Kho dữ liệu an toàn

-T07-

(Secure Vault: Securely store and deliver client data (sensitive data) in the Android environment

Nhân viên thực hiện:

Đỗ Minh Đức

Hoàng Minh Đức

Cao Đăng Đạt

Mentor hướng dẫn:

Nguyễn Quốc Đạt

Mục lục

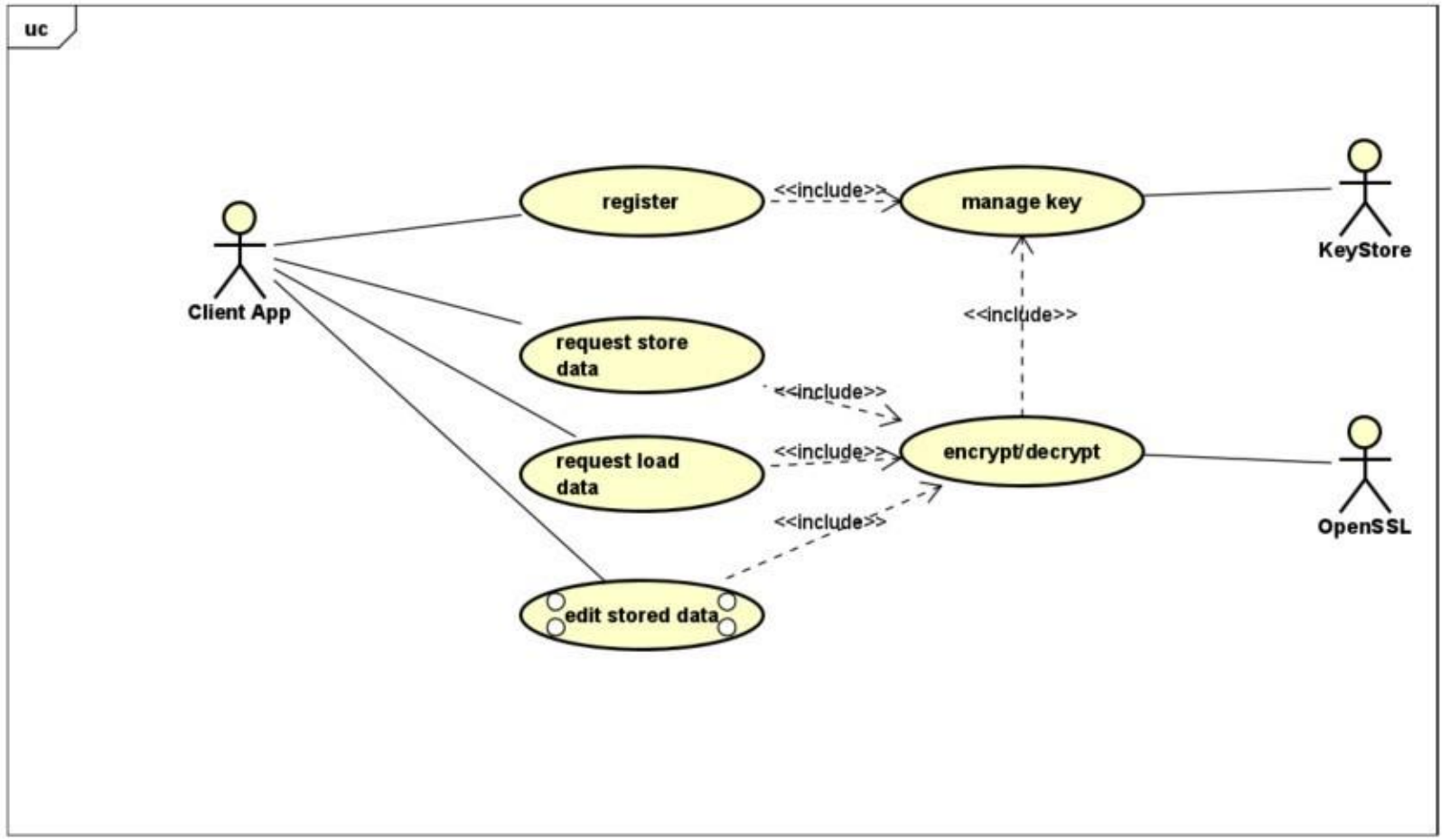
1. Đặt vấn đề
2. Phát biểu bài toán
3. Phân tích thiết kế
4. Phương pháp thực hiện
5. Kết quả thực nghiệm
6. Demo
7. Kết luận

1. Đặt vấn đề

- Môi trường Android đang phát triển nhanh chóng, tạo ra thách thức trong việc bảo vệ dữ liệu.
- Mục tiêu của chúng ta là xây dựng một Kho Dữ liệu An Toàn để lưu trữ và truyền dữ liệu nhạy cảm một cách an toàn và bảo mật.

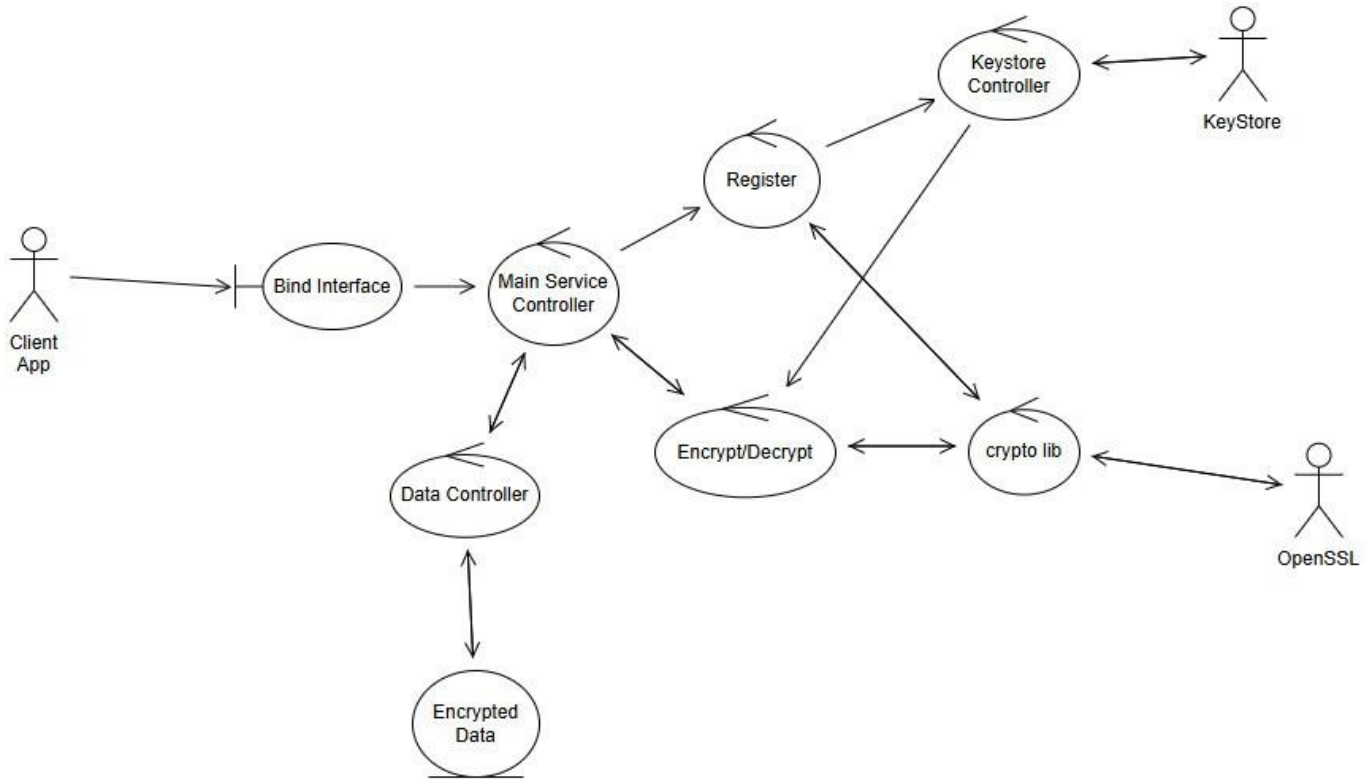
2. Phát biểu bài toán

- Mục tiêu: Xây dựng một dịch vụ lưu trữ an toàn trên Android.
- Chức năng: Dịch vụ sẽ hoạt động như một kho lưu trữ an toàn cho dữ liệu nhạy cảm của người dùng, quản lý khóa, mã hóa, giải mã và lưu trữ dữ liệu theo yêu cầu của ứng dụng.
- Yêu cầu: Đảm bảo tính bảo mật và tính toàn vẹn của dữ liệu trong quá trình lưu trữ và truy cập.



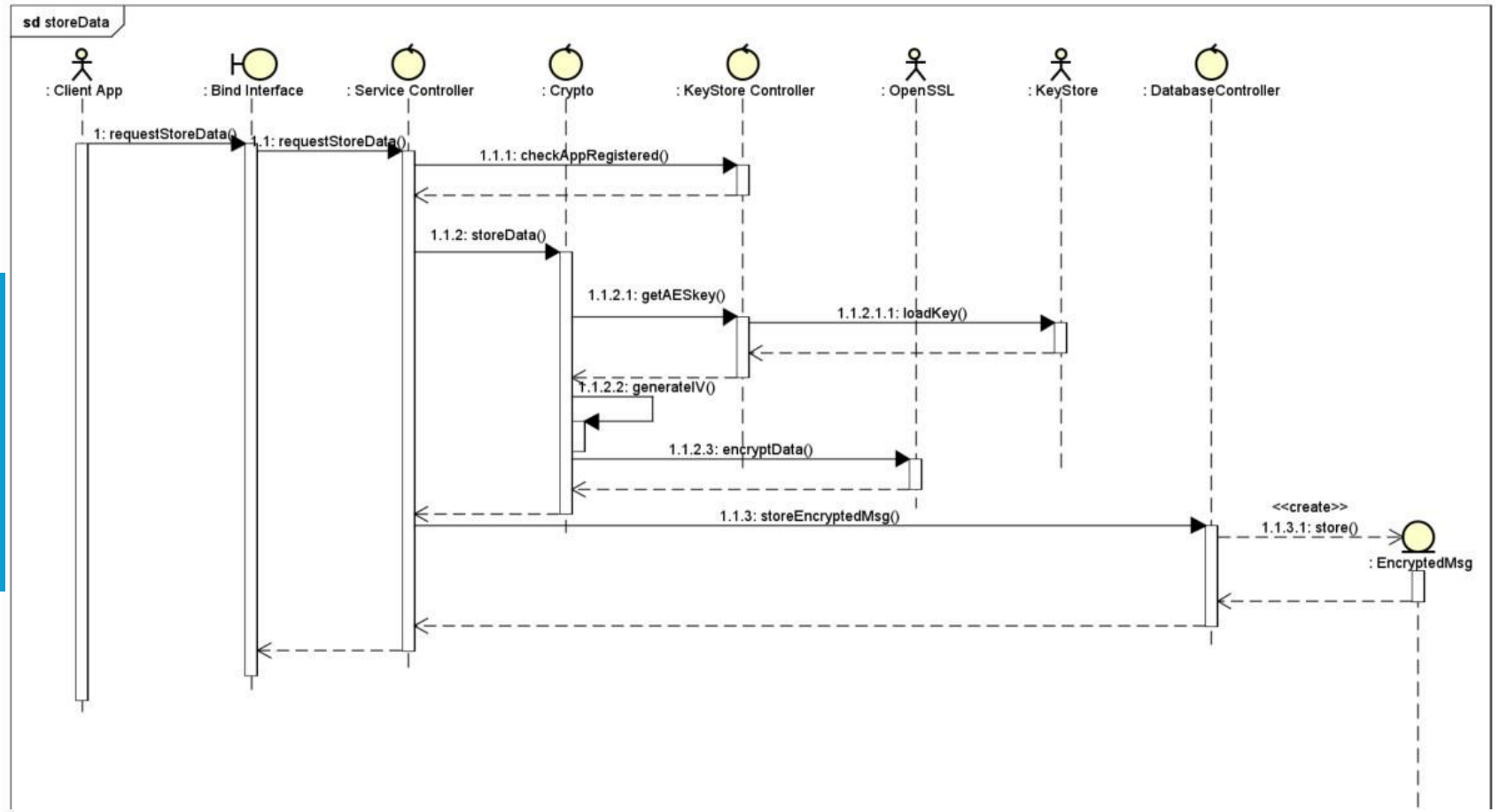
3. Phân tích thiết kế

Hình 3.1. Usecase Tổng quan



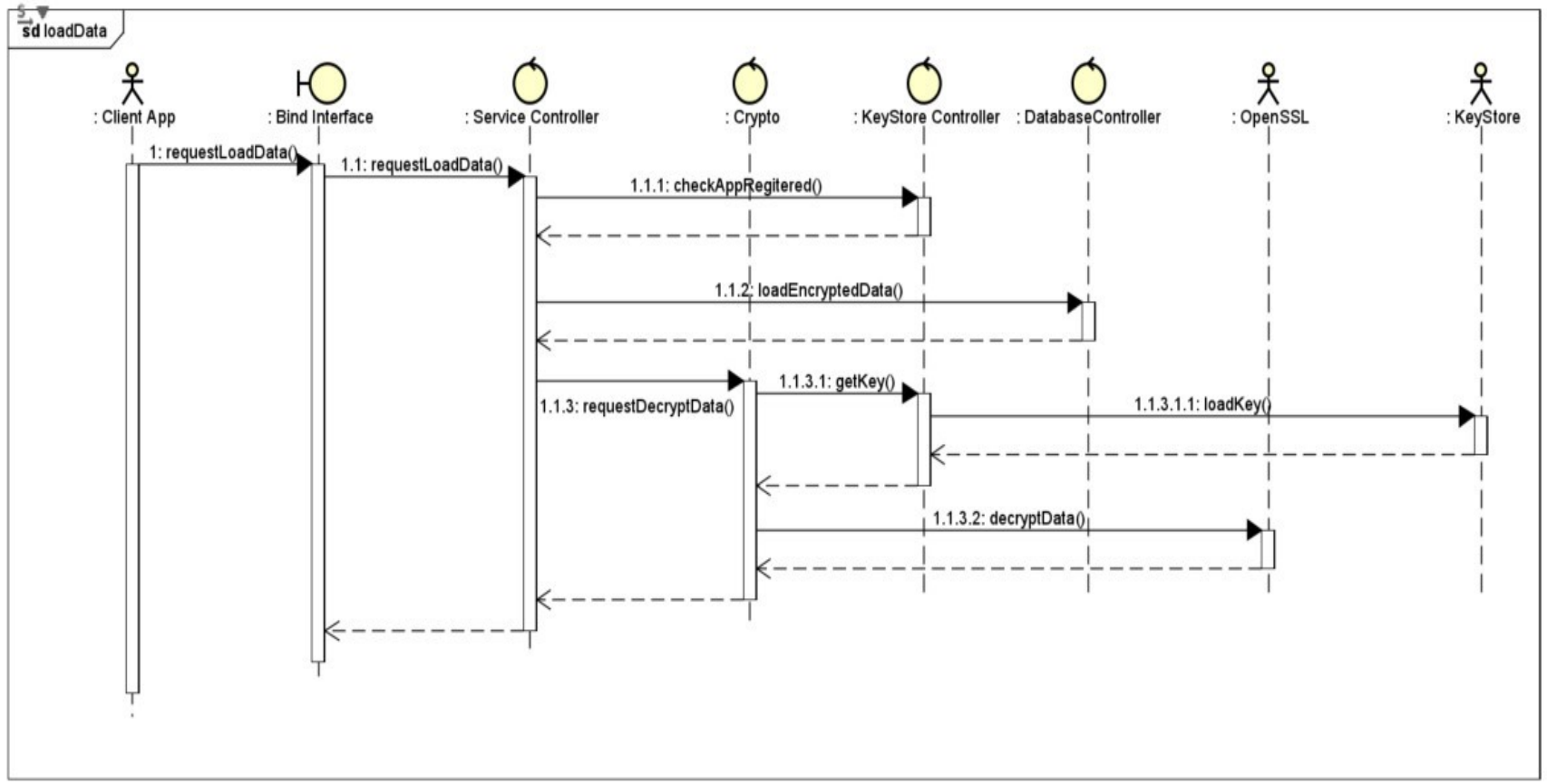
3. Phân tích thiết kế

Hình 3.2. Robustness diagram



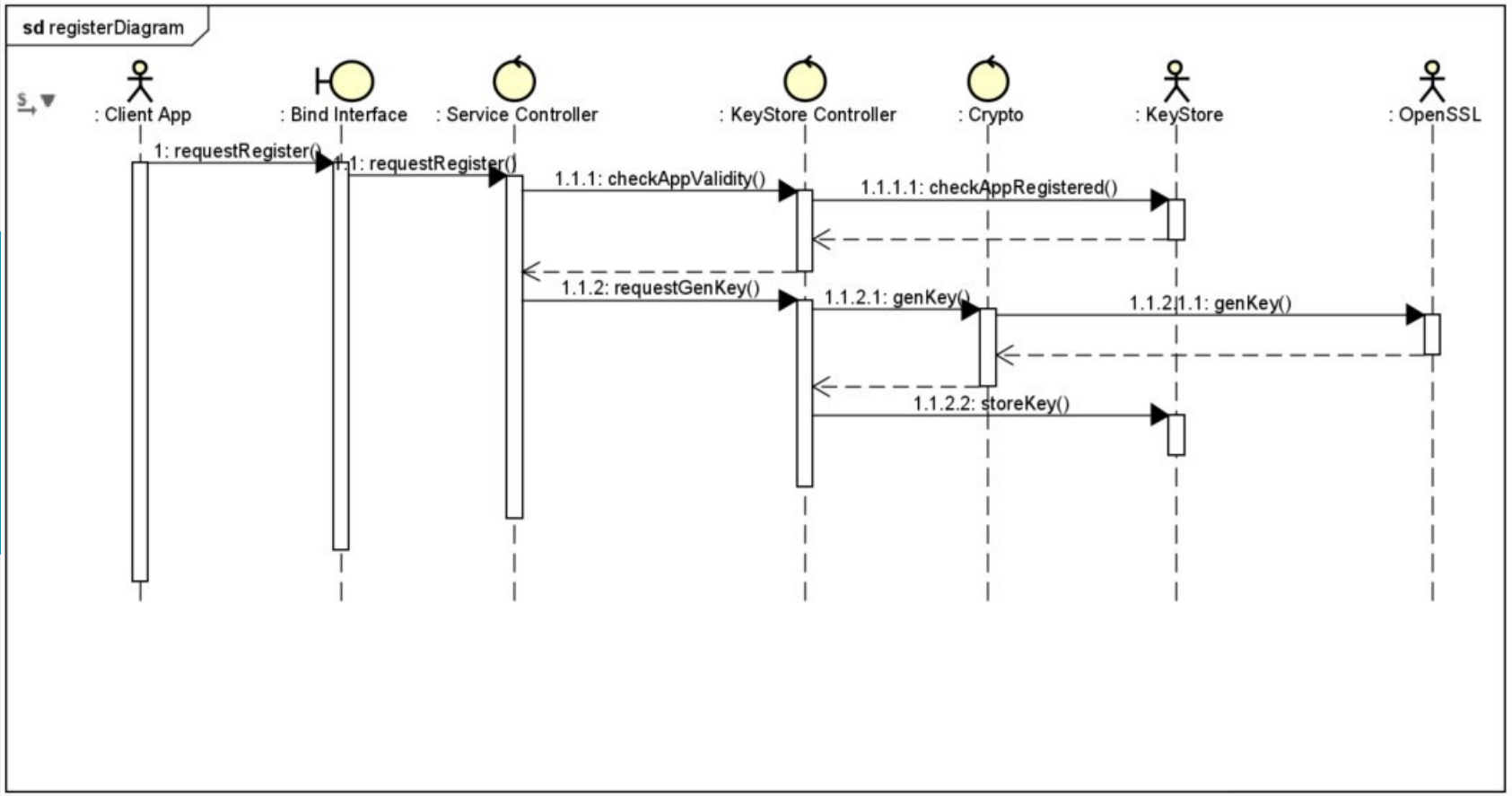
3. Phân tích thiết kế

Hình 3.3. Luồng UC lưu dữ liệu



3. Phân tích thiết kế

Hình 3.4. Luồng UC lấy dữ liệu

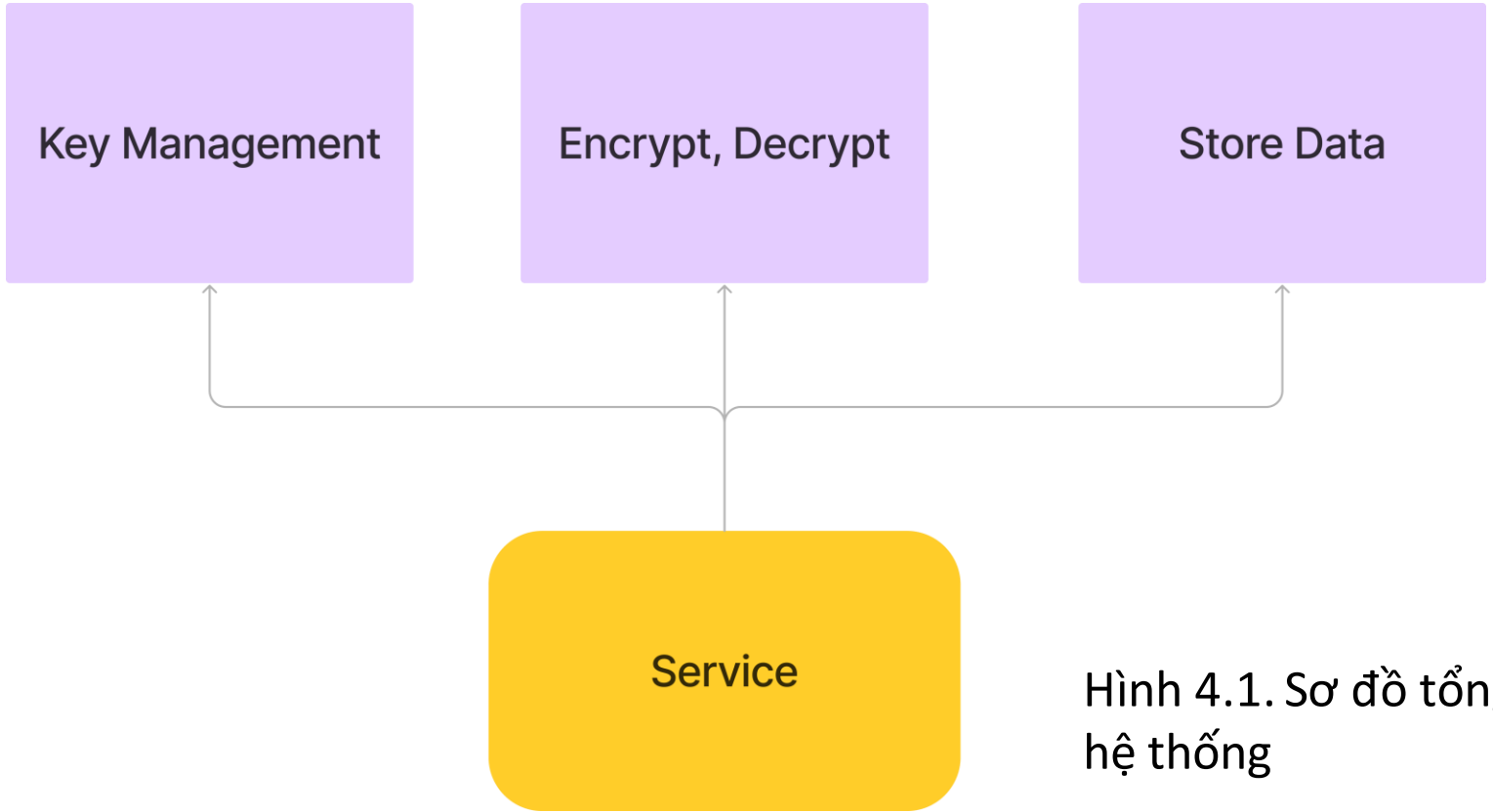


3. Phân tích thiết kế

Hình 3.5. Luồng UC đăng ký

4. Phương pháp thực hiện

A. Tổng quan hệ thống



Hình 4.1. Sơ đồ tổng quan hệ thống

4. Phương pháp thực hiện

- Công nghệ sử dụng: Android Studio, Java Native Interface, Android Keystore
- Ngôn ngữ sử dụng: C++, Kotlin, Java
- Thư viện sử dụng: OpenSSL

4. Phương pháp thực hiện

B. Module quản lý khóa:

Nhiệm vụ:

- Quản lý việc tạo, lưu trữ và quản lý khóa mã hóa cho dữ liệu.

Tính năng:

- Tạo khóa
- Lưu trữ khóa.
- Quản lý quyền truy cập vào khóa.
- Định kỳ thay đổi khoá mã hóa mới

4. Phương pháp thực hiện

C. Module mã hóa, giải mã dữ liệu:

Nhiệm vụ:

- Bảo vệ dữ liệu nhạy cảm bằng cách mã hóa trước khi lưu trữ và giải mã khi cần truy xuất.

Tính năng:

- Mã hóa dữ liệu
- Giải mã dữ liệu
- Quản lý quy trình mã hóa/giải mã

4. Phương pháp thực hiện

C. Module mã hóa, giải mã dữ liệu:

Thuật toán:

AES - GCM mode: 256bit key - 128bit IV

Ưu điểm:

- Dễ triển khai
- Tích hợp xác thực tính toàn vẹn kết hợp mã hoá
- Có hiệu suất cao

4. Phương pháp thực hiện

D. Module quản lý dữ liệu:

Nhiệm vụ:

- Đảm bảo tính bảo mật và tính toàn vẹn của dữ liệu trong quá trình lưu trữ và truy xuất.
- Quản lý quyền truy cập vào dữ liệu để đảm bảo chỉ những người được ủy quyền mới có thể truy xuất và sử dụng dữ liệu.

Tính năng:

- Lưu trữ dữ liệu
- Quản lý quyền truy cập
- Xử lý yêu cầu truy xuất dữ liệu

Phân công kế hoạch

Công việc	Cao Đăng Đạt	Hoàng Minh Đức	Đỗ Minh Đức
Phân tích thiết kế	x	x	
Module quản ký khoá			x
Module mã hoá	x		
Module service		x	
Xây dựng Client App + Demo		x	
Test	x	x	x

5. Kết quả thực nghiệm

ID	Chức năng	Mô tả	Kết quả kì vọng	Kết quả thực tế
1	Đăng ký lưu	Ứng dụng chưa đăng ký lưu dữ liệu, người dùng yêu cầu đăng ký lưu	Đăng ký lưu thành công	Đăng ký lưu thành công
2	Đăng ký lưu	Ứng dụng đã đăng ký lưu, người dùng yêu cầu đăng ký lưu	Đăng ký thất bại, thông báo đã đăng ký	Đăng ký thất bại, thông báo đã đăng ký
3	Lưu dữ liệu	Ứng dụng khách gửi dữ liệu chưa được lưu đến service và yêu cầu lưu	Lưu thành công	Lưu thành công
4	Lưu dữ liệu	Ứng dụng khách gửi dữ liệu đã được lưu đến service và yêu cầu lưu	Lưu thất bại	Lưu thất bại
5	Load dữ liệu	Ứng dụng khách load 1 loại dữ liệu đã được lưu	Load thành công	Load thành công
6	Load dữ liệu	Ứng dụng khách load 1 loại dữ liệu chưa được lưu	Load thất bại	Load thất bại

5. Kết quả thực nghiệm

ID	Chức năng	Mô tả	Kết quả kì vọng	Kết quả thực tế
7	Sửa dữ liệu	Ứng dụng khách gửi dữ liệu và yêu cầu sửa dữ liệu (dữ liệu cần sửa đã được lưu)	Sửa thành công	Sửa thành công
8	Sửa dữ liệu	Ứng dụng khách gửi dữ liệu và yêu cầu sửa dữ liệu (dữ liệu cần sửa chưa được lưu)	Sửa thất bại	Sửa thất bại
9	Xóa dữ liệu	Ứng dụng khách gửi yêu cầu xóa dữ liệu với loại dữ liệu đã được lưu	Xóa thành công	Xóa thành công
10	Xóa dữ liệu	Ứng dụng khách gửi yêu cầu xóa dữ liệu với loại dữ liệu đã được lưu	Xóa thất bại	Xóa thất bại
11	Đổi key định kỳ	Cứ cách 1 khoảng thời gian nhất định, toàn bộ key mã hóa sẽ được làm mới và tất cả các dữ liệu sẽ được mã hóa lại bằng key mới.	Thay đổi key thành công, dữ liệu được mã hóa bằng key mới	Thay đổi key thành công, dữ liệu được mã hóa bằng key mới

5. Kết quả thực nghiệm

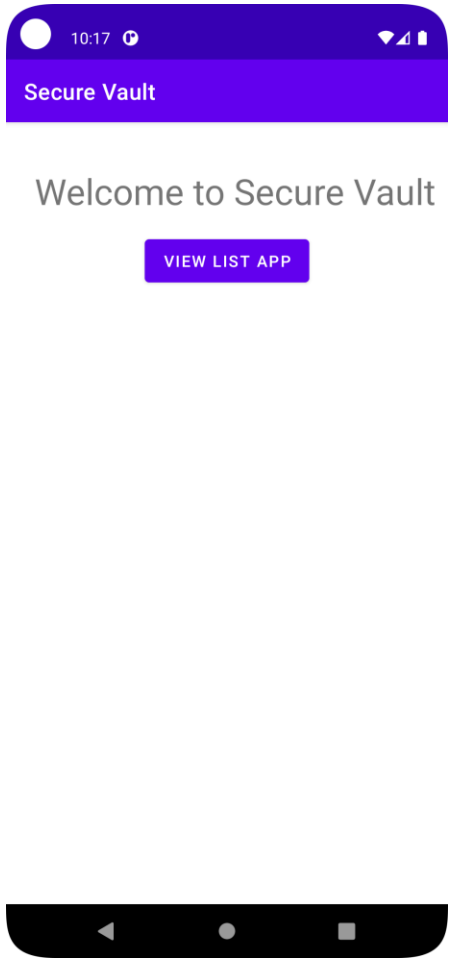
Đánh giá tính bảo mật

- Dữ liệu được mã hoá bằng thuật toán AES GCM với key 256bit

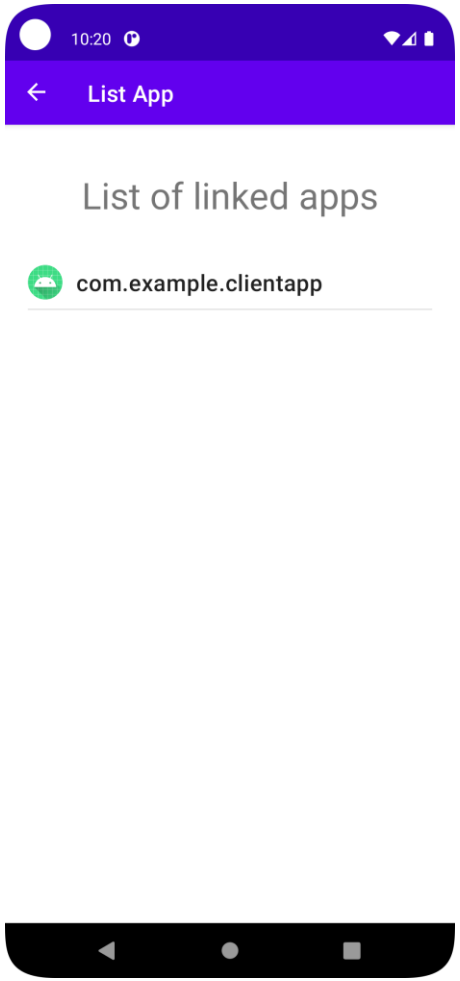
=> Đảm bảo dữ liệu được mã hoá mức độ cao và kiểm tra tính toàn vẹn

- Dữ liệu mã hoá được lưu trong internal storage
- Keystore được lưu ở app-specific external storage

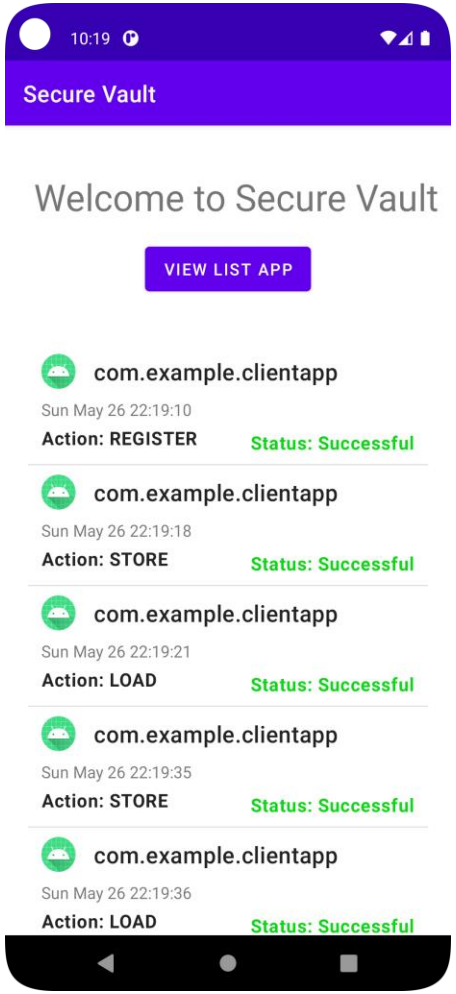
=> Dữ liệu và key không thể truy cập bởi các ứng dụng khác



Hình 5.1: Giao diện trang chủ



Hình 5.2: Giao diện danh mục ứng dụng



Hình 5.3: Giao diện Log

6. Demo

7. Kết luận

Đóng góp:

- Xây dựng một mô-đun quản lý khóa an toàn
- Mô-đun mã hóa và giải mã dữ liệu đã được triển khai một cách thành công.

Hướng nghiên cứu tương lai:

- Nghiên cứu và ứng dụng các thuật toán mã hóa mới
- Tối ưu hóa hiệu suất của hệ thống
- Khảo sát và triển khai các phương pháp bảo vệ dữ liệu tiên tiến


THANK YOU