ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP. HỒ CHÍ MINH

NHẬP MÔN MÃ HÓA MẬT MÃ

BÁO CÁO ĐỒ ÁN 1

Giáo viên hướng dẫn : Thầy Nguyễn Văn Quang Huy
Thầy Ngô Đình Hy
Thầy Nguyễn Đình Thức

Sinh viên thực hiện:

Nguyễn Thanh Quân	19127525
Phùng Anh Khoa	19127449
Lâm Chí Văn	19127625
Tô Gia Hảo	19127392
Thái Duy Nguyễn	19127054

THÀNH PHỐ HỒ CHÍ MINH THÁNG 1 2022

Mục lục:

I) Khái quát:

1. Tổng quát về ứng dụng:

Ứng dụng lưu trữ file an toàn của nhóm SOD sử dụng 3 ngôn ngữ lập trình (Python, C#, Java)

- Server: Sử dụng Java - Spring boot và Heroku
- Client: Sử dụng C# - Winform
- Mã hoá giải mã (RSA): Sử dụng Python
- Cloud database: Sử dụng MongoDB Atlas và Cloudinary

Server của ứng dụng được deploy lên Heroku - trang web giúp chạy server trực tuyến và sử dụng Cloud database nhằm cho phép người dùng ở các nơi khác nhau đều có thể tương tác với nhau. Heroku linh hoạt và dễ sử dụng, cung cấp cho một con đường đơn giản nhất để đưa sản phẩm tiếp cận người dùng. Nó giúp các nhà phát triển tập trung vào phát triển sản phẩm mà không cần quan tâm đến việc vận hành máy chủ hay phần cứng…

Ảnh sẽ được lưu trữ trên Cloudinary- một cloud-based service, nó cung cấp một giải pháp quản lý hình ảnh bao gồm upload, lưu trữ, thao tác, tối ưu hóa và delivery. Thông tin người dùng, đường link dẫn tới ảnh trên Cloudinary thì sẽ được lưu trên MongoDB - một cơ sở dữ liệu đa nền tảng, hoạt động trên các khái niệm Collection và Document, nó cung cấp hiệu suất cao, tính khả dụng cao và khả năng mở rộng dễ dàng.

Việc mã hoá giải mã sử dụng ngôn ngữ lập trình Python vì Python có các thư viện hỗ trợ tốt cho việc tương tác với ảnh.

Client sử dụng C# winform, do có tốc độ xử lý dữ liệu nhanh chóng, đảm bảo an toàn, bảo mật thông tin, có thể chạy trên các phiên bản Windows khác nhau và thao tác trên nhiều giao diện.

2. Khả năng lưu trữ hình ảnh của ứng dụng:

Mỗi lần upload ảnh, người dùng chỉ được upload 1 tấm, và ảnh này sẽ được được mã hoá từng điểm ảnh bằng public key của người dùng.
Sau đó tấm ảnh đã được mã hoá sẽ được gửi lên Cloudinary và đồng thời gửi đường link tới ảnh với id user lên MongoDB để lưu trữ.

3. Xem danh sách tệp ảnh thuộc về mình:

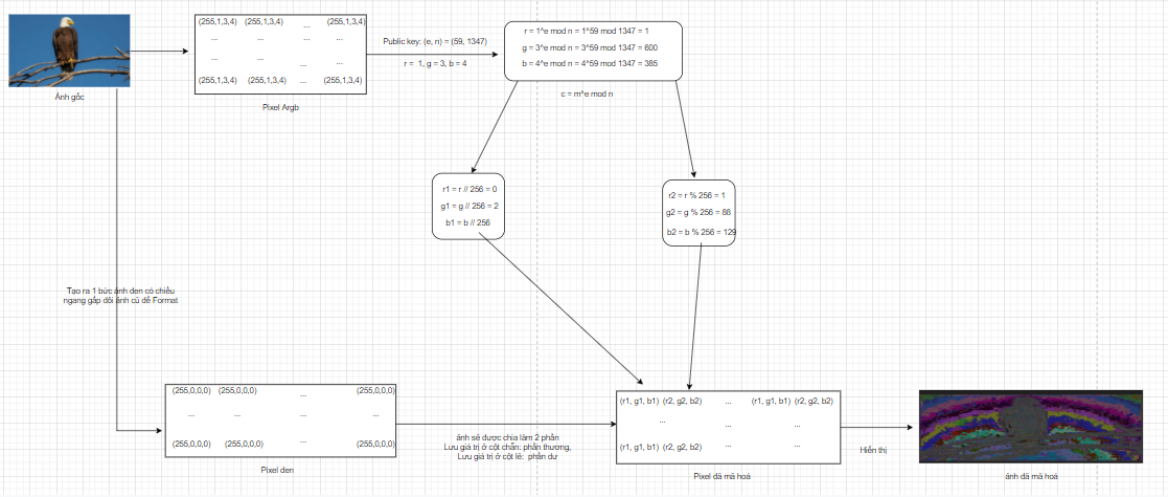
Khi người dùng đăng nhập vào hệ thống thì hệ thống sẽ hiển thị danh sách các hình ảnh của người dùng đã tải lên ứng dụng trước đó.

4. Gửi ảnh tới một user có ID cụ thể:

User1 chọn các ảnh muốn gửi, hệ thống lúc này sẽ yêu cầu người dùng nhập đúng private key để có thể tiến hành giải mã các ảnh đó.
Tiếp theo, hệ thống sẽ yêu cầu người dùng nhập ID của User2 mà người dùng muốn gửi tới, khi có được ID của User2 thì hệ thống sẽ hỏi Server Public Key của User2 để tiến hành mã hoá ảnh, sau đó gửi ảnh lên Cloudinary và cuối cùng là lưu trữ ID user2 và link tới ảnh lên MongoDB

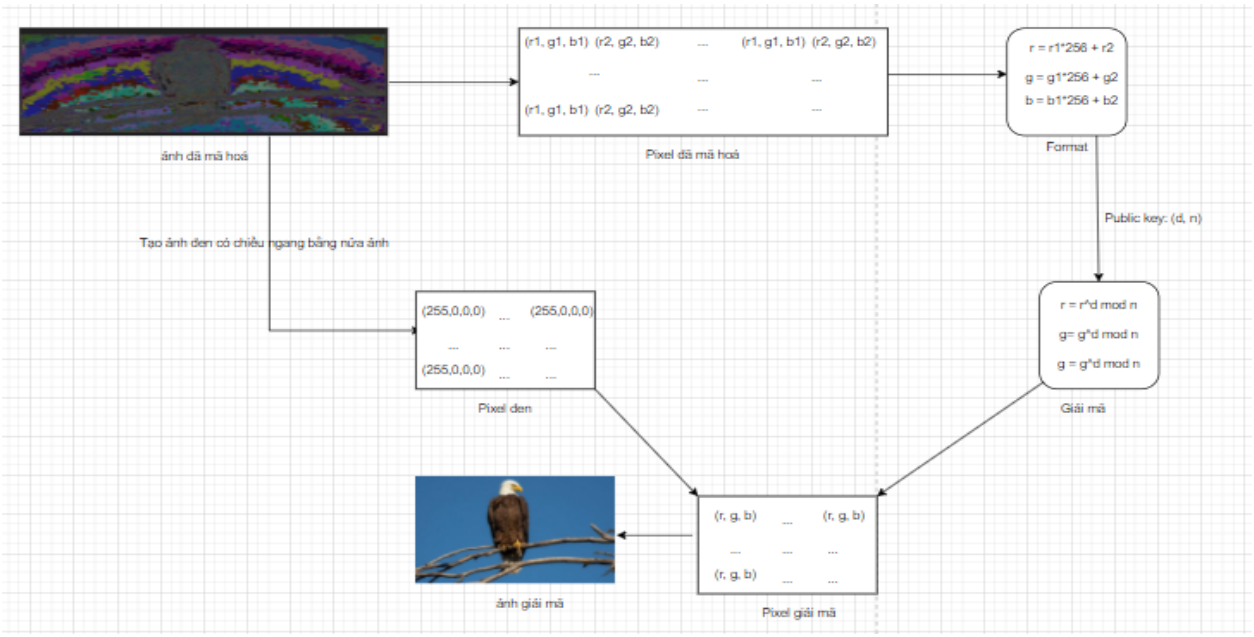
5. Mã hoá ảnh:

Việc mã hóa hình ảnh như đã nói ở trên sẽ sử dụng ngô ngữ Python.
Thực hiện đọc các pixel từ hình ảnh, Đọc giá trị R, G, B ở mỗi pixel sau đó mã hóa 3 giá trị này bằng khóa công khai. Các giá trị sau khi mã hóa có thể lớn hơn 255 nên ta sẽ tạo 1 mảng có kích thước gấp 2 lần mảng pixel bên trên.
Thực hiện chia lấy thương và dư cho 256 cho từng cặp [R, G, B], các cột có index chẵn (0, 2, 4, 6, …) sẽ là giá trị phần thương và các cột có index lẻ (1, 3, 5, 7, …) sẽ lưu giữ giá trị phần dư.
Khi thực hiện xong, ta format lại mảng này và lưu nó trở thành 1 file ảnh với định dạng .bmp.

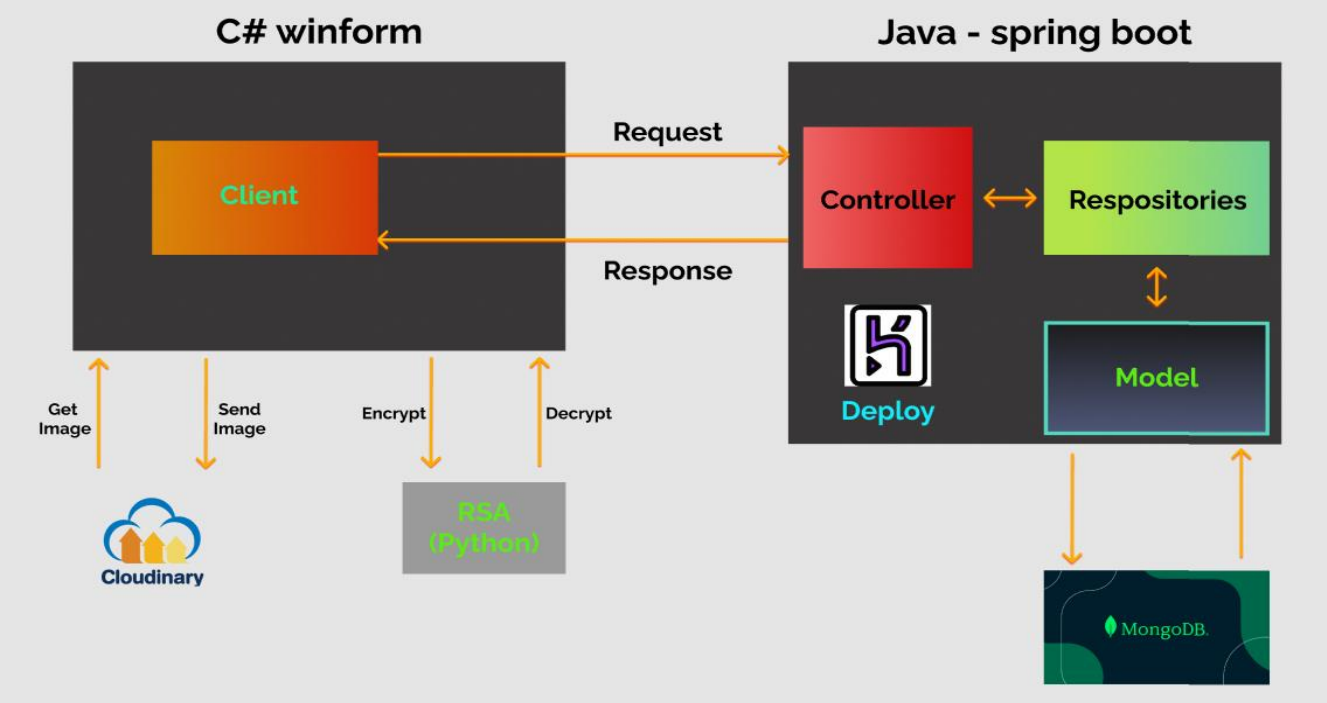


6. Giải mã ảnh:

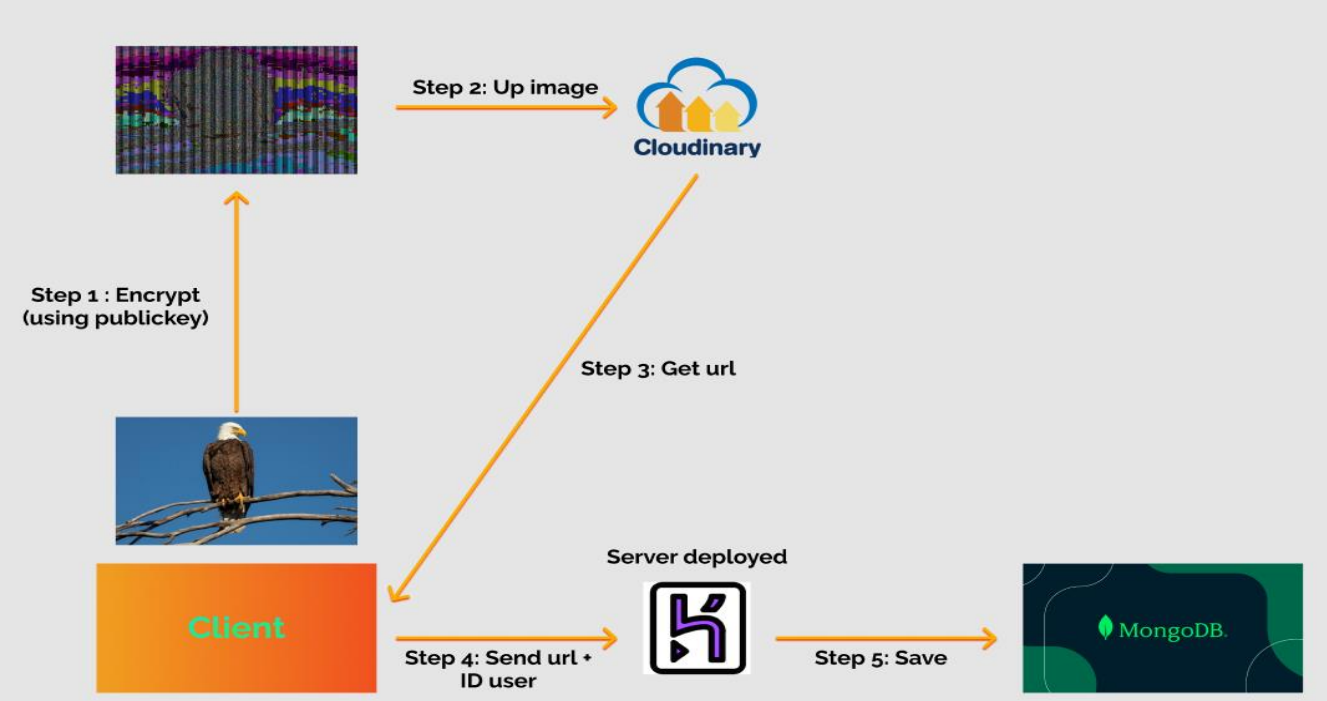
Ta sẽ viết hàm return_Ori (a, b), có tác dụng trả về bộ ba giá trị [R, G, B] đã được mã hóa ban đầu khi chưa chia lấy thương và dư cho 256.Giá trị đầu vào là cột chẵn và cột lẻ tương đương.
Giải mã từng vị trí index 1, với giá trị [R, G, B] cần giải mã đã được biến đổi về dạng khi vừa được mã hóa bằng khóa bí mật.
Khi thực hiện xong ta sẽ format lại mảng này và lưu nó trở thành 1 file ảnh với định dạng dạng ban đầu của ảnh.



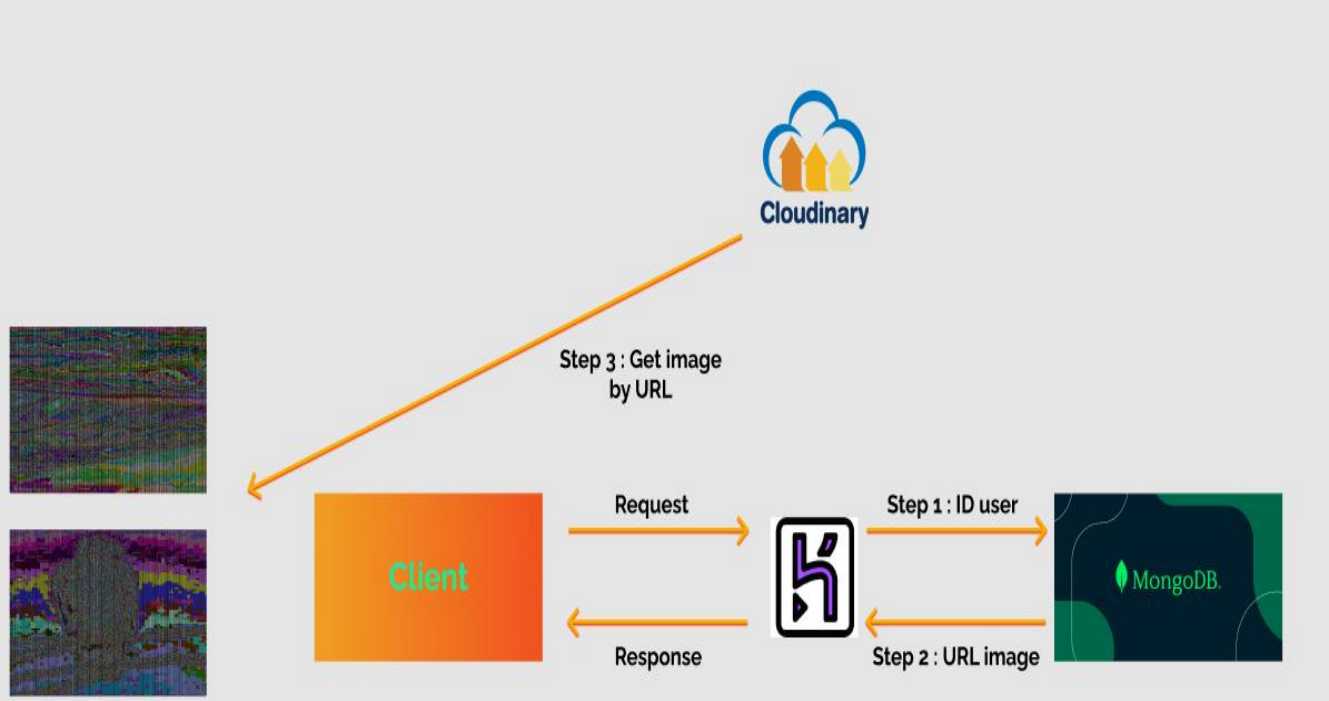
II) Protocol sử dụng ứng dụng:



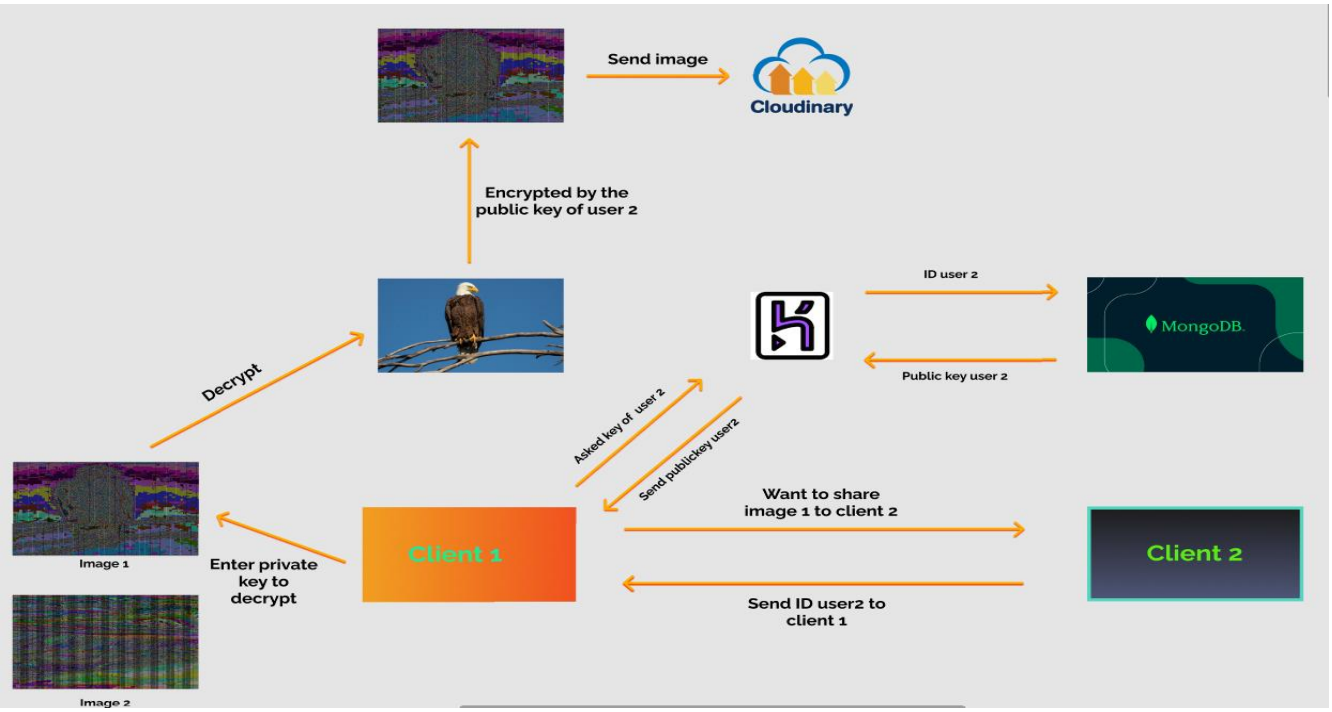
Tổng quan



Upload hình ảnh



Lấy hình ảnh

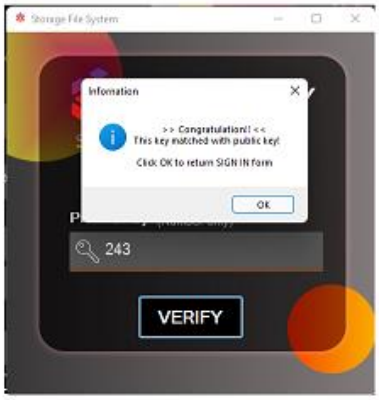


Chia sẻ hình ảnh

III) Hướng dẫn sử dụng chức năng ứng dụng:

1. Đăng ký tài khoản ứng dụng:

- Người dùng nếu chưa có tài khoản để đăng nhập ứng dụng sẽ bấm vào nút “SIGN UP” ở dưới thanh nhập password để mở màn hình đăng ký.
- Tiến hành đăng ký tài khoản bằng cách điền các thông tin cần thiết vào các hộp thoại xuất hiện trên màn hình
- Sau khi đã đăng ký tài khoản xong và nhấn “REGISTER”, sẽ xuất hiện hộp thoại “VERIFY KEY” để người dùng nhập Private key, nếu nhập sai thì ứng dụng sẽ yêu cầu người dung nhập lại
- Lưu ý: Username và Public key mà người dùng đăng ky phải là duy nhất, tức là không được trùng với các Username và Public Key của các người dùng khác đã dùng để đăng ký tài khoản.



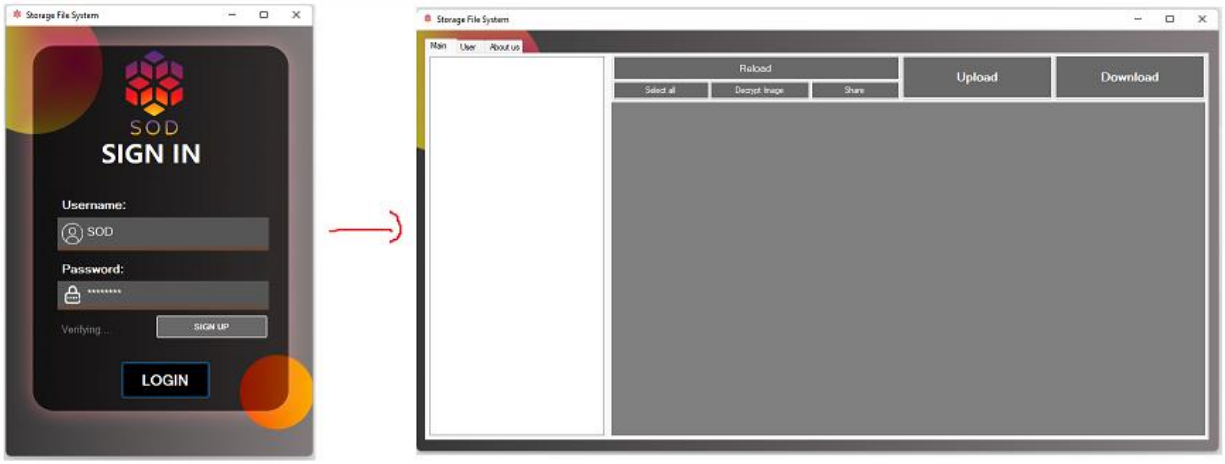
2. Đăng nhập

Lúc mở ứng dụng

Không nhập hết

Đang kiểm tra tài khoản

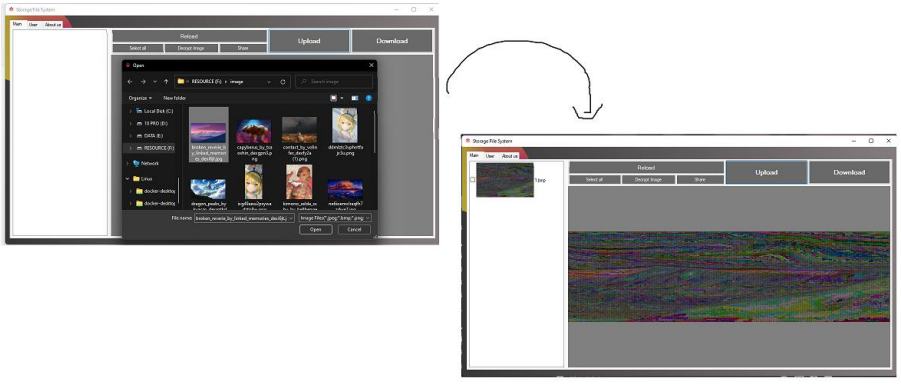
Đăng nhập thất bại



Đăng nhập thành công

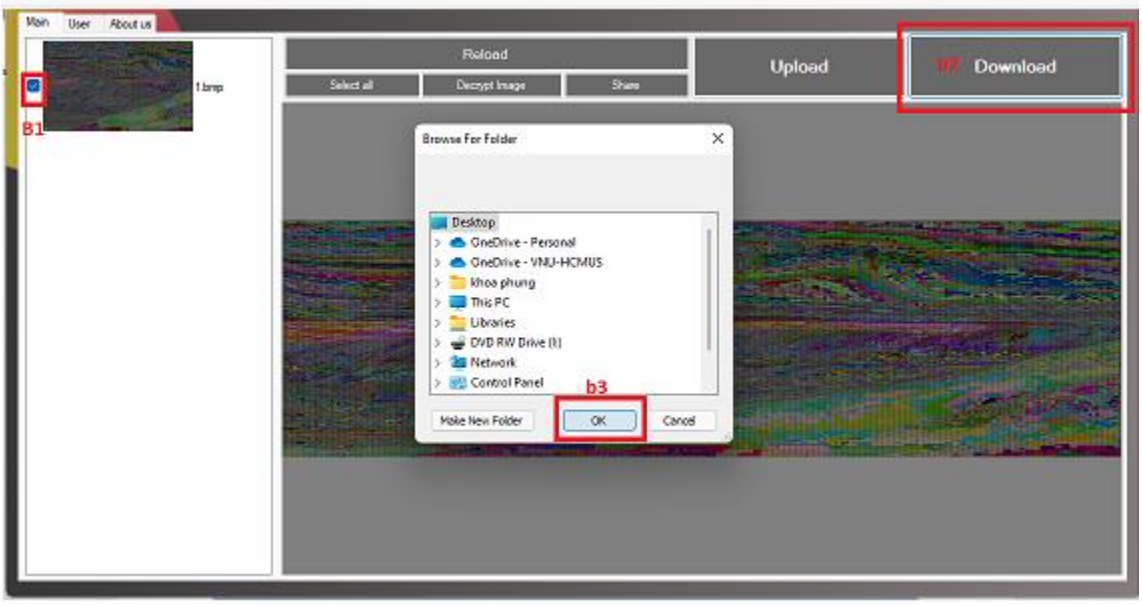
3. Upload hình ảnh lên ứng dụng:

- Sau khi đã đăng ký hoặc đăng nhập thành công vào ứng dụng, người dùng sử dụng chức năng Upload hình ảnh bằng cách nhấn nút 'UPLOAD" trên màn hình chính của ứng dụng. Lúc này người dùng sẽ chọn hình ảnh đã được lưu vào bộ nhớ máy tính để tiến hành Upload lên ứng dụng, và mỗi lần upload lên ứng dụng chỉ được 1 ảnh mỗi lần
- Khi người dùng Upload ảnh lên, lức này ảnh sẽ được ứng dụng mã hóa bằng Public key àm người dùng đã sử dụng để đăng ký tài khoản



4. Download hình ảnh đã được mã hóa về máy:

Người dùng chọn hình ảnh cần tải về từ ứng dụng bằng cách bấm tick vào ô bên trái của bức ảnh , sau đó nhấn vào nút “DOWNLOAD” trên màn hình chính của ứng dụng để tiến hành tải hình ảnh về máy.

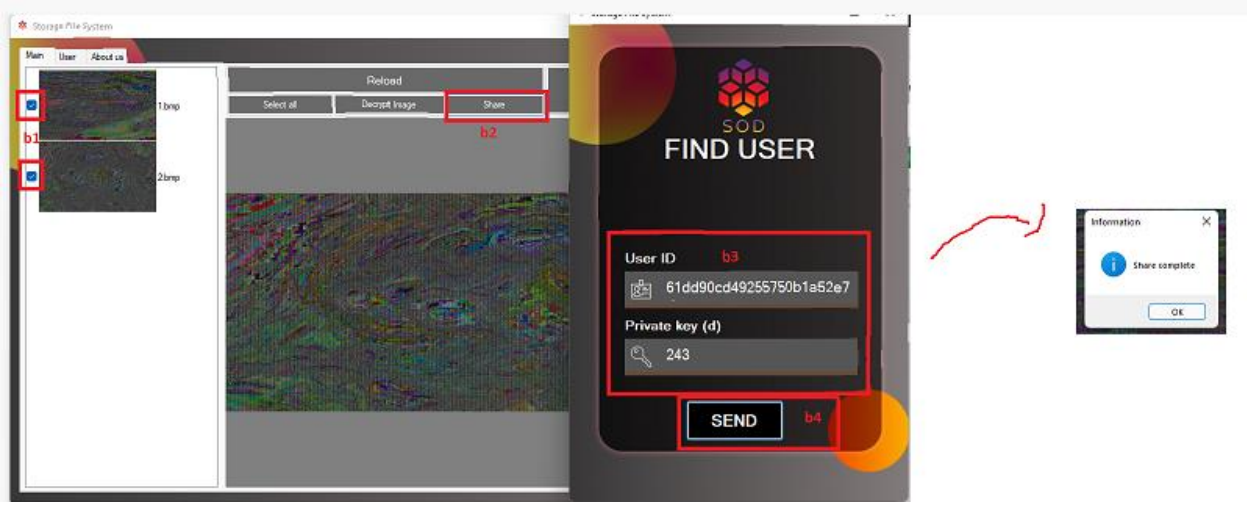


5. Giải mã ảnh:

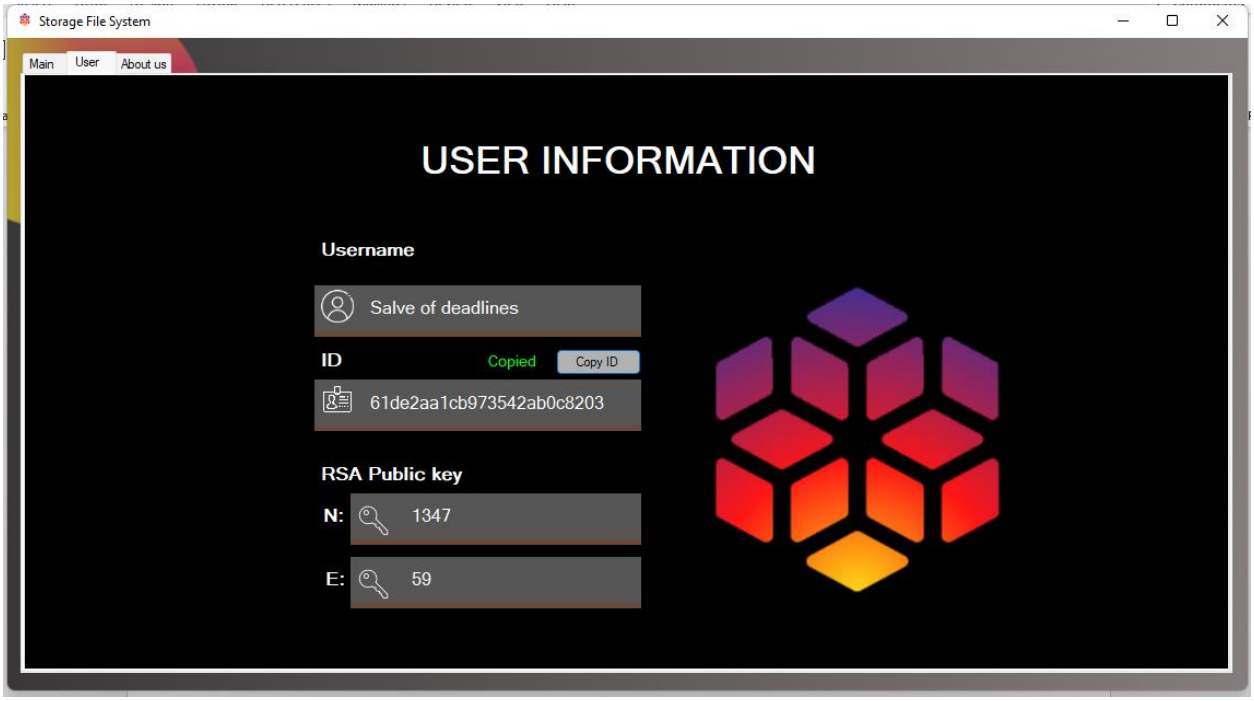
- Nếu muốn giải mã ảnh của mình trên ứng dụng, người dùng sẽ nhấn vào nút “Decrypt Image”, và sẽ nhập Private key của người dùng đẻ có thể tiến hành giải mã được ảnh.
- Lưu ý: Khi giải mã ảnh, ứng dụng sẽ giải mã tất cả các ảnh có trong tài khoản của người dùng và nếu private key sai thì ảnh sẽ bị giải mã sai

6. Chia sẻ ảnh thông qua ID của người dùng:

- Người dùng có thể chia sẽ hình ảnh cho nhau thông qua ID bằng cách nhấn vào nút “SHARE” trên màn hình chính, sau đó nhập ID của người muốn chia sẻ tới
- Người dùng có thể xem ID của mình bằng cách nhấn vào nút “USER” ở góc trái màn hình để vào được phần User Information. Người dùng cũng có thể copy ID của mình để gửi cho người dùng khác, không cần phải ghi nhớ từng ký tự trong ID.



7. Xem thông tin và copy ID



IV) Restful API:

1. Đăng ký:

POST: https://slave-of-deadlines.herokuapp.com/customers/register

```
{
    “fullname”: “Slave of deadlines”,
    “username”: “SOD”,
    “password”: “MHMM_RSA”,
    “e”: 1347,
    “n”: 59
}
```

2. Đăng nhập

POST: https://slave-of-deadlines.herokuapp.com/customers/login

```
{
    “username”: “SOD”,
    “password”: “MHMM_RSA”
}
```

3. Kiểm tra public key

GET: https://slave-of-deadlines.herokuapp.com/customers/username/:username

4. Kiểm tra username

GET: https://slave-of-deadlines.herokuapp.com/customers/publickey/:e&n

5. Nhận public key bằng ID user

GET: https://slave-of-deadlines.herokuapp.com/customers/pubkey/:id

6. Upload một ảnh

POST: https://slave-of-deadlines.herokuapp.com/photos/one

```
{
    “id”: ”6fasfsa6fasf89”,
    “urlFile”:
“https://res.cloudinary.com/cryption/image/upload/v1641934100/hahlil0l5vxqshoahwbd.png”
}
```

7. Upload nhiều ảnh

POST: https://slave-of-deadlines.herokuapp.com/photos/multiple

```
[
    {
        “id”: ”6fasfsa6fasf89”,
        “urlFile”:
“https://res.cloudinary.com/cryption/image/upload/v1641934100/hahlil0l5vxqshoahwbd.png”
    },
    {
        “id”: ”6fasfsa6fasf8g9”,
        “urlFile”:
“https://res.cloudinary.com/cryption/image/upload/v1641934100/hahlil0l5vxqshoahwbd.png”
    }
]
```

8. **Lấy ảnh bằng ID user**

GET: https://slave-of-deadlines.herokuapp.com/photos/:id

V) Các ưu điểm và khuyết điểm của ứng dụng

✓ Ưu điểm:

- Bảo mật, vì mỗi người dùng có 1 account riêng và có 1 khóa riêng, không ai trùng ai
- Ảnh được mã hóa khá ở client, bên server chỉ lưu ảnh mã hóa -> đảm bảo dữ liệu của người dùng
- App có giao diện bắt mắt, hấp dẫn

❖ Khuyết điểm:

- Ứng dụng chỉ mã hóa và giải mã với key n < 65000, với key lớn hơn thì mã hóa được nhưng không giải mã ngược được
- App load khá lâu (vì server free nên tốc độ chậm -> mỗi lần fetch dữ liệu từ server tốn thời gian)
- Không có chức năng Reset password khi người dùng quên tài khoản
- Khi 1 user khác gửi ảnh cho 1 user, bên user nhận sẽ không nhận được thông báo và app không tự reload khi có hành động send ảnh qua lại giữa các user. Vì vậy user nhận phải nhấn nút Reload thì mới cập nhập lại app
- Mất khá nhiều thời gian để mã hóa ảnh.
- Không xử lý được key có giá trị lớn.

VI) CODE:

Server: (Đã được deplopy lên heroku nên không cần phải chạy)
https://github.com/wander23/Server-MHMM

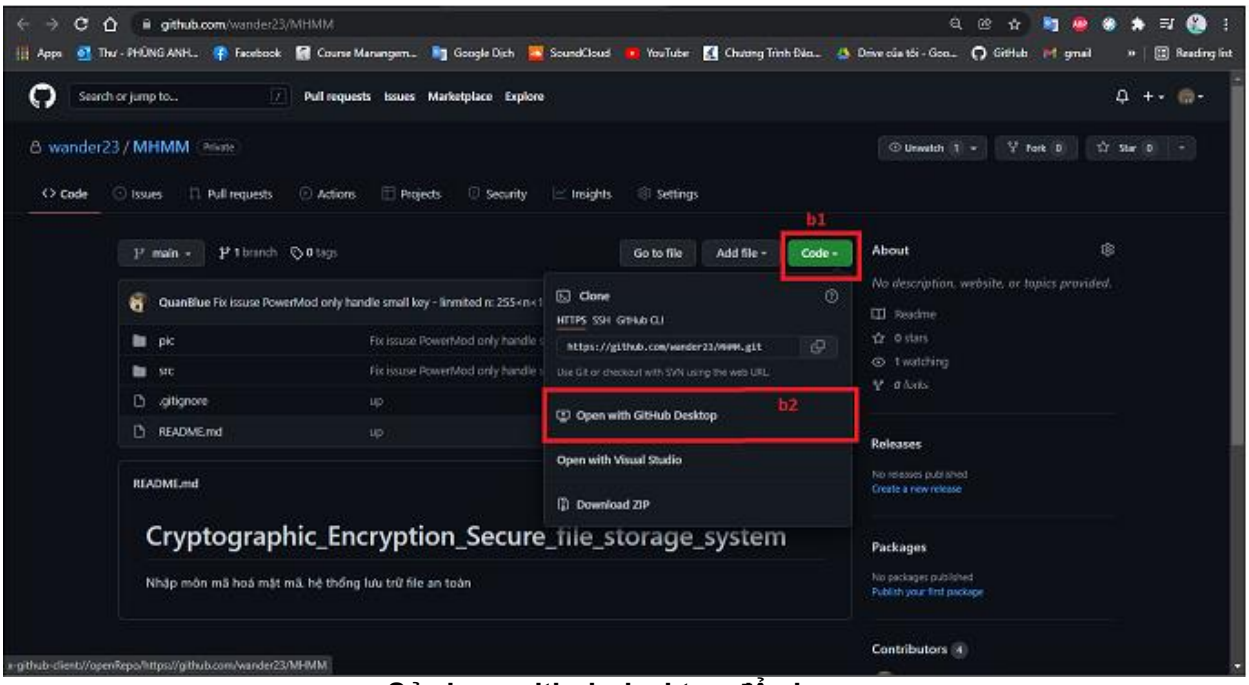
Client:
https://github.com/wander23/MHMM

Video Demo:
https://youtu.be/Y2-wst1yYg0

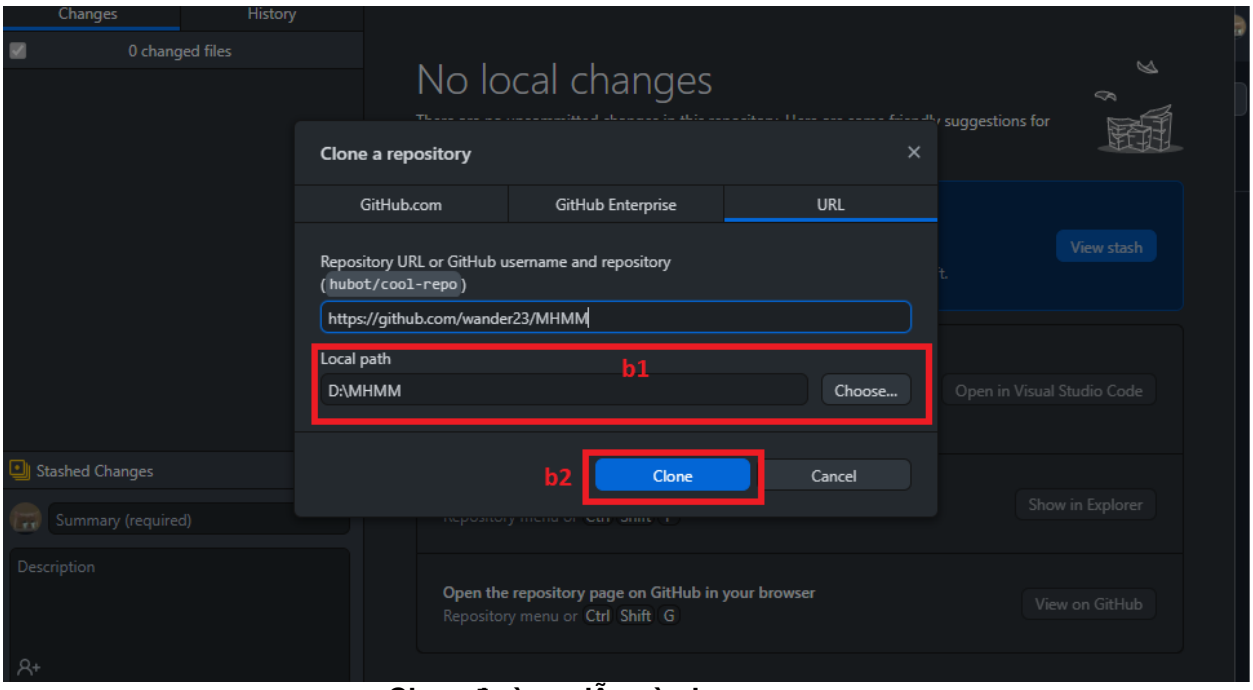
VII) Hướng dẫn biên dịch:

1/ Bấm vào link client github: https://github.com/wander23/MHMM

2/ Clone code

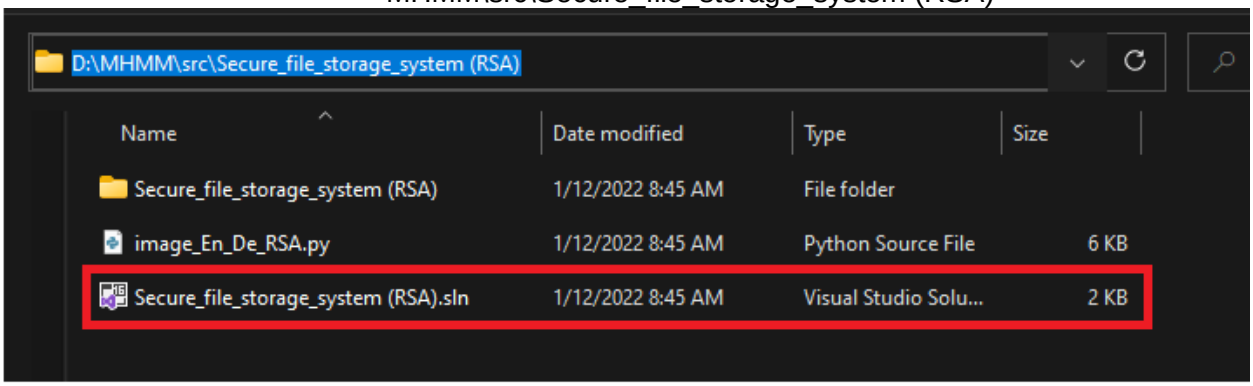


Sử dụng github desktop để clone



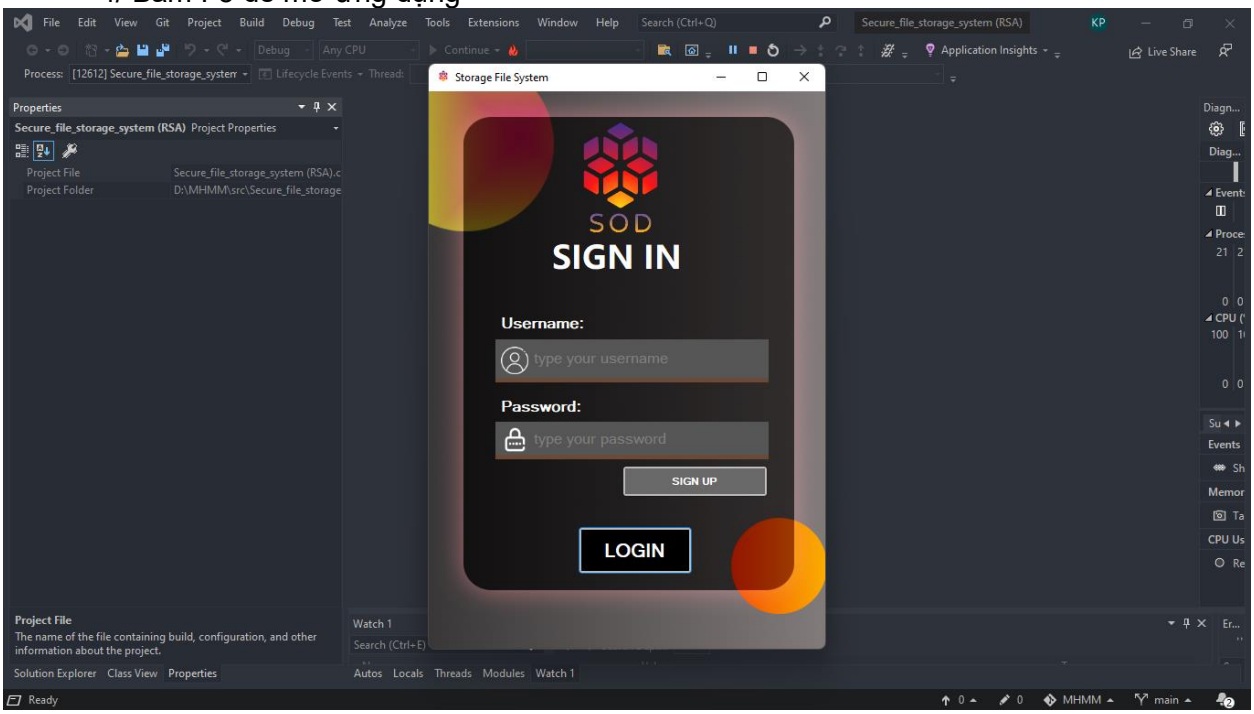
Chọn đường dẫn và clone

3/ Vào thư mục theo đường dẫn đã clone trước đó và vào theo thứ tự:
MHMM\src\Secure_file_storage_system (RSA)



Bật solution

4/ Bấm F5 để mở ứng dụng



Lưu ý: Nếu gặp lỗi không sử dụng được các dependency sau thì update hoặc reinstall