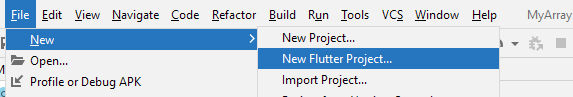
**TẠO ỨNG DỤNG FLUTTER ĐẦU TIÊN**

Trong bài này sẽ hướng dẫn các bạn tạo một **Flutter Application** đầu tiên trên Android Studio, quan đó giúp các bạn hiểu cơ bản cấu trúc của một project ứng dụng Flutter

**Bước 1** − Mở Android Studio

**Bước 2** − Tạo Flutter Project mới: Chon **Start a New Flutter Project** hoặc từ menu  **File → New → New Flutter Project**



**Bước 3** − Chọn đường dẫn thư mục **Flutter SDK**

A screenshot of a computer

Description automatically generated

**Bước 4** − Có nhiều loại project Flutter khác nhau chúng ta chọn  **Application** .

Đặt tên và mô tả cho Project sau đó chọn **Next**.

Bạn đặt tên cho project là **hello\_app** hoặc tên bất kì., nơi lưu trữ project và mô tả của ứng dụng

Điền package\_name cho ứng dụng

A screenshot of a computer

Description automatically generated

**Bước 6** − Nhấn Finish và đợi một lúc để Android Studio tiến hành việc tạo project

Sau khi tạo xong, chúng ta có thể thấy cấu trúc của một project Flutter như bên dưới

A picture containing text, screenshot, mobile phone, software

Description automatically generated

Mình giải thích qua các thành phần của một Project Flutter

* **android** − Thư mục code sinh tự động cho ứng dụng Android
* **ios** − Thư mục code sinh tự động cho ứng dụng iOS
* **lib** − Main folder chứa Dart code được viết khi sử dụng flutter framework
* **lib/main.dart** − File đầu tiên là điểm khởi đầu của ứng dụng Flutter application
* **test** − Folder chứa Dart code để test flutter application
* **test/widget\_test.dart** − Sample code
* **.gitignore** − Git version control file - File này chứa cấu hình cho project git
* **.metadata** − sinh tự động bởi flutter tools
* **.packages** − sinh tự động để theo dõi flutter packages
* **.iml** − project file của Android studio
* **pubspec.yaml** − Được sử dụng  **Pub**, Flutter package manager
* **pubspec.lock** − Sinh tự động bởi Flutter package manager, **Pub**
* **README.md** − Project description được viết theo cấu trúc Markdown

**Bước 7** − Mặc định Android Studio đã tạo sẵn cho chúng ta code ở  *lib/main.dart file* , tuy nhiên chúng ta xoá đi và viết lại đoạn code dưới đây để dễ hiểu hơn

import 'package:flutter/material.dart';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {

// This widget is the root of your application.

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'Hello World Demo Application',

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: MyHomePage(title: 'Home page'),

);

}

}

class MyHomePage extends StatelessWidget {

MyHomePage({Key? key, required this.title}) : super(key: key);

final String title;

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text(this.title),

),

body: Center(

child:

Text(

'Hello World',

)

),

);

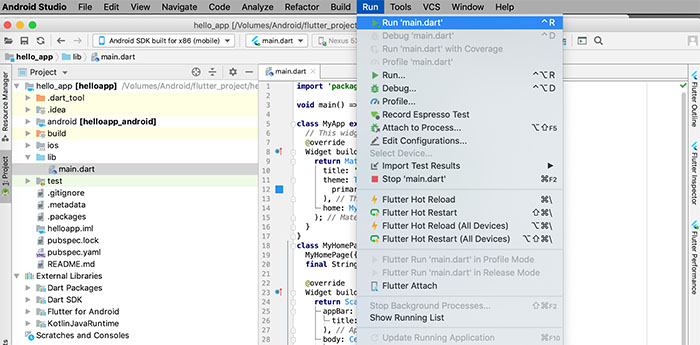
}

}

Mình sẽ giải thích ý nghĩa chi tiết của đoạn code trên

* **Dòng 1** − import flutter package, tên là *material*. **Material** là một flutter package được sử dụng để tạo giao diện người dùng theo Material design cho Android.
* **Dòng 2** − là điểm khởi đầu của Flutter application là hàm main của ứng dụng. Phương thức *runApp* được gọi và truyền vào đối tượng của lớp *MyApp*. Mục đích chính của phương thức *runApp* là đưa giao diện widget vào hiển thị trên màn hình
* **Dòng 5-17** − Widget được sử dụng để tạo UI (giao diện người dùng) qua flutter framework. *StatelessWidget* là một widget, nó không bao gồm trạng thái nào của widget. *MyApp* kế thừa *StatelessWidget* và ghi đè phương thức *build method*. Mục đích của phương thức *build* là tạo một phần UI cho ứng dụng. Ở đây, phương thức *build* sử dụng *MaterialApp*, một widget để tạo layout UI gốc cho ứng dụng. Bao gồm 3 thành phần chính là - *title (tiêu đề), theme (chủ đề)* và *home (trang chủ hay phần màn hình)*.*title* là tiêu đề của ứng dụng*theme* là chủ đề của widget. Ở đây, chúng ta set chủ đề là *blue* đó là màu sắc chủ đạo của ứng dụng thông qua class *ThemeData* và các thuộc tính của nó ví dụ *primarySwatch*.home phần màn hình của ứng dụng, nó được tạo bởi một widget khác, **MyHomePage**
* **Dòng 19 - 38** − *MyHomePage* tương tự *MyApp* ngoại trừ nó trả về *Scaffold* Widget. *Scaffold* là môt top level widget đứng sau *MaterialApp* widget được sử dụng để tạo UI theo material design. Nó có hai thuộc tính quan trọng nhất là *appBar* để hiển thị phần đầu của ứng dụng và body để hiển thị nôi dung chính của ứng dụng. *AppBar* là một widget khác để tạo phần đầu của ứng dụng và sử dụng các thuộc tính của *appBar*. Trong các thuộc tính của *body*, chúng ta sử dụng một *Center* widget, đây là một widget con. *Text* là một widget cuối cùng, phổ biến để hiển thị một văn bản giữa màn hình

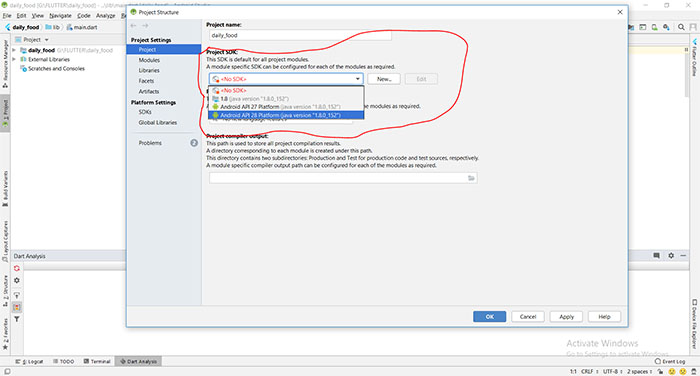
**Bước 8** − Bây giờ chúng ta bắt đầu chạy ứng dụng bằng lệnh **Run → Run main.dart**



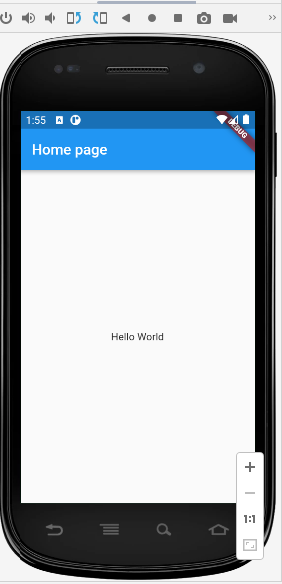
Lưu ý bạn cần kết nối với một thiết bị Android thật thông qua cab USB (bật chế độ nhà phát triển) hoặc kết nối với một máy aỏ Android Emulator

Trong trường hợp Android Studio báo lỗi không kết nối được với thiết bị (**flutter run: No connected devices**) có thể là do bạn chưa chọn API Android cho project

Các bạn chọn**File → Project Structure**và thêm vào SDK Android mới nhất



Kết quả khi chạy Project Flutter trên máy ảo Android Emulator



Như vậy là mình đã hướng dẫn các bạn **tạo ứng dụng Flutter đầu tiên**, giải thích các thành phần chính của project cũng như code của file main. Trong bài tiếp theo mình sẽ hướng dẫn các bạn tìm hiểu kiến trúc của ứng dụng **Flutter**