ĐỒ ÁN 1 - MÔN HỌC AN NINH MÁY TÍNH

I. QUY ĐỊNH CHUNG

- Mỗi **nhóm gồm 3 sinh viên**.
- **Phân công rõ ràng**: trong báo cáo cần ghi rõ vai trò chính của từng bạn (VD: bạn A phụ trách giao diện và tính năng đăng ký; bạn B phụ trách mã hóa AES & QR code; bạn C lo xử lý OTP, MFA và log).
- **Ngôn ngữ lập trình**: tự chọn, khuyến khích Python, Java, hoặc C#. Yêu cầu viết code rõ ràng, có chú thích. Hạn chế copy-paste thư viện không rõ nguồn gốc.
- **Giao diện**: Có thể dùng console nhưng **khuyến khích có GUI** để dễ thao tác và trình bày demo.
- **Cấu trúc thư mục bài nộp**:

```
/MSSV1_MSSV2_MSSV3
  ├── main.py (hoặc main.java, main.cs)
  ├── modules/ (các chức năng chia theo mô-đun)
  ├── gui/ (giao diện nếu có)
  ├── data/ (file test, QR code, sig,...)
  ├── report/ (báo cáo .pdf hoặc .docx)
  └── README.md (hướng dẫn cài đặt & chạy thử)
```

II. MỤC TIÊU

Mục tiêu của đồ án là xây dựng **một ứng dụng mô phỏng hệ thống bảo mật** theo mô hình thực tế: người dùng đăng ký, tạo khoá, mã hoá & ký tập tin, xác thực đa yếu tố, và quản trị tài khoản. Các bước cần thực hiện bao gồm:

1. **Phân tích yêu cầu & chia nhóm công việc**.
2. **Thiết kế kiến trúc tổng thể**: modules, UI, dữ liệu lưu trữ (JSON/XML hay CSDL).
3. **Chọn và tích hợp thư viện mã hoá an toàn** (không tự viết thuật toán từ đầu).
4. **Phát triển từng chức năng theo checklist mục III bên dưới**.
5. **Kiểm thử từng phần & kết hợp thành hệ thống hoàn chỉnh**.
6. **Viết báo cáo tổng kết quá trình & demo**.
7. **Thực hiện vấn đáp bảo vệ đồ án**.

III. CÁC CHỨC NĂNG BẮT BUỘC

Lưu ý: Hệ thống cần có menu hoặc giao diện rõ ràng để truy cập các chức năng sau khi đăng nhập thành công. Với mỗi chức năng:

- **Phải có kiểm thử rõ ràng**: file test, ảnh chụp giao diện, log đầu ra.
- **Ghi chú chi tiết trong báo cáo**: từng chức năng đã dùng thuật toán/bảo mật gì.

1. **Đăng ký tài khoản người dùng**

Yêu cầu:

- Form nhập thông tin: Email, họ tên, ngày sinh, số điện thoại, địa chỉ, passphrase.
- Tự động kiểm tra:
  - Email không được trùng với tài khoản đã có.
  - Passphrase phải có độ mạnh tối thiểu (gợi ý: ít nhất 8 ký tự, có chữ hoa, số, ký hiệu).
- Tạo Salt ngẫu nhiên → kết hợp với passphrase → hash bằng SHA-256 → lưu xuống.
- Lưu thông tin vào: `users.json`, `users.xml`, hoặc bảng `users` trong CSDL.

Kiểm chứng:

- Ảnh chụp giao diện đăng ký.
- Log ghi lại hành động đăng ký.
- File `users.json` (hoặc CSDL) có dữ liệu đúng định dạng.

2. **Đăng nhập & Xác thực đa yếu tố (MFA)**

Yêu cầu:

- Nhập email và passphrase → kiểm tra hash (với Salt tương ứng).
- Nếu đúng → gửi mã OTP hoặc sinh mã TOTP (dùng `pyotp`, `qrcode`).
- Gửi OTP:
  - Sinh mã ngẫu nhiên 6 chữ số.
  - Lưu thời gian tạo và hết hạn.
  - Gửi qua email (SMTP hoặc giả lập).
- Nếu người dùng chọn TOTP:
  - Hiển thị QR để quét vào Google Authenticator.
  - Kiểm tra mã 6 chữ số do ứng dụng sinh ra.

Kiểm chứng:

- Chụp màn hình xác thực MFA.
- Log ghi thời gian tạo OTP, email đã gửi.

3. **Quản lý khoá RSA cá nhân**

Yêu cầu:

- Sau khi đăng ký thành công, người dùng có thể:
  - Tạo cặp khoá RSA (2048 bit).
  - Private Key → mã hoá bằng AES (key AES sinh từ passphrase).
  - Public Key → lưu công khai, gắn với email, ngày tạo.

    - Tự động đặt hạn dùng 90 ngày.

Kiểm chứng:

- Lưu file `.pem` hoặc chuỗi base64.
- Có chức năng "xem trạng thái khóa" hiển thị hạn dùng.

4. **QR Code Public Key**

Yêu cầu:

- Tạo mã QR chứa: email, ngày tạo, public key (mã hóa base64 hoặc nén).
- Lưu QR code thành file `.png`.
- Có chức năng đọc QR từ file → hiển thị thông tin, lưu vào danh sách public key.

Kiểm chứng:

- Ảnh mã QR.
- Test quét mã từ ảnh.

5. **Cập nhật thông tin tài khoản**

Yêu cầu:

- Cho phép sửa họ tên, ngày sinh, địa chỉ, SĐT.
- Cho phép đổi passphrase:
    - Nếu thay đổi → phải giải mã private key → mã hóa lại bằng AES mới.
    - Đảm bảo không mất dữ liệu khóa.

Kiểm chứng:

- Giao diện chỉnh sửa thông tin.
- Log thay đổi passphrase (không lưu pass cũ/mới).

6. **Mã hoá tập tin gửi người khác**

Yêu cầu:

- Chọn file + email người nhận (phải có public key).
- Hệ thống:
    - Sinh Ksession (AES key).
    - Mã hoá file bằng AES (CBC hoặc GCM).
    - Mã hoá Ksession bằng RSA public key người nhận.
    - Gắn metadata đã mã hoá (người gửi, tên file, timestamp).
    - Tùy chọn lưu: gộp thành 1 file `.enc`, hoặc tách file `.key`.

Kiểm chứng:

- File mã hoá.
- Metadata đúng định dạng.
- Log hành vi.

7. **Giải mã tập tin**

Yêu cầu:

- Chọn file `.enc` (và `.key` nếu tách riêng).
- Dùng passphrase → giải mã private key → giải mã Ksession → giải mã file.
- Hiển thị kết quả, metadata, lưu file gốc.

Kiểm chứng:

- So sánh nội dung trước/sau mã hoá.
- Log chi tiết.

8. **Ký số tập tin**

Yêu cầu:

- Chọn file bất kỳ → băm SHA-256 → ký bằng private key → tạo file chữ ký `.sig`.

Kiểm chứng:

- File `.sig` rõ định dạng.
- Có log người ký, thời gian.

9. **Xác minh chữ ký**

Yêu cầu:

- Chọn file và `.sig`.
- Kiểm tra chữ ký với tất cả public key đã lưu.
- Nếu hợp lệ → hiển thị email người ký, ngày ký.
- Nếu sai → thông báo rõ.

Kiểm chứng:

- Trường hợp đúng/sai.
- Log xác minh.

10. **Phân quyền tài khoản**

Yêu cầu:

- Có cờ `role` trong dữ liệu user (`admin` hoặc `user`).

- `admin` có thể:
    - Xem danh sách tài khoản.
    - Khoá/mở tài khoản.
    - Xem log hoạt động toàn hệ thống.

Kiểm chứng:

- Giao diện quản trị.
- Log thao tác của admin.

11. **Ghi log bảo mật**

Yêu cầu:

- Ghi vào file `security.log` hoặc bảng `log_activity`.
- Ghi chi tiết: thời gian, email, hành động, trạng thái (thành công/thất bại).

Kiểm chứng:

- Trích log file.

12. **Chia nhỏ tập tin lớn**

Yêu cầu:

- Nếu file > 5MB → chia block 1MB → mã hoá từng block.
- Sử dụng AES-GCM để đảm bảo toàn vẹn.
- Gộp lại thành 1 file `.enc`.

Kiểm chứng:

- So sánh dung lượng, block đã mã hoá.
- Log tiến trình.

13. **Kiểm tra trạng thái khoá**

Yêu cầu:

- Hiển thị:
    - Ngày tạo.
    - Hạn dùng.
    - Trạng thái: Còn hạn / Gần hết hạn / Hết hạn.
- Cho phép tạo mới hoặc gia hạn.

Kiểm chứng:

- Giao diện quản lý khóa.

- Test tự động thay đổi trạng thái.

14. **Tìm kiếm public key**

Yêu cầu:

- Cho phép nhập email → tìm public key nếu có.
- Hiển thị kết quả: email, QR code, ngày tạo, thời hạn còn lại.

Kiểm chứng:

- Giao diện tìm kiếm.
- Test tìm không có → thông báo rõ ràng.

15. **Giới hạn đăng nhập**

Yêu cầu:

- Sau 5 lần nhập sai → tài khoản bị khoá 5 phút.
- Ghi log mỗi lần sai.
- Hiển thị thời gian chờ.

Kiểm chứng:

- Test cố tình nhập sai.
- Log số lần sai.

16. **Tùy chọn định dạng lưu file**

Yêu cầu:

- Cho phép chọn:
    - Gộp `.enc` và khóa vào 1 file.
    - Tách `.enc` + `.key`.
- Khi giải mã → tự động nhận diện định dạng.

Kiểm chứng:

- Test cả 2 dạng.

17. **Khôi phục tài khoản**

Yêu cầu:

- Khi đăng ký → cho phép tạo mã khôi phục (hiển thị 1 lần).
- Khi quên passphrase:
    - Nhập mã khôi phục → cho đổi passphrase mới.

- Phải giải mã lại private key → mã hoá bằng passphrase mới.

Kiểm chứng:

- Tình huống test quên mật khẩu.
- Mã khôi phục chỉ hiển thị 1 lần.

IV. GỢI Ý CÔNG NGHỆ

Nhiệm vụ	Công nghệ cụ thể
Lưu trữ user/pass, khóa	`sqlite3` hoặc `JSON` lưu SHA-256(pass + salt), public/private key
Tạo OTP, TOTP	`pyotp` (Python), hoặc dùng web API gửi mail OTP
QR code	`qrcode.make()` trong Python; `ZXing` (Java), `QRCoder` (C#)
AES (CBC, GCM), RSA	`cryptography.fernet`, `pycryptodome`, hoặc `BouncyCastle`
GUI	`Tkinter`, `PyQt5`, `JavaFX`, hoặc `WPF`
Log	Ghi vào file `.log`, hoặc bảng `log_activity` trong CSDL

V. BÀI NỘP GỒM

- **Mã nguồn đầy đủ**, chia thành module, có `README.md` ghi rõ cách chạy.
- **Báo cáo chi tiết**:
  - Tên nhóm, MSSV, email, phân công cụ thể
  - Mô tả sơ đồ kiến trúc hệ thống (dạng hình vẽ hoặc bảng)
  - Giải thích các kỹ thuật bảo mật đã dùng cho từng phần
  - Ảnh chụp giao diện, ví dụ test: QR code, file mã hoá, chữ ký...
- **Video demo** (không bắt buộc nhưng được khuyến khích, 3–5 phút, trình bày quy trình sử dụng).
- **File dữ liệu test**:
  - File `.enc`, `.key`, `.sig`
  - QR code mẫu
  - Public/private key test
  - Log hoạt động

VI. THANG ĐIỂM ĐÁNH GIÁ

STT	Chức năng	Tiêu chí đánh giá chi tiết	Điểm tối đa (trên 10)	Điểm đạt	Ghi chú
1	Đăng ký tài khoản người dùng	Đầy đủ thông tin, hash passphrase đúng, lưu dữ liệu	1.0		
2	Đăng nhập & xác thực đa yếu tố (MFA)	Đăng nhập chính xác, OTP/TOTP đúng, giới hạn thời gian	1.2		

STT	Chức năng	Tiêu chí đánh giá chi tiết	Điểm tối đa (trên 10)	Điểm đạt	Ghi chú
3	Quản lý khoá RSA cá nhân	Tạo khoá, mã hoá private key, lưu public key, gia hạn khoá	1.2		
4	Tạo và quét QR Code cho public key	Tạo QR đúng, lưu ảnh, quét và lưu dữ liệu chính xác	0.9		
5	Cập nhật thông tin tài khoản	Sửa thông tin, đổi passphrase bảo toàn khoá AES	0.9		
6	Mã hóa tập tin gửi người khác	Sinh khoá AES phiên, mã hoá file & khoá đúng, metadata	1.2		
7	Giải mã tập tin	Giải mã private key, khoá phiên, file chính xác	1.2		
8	Ký số tập tin	Hash SHA-256, ký RSA đúng, tạo file chữ ký	0.9		
9	Xác minh chữ ký	Xác minh đúng, thông báo chính xác người ký	0.9		
10	Phân quyền tài khoản	Phân quyền đúng, admin có quyền quản lý	0.9		
11	Ghi log bảo mật	Ghi log đầy đủ các sự kiện quan trọng	0.7		
12	Chia nhỏ tập tin lớn khi mã hóa	Chia block, mã hoá AES-GCM đảm bảo toàn vẹn	0.7		
13	Kiểm tra trạng thái khóa	Hiển thị, gia hạn, tạo mới khoá đúng	0.7		
14	Tìm kiếm public key	Tìm kiếm chính xác, hiển thị đủ thông tin	0.6		
15	Giới hạn số lần đăng nhập	Khoá tài khoản đúng khi đăng nhập sai	0.6		
16	Tùy chọn định dạng lưu file	Lưu file gộp/tách đúng, tự nhận diện giải mã	0.6		
17	Khôi phục tài khoản	Tạo mã khôi phục, dùng mã để đổi passphrase	0.7		
	Báo cáo & tài liệu	Phân công rõ, mô tả kỹ thuật, ảnh minh hoạ, file README	0.8		
	Demo (Video/Thuyết trình)	Trình bày rõ ràng, demo chức năng chính	0.6		