**1. Đặt file credentials\_oauth.json đúng chỗ**

File credentials\_oauth.json là "chìa khóa" để ứng dụng có thể bắt đầu quá trình xác thực với Google.

* **Khi đóng gói:** Bạn cần dùng một công cụ như PyInstaller và phải chỉ cho nó biết để **đính kèm** file này vào trong file .exe hoặc đặt nó bên cạnh.
* **Cách đơn giản nhất:** Sau khi đóng gói, hãy đảm bảo rằng file client\_app.exe và file credentials\_oauth.json **luôn nằm trong cùng một thư mục** khi bạn gửi cho các trường. Code đã được lập trình để tự tìm file này.

**2. Lệnh đóng gói (Ví dụ với PyInstaller)**

Khi bạn sử dụng PyInstaller, hãy dùng các tùy chọn --onefile (để tạo một file duy nhất), --windowed (để ẩn cửa sổ dòng lệnh màu đen), và quan trọng nhất là --add-data để đính kèm file credentials\_oauth.json.

Đây là một ví dụ về câu lệnh bạn có thể chạy trong terminal:

Bash

pyinstaller --onefile --windowed --add-data "credentials\_oauth.json;." client\_app.py

* --onefile: Tạo ra một file client\_app.exe duy nhất.
* --windowed: Khi chạy file .exe, sẽ không có cửa sổ console màu đen hiện ra.
* --add-data "credentials\_oauth.json;.": Lệnh này bảo PyInstaller lấy file credentials\_oauth.json và đặt nó vào thư mục gốc (.) bên trong gói ứng dụng.

**## Trải nghiệm của người dùng (Trường học) sẽ như thế nào?**

Bạn có thể yên tâm rằng với phiên bản code mới nhất, quy trình cho các trường học sẽ rất đơn giản:

1. **Lần chạy đầu tiên:**
   * Họ chạy file client\_app.exe.
   * Một cửa sổ trình duyệt (Chrome, Cốc Cốc...) sẽ **tự động mở ra**, yêu cầu họ đăng nhập vào tài khoản Google của trường.
   * Họ đăng nhập và nhấn **"Cho phép" (Allow)** để cấp quyền cho ứng dụng.
   * Sau đó, họ có thể đóng trình duyệt và sử dụng ứng dụng bình thường.
2. **Từ các lần chạy sau:**
   * Họ chỉ cần mở file client\_app.exe.
   * Ứng dụng sẽ **chạy ngay lập tức** mà không cần phải đăng nhập hay xác thực lại. Tất cả thông tin đã được lưu an toàn trong thư mục C:\Users\<Tên người dùng>\.auto\_report\_client.