**BÁO CÁO KIỂM TRA GIỮA KỲ**

SẢN PHẨM:

**https://github.com/PLAQUY/GiuaKyDNT**

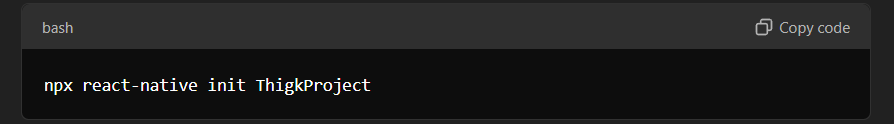
**1. Những nội dung đã hoàn thành:**

* **Xây dựng giao diện cơ bản:**  
  Trong React Native, bạn sẽ sử dụng các component như TextInput (cho form nhập), Button (cho nút thêm sản phẩm), và FlatList (cho danh sách sản phẩm).
* **Kết nối Firebase:**  
  Bạn có thể sử dụng thư viện react-native-firebase để kết nối React Native với Firebase.
* **Chức năng thêm, sửa, xóa:**  
  Tạo các hàm để tương tác với Firebase, bao gồm thêm, sửa và xóa dữ liệu từ cơ sở dữ liệu Firestore của Firebase.
* **Hiển thị danh sách sản phẩm:**  
  Sử dụng FlatList hoặc ScrollView để hiển thị danh sách các sản phẩm, bao gồm các thông tin như Tên sản phẩm, Loại, và Giá, lấy từ Firestore.

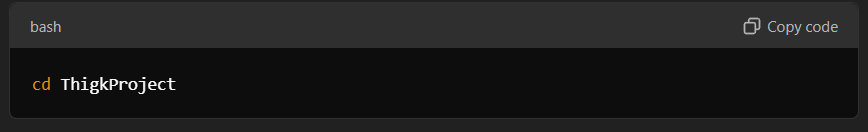
**2. Các bước thực hiện:**

**Bước 1: Tạo một project React Native:**

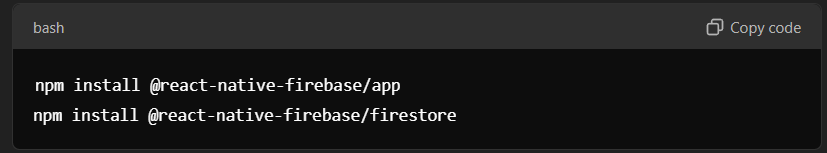
1. Mở terminal và chạy lệnh sau để tạo project React Native mới:



1. Di chuyển vào thư mục project:

****

**Bước 2: Thực hiện kết nối Firebase với React Native:**

1. **Đăng nhập vào Firebase và tạo project:**
   * Đăng nhập vào Firebase Console và tạo một project mới với tên là "thigk" như trong hướng dẫn.
2. **Thêm Firebase vào project React Native:**
   * **Cài đặt Firebase SDK:** Chạy các lệnh sau để cài đặt Firebase cho React Native: 

**Cấu hình Firebase trong ứng dụng:**

* + **iOS:**  
    Mở ios/ThigkProject.xcworkspace bằng Xcode, thêm file GoogleService-Info.plist vào project iOS của bạn trong thư mục ios.
  + **Android:**  
    Thêm file google-services.json vào thư mục android/app. Sau đó, cập nhật các file android/build.gradle và android/app/build.gradle để tích hợp Firebase vào Android.

**Bước 3: Tạo cơ sở dữ liệu trên firebase**

* Trên firebase console, chọn Build -> firestore database
* Tạo một bảng dữ liệu tên “sanpham” bao gồm các thuộc tính “TenSP”, “Gia”, “Loai”

**Bước 4: Tạo giao diện**

* Tại file main.dart:
  + MaterialApp: Tạo widget chính của ứng dụng, bao gồm tiêu đề, thêm và trang chính
    - ThemeData: định nghĩa kiểu dáng, màu sắc cho các button, TextFormField,..
    - Home: gọi class Home\_Screen làm trang chính cho ứng dụng
* Tạo một class HomeScreen là một Widget ở trạng thái Stateful, đây là nơi chứa giao diện của ứng dụng
  + Tạo một Scaffold bao gồm AppBar, body
    - appBar: Hiển thị tiêu đề
    - body:
      * chứa form nhập liệu: sử dụng TextFormField để làm input nhập các giá trị đầu vào
      * button Thêm sản phẩm
      * sử dụng ListView để hiển thị danh sách sản phẩm: trong listView sẽ chia thành 3 cột:
        + Cột thứ nhất: chiếm 1/5 chiều rộng listview, để hiển thị hình ảnh
        + Cột thứ hai: chiếm 3/5 chiều rộng listview, để hiển thị các thông tin sản phẩm
        + Cột thứ ba: chiếm 1/5 chiều rộng litsview, chứa 2 IconButton là edit và delete

**Bước 5: Thêm chức năng cho ứng dụng**

* Thêm chức năng Thêm sản phẩm
* Thêm chức năng xóa sản phẩm
* Thêm chức năng edit sản phẩm:
  + Tạo một showDialog để hiển thị form edit sản phẩm
  + Tạo nút Cancel và Update
  + Thêm action cho nút Update