

FirebaseOptions cannot be null when creating the default app

Asked 3 months ago Active 2 months ago Viewed 11k times



12



2



I am trying to try a sample project in Flutter integration email and google based login, and planning to use firebase initialisation for doing it while I have followed all the steps as mentioned in tutorials I am getting this error as soon as firebase is attempted to be initialised.

```
"FirebaseOptions cannot be null when creating the default app."
  at Object.throw_ [as throw] (http://localhost:7357/dart_sdk.js:5063:11)
  at Object.assertFailed (http://localhost:7357/dart_sdk.js:4988:15)
  at firebase_core_web.FirebaseCoreWeb.new.initializeApp
  (http://localhost:7357/packages/firebase_core_web/firebase_core_web.dart.lib.js:25

    at initializeApp.next (<anonymous>)
    at http://localhost:7357/dart_sdk.js:40192:33
    at _RootZone.runUnary (http://localhost:7357/dart_sdk.js:40062:59)
    at _FutureListener.thenAwait.handleValue
  (http://localhost:7357/dart_sdk.js:34983:29)
    at handleValueCallback (http://localhost:7357/dart_sdk.js:35551:49)
    at Function._propagateToListeners
  (http://localhost:7357/dart_sdk.js:35589:17)
    at _Future.new._[_completeWithValue]
  (http://localhost:7357/dart_sdk.js:35437:23)
    at async._AsyncCallbackEntry.new.callback
  (http://localhost:7357/dart_sdk.js:35458:35)
    at Object._microtaskLoop (http://localhost:7357/dart_sdk.js:40330:13)
    at _startMicrotaskLoop (http://localhost:7357/dart_sdk.js:40336:13)
    at http://localhost:7357/dart_sdk.js:35811:9
```

Here is my index.html

```
<!DOCTYPE html>
<html>
<head>
  <!--
    If you are serving your web app in a path other than the root, change the
    href value below to reflect the base path you are serving from.

    The path provided below has to start and end with a slash "/" in order for
    it to work correctly.

    For more details:
    * https://developer.mozilla.org/en-US/docs/Web/HTML/Element/base

    This is a placeholder for base href that will be replaced by the value of
    the `--base-href` argument provided to `flutter build`.
  -->
  <base href="$FLUTTER_BASE_HREF">

  <meta charset="UTF-8">
  <meta content="IE=Edge" http-equiv="X-UA-Compatible">
  <meta name="description" content="A new Flutter project.">
  <meta name="google-signin-client_id" content="619218114547-
```

```
xxxx.apps.googleusercontent.com">
  <!-- iOS meta tags & icons -->
  <meta name="apple-mobile-web-app-capable" content="yes">
  <meta name="apple-mobile-web-app-status-bar-style" content="black">
  <meta name="apple-mobile-web-app-title" content="signin_example">
  <link rel="apple-touch-icon" href="icons/Icon-192.png">

  <!-- Favicon -->
  <link rel="icon" type="image/png" href="favicon.png"/>

  <title>signin_example</title>
  <link rel="manifest" href="manifest.json">
</head>
<body>
  <!-- This script installs service_worker.js to provide PWA functionality to
  application. For more information, see:
  https://developers.google.com/web/fundamentals/primers/service-workers -
  -->

  <script>
    var serviceWorkerVersion = null;
    var scriptLoaded = false;
    function loadMainDartJs() {
      if (scriptLoaded) {
        return;
      }
      scriptLoaded = true;
      var scriptTag = document.createElement('script');
      scriptTag.src = 'main.dart.js';
      scriptTag.type = 'application/javascript';
      document.body.append(scriptTag);
    }

    if ('serviceWorker' in navigator) {
      // Service workers are supported. Use them.
      window.addEventListener('load', function () {
        // Wait for registration to finish before dropping the <script> tag.
        // Otherwise, the browser will load the script multiple times,
        // potentially different versions.
        var serviceWorkerUrl = 'flutter_service_worker.js?v=' +
serviceWorkerVersion;
        navigator.serviceWorker.register(serviceWorkerUrl)
          .then((reg) => {
            function waitForActivation(serviceWorker) {
              serviceWorker.addEventListener('statechange', () => {
                if (serviceWorker.state == 'activated') {
                  console.log('Installed new service worker.');
                  loadMainDartJs();
                }
              });
            }
            if (!reg.active && (reg.installing || reg.waiting)) {
              // No active web worker and we have installed or are installing
              // one for the first time. Simply wait for it to activate.
              waitForActivation(reg.installing || reg.waiting);
            } else if (!reg.active.scriptURL.endsWith(serviceWorkerVersion)) {
              // When the app updates the serviceWorkerVersion changes, so we
              // need to ask the service worker to update.
              console.log('New service worker available.');
              reg.update();
              waitForActivation(reg.installing);
            }
          });
      });
    }
  </script>
</body>
</html>
```

```

    } else {
      // Existing service worker is still good.
      console.log('Loading app from service worker.');
```

`loadMainDartJs();`

```

    }
  });

  // If service worker doesn't succeed in a reasonable amount of time,
  // fallback to plain <script> tag.
  setTimeout(() => {
    if (!scriptLoaded) {
      console.warn(
        'Failed to load app from service worker. Falling back to plain
<script> tag.',
      );
      loadMainDartJs();
    }
  }, 4000);
});
} else {
  // Service workers not supported. Just drop the <script> tag.
  loadMainDartJs();
}
</script>
<script type="module">
  // Import the functions you need from the SDKs you need
  import { initializeApp } from
'https://www.gstatic.com/firebasejs/9.6.0/firebase-app.js';
  import { getAnalytics } from
'https://www.gstatic.com/firebasejs/9.6.0/firebase-analytics.js';
  // TODO: Add SDKs for Firebase products that you want to use
  // https://firebase.google.com/docs/web/setup#available-libraries

  // Your web app's Firebase configuration
  // For Firebase JS SDK v7.20.0 and later, measurementId is optional
  const firebaseConfig = {
    apiKey: "xxx",
    authDomain: "xxx",
    projectId: "xx",
    storageBucket: "exxx",
    messagingSenderId: "xxx",
    appId: "xxx",
    measurementId: "xxx"
  };

  // Initialize Firebase
  const app = initializeApp(firebaseConfig);
  const analytics = getAnalytics(app);
</script>
</body>
</html>

```

Part of code from main.dart file where firebase initialisation is happening

```

void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp();
  runApp(SignUpApp());
}

```

Pubsec.yaml file of the project

```
name: signin_example
description: A new Flutter project.

# The following line prevents the package from being accidentally published to
# pub.dev using `flutter pub publish`. This is preferred for private packages.
publish_to: 'none' # Remove this line if you wish to publish to pub.dev

https://developer.apple.com/library/archive/documentation/General/Reference/InfoPl

version: 1.0.0+1

environment:
  sdk: ">=2.15.0-268.18.beta <3.0.0"

dependencies:
  flutter:
    sdk: flutter

# The following adds the Cupertino Icons font to your application.
# Use with the CupertinoIcons class for iOS style icons.
cupertino_icons: ^1.0.2
firebase_auth: ^3.3.0
firebase_core: ^1.10.2
google_sign_in: ^5.2.1
shared_preferences: ^2.0.9

dev_dependencies:
  flutter_test:
    sdk: flutter

  flutter_lints: ^1.0.0

# The following section is specific to Flutter.
flutter:

  assets:
    - assets/images/

  uses-material-design: true
```

Can anyone help me understand what is the issue here?

[firebase](#) [flutter](#) [dart](#) [web](#)

Share Follow

asked Dec 5, 2021 at 9:23



[paras 231096](#)

123 1 1 6

Also see: stackoverflow.com/questions/70221915/... Unanswered for now, but with the same problem.



– [Frank van Puffelen](#) Dec 5, 2021 at 15:27

2 Answers

| Active | Oldest | Votes |
|--------|--------|-------|
|--------|--------|-------|

UPDATE:

27 For your `firebase_core` version it seems to be sufficient to pass the `FirebaseOptions` once you initialize firebase in your flutter code (and you don't need any script tags in your `index.html`):



```
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    // Replace with actual values
    options: FirebaseOptions(
      apiKey: "XXX",
      appId: "XXX",
      messagingSenderId: "XXX",
      projectId: "XXX",
    ),
  );
  runApp(MyApp());
}
```

Alternatively, one can also follow the updated [official documentation](#) and use the Firebase CLI to automatically create a `firebase_options.dart` file in your `lib` folder that will define the correct `FirebaseOptions` for you. The steps are:

1. Install Flutterfire CLI:

```
dart pub global activate flutterfire_cli
```

2. Configure Flutterfire (run in your project's root and go through the wizard selecting the correct Firebase project and target platforms):

```
flutterfire configure
```

3. Import the automatically generated `firebase_options.dart` file in your `main.dart` file and initialize Firebase as follows:

```
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    options: DefaultFirebaseOptions.currentPlatform,
  );
  runApp(MyApp());
}
```

PREVIOUS ANSWER:

From the [docs](#):

The only way to currently add the Firebase SDKs to your Flutter web project is by importing the scripts from the Firebase content delivery network (CDN).

Therefore, please try to use script tags to import the relevant firebase components with version 8.6.1 as shown below (more info [here](#)):

```
<script src="https://www.gstatic.com/firebasejs/8.6.1/firebase-app.js">
</script>
<script src="https://www.gstatic.com/firebasejs/8.6.1/firebase-analytics.js">
</script>
<script>
  var firebaseConfig = {
    apiKey: "xxx",
    authDomain: "xxx",
    projectId: "xx",
    storageBucket: "exxx",
    messagingSenderId: "xxx",
    appId: "xxx",
    measurementId: "xxx"
  };

  firebase.initializeApp(firebaseConfig);
  firebase.analytics();
</script>
```

Share Follow

edited Dec 25, 2021 at 9:13

answered Dec 5, 2021 at 11:55



[hnnngwdlch](#)

1,016 6 10

On switching to 8.6.1 getting these errors now `FlutterFire for Web is explicitly tested against Firebase JS SDK version "8.10.0" but your currently specifying "8.6.1" by either the imported Firebase JS SDKs in your web/index.html file or by providing an override . It is recommended that you change all of the versions of the Firebase JS SDK version "8.10.0" ` On updating it to 8.10.0 then or remaining at 8.6.1 as well getting this `NoSuchMethodError: tried to call a non-function, such as null: 'dart.global.firebase.auth' null ` Can you please suggest next? – [paras 231096](#) Dec 5, 2021 at 14:19

Should I downgrade firebase dependencies in pubsec.yaml ? `` cupertino_icons: ^1.0.2 firebase_auth: ^3.3.0 firebase_core: ^1.10.2 google_sign_in: ^5.2.1 shared_preferences: ^2.0.9 `` – [paras 231096](#) Dec 5, 2021 at 14:25

Just updated my answer – [hnnngwdlch](#) Dec 5, 2021 at 16:12

Thanks, it helped me. – [My Car](#) Feb 23 at 12:58



5

Adding web features to flutter apps, we need to provide values in options argument of `Firebase.initializeApp(options:FirebaseOptions...)` .



These values are what we get when we add web app in our Firebase console. What I did was created `config/config.dart` file in `lib` folder. Where:

```
class Configurations {
  static const _apiKey = "Your values";
  static const _authDomain = "Your values";
  static const _projectId = "Your values";
  static const _storageBucket = "Your values"
  static const _messagingSenderId = "Your values"
  static const _appId = "Your values"

  //Make some getter functions
  String get apiKey => _apiKey;
  String get authDomain => _authDomain;
  String get projectId => _projectId;
  String get storageBucket => _storageBucket;
  String get messagingSenderId => _messagingSenderId;
  String get appId => _appId;
}
```

Then add the config folder to `.gitignore`.

```
lib/config
```

Now Import the file into relevant sections and then

```
import 'config/config.dart';

final configurations = Configurations();
Future<void> init() async {
  await Firebase.initializeApp(
    options: FirebaseOptions(
      apiKey: configurations.apiKey,
      appId: configurations.appId,
      messagingSenderId: configurations.messagingSenderId,
      projectId: configurations.projectId));
}
```

This should now work.

Share Follow

edited Dec 16, 2021 at 6:51



Jeremy Caney

5,622 30 41 69

answered Dec 15, 2021 at 11:15



nibukdk93

134 11

Thanks for your contribution, it helped me – [Sana'a Al-ahdal](#) Jan 16 at 10:47
