỗi xảy ra, trả về None.

Hàm `create_email(sender, receiver, subject, text, attachment)`

- Nhiệm vụ chính: tạo một email.
- Đầu vào:
 - sender: kiểu string, là địa chỉ email của người gửi.
 - receiver: kiểu string, là địa chỉ email của người nhận.
 - subject: kiểu string, là tiêu đề email.
 - text: kiểu string, là nội dung dạng chữ của email.
 - attachment: kiểu string, là đường dẫn đến tập tin đính kèm email nếu có, nếu không có tập tin đính kèm thì để giá trị là None.
- Đầu ra:
 - Nếu không xảy ra lỗi, trả về đối tượng kiểu dictionary chứa dữ liệu email.
 - Nếu có lỗi xảy ra, trả về None.

Hàm send_email(service, user_id, body)

- Nhiệm vụ chính: gửi một email thông qua dịch vụ Gmail.
- Đầu vào:
 - service: dịch vụ gửi email, trong phần mềm này là Gmail.
 - user_id: địa chỉ email của người sử dụng dịch vụ, nếu người sử dụng dịch vụ là người dùng đã được cấp quyền ở tham số service thì để giá trị là 'me'.
 - body: kiểu dictionary, là dữ liệu của email cần gửi.
- Đầu ra:
 - Nếu gửi email thành công, trả về một dictionary chứa thông tin về email đã gửi.
 - Nếu có lỗi xảy ra, trả về None.

Hàm restart(time_str)

- Nhiệm vụ chính: lên lịch khởi động lại máy tính vào thời điểm nhất định trong cùng ngày.
- Đầu vào:
 - time_str: kiểu string, là thời điểm khởi động lại máy, định dạng hh:mm:ss.
- Đầu ra:
 - Nếu lên lịch khởi động lại thành công, trả về True.
 - Nếu thời điểm khởi động lại là sớm hơn so với thời gian thực thi hàm hoặc người dùng máy tính hiện tại không đồng ý khởi động lại, hoặc có lỗi xảy ra trong quá trình thực thi hàm, trả về giá trị None.

Hàm shutdown(time_str)

- Nhiệm vụ chính: lên lịch tắt máy tính vào thời điểm nhất định trong cùng ngày.
- Đầu vào:
 - time_str: kiểu string, là thời điểm tắt máy, định dạng hh:mm:ss.
- Đầu ra:
 - Nếu lên lịch tắt máy thành công, trả về True.
 - Nếu thời điểm tắt máy là sớm hơn so với thời gian thực thi hàm hoặc người dùng máy tính hiện tại không đồng ý tắt máy, hoặc có lỗi xảy ra trong quá trình thực thi hàm, trả về giá trị None.

Hàm `copy_file(src_path, dest_path)`

- Nhiệm vụ chính: sao chép tập tin.
- Đầu vào:
 - `src_path`: kiểu string, là đường dẫn đến tập tin nguồn.
 - `dest_path`: kiểu string, là đường dẫn đến tập tin đích.
- Đầu ra:
 - Nếu thực hiện sao chép thành công, trả về True và đường dẫn đích.
 - Nếu có lỗi xảy ra như tập tin nguồn không tồn tại hay lỗi trong quá trình sao chép, trả về None kèm theo thông tin lỗi dạng string.

Hàm `take_screenshot(save_dir)`

- Nhiệm vụ chính: chụp ảnh toàn màn hình máy tính và lưu vào tệp tin png.
- Đầu vào:
 - `save_dir`: kiểu string, là đường dẫn đến thư mục lưu ảnh chụp màn hình.
- Đầu ra:
 - Nếu chụp ảnh màn hình thành công, trả về True và đường dẫn đến tệp tin hình ảnh.
 - Nếu có lỗi xảy ra, trả về None và thông tin lỗi.

Hàm `list_processes()`

- Nhiệm vụ chính: lấy danh sách các tiến trình đang chạy trên máy tính.
- Đầu vào: không có.
- Đầu ra:
 - Nếu lấy danh sách các tiến trình thành công, trả về một dictionary chứa 3 khoá: `name`, `id`, `num_of_thread`, mỗi khoá gắn với một list các giá trị tương ứng theo thứ tự.
 - Nếu có lỗi xảy ra, trả về None, kèm theo thông tin lỗi dưới dạng string.

Hàm `kill_process(pid)`

- Nhiệm vụ chính: buộc dừng một tiến trình với PID xác định.
- Đầu vào:
 - `pid`: kiểu string, là ID của tiến trình cần dừng.
- Đầu ra:
 - Nếu dừng tiến trình thành công, trả về True và tên tiến trình đã dừng.
 - Nếu dừng tiến trình không thành công, trả về None và thông tin lỗi dưới dạng string.

Hàm `key_log(time)`

- Nhiệm vụ chính: bắt phím nhấn từ bàn phím trong một khoảng thời gian nhất định và lưu vào một tập tin txt.
- Đầu vào:
 - `time`: kiểu int, là số giây thực hiện bắt phím nhấn.
- Đầu ra:
 - Nếu quá trình bắt phím thành công, trả về True.
 - Nếu xảy ra lỗi trong quá trình bắt phím, trả về None kèm theo thông tin lỗi dưới dạng string.

Hàm `webcam_shot()`

- Nhiệm vụ chính: chụp một bức ảnh từ webcam và lưu vào tập tin png.
- Đầu vào: không có.
- Đầu ra:
 - Nếu chụp ảnh từ webcam và lưu thành công, trả về True kèm theo đường dẫn đến tập tin hình ảnh.
 - Nếu có lỗi xảy ra, trả về None kèm theo thông tin lỗi dưới dạng string.

Hàm `parse_data(full_path)`

- Nhiệm vụ chính: phân tích đường dẫn đến một registry key thành các thành phần hive, registry key, registry value.
- Đầu vào:
 - `full_path`: kiểu string, là đường dẫn của registry key.
- Đầu ra:
 - Nếu không có lỗi xảy ra, trả về một dictionary chứa 3 khoá: `hive`, `reg_key` và `reg_value`, mỗi khoá ứng với một giá trị tương ứng kiểu string.
 - Nếu có lỗi xảy ra, trả về dictionary với 3 khoá như trên, nhưng mỗi khoá đều ứng với giá trị None.

Hàm `set_value(full_path, value, value_type)`

- Nhiệm vụ chính: gán một giá trị cho một registry key.
- Đầu vào:
 - `full_path`: kiểu string, là đường dẫn đến registry key.
 - `value`: kiểu string, là giá trị cần gán.

- `value_type`: kiểu string, là kiểu giá trị cần gán.
- Đầu ra:
 - Nếu gán giá trị thành công, trả về True.
 - Nếu gán giá trị thất bại, trả về False.

Hàm `delete_value(full_path)`

- Nhiệm vụ chính: xoá một giá trị khỏi một registry key.
- Đầu vào:
 - `full_path`: kiểu string, là đường dẫn đến registry key.
- Đầu ra:
 - Nếu xoá giá trị thành công, trả về True.
 - Nếu xoá giá trị thất bại, trả về False.

Hàm `webcam_record(record_time)`

- Nhiệm vụ chính: ghi hình từ webcam và lưu vào tập tin mp4.
- Đầu vào:
 - `record_time`: kiểu int, là số giây thực hiện ghi hình. Nếu số giây này lớn hơn 30 giây, sẽ được giới hạn thành 30 giây để quá trình gửi email phản hồi diễn ra thuận lợi.
- Đầu ra:
 - Nếu quá trình ghi hình diễn ra thành công, trả về True kèm theo đường dẫn đến tập tin video.
 - Nếu có lỗi xảy ra, trả về None kèm theo thông tin lỗi dưới dạng string.

Hàm `prepare_for_listen()`

- Nhiệm vụ chính: chuẩn bị các bước cần thiết trước khi tiến hành lắng nghe email yêu cầu, các bước chuẩn bị bao gồm:
 - Kiểm tra tập tin chứa thông tin tài khoản Google, nếu tập tin này bị sai hoặc không tồn tại thì yêu cầu người dùng đăng nhập và lưu thành tập tin mới.
 - Xây dựng kết nối đến dịch vụ Gmail.
 - Nếu có lỗi xảy ra trong 2 bước trên, xoá tập tin thông tin tài khoản và thực hiện đăng nhập lại cho đến khi thành công kết nối đến dịch vụ Gmail.
- Đầu vào: không có.
- Đầu ra: không có.

Hàm listen()

- Nhiệm vụ chính: thực hiện vòng lặp đọc và thực thi email yêu cầu. Các email yêu cầu nếu có sẽ được thực thi lần lượt theo thứ tự email mới nhất đến cũ nhất.
- Đầu vào: không có.
- Đầu ra: không có.

Các hàm do_X_request(email)

- Nhiệm vụ chính: thực thi một email yêu cầu một hoạt động X, và gửi trả email phản hồi.
- Đầu vào:
 - email: kiểu dictionary, là dữ liệu email yêu cầu, chứa 3 khoá: subject, sender, context.
- Đầu ra: không có giá trị trả về.

5. Hướng dẫn sử dụng

5.1. Chương trình nhận email và điều khiển máy tính

- Để điều khiển được máy tính qua email, cần khởi chạy chương trình PC Remote Control trên máy cần điều khiển.

Lưu ý: cần đảm bảo các file credentials.json, README.md và thư mục icons nằm cùng thư mục với file thực thi PC Remote Control.exe

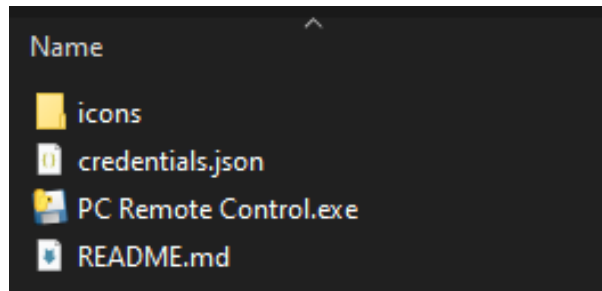


Figure 3: Mở file PC Remote Control.exe

- Nếu chương trình không tìm thấy thông tin tài khoản, cần đăng nhập tài khoản Google.

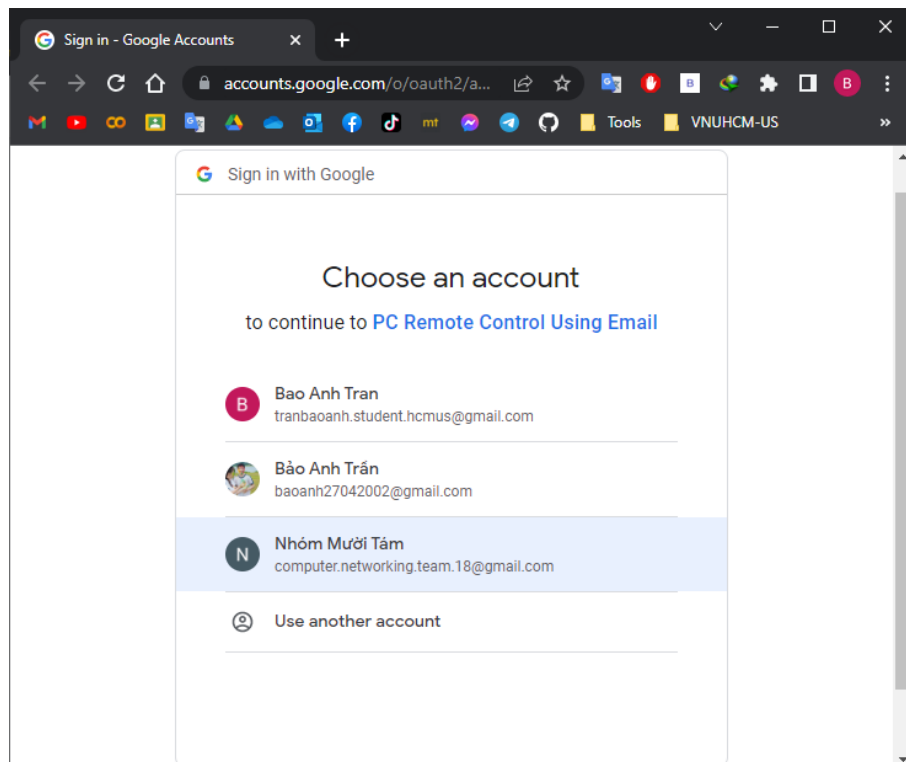


Figure 4: Chọn hoặc đăng nhập tài khoản Google

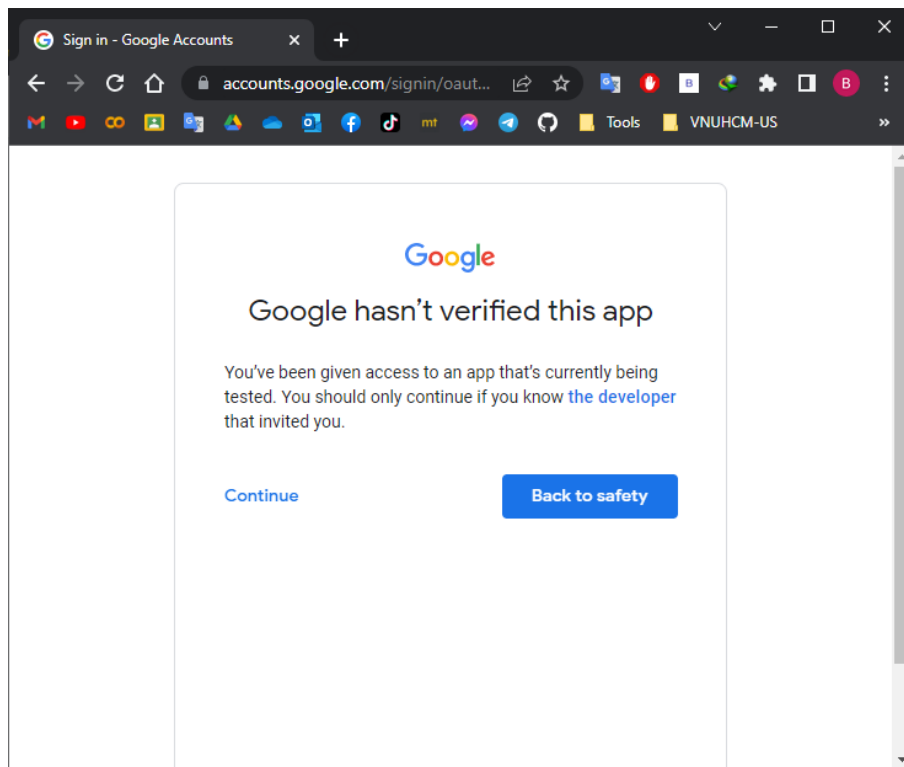


Figure 5: Chọn Continue để tiếp tục

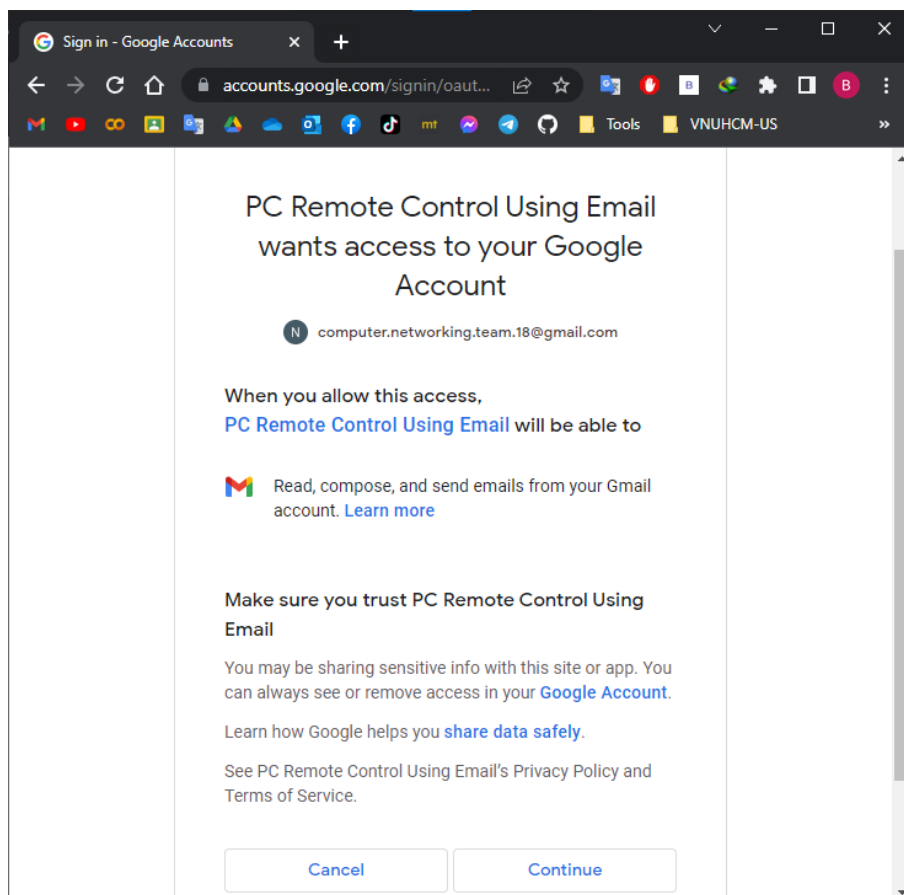


Figure 6: Chọn Continue để cấp quyền truy cập Gmail cho ứng dụng

- Sau khi đăng nhập thành công, đây là giao diện chính của chương trình

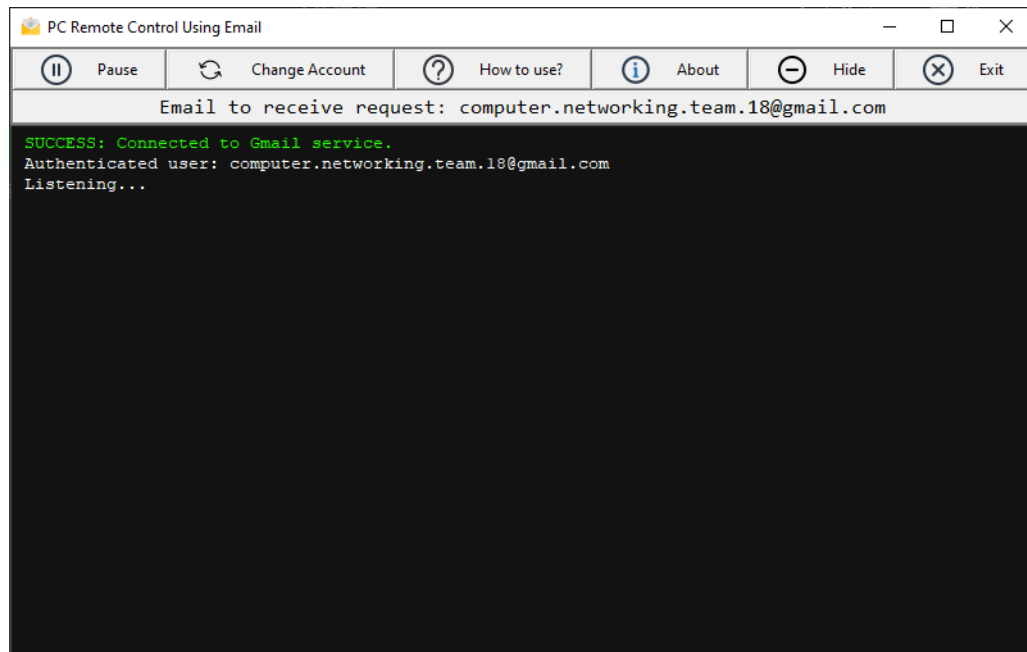


Figure 7: Giao diện chính của chương trình

Giao diện của chương trình tương đối đơn giản với một khu vực hiển thị các thông báo, thông tin địa chỉ nhận email yêu cầu và 6 phím chức năng:

- Pause/Continue: để tạm dừng và tiếp tục lắng nghe email yêu cầu.
- Change Account: để thay đổi tài khoản Google.
- How to use?: để xem cấu trúc email yêu cầu.
- About: thông tin nhà phát triển.
- Hide: ẩn cửa sổ chương trình, tương tự khi ấn nút X ở góc phải cửa sổ.
- Exit: thoát khỏi chương trình.

Ngoài cửa sổ chính, chương trình còn cung cấp một biểu tượng nhỏ ở góc phải dưới màn hình trên thanh taskbar với các chức năng tương tự như trên cửa sổ chính.



Figure 8: Biểu tượng ứng dụng trên thanh taskbar

- Như vậy là chương trình nhận email yêu cầu đã bắt đầu hoạt động và sẵn sàng thực thi các yêu cầu được gửi đến.

5.2. Chương trình gửi email yêu cầu

- Để khởi chạy chương trình gửi email yêu cầu, mở tập tin Request Email Generator.exe.
Lưu ý: cần đảm bảo thư mục data, tệp tin credentials.json và Request Email Generator.exe nằm trong cùng một thư mục.

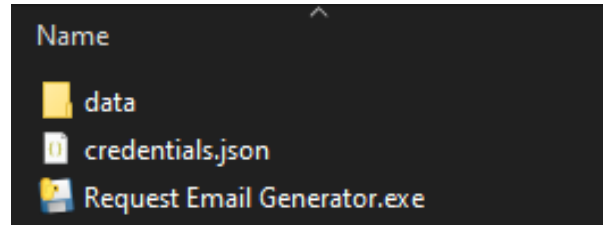


Figure 9: Mở file Request Email Generator.exe

- Sau khi khởi chạy chương trình, chọn Sign in with Google để đăng nhập vào tài khoản Google.

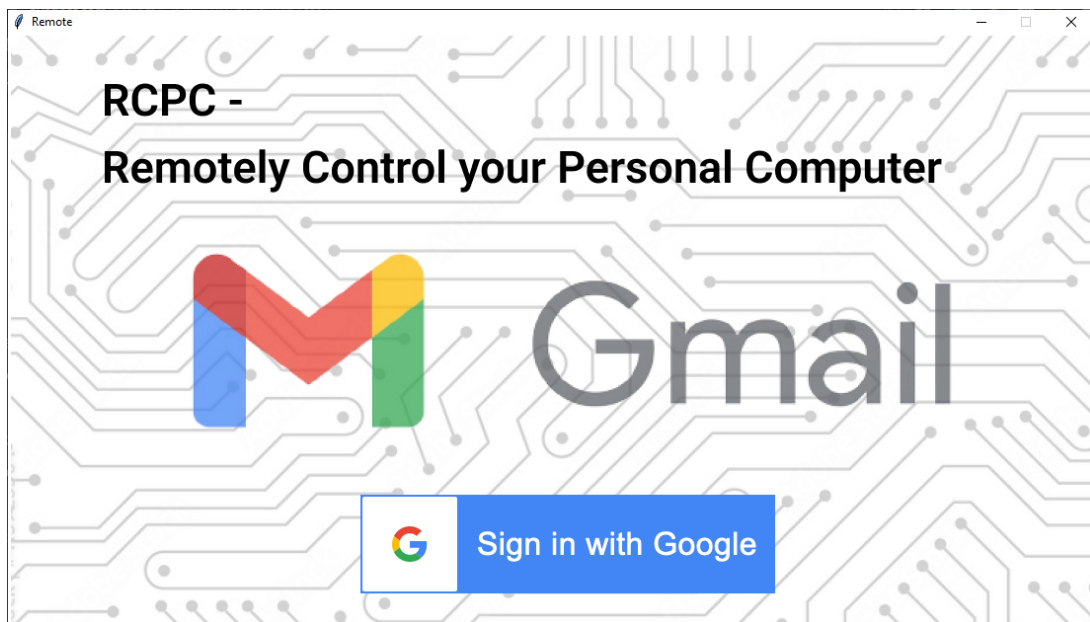


Figure 10: Nhấn Sign in with Google để đăng nhập tài khoản Google

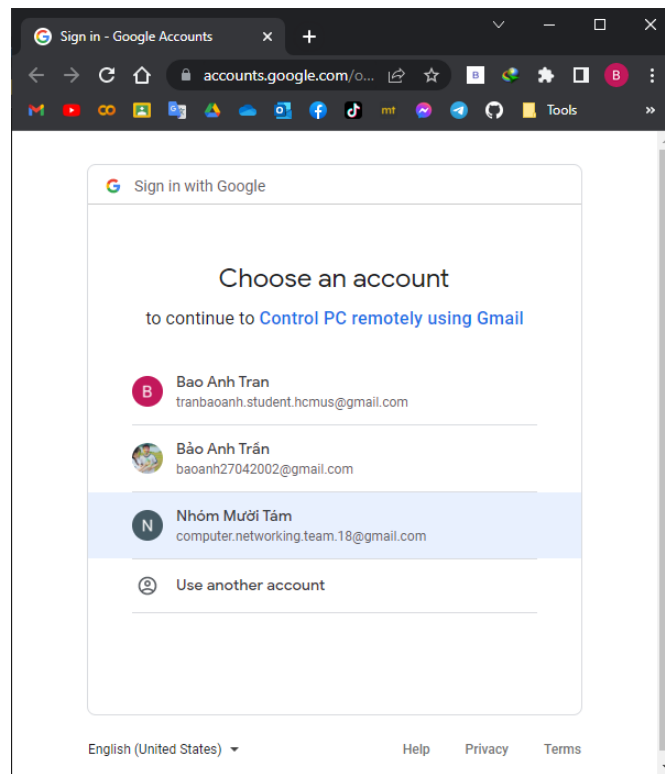


Figure 11: Chọn hoặc đăng nhập tài khoản Google

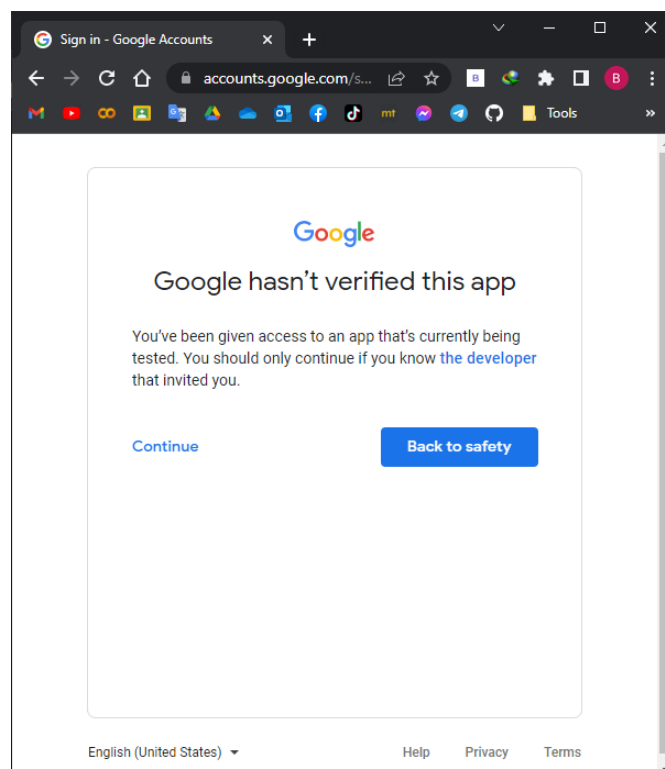


Figure 12: Chọn Continue để tiếp tục

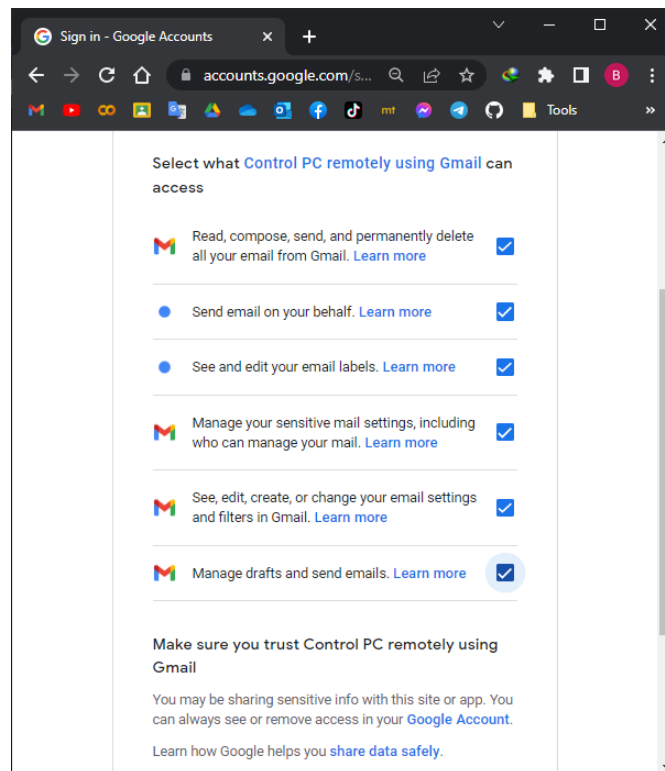


Figure 13: Đánh dấu Tick để cấp quyền cho ứng dụng sau đó nhấn Continue

- Sau khi đăng nhập thành công, đây là giao diện chính của chương trình.

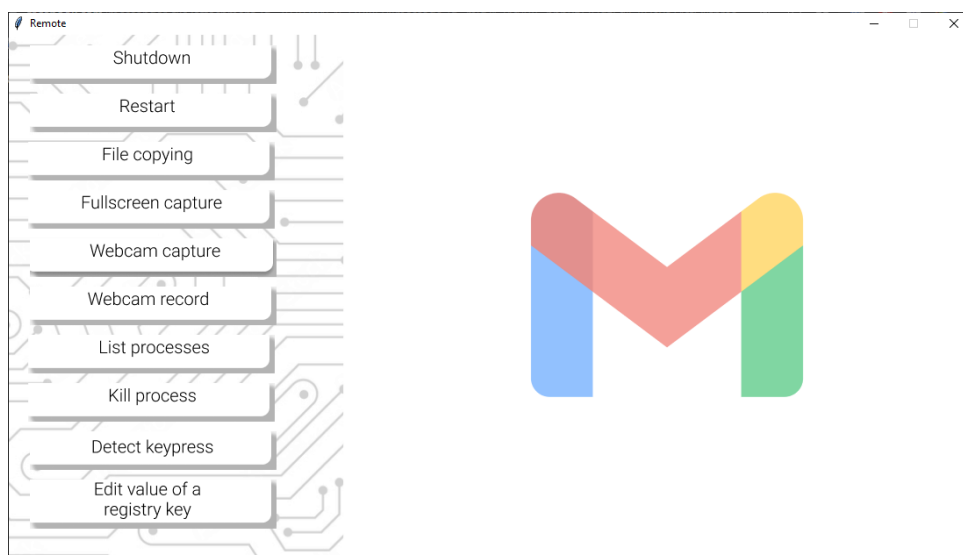


Figure 14: Giao diện chính của chương trình gửi email yêu cầu

- Để sử dụng chương trình, chỉ cần chọn chức năng cần thực thi trong các chức năng bên phía trái của sổ chương trình, sau đó nhập các thông tin cần thiết và nhấn SEND để gửi email yêu cầu.

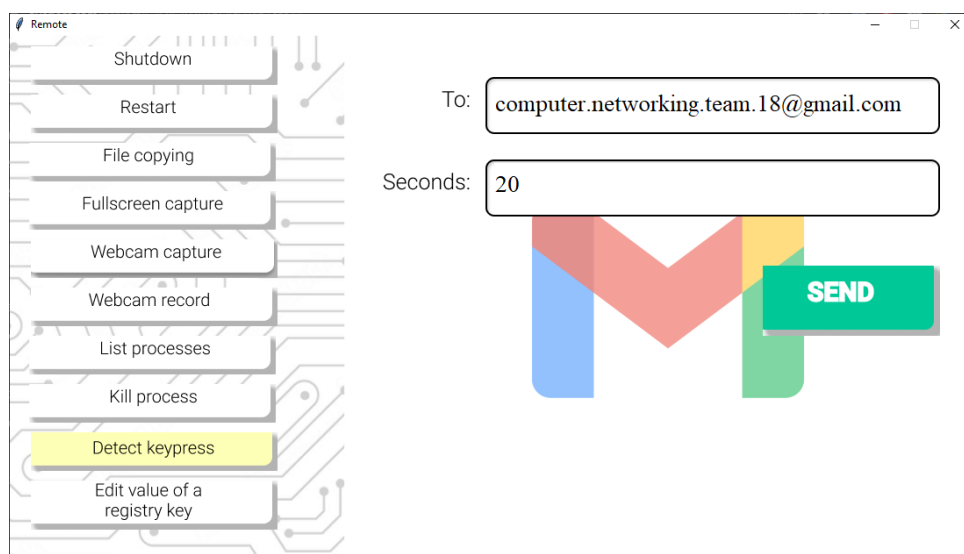


Figure 15: Chọn chức năng và gửi email yêu cầu

6. Cấu trúc Email

Nhằm mang đến sự tiện lợi cho người sử dụng trên nền tảng web gmail, nhóm em sẽ cung cấp cấu trúc email gửi yêu cầu xử lý. Với mỗi chức năng, sẽ tương ứng với: Tiêu đề - Nội dung của thư điện tử.

Lưu ý: dòng đầu tiên của nội dung email phải là mã nhận diện **019250304**, sau đó mới đến các nội dung tiếp theo nếu có.

- Tắt nguồn máy tính vào thời điểm cố định:
SHUTDOWN - <Thời điểm tắt nguồn có cấu trúc hh:mm:ss>.
- Khởi động lại máy tính vào thời điểm cố định:
RESTART - <Thời điểm khởi động lại có cấu trúc hh:mm:ss>.
- Sao chép tập tin đến một địa dẫn khác:
COPY FILE - <Đường dẫn cũ của tập tin>
 <Đường dẫn mới của tập tin>.
- Chụp toàn màn hình máy tính:
SCREEN CAPTURE - <Bất kì/Trống>.
- Chụp ảnh từ webcam của máy tính:
WEBCAM CAPTURE - <Bất kì/Trống>.
- Ghi hình từ webcam máy tính trong khoảng thời gian:
WEBCAM RECORD - <Số giây cần ghi hình>.
- Bắt phím do người dùng nhấn trong khoảng thời gian:
KEYPRESS - <Số giây cần bắt phím nhấn>.
- Cập nhật/xóa giá trị của một registry key:
REGISTRY KEY - [delete/set]
 <Đường dẫn của registry key>
 <Giá trị> (nếu cập nhật bằng lệnh [set])
 <Kiểu giá trị> (nếu cập nhật bằng lệnh [set]).
- Liệt kê các tiến trình đang chạy trên máy tính:
LIST PROCESSES - <Bất kì/Trống>.
- Dừng một tiến trình của máy tính theo PID:
KILL PROCESS - <Mã ID của tiến trình đó>.

7. Đánh giá mức độ hoàn thành

Chức năng	Đánh giá
Liệt kê các tiến trình đang chạy trên máy tính	✓
Dừng một tiến trình của máy tính theo PID	✓
Chụp toàn màn hình máy tính	✓
Chụp ảnh từ webcam của máy tính	✓
Ghi hình từ webcam máy tính trong khoảng thời gian cố định tối đa 30 giây (do giới hạn tốc độ gửi email của Gmail API)	✓
Lên lịch tắt nguồn máy tính vào thời điểm cố định	✓
Lên lịch khởi động lại máy tính vào thời điểm cố định	✓
Bắt phím do người dùng nhấn trong khoảng thời gian cố định (giây)	✓
Sao chép tập tin đến một địa dẫn khác	✓
Cập nhật/xóa giá trị của một registry key khi biết đường dẫn của registry key	✓
Giao diện người dùng	✓

Tài liệu tham khảo

- [1] Google Developers. *Develop on Google Workspace*. URL: <https://developers.google.com/workspace/guides/get-started>.
- [2] Google Developers. *Gmail API - Python Quickstart*. URL: <https://developers.google.com/gmail/api/quickstart/python>.