

**Phòng thí nghiệm An toàn Thông tin**

**Trường Đại học Công nghệ Thông tin (UIT)**

# BÁO CÁO THỰC HÀNH

## Môn học: Bảo mật Web và Ứng dụng Kỳ báo cáo: Buổi 05 (Session 05)

**Tên chủ đề: Basic Android Secure Programming**

*GVHD: Ngô Đức Hoàng Sơn Ngày báo cáo: 14/5/2024* **Nhóm: 09**

1. **THÔNG TIN CHUNG:**

*(Liệt kê tất cả các thành viên trong nhóm)*

Lớp: NT213.O22.ATCL.2

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Hồ Ngọc Thiện | 21522620 | [21522620@gm.uit.edu.vn](mailto:21522620@gm.uit.edu.vn) |

## NỘI DUNG THỰC HIỆN:1

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Kết quả tự đánh giá** |
| 1 | Kịch bản 01 | 100% |
| 2 | Kịch bản 02 | 100% |
| 3 | Kịch bản 03 | 100% |
| 4 | Kịch bản 04 | 100% |
| 5 | Kịch bản 05 | 100% |
| 6 | Kịch bản 06 | 100% |
| 7 | Kịch bản 07 | 100% |

**Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.**

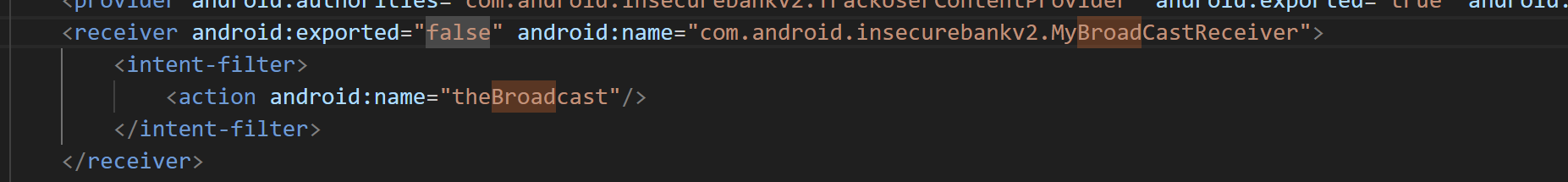
1 Ghi nội dung công việc, các kịch bản trong bài Thực hành

**2**

# BÁO CÁO CHI TIẾT

Yêu cầu 1: Sinh viên tiếp tục sửa lỗi Broadcast Receiver

Cũng giống như sửa lỗi ở activity PostLogin, ta cũng chuyển thuộc tính exported của Broadcast Receiver của ta thành “false”



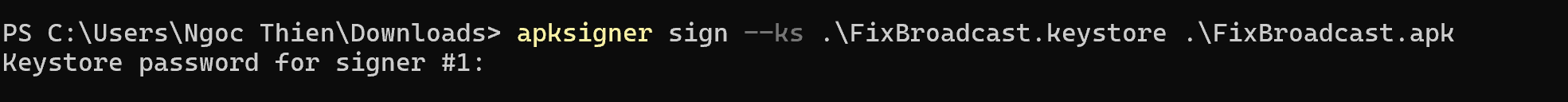
Tiến hành build và kí.

A black screen with white text

Description automatically generated

A screenshot of a computer program

Description automatically generated



Sau đó thử chạy lại đoạn code exploit và kiểm tra phần tin nhắn xem còn có thể tương tác với Broadcast Receiver này nữa không

A computer screen shot of a program code

Description automatically generated

A screenshot of a computer

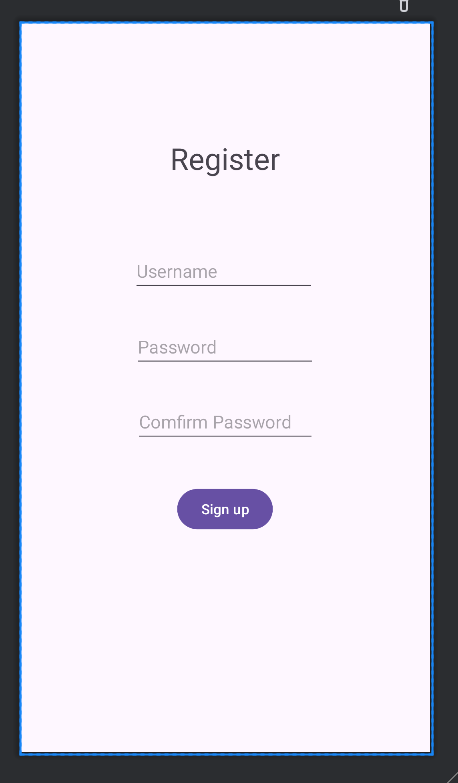
Description automatically generated

Phần tin nhắn không hiện tin nhắn đổi password nữa, vậy ta đã sửa lỗi thành công.

A text on a white surface

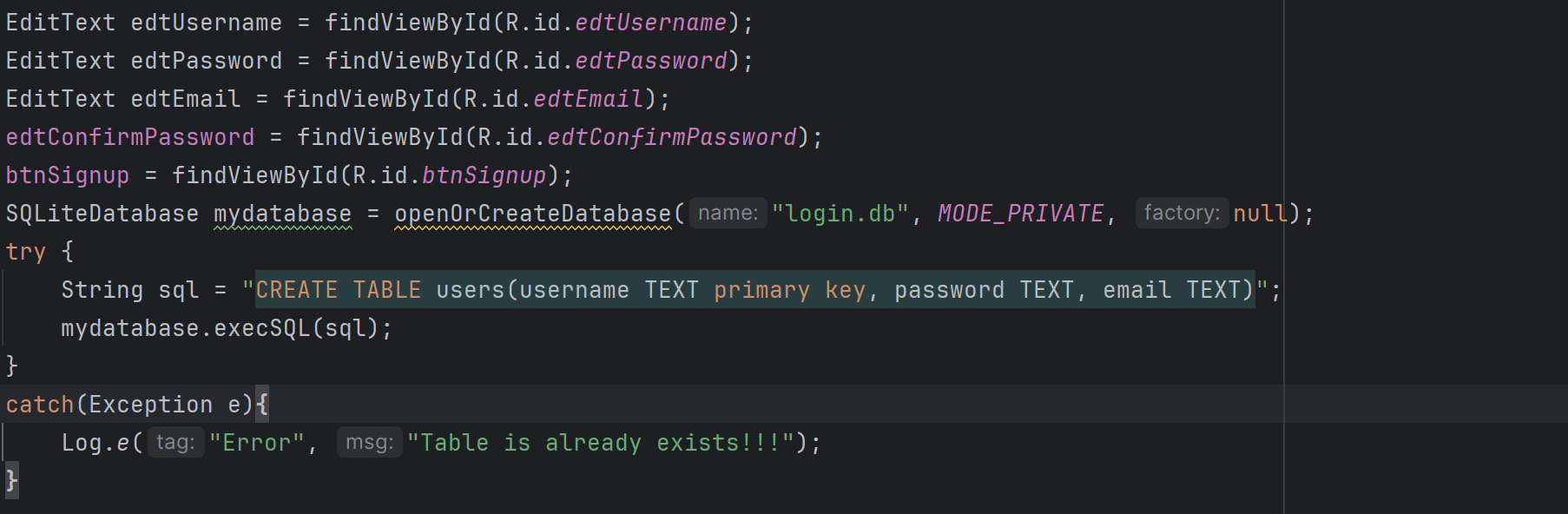
Description automatically generated

Giao diện của register



Phần code:

Ta sử dụng SQLite để tạo 1 database local (lưu trữ trong thiết bị của người dùng)



Dưới đây là phần kiểm tra định dạng của username và password đang đăng kí, đồng thời INSERT thông tin đăng nhập vào database của người dùng.

A screen shot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

Giao diện của trang đăng nhập

A screenshot of a login screen

Description automatically generated

Còn đây là phần code xử lý đăng nhập, trong đó bao gồm kiểm tra thông tin đăng nhập có chứa trong database hay không, nếu có thực hiện truy cập và gửi username đến trang chào người dùng.

A screenshot of a computer

Description automatically generated

Giao diện của trang chào người dùng

A screenshot of a phone

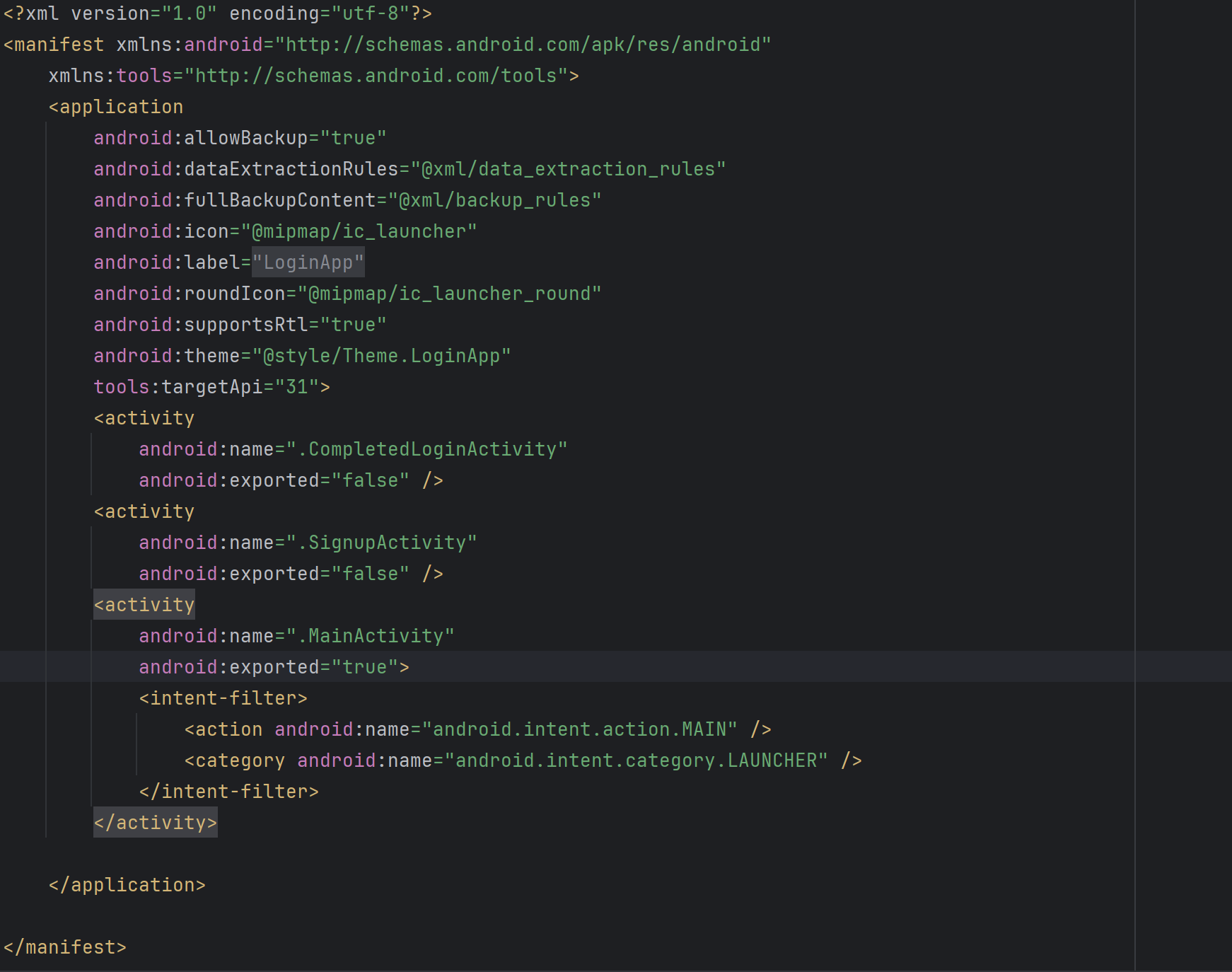
Description automatically generated

Phần code xử lý bao gồm việc nhận dữ liệu và 1 nút để trở về màn hình chính

A computer screen shot of a program code

Description automatically generated

Cấu hình file AndroidManifest.xml



Kết quả

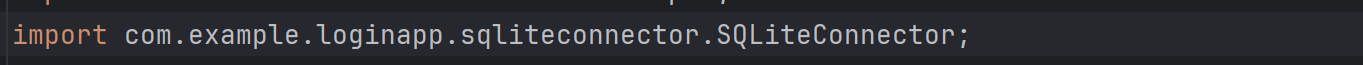
A screenshot of a computer

Description automatically generated

A black text on a white background

Description automatically generated

Thêm dòng sau để import SQLiteConnector cho trang login và register



Code của trang register, ta tạo đối tượng SQLiteConnector để thực hiện các thao tác trên database.

A screen shot of a computer code

Description automatically generated

Sau đó kiểm tra định dạng và sử dụng hàm **addUser(User user)** để thêm user mới vào database.

A screenshot of a computer program

Description automatically generated

Còn ở trang Login, ta sử dụng hàm checkUser(String username, String password) để kiểm tra xem user có tồn tại ở trong database hay không.

A screenshot of a computer program

Description automatically generated

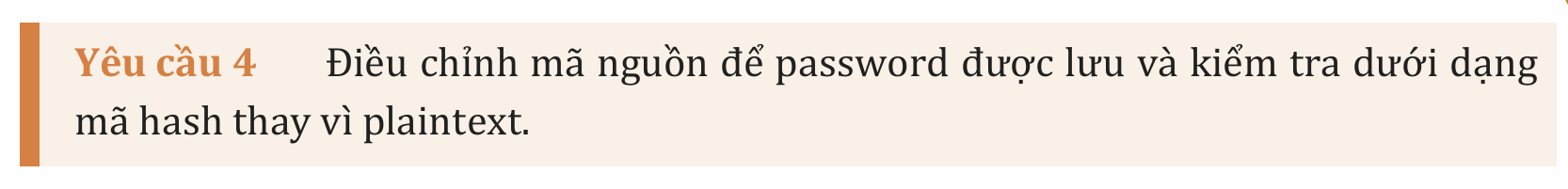
Kết quả

A screenshot of a login form

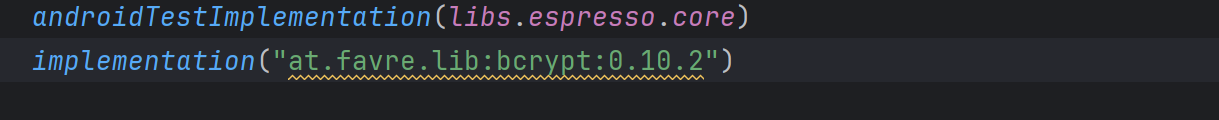
Description automatically generated

A screenshot of a phone

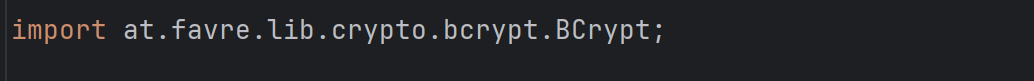
Description automatically generated



Trong build.gradle, ta thêm dòng sau để có thể sử dụng mã hóa bcrypt cho yêu cầu này



Đồng thời import dòng sau vào các file cần thiết



Ở file SQLiteConnector, ta thêm 1 hàm mới như sau để lấy mật khẩu đã được hash về để kiểm tra:

A screenshot of a computer program

Description automatically generated

Ở Register, ta sửa như hình dưới.

A black screen with text

Description automatically generated

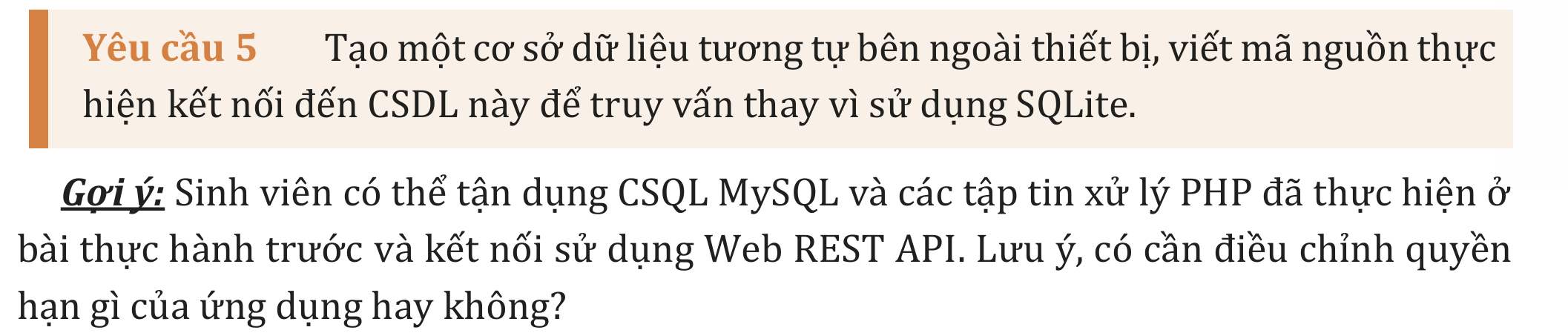
Ở Login, ta sửa như hình dưới, sử dụng hàm **getHashPassword(String username)** để lấy thông tin mật khẩu trên database, sau đó sử dụng **verifyer()** để kiểm tra tính chính xác.

A screen shot of a computer program

Description automatically generated

A screenshot of a black screen

Description automatically generated



Thêm dòng sau để thêm thư viện của Mysql vào project

A screen shot of a computer code

Description automatically generated

Sau đó, tạo 1 file java clas ConnecionDB như sau:

A screen shot of a computer program

Description automatically generated

Tuy nhiên sau khi thử và biên dịch thì gặp lỗi như sau, em sẽ cố gắng khắc phục trong tương lai

A black screen with a black border

Description automatically generated



Khi không bật tính năng Proguard:

A screenshot of a computer

Description automatically generated

Thuộc tính **isMinifyEnabled** được set là false, ta thử decompile bằng apktool xem ta nhận được gì

A screenshot of a computer

Description automatically generated

Đây là thư mục ta có được sau khi decompile

A screenshot of a computer

Description automatically generated

Ta dễ dàng có được các file smali, chính là mã nguồn của chương trình thực tế

A screenshot of a computer

Description automatically generated

Khi sử dụng Proguard, ta chuyển trạng thái của thuộc tính isMinifyEnabled thành true:

A screen shot of a computer program

Description automatically generated

Ta có thể cấu hình file proguard-rules.pro như sau:

A computer screen shot of a black screen

Description automatically generated

Thử decompile lại file apk, ta nhận được thư mục như sau:

A screenshot of a computer

Description automatically generated

Tuy nhiên, khi truy cập vào folder smali, ta không còn thấy đường dẫn com.example.loginapp nữa, thay vào đó là các thư mục được Proguard tạo ra để tránh decompile

A screenshot of a computer

Description automatically generated

Như vậy, nhà phát triển có thể sử dụng proguard như 1 công cụ để che giấu mã nguồn, đồng thời giúp tinh giảm dung lượng của chương trình.

---

***Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này***

## thường gặp Nhóm 07

**19**

**YÊU CẦU CHUNG**

* Sinh viên tìm hiểu và thực hành theo hướng dẫn.
* Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
* Sinh viên báo cáo kết quả thực hiện và nộp bài.

## Báo cáo:

* File .DOCX và .PDF. Tập trung vào nội dung, không mô tả lý thuyết.
* Nội dung trình bày bằng Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.
* Đặt tên theo định dạng: [Mã lớp]-SessionX\_GroupY. (trong đó X là Thứ tự buổi Thực hành, Y là số thứ tự Nhóm Thực hành đã đăng ký với GVHD-TH).

*Ví dụ: [*NT101.K11.ATCL*]-Session1\_Group3.*

* Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
* Không đặt tên đúng định dạng – yêu cầu, sẽ **KHÔNG** chấm điểm bài Lab.
* Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

**Đánh giá**: Sinh viên hiểu và tự thực hiện được bài thực hành. Khuyến khích:

* Chuẩn bị tốt.
* Có nội dung mở rộng, ứng dụng trong kịch bản phức tạp hơn, có đóng góp xây

dựng bài thực hành.

*Bài sao chép, trễ, … sẽ được xử lý tùy mức độ vi phạm.*

**HẾT**