

NAME : Nilanchala Panda  
Roll No . : 41

Div . : D15A  
Batch : B

## EXPERIMENT NO 6

**Aim:** To connect firebase database with flutter UI

### Theory:

Firebase is a great backend solution for anyone that wants to use authentication, databases, cloud functions, ads, and countless other features within an app.

In this article, you will create a Firebase project for iOS and Android platforms using Flutter.

## Prerequisites

To complete this tutorial, you will need:

- A Google account to use Firebase.
- Developing for iOS will require XCode.
- To download and install Flutter.
- To download and install Android Studio and Visual Studio Code.
- It is recommended to install plugins for your code editor:
  - `Flutter` and `Dart` plugins installed for Android Studio.
  - `Flutter` extension installed for Visual Studio Code.

This tutorial was verified with Flutter v2.0.6, Android SDK v31.0.2, and Android Studio v4.1.

## Creating a New Flutter Project

This tutorial will require the creation of an example Flutter app.

Once you have your environment set up for Flutter, you can run the following to create a new application:

```
flutter create flutterfirebaseexample
```

1.

[Copy](#)

Navigate to the new project directory:

```
cd flutterfirebaseexample
```

```
1.
```

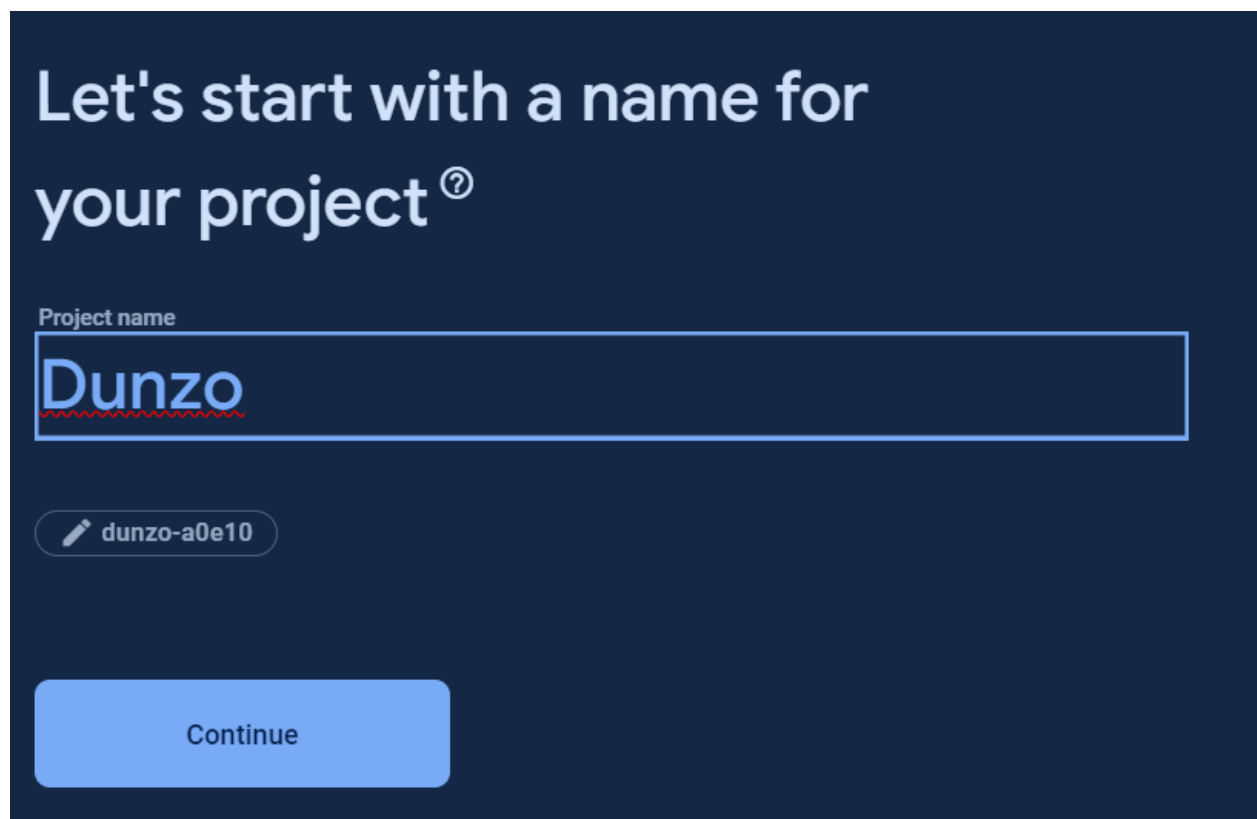
[Copy](#)

Using `flutter create` will produce a demo application that will display the number of times a button is clicked.

Now that we've got a Flutter project up and running, we can add Firebase.

## Creating a New Firebase Project

First, log in with your Google account to manage your Firebase projects. From within the Firebase dashboard, select the Create new project button and give it a name:

A screenshot of the Firebase project creation interface. The background is dark blue. At the top, the text "Let's start with a name for your project" is displayed in white, with a question mark icon to the right. Below this, there is a text input field with the label "Project name" above it. The field contains the text "Dunzo" in a large, white, sans-serif font. Below the input field, there is a button with a pencil icon and the text "dunzo-a0e10". At the bottom, there is a large, light blue button with the text "Continue" in white.

```
dependencies:
  flutter:
    sdk: flutter
  curved_navigation_bar: ^1.0.3
  firebase_core: ^2.24.2
  firebase_auth: ^4.16.0
  cloud_firestore: ^4.14.0

  image_picker: ^1.0.7
  firebase_storage: ^11.6.6
  # flutter_stripe: ^10.0.0
  http: 1.2.0
  random_string: ^2.3.1
  shared_preferences: ^2.2.2

# The following adds the CupertinoIcons
# Use with the CupertinoIcons
cupertino_icons: ^1.0.2
```

Be sure to move this file within Xcode to create the proper file references.

There are additional steps for installing the Firebase SDK and adding initialization code, but they are not necessary for this tutorial.

That's it!

## Conclusion

In this article, you learned how to set up and ready our Flutter applications to be used with Firebase.

Flutter has official support for Firebase with the FlutterFire set of libraries.