

Firestore Initialize - New Clinic

After you created a new clinic, on the manage page in the management project you can edit the clinic properties. In the form that will open, if there is no firebase config already entered into the clinic, you will see a message with the steps to follow (this guide will do the same with more details).

The screenshot shows a form titled "Firestore Properties:". The main section is "Firestore Project Init Guide", which contains the following text: "It seems like you didn't set any Firestore project for this clinic yet. Open the Pdf Guide or follow the next steps:". Below this is a button labeled "Download Pdf Guide". A yellow callout box on the left says "We are in this guide" with an arrow pointing to the button. The rest of the guide text includes: "Please go to [Firestore Console](#) and create a project.", "After that create web-app and copy the config to the field 'Firestore Props' below.", "Create a **Realtime database** and set the rules (**read and write**) to True. (You can follow this [video](#))", "In the Firestore project look for **Authetication**, setup our Firestore Authetication method to be Email/Password and Enable the first option.", and "Don't forget to check the 'Firestore Props Changed' checkbox!". Below the guide is a section for "Firestore Config" with the instruction: "Copy the firestore config propertis from your firestore app settings."

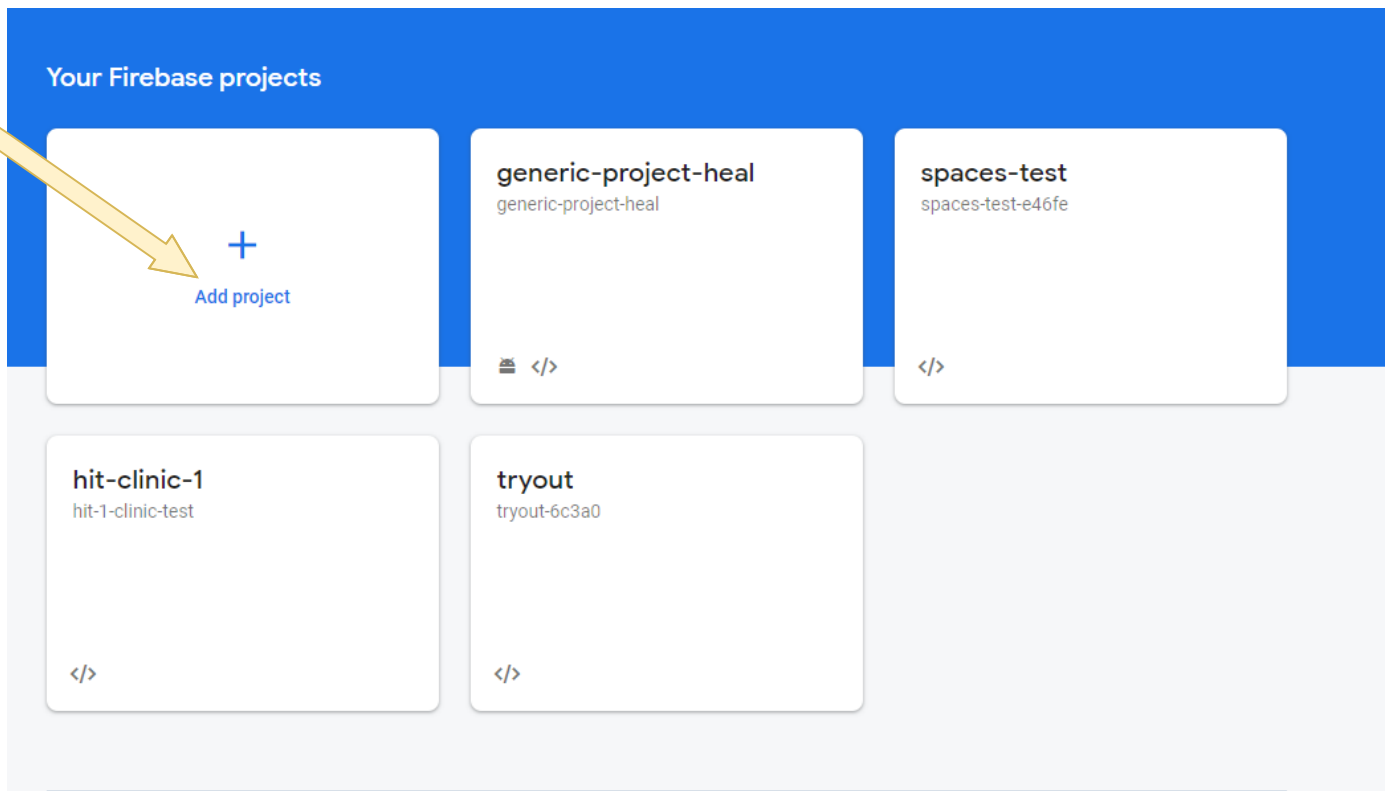
Step 1: Generate a new firebase project

Go to the firebase console and press add project.

Follow the steps: enter the project name (you can use the clinic name), choose the user that will control the firebase, and let the project generate.

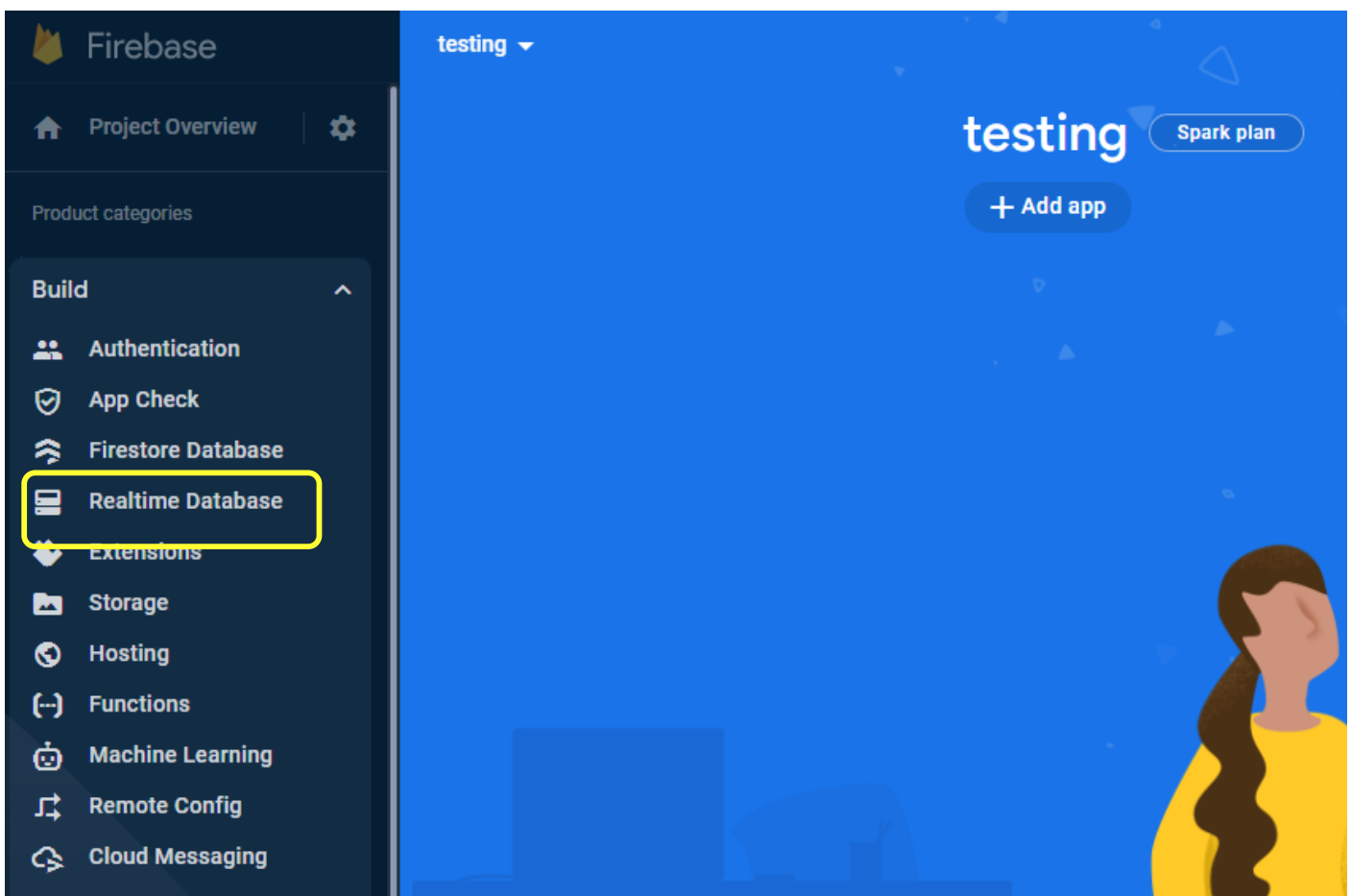
After that firebase will redirect you to the project console.

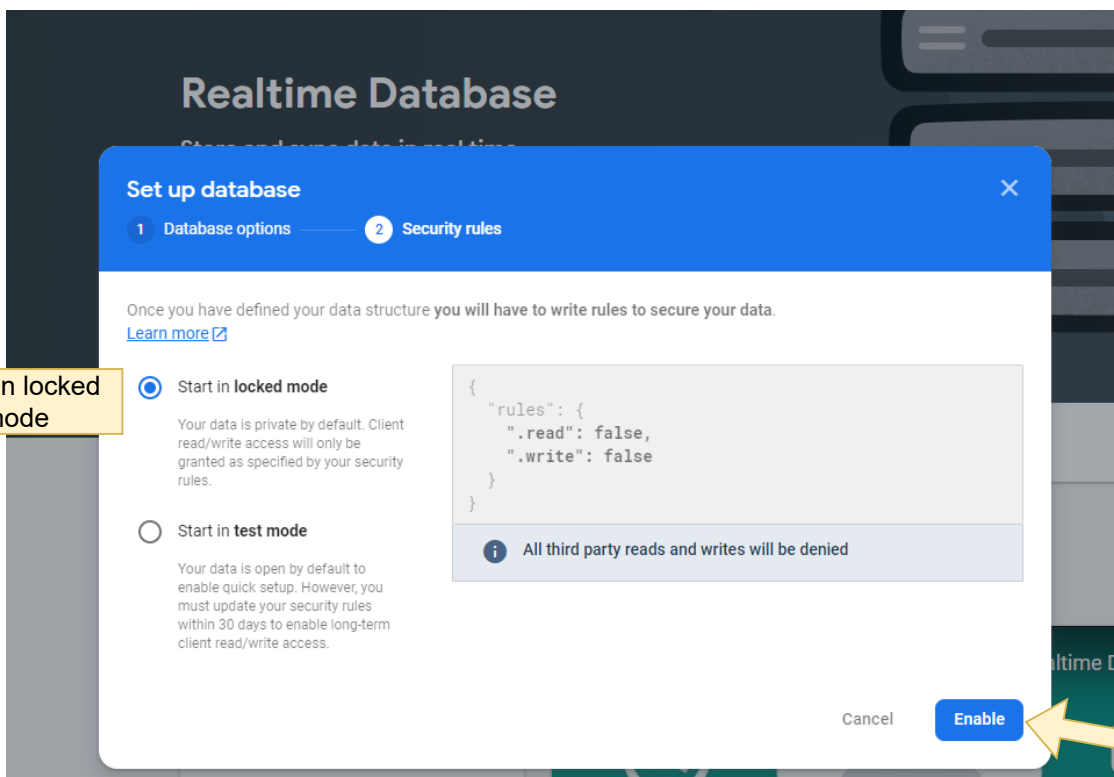
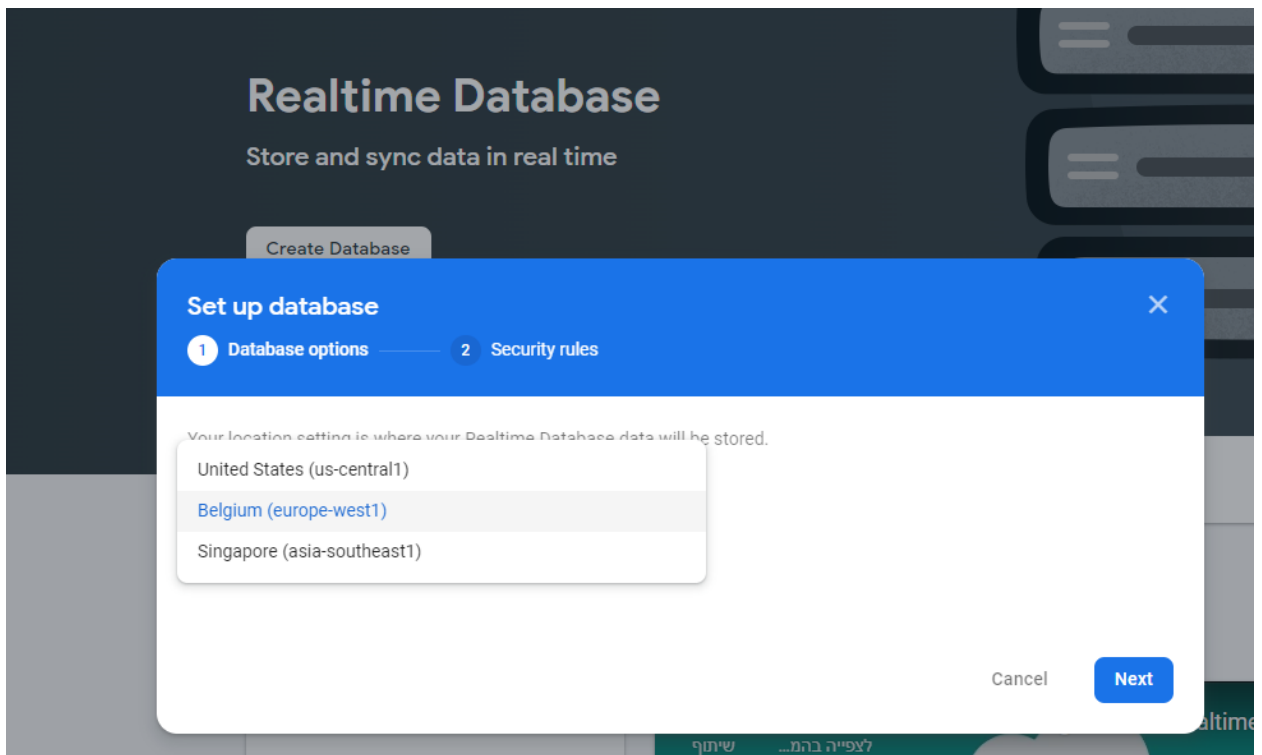
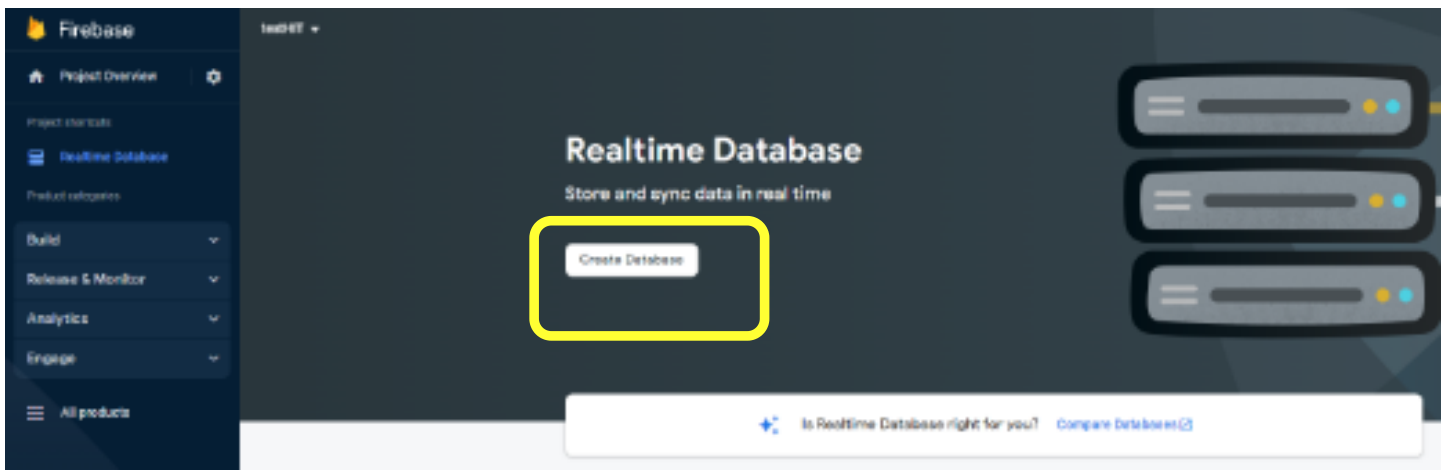
[Firestore console link.](#)



Step 2: Creating Realtime Database

Let's create our new Realtime database.
Under the Build section, we can find Realtime Database.
Go to the section and press create a database, choose your database location (Europe for us - Belgium, Ireland and etc)





Start in locked mode

And enable the database

We are not done yet.
We need to change the Rules of our new database.
In the Realtime database section now you can see our database.
You can see you have an Rules option on the top bar.

Realtime Database

Data **Rules** Backups Usage

Edit rules Monitor rules

★ Default security rules are locked from access

```
1 {
2   "rules": {
3     ".read": false,
4     ".write": false
5   }
6 }
```

We will change this config into this:

```
{
  "rules": {
    ".read": true,
    ".write": true,
    "Users": {
      ".indexOn": ["first_name", "last_name", "role"]
    }
  }
}
```

Rules playground

Learn more Dismiss

Realtime Database

Data **Rules** Backups Usage

Edit rules Monitor rules

Now we press Publish

unpublished changes **Publish** Discard

★ Default security rules are locked from access

```
1 {
2   "rules": {
3     ".read": true,
4     ".write": true,
5     "Users": {
6       ".indexOn": ["first_name", "last_name", "role"]
7     }
8   }
9 }
```

After that our Realtime database should look like this:

testHIT

Realtime Database

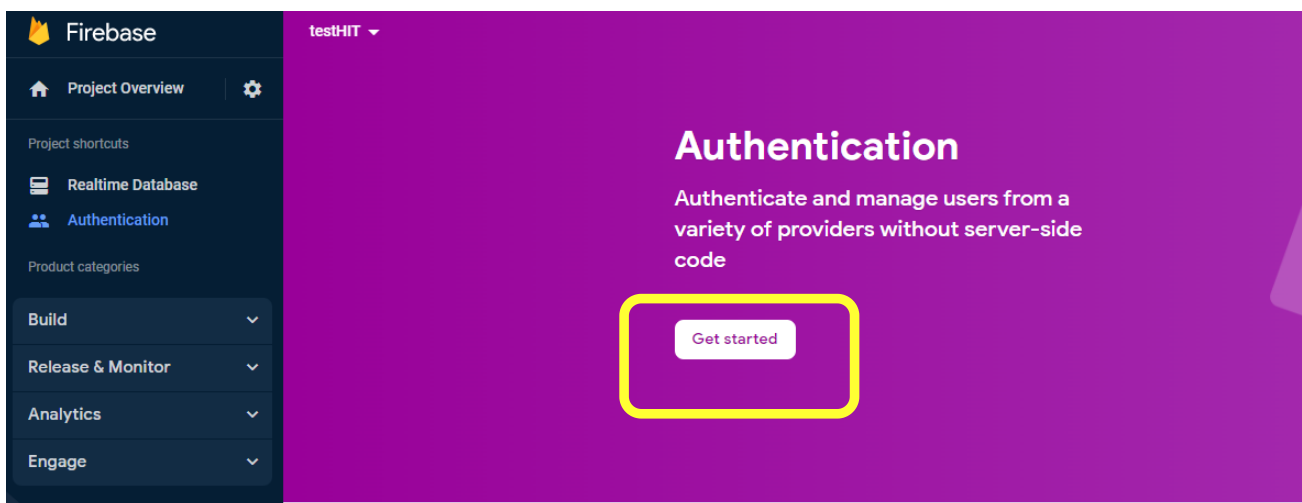
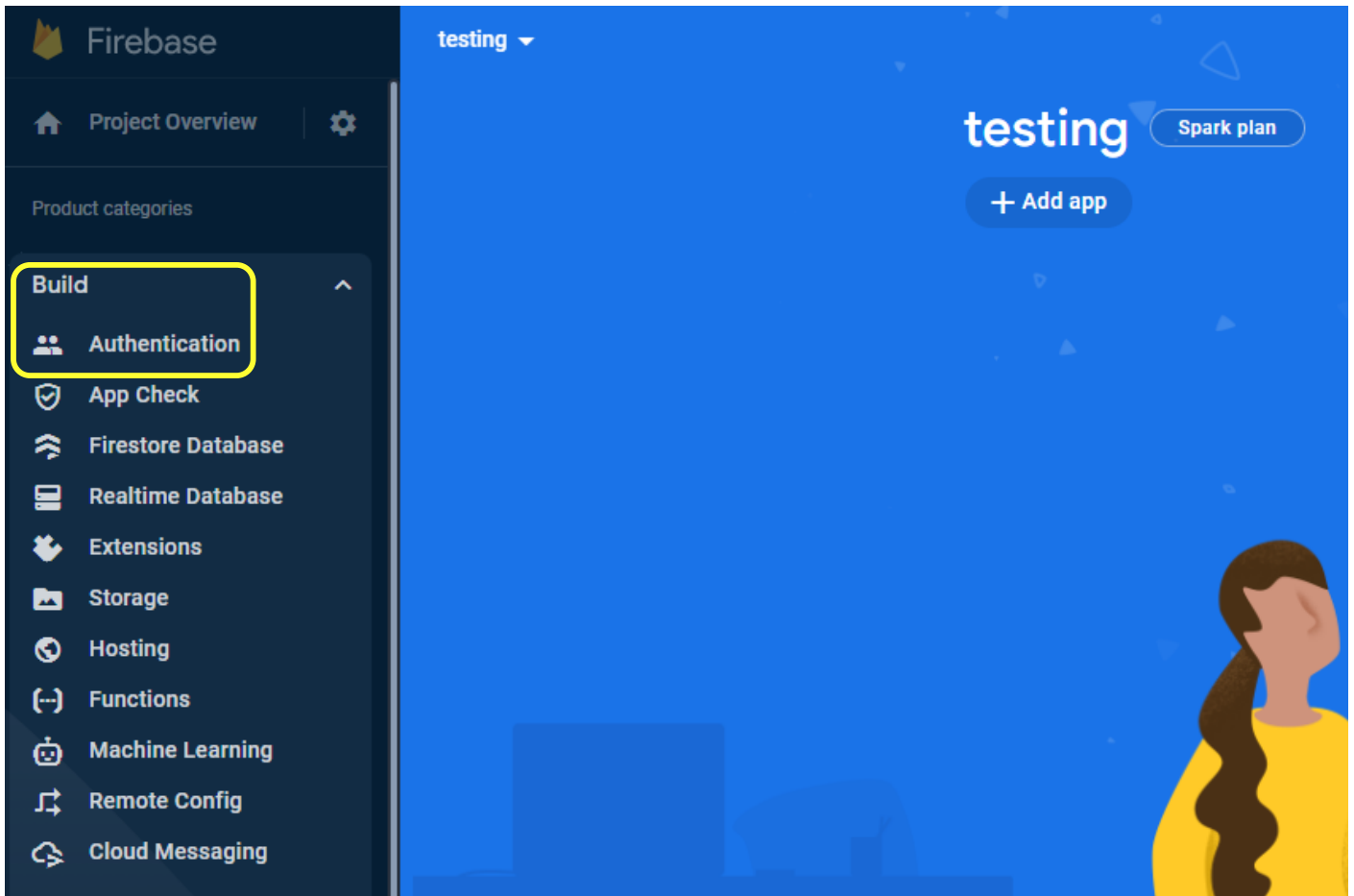
Data **Rules** Backups Usage

Protect your Realtime Database resources from abuse, such as billing fraud or phishing [Configure App Check](#) ✕

```
https://testhit-a46cc-default-rtdb.europe-west1.firebaseio.com
https://testhit-a46cc-default-rtdb.europe-west1.firebaseio.com/:null
```

Step 3: Setup Authentication

Under the Build section on the sidebar, you can find 'Authentication' press on it.



testing ▾

Authentication

Users Sign-in method Templates Usage Settings

Get started with Firebase Auth by adding your first sign-in method

Native providers

- Email/Password
- Phone
- Anonymous

Additional providers

- Google
- Game Center
- Microsoft
- Facebook
- Apple
- Twitter
- Play Games
- GitHub
- Yahoo

Custom providers

- OpenID Connect
- SAML

Advanced

- SMS Multi-factor Authentication

Here we want to choose Email/Password as our authentication method

Sign-in providers

Mark the first option to be Enable, and save.

Email/Password Enable

Allow users to sign up using their email address and password. Our SDKs also provide email address verification, password recovery, and email address change primitives. [Learn more](#)

Email link (passwordless sign-in) Enable

Cancel **Save**

After that our Authentication page should look like this:
With Add user button

testHIT ▾

Authentication

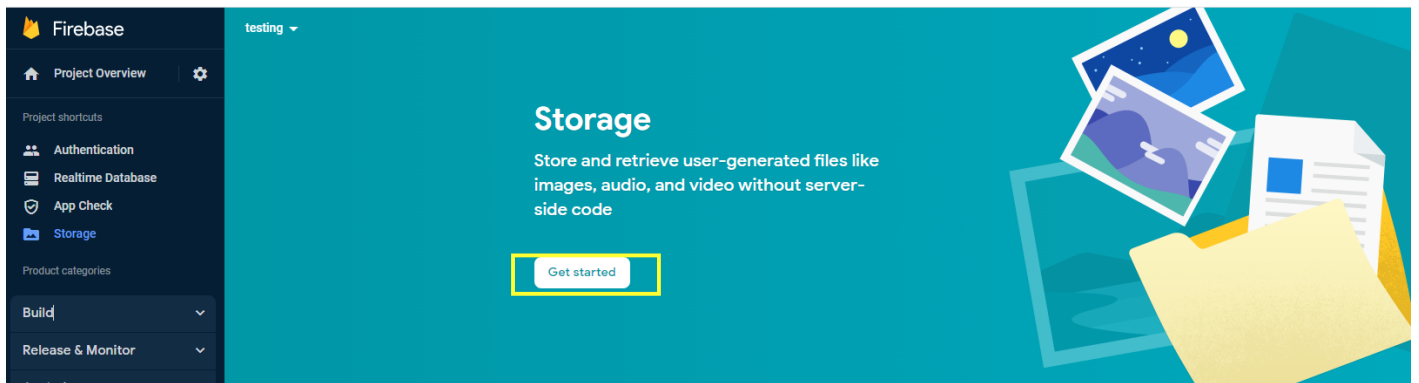
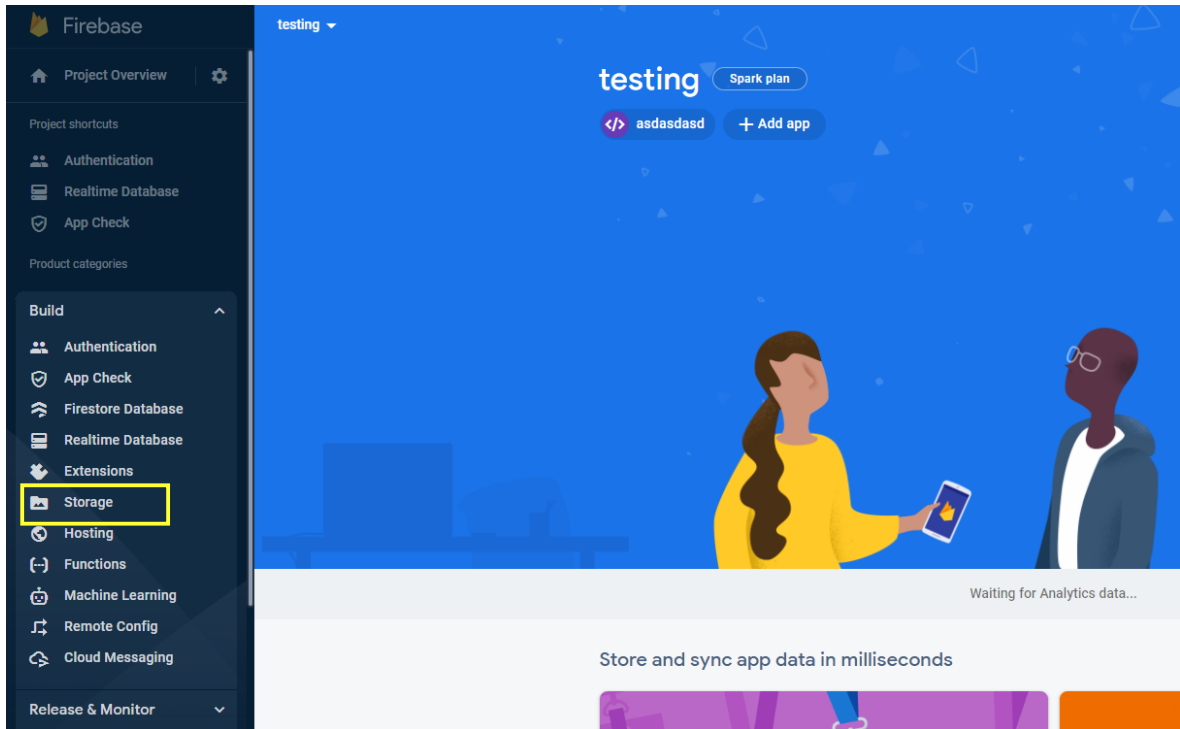
Users Sign-in method Templates Usage Settings

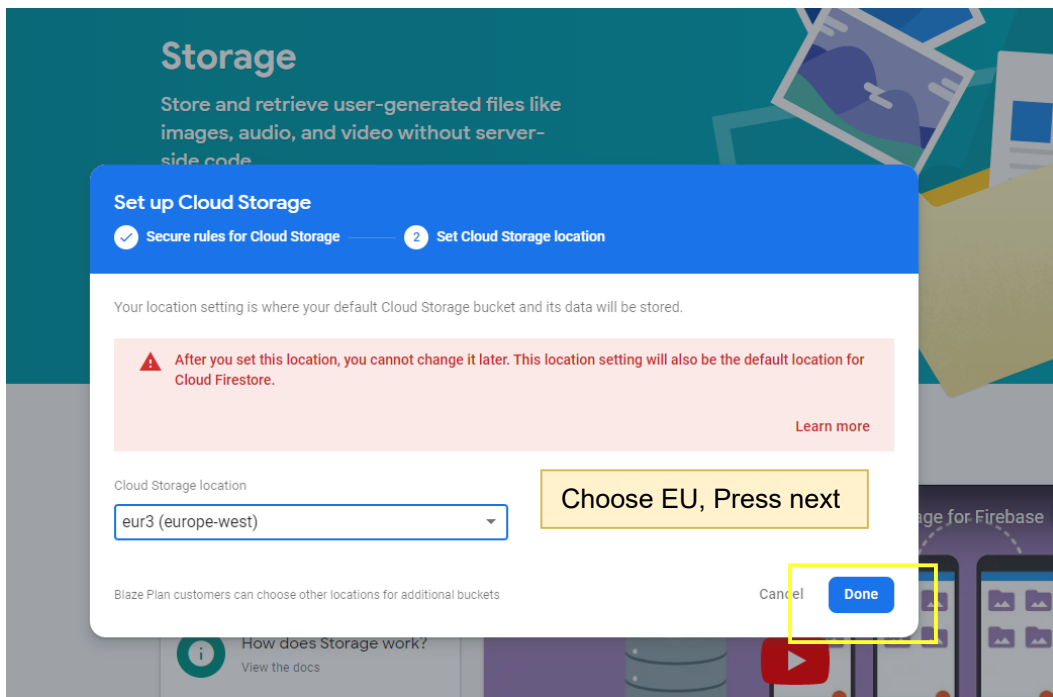
Add user ↻ ⋮

Identifier	Providers	Created ↓	Signed In	User UID
No users for this project yet				

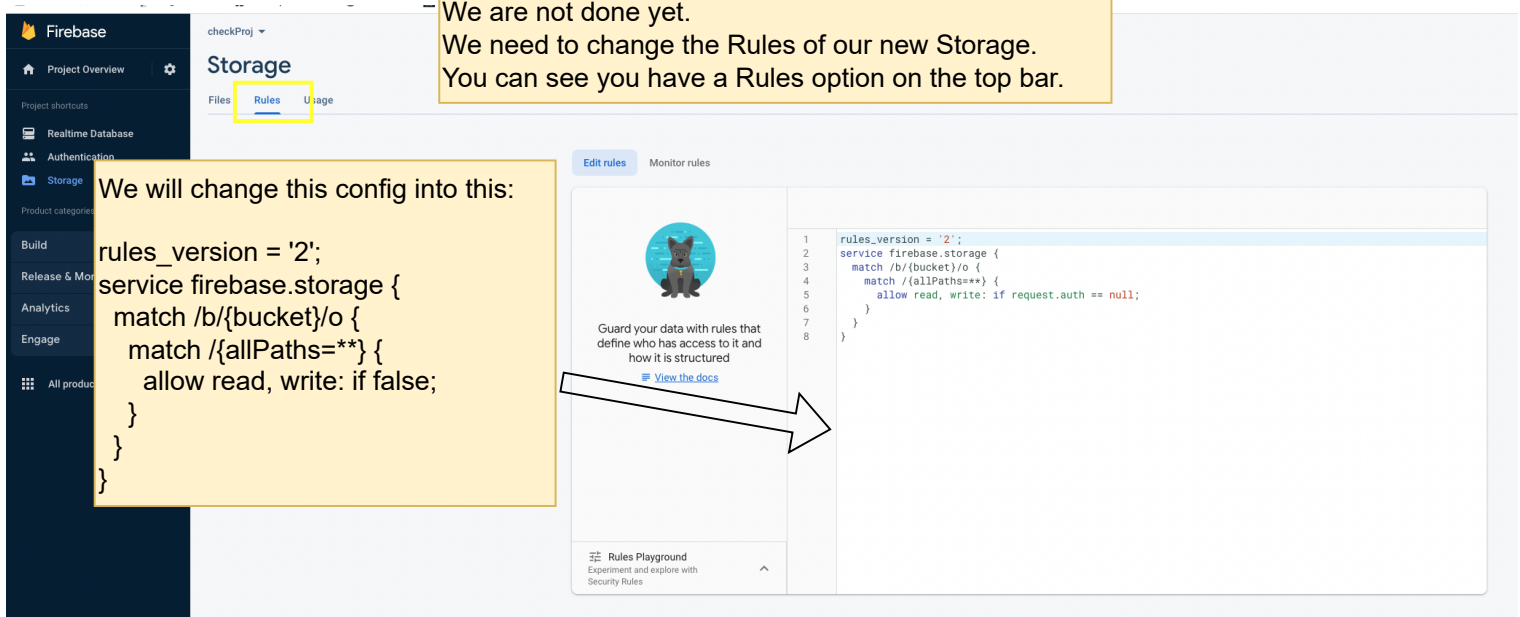
Step 4: Steup storage

Under the Build section on the sidebar, you can find 'Storage' press on it.





We are not done yet.
We need to change the Rules of our new Storage.
You can see you have a Rules option on the top bar.

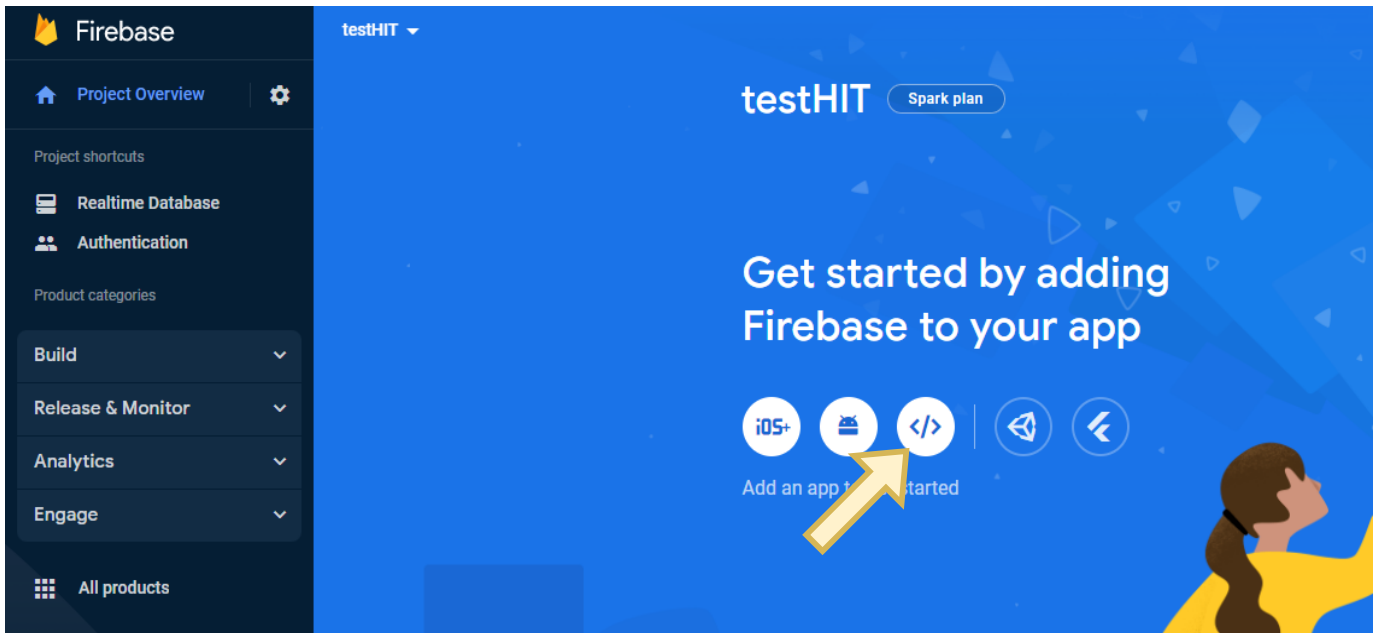


Step 5: Create a web app for the new project

Now you are in the new project console, we want to create a new web app.

Press on the firebase Logo in the top left corner will redirect to the project console page.

Now can choose to create our new web app.



2 Add Firebase SDK

Use npm [?](#) Use a <script> tag [?](#)

If you're already using [npm](#) and a module bundler such as [webpack](#) or [Rollup](#), you can run the following command to install the latest SDK:

```
$ npm install firebase
```

Create new web app, enter name and generate the app.
In the end you will get this config file.

Then, initialize Firebase and begin using the SDKs for the products you'd like to use.

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// 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: "AIzaSyAkKU3tIrfkNEsqf_puOpNADyW40juaLpg",
  authDomain: "testhit-a46cc.firebaseio.com",
  databaseURL: "https://testhit-a46cc-default-rtdb.europe-west1.firebaseio.com",
  projectId: "testhit-a46cc",
  storageBucket: "testhit-a46cc.appspot.com",
  messagingSenderId: "692378054216",
  appId: "1:692378054216:web:e0744ebed31f1c6f4cb27d",
  measurementId: "G-M5X52G2YFQ"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
```

We will copy this section without the brackets !
We should make sure we have the next keys:
apiKey, authDomain, databaseURL, storageBucket

Note: This option uses the [modular JavaScript SDK](#), which provides reduced SDK size.

Learn more about Firebase for web: [Get Started](#), [Web SDK API Reference](#), [Samples](#)

We will copy this config into our clinic edit form and paste it

The screenshot shows the 'uilib' clinic configuration form in the Firebase console. The form includes fields for 'Email Address' (uilib@test.com) and 'Clinic Name' (uilib). Under 'Firebase Properties:', there is a 'Firebase Project Init Guide' dropdown and a 'Firebase Config' field containing the following JSON:

```
rtdb.europe-west1.firebaseio.com",  
projectId: "generic-management-hit",  
storageBucket: "generic-management-hit.appspot.com",  
messagingSenderId: "110025314903",  
appId: "1:110025314903:web:77264c5402735b14670912",  
measurementId: "G-5QGGTD28PV"
```

Below the config is an 'Api key post' field. Under 'Chat Firebase Properties:', there is a 'Firebase Chat Config' field with instructions to copy chat firebase config properties from the app settings.

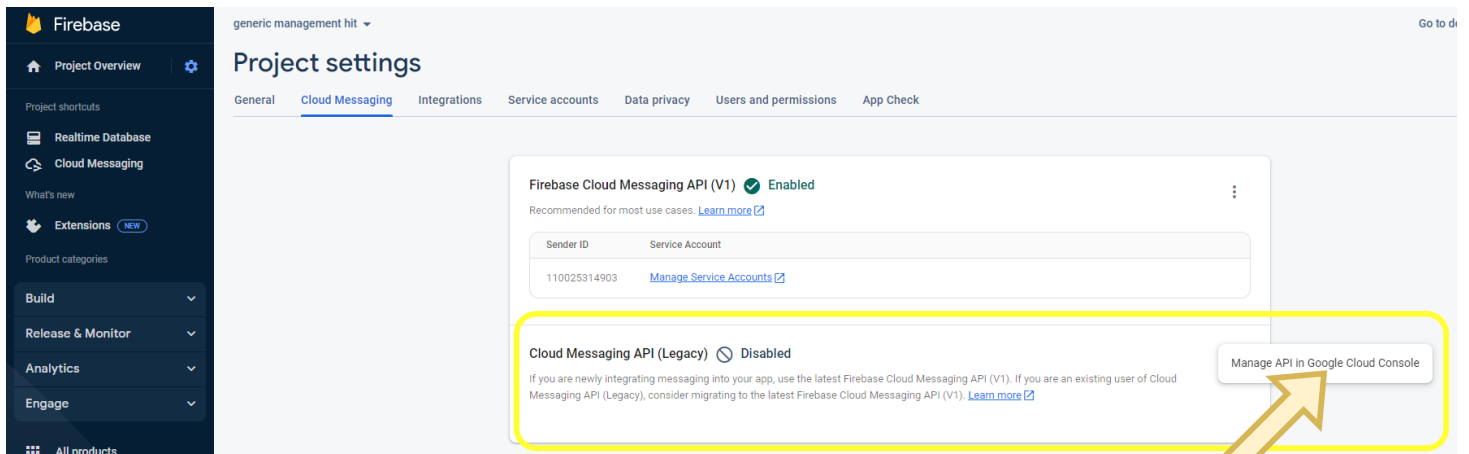
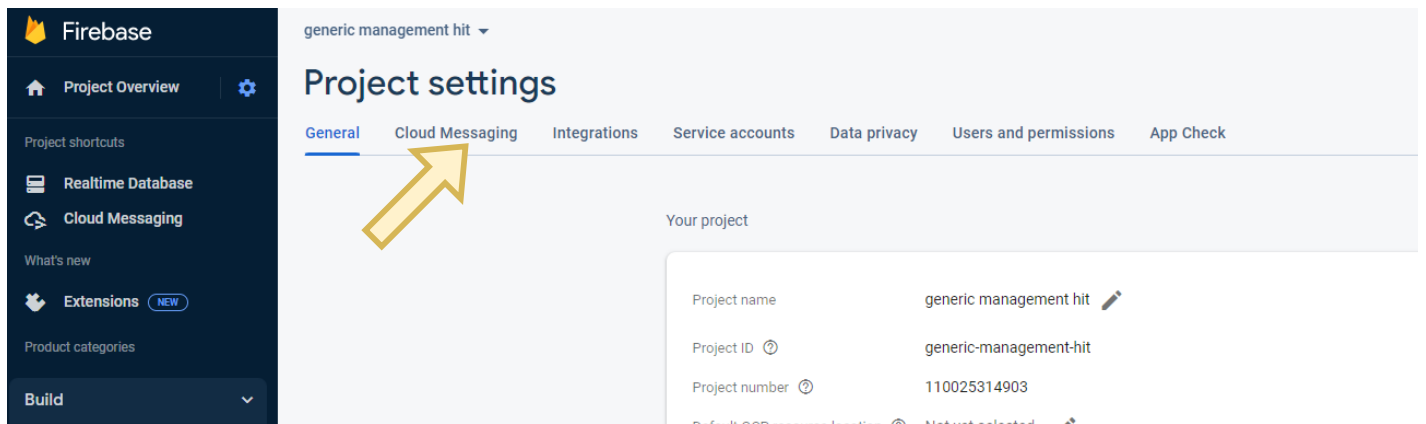
Step 6: Create second Firebase for Chat

Now we will create another firebase as mentioned. This time we will create a web app and an android app.

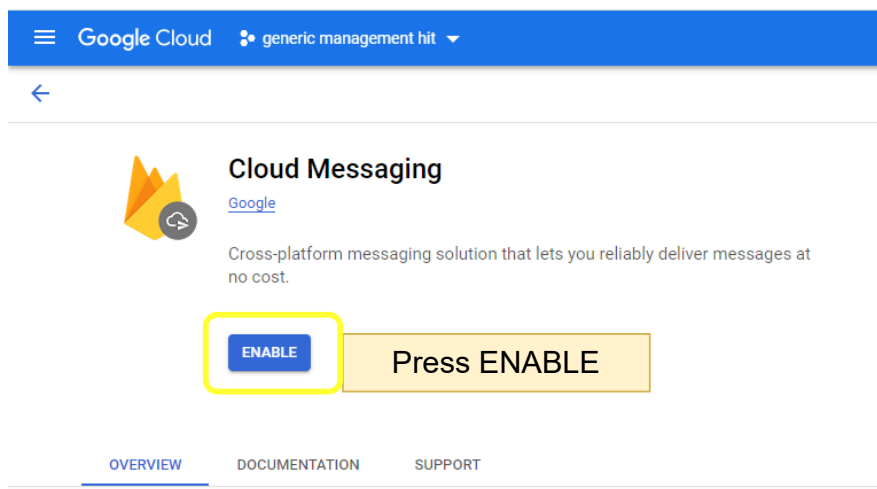
Create new android app

The screenshot shows the 'generic management hit' project page in the Firebase console. A yellow arrow points to the '+ Add app' button. A text box with the text 'Press add app' is overlaid on the image.

Now we will generate cloud messaging API key
Under project settings we can find Cloud Messaging



Now we want to enable cloud messaging.
In the right side of the Cloud Messaging API we will have 3 dots we will press on them and then on the button that will shown



Now we have cloud messaging key and we will copy it to the clinic form

The screenshot shows the 'Project settings' page for a Firebase project named 'generic management hit'. The 'Cloud Messaging' tab is selected. It displays two API configurations: 'Firebase Cloud Messaging API (V1)' and 'Cloud Messaging API (Legacy)', both of which are 'Enabled'. The 'Legacy' API section is highlighted with a yellow box, showing a 'Server key' with a long alphanumeric string. A blue 'Add server key' button is visible at the bottom right of the configuration area.

This screenshot shows a Django management console. On the left, there is a table with columns 'Clinic Name', 'Clinic Email', and 'Account'. The table contains five rows of clinic data. On the right, a 'Chat Firebase Properties' form is displayed. It includes sections for 'Firebase Config' and 'Api key post', both containing configuration details. A yellow arrow points from the 'Server key' in the previous screenshot to the 'Api key post' field in this form.

Clinic Name	Clinic Email	Account
clinic1	clinic1@example.com	✓
riri	riri@example.com	✓
clinic3	clinic3@example.com	✓
samuel	samuel@example.com	✓
uilib	uilib@test.com	✓

Now can choose the applications and the measurements for this project.

Make sure to mark the Firebase Props Changed check box

Now we can submit, the management project will generate a new Django project, and will register the clinic into the DB.

This is a close-up of the 'Chat Firebase Properties' form. It shows the 'Firebase Chat Config' section with a 'storageBucket' field. Below this, there are checkboxes for 'accounts', 'chat', 'dashboard', 'documents_share', 'measurements', 'medications', and 'questionnaire'. The 'accounts' and 'chat' checkboxes are checked. There is a 'Save Measurement' button and a 'Measurements' dropdown menu. At the bottom, a yellow box contains the text 'If you set the Firebase props please check this box:' followed by a checked checkbox labeled 'Firebase Props Changed'. A blue 'Submit' button is at the very bottom. A yellow arrow points to the 'Submit' button.

Clinics

Generating Project . . .

Clinic Name	Clinic Email	Accounts	Dashboard	Measurements	Medications	Questionnaire	Chat	Edit
react123	test@example.com	✓	✓	✓	✓	X	X	
ui-lib	uilib@test.com	✓	✓	✓	X	X	✓	
mvt	memberview@test.com	✓	✓	✓	✓	✓	✓	
spaces	spaces@test.com	✓	✓	X	X	✓	✓	
owners	owners@test.com	✓	✓	X	X	X	✓	
samuel	samuel@exmaple.com	✓	✓	X	X	X	X	

At the end of the process you will see this success message appears. In this message you can see the password for the clinic to login to the new project the username will be the clinic email.

Clinics

Finished to create your new project!

The password for clinic firebase login: **Password**
'+kz/F8BhC

Close

Clinic Name	Clinic Email	Accounts	Dashboard	Measurements	Medications	Questionnaire	Chat	Edit
react123	test@example.com	✓	✓	✓	✓	X	X	
ui-lib	uilib@test.com	✓	✓	✓	X	X	✓	
mvt	memberview@test.com	✓	✓	✓	✓	✓	✓	
spaces	spaces@test.com	✓	✓	X	X	✓	✓	
owners	owners@test.com	✓	✓	X	X	X	✓	
samuel	samuel@exmaple.com	✓	✓	X	✓	✓	X	

Finally the new clinic DB should look like this:

testHIT ▾

Realtime Database

Data Rules Backups Usage

Protect your Realtime Database resources from abuse, such as

<https://testhit-a46cc-default-rtdb.europe-west1.firebaseio.com>

- Features
- Users
- checkMes

And the Authentication:

testHIT ▾

Authentication



Project shortcuts


- Realtime Database
- Authentication



Product categories

- Build
- Release & Monitor
- Analytics
- Engage


Users Sign-in method Templates Usage Settings

Search by email address, phone number, or user UID [Add user](#)  

Identifier	Providers	Created ↓	Signed In	User UID
samuel@exmaple.com		Sep 3, 2022	Sep 3, 2022	a7azaAF40Mc5dMcUW6cju3Eib5...

Rows per page: 50 1 - 1 of 1  

And the folder created under the clinic email name

Name	Date modified	Type	Size
 samuel@exmaple.com	13/08/2022 20:34	File folder	