

Get Started

Firebase is back at Google I/O on May 10! Register now

# Get Started with Firebase Authentication on



- 1. If you haven't already, install the Firebase JS SDK and initialize Firebase.
- 2. Add the Firebase Authentication JS SDK and initialize Firebase Authentication:



#### (Optional) Prototype and test with Firebase Local Emulator Suite

Before talking about how your app authenticates users, let's introduce a set of tools you can use to prototype and test Authentication functionality: Firebase Local Emulator Suite. If you're deciding among authentication techniques and providers, trying out different data models with public and private data using Authentication and Firebase Security Rules, or prototyping sign-in UI designs, being able to work locally without deploying live services can be a great idea.

An Authentication emulator is part of the Local Emulator Suite, which enables your app to interact with emulated database content and config, as well as optionally your emulated project resources (functions, other databases, and

- 1. Adding a line of code to your app's test config to connect to the emulator.
- 2. From the root of your local project directory, running firebase emulators:start.
- 3. Using the Local Emulator Suite UI for interactive prototyping, or the Authentication emulator REST API for non-

A detailed guide is available at Connect your app to the Authentication emulator. For more information, see the Local

Create a form that allows new users to register with your app using their email address and a password. When a user completes the form, validate the email address and password provided by the user, then pass them to the createUserWithEmailAndPassword method:

English ▼

Add and initialize the Authentication

(Optional) Prototype and test with Firebase Local Emulator Suite

Go to console

Sign up new users

Set an authentication state observer and get user data

Next steps

Recommended for you

Manage Users in Firebase Updated 12 thg 4, 2023

```
Manage Users
     Password Authentication
    Email Link Authentication
     Google
    Facebook Login
    Sign in with Apple
    Twitter
    GitHub
    Microsoft
     Yahoo
    Phone Number
    OpenID Connect
    SAML
    Use a Custom Auth System
iOS+

    Android

▶ Flutter
▼ Web
    Sign in with a pre-built UI
    Get Started
    Manage Users
    Password Authentication
    Email Link Authentication
    Google
    Facebook Login
    Sign in with Apple
    Twitter
```

GitHub

Microsoft

Phone Number

OpenID Connect

SAML

Use a Custom Auth System

Users in Firebase Projects

- IOS+
- Android
- Flutter Web

Sign in with a pre-built UI

#### Get Started

Manage Users

Password Authentication

Email Link Authentication

Google

Facebook Login

Sign in with Apple Twitter

GitHub

Microsoft

Phone Number

OpenID Connect

Use a Custom Auth System

Users in Firebase Projects

- iOS+
- Android Flutter
- Web

Sign in with a pre-built UI

#### Get Started

Manage Users

Password Authentication

Email Link Authentication

Google

Facebook Login Sign in with Apple

Twitter

Microsoft

Yahoo

Phone Number OpenID Connect

Use a Custom Auth System

Anonymous Authentication



### Sign in existing users

Create a form that allows existing users to sign in using their email address and password. When a user completes the form, call the signInWithEmailAndPassword method:

```
(modular)
               (namespaced)
★ Learn more about the tree-shakeable Web v9 modular SDK and upgrade from version 8.
                                                                                                  • 🗇
import { getAuth, signInWithEmailAndPassword } from "firebase/auth";
const auth = getAuth();
\verb|signInWithEmailAndPassword(auth, email, password)|\\
  .then((userCredential) => {
    // Signed in
   const user = userCredential.user;
    // ...
  .catch((error) => {
   const errorCode = error.code;
    const errorMessage = error.message;
                                                                          auth_signin_password.js 🔘
```

#### Set an authentication state observer and get user data

For each of your app's pages that need information about the signed-in user, attach an observer to the global authentication object. This observer gets called whenever the user's sign-in state changes.

Attach the observer using the onAuthStateChanged method. When a user successfully signs in, you can get information about the user in the observer.

```
Web version 9
               Web version 8
  (modular)
                (namespaced)
 ★ Learn more about the tree-shakeable Web v9 modular SDK and upgrade from version 8.
                                                                                                  • •
import { getAuth, onAuthStateChanged } from "firebase/auth";
const auth = getAuth();
onAuthStateChanged(auth, (user) => {
  if (user) {
    // User is signed in, see docs for a list of available properties
    // https://firebase.google.com/docs/reference/js/firebase.User
    const uid = user.uid;
  } else {
    // User is signed out
});
                                                                           auth_state_listener.js ()
```

#### Next steps

Learn how to add support for other identity providers and anonymous guest accounts:

- · Google Sign-in
- Facebook Login
- Twitter Login
- · GitHub Login
- · Anonymous sign-in

Users in Firebase Projects

- iOS+
- Android
- Flutter

Sign in with a pre-built UI Get Started

### Manage Users

Password Authentication Email Link Authentication

Google

Facebook Login

Sign in with Apple

Twitter

GitHub

Microsoft

Yahoo Phone Number

- i0S+
- Android
- Flutter
- ▼ Web

Sign in with a pre-built UI

#### Get Started

Manage Users

Password Authentication

Email Link Authentication

Google

Facebook Login

Sign in with Apple

GitHub

Microsoft

Yahoo Phone Number

OpenID Connect

SAML

Was this helpful?

Send feedback

## Recommended for you ①

#### Manage Users in Firebase

Firebase gives you the tools and infrastructure you need to build...

Updated 12 thg 4, 2023

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 4.0 License, and code samples are licensed under the Apache 2.0 License. For details, see the Google Developers Site Policies. Java is a registered trademark of Oracle and/or its affiliates

Last updated 2023-04-11 UTC.

Learn Stay connected Support Guides Blog Contact support Stack Overflow Reference Firebase Summit Samples Twitter Slack community Libraries YouTube Google group GitHub Release notes FAQs

Google Developers Android Chrome Firebase Google Cloud Platform All products