**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN – ĐHQG HCM**

**KHOA CÔNG NGHÊ THÔNG TIN**

Logo

Description automatically generated

**ĐỒ ÁN 1**

**Môn: An ninh máy tính**

***Chủ đề:***

*Ứng dụng chia sẻ file an toàn*

Tp. Hồ Chí Minh – 07/2022

Thành viên

|  |  |  |
| --- | --- | --- |
| Mã sô sinh viên | Họ và tên | Email |
| 19127392 | Tô Gia Hảo | 19127392@student.hcmus.edu.vn |
| 19127525 | Nguyễn Thanh Quân (Trưởng nhóm) | 19127525@student.hcmus.edu.vn |
| 19127625 | Lâm Chí Văn | 19127625@student.hcmus.edu.vn |

Phân công

|  |  |  |
| --- | --- | --- |
| Tên | Công việc | Đánh giá |
| Tô Gia Hảo | Task 6: Ký trên tập tin | 100% |
|  | Task 7: Xác nhận chữ kí | 100% |
| Nguyễn Thanh Quân | Xây dựng giao diện cho ứng dụng (full responsive) | 100% |
|  | Kết nối MongoDB vào dự án | 100% |
|  | Task 1: đăng kí tài khoản người dùng | 100% |
|  | Task 2: Phát sinh cặp khóa bất đối xứng | 100% |
|  | Task 3: Cập nhập thông tin tài khoản | 100% |
| Lâm Chí Văn | Task 4: Mã hóa tập tin |  |
|  | Task 5: Giải mã tập tin |  |

Tổng quan:

Vì ứng dụng chuyên về Cryptography nên nhóm quyết định dùng Python để viết ứng dụng, vì Python có khá nhiều thư viện hỗ trợ cho Cryptography

Thay vì chỉ làm giao diện Console, nhóm quyết định làm GUI cho dễ tương tác và sinh động vì vậy chọn Flask sẽ làm Framework để viết App

Giao diện GUI: HTML, CSS, JavaScript và Bootstrap 5

Cơ sở dữ liệu: MongoDB

Task 1: Đăng kí tài khoản người dùng

Graphical user interface

Description automatically generated

Hình 1: Giao diện đăng kí tài khoản

Khi đăng kí, chương trình bắt buộc nhập đầy đủ thông tin vào các trường có trong form Đăng kí

Với trường hợp đăng kí bằng gmail đã đăng kí trước đó, chương trình sẽ thông báo lỗi

Graphical user interface, text, application, chat or text message, website

Description automatically generated

Hình 2: lỗi đăng kí trùng tài khoản đã tồn tại

Mật khẩu được lưu trữ dưới hash có kết hợp salt nên mật khẩu sẽ có dạng

Salt = random(32 byte)

Hash\_password = hash(password + salt)

Mật khẩu được lưu vào database sẽ có dạng: passphase = salt + hash\_password

Vì mật khẩu được lưu vào database với cấu trúc như trên nên không cần phải lưu salt vào database.

Khi authen chỉ cần tách passphase ra thì ta sẽ có được salt và hash\_password

salt = passphase[:32]

hash\_password = passphase[32:]

Với việc tách như vậy, khi authen chỉ cần so sánh input\_passphase với passphase (tương đương với email input) trong database

Với input\_passphase = hash(input\_password + salt) (salt này mới lấy từ passphase lưu trong database ở trên)

Graphical user interface, application

Description automatically generated

Hình 2: Giao diện đăng nhập

Task 2: Phát sinh cặp khóa bất đối xứng

phát sinh một cặp khoá(Kpublic, Kprivate)có độ dài là 2048 bit

Vì vậy sử dụng thư viện Crypto để generate cặp khóa có độ dài 2048 bit và lấy cặp khóa dưới dạng PEM

Text

Description automatically generated

Hình 3: hàm phát sinh cặp khóa RSA

Sau khi có được cặp khóa RSA dưới dạng PEM, tiến hành mã hóa Khóa bí mật bằng thuật toán AES với mode CBC với nội dung là Khóa bí mật và Ksecret trong thuật toán AES là passphase của người dùng

Text

Description automatically generated

Hình 4: Mã hóa Khóa bí mật RSA bằng thuật toán AES

Sau khi có tạo cặp khóa và mã hóa khóa bí mật, ta lưu khóa công khai (PEM) và khóa bí mật đã được mã hóa vào database dưới dạng byte

Text

Description automatically generated

Hình 5: hàm xử lí cặp khóa RSA vừa phát sinh ở Hình 3

Task 3: cập nhập thông tin tài khoản

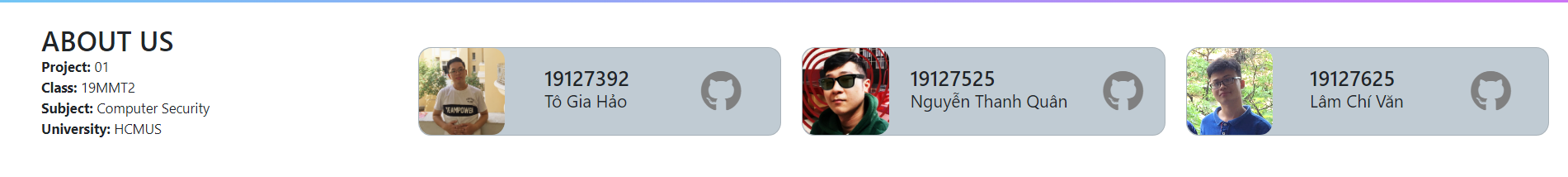
Sau khi đăng nhập thành công, chương trình sẽ lưu thông tin người dùng vào Session sau đó sẽ tự động redirect về trang chủ

Việc lưu thông tin người dùng vào Session sẽ phục vụ cho việc Authorize (nếu có session user -> người dùng đã đăng nhập thành công, ngược lại: người dùng chưa đăng nhập)

Khi nhấn vào nút Logout, chương trình sẽ tự động xóa session user, để đảm bảo tránh trường hợp không đăng nhập vẫn vào được trang chủ

Graphical user interface, application

Description automatically generated



Hình 6: Giao diện trang chủ

Nhấn nút Change Information để thay đổi thông tin người dùng

Graphical user interface, application

Description automatically generated

Hình 7: Popup thay đổi thông tin chi tiết

Khi thay đổi các thông tin (trừ password) ứng dụng sẽ cập nhập thẳng lên database mà không cần xử lí

Khi thay đổi password, chương trình sẽ:

Tạo lại passphase mới như Task 1 với new\_passphase = new\_salt + hash\_new\_password

Giải mã khóa bí mật (đã được mã hóa) với khóa Ksecrect là passphase cũ (line 14)

Sau đó mã hóa lại khóa bí mật bằng thuật toán AES với Ksecret là passphase vừa tạo

Text

Description automatically generated

Hình 8: Quá trình xử lý, thay đổi passphase mới

Task 6:

Graphical user interface, text, application, chat or text message

Description automatically generated

Hình 9: Giao diện Sign on File

Task 7:

Graphical user interface, text, application, chat or text message

Description automatically generated