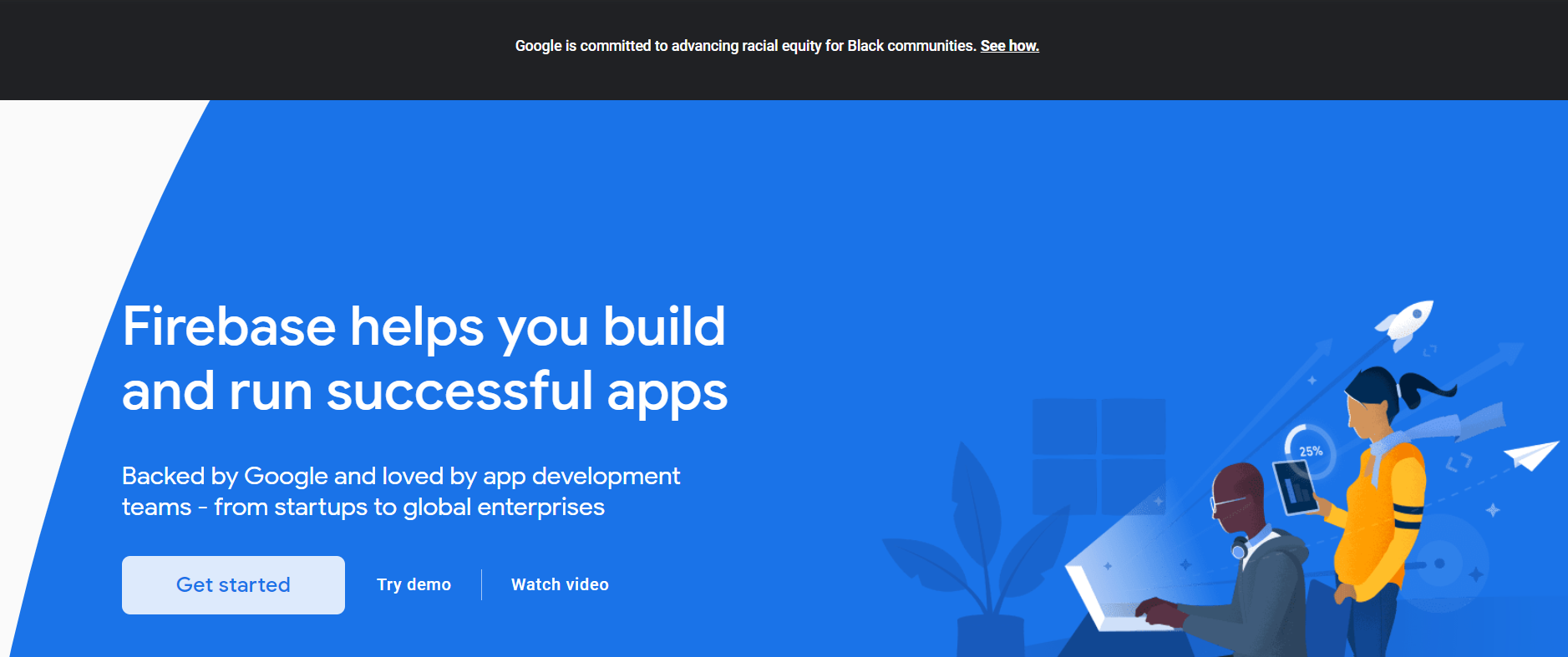
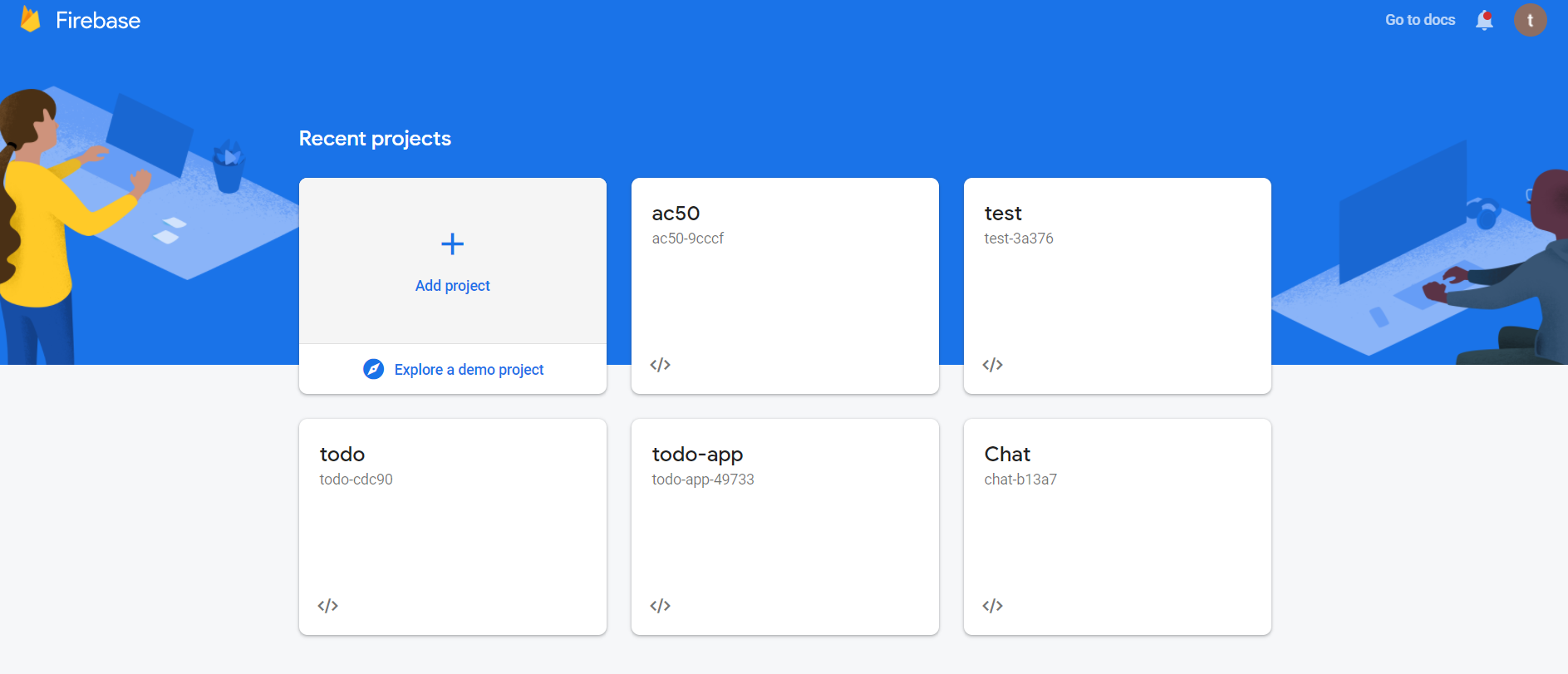
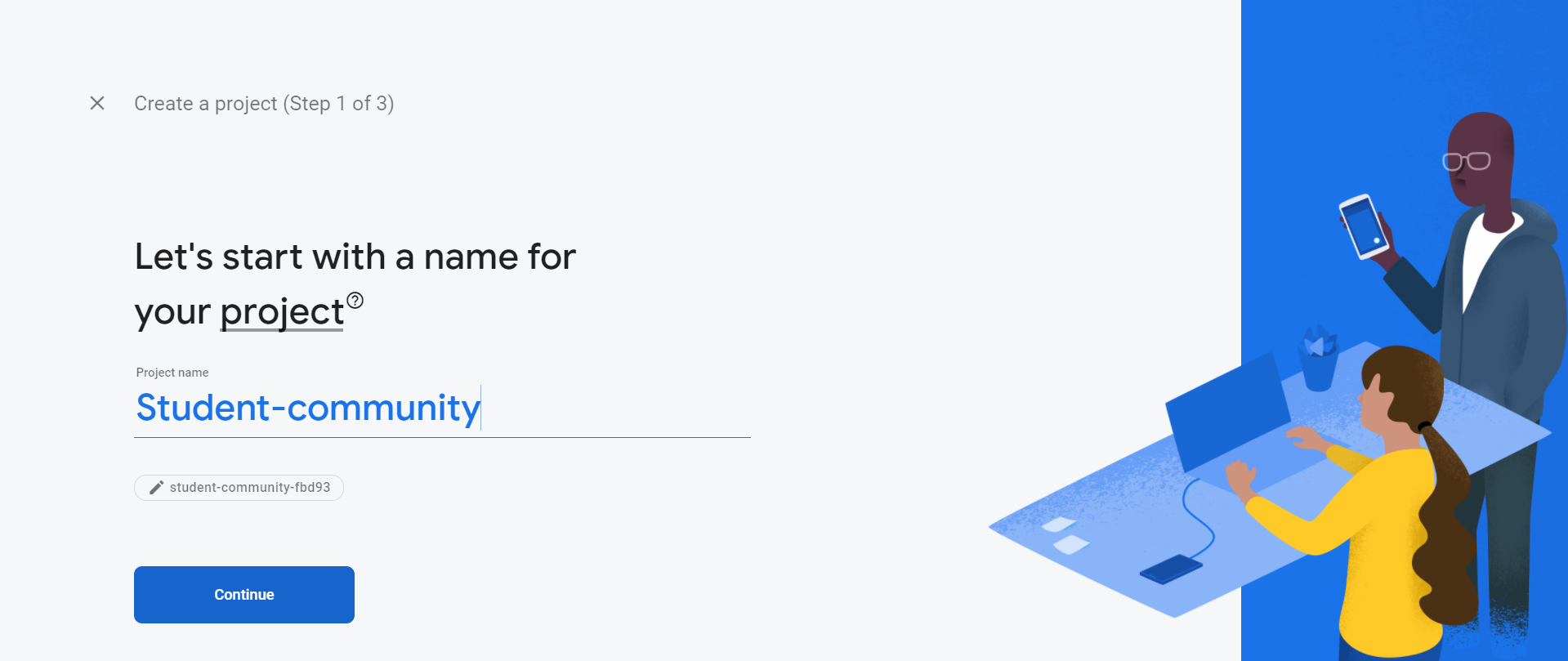
**Firebase Authentication for Login and Sign-up.**



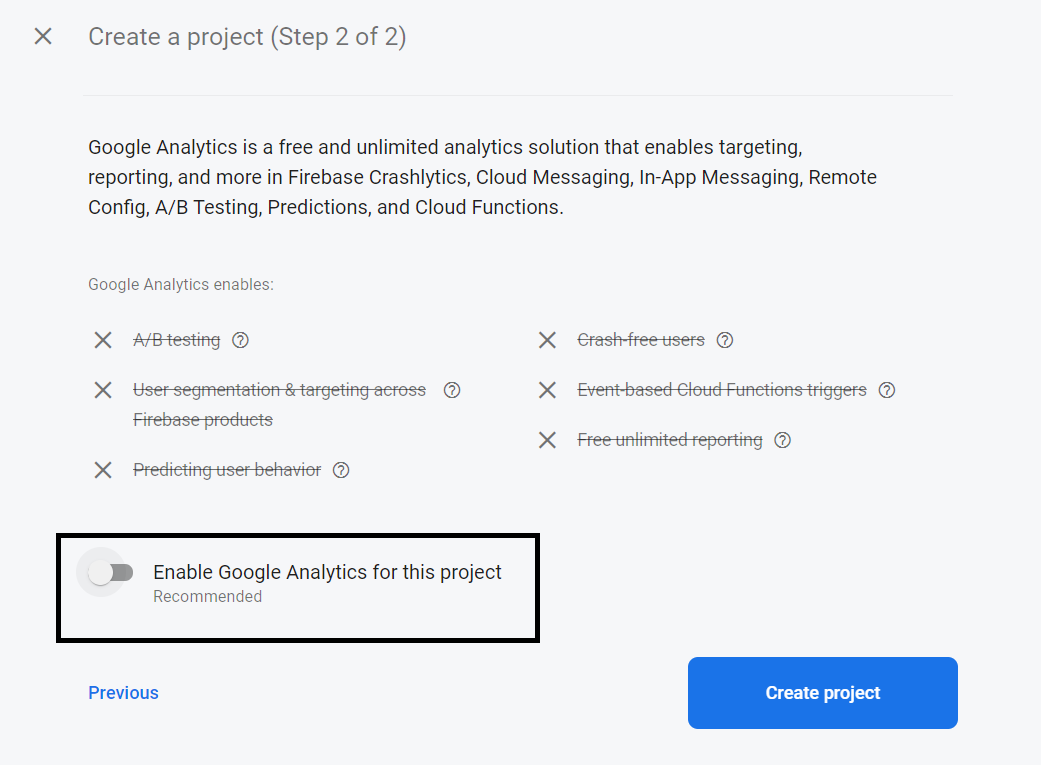
Go to Firebase and then click on **Get started**.



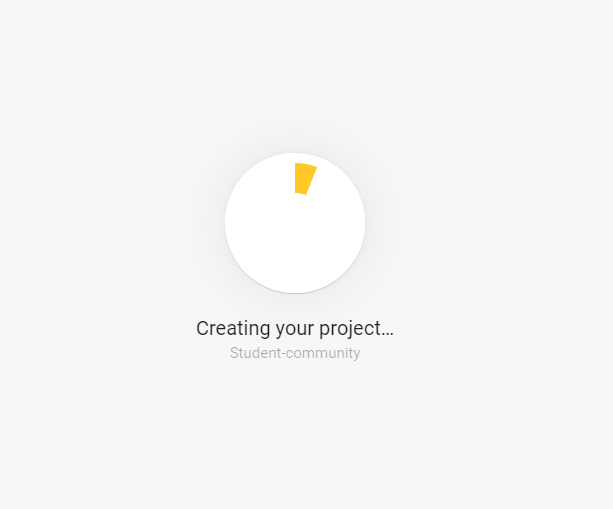
Then click on **Add project** to start a new project.



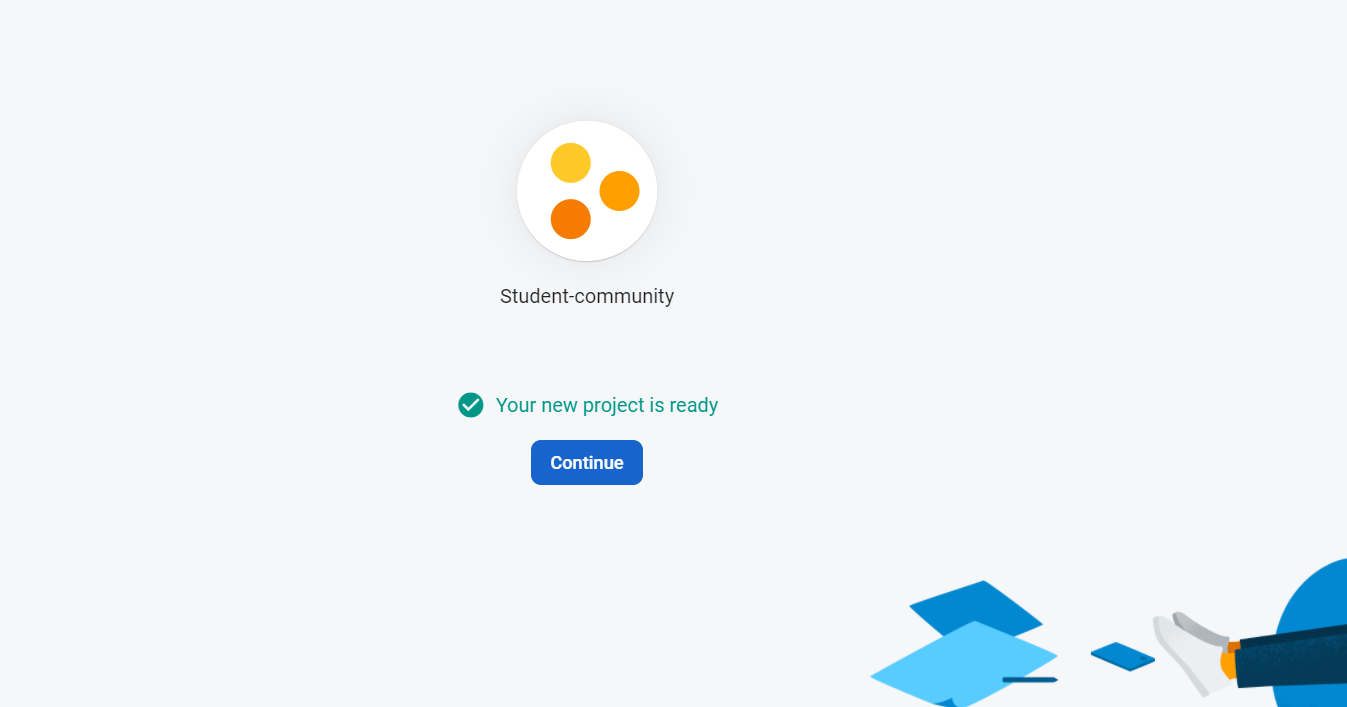
Give your Project a name, and then click on Continue.



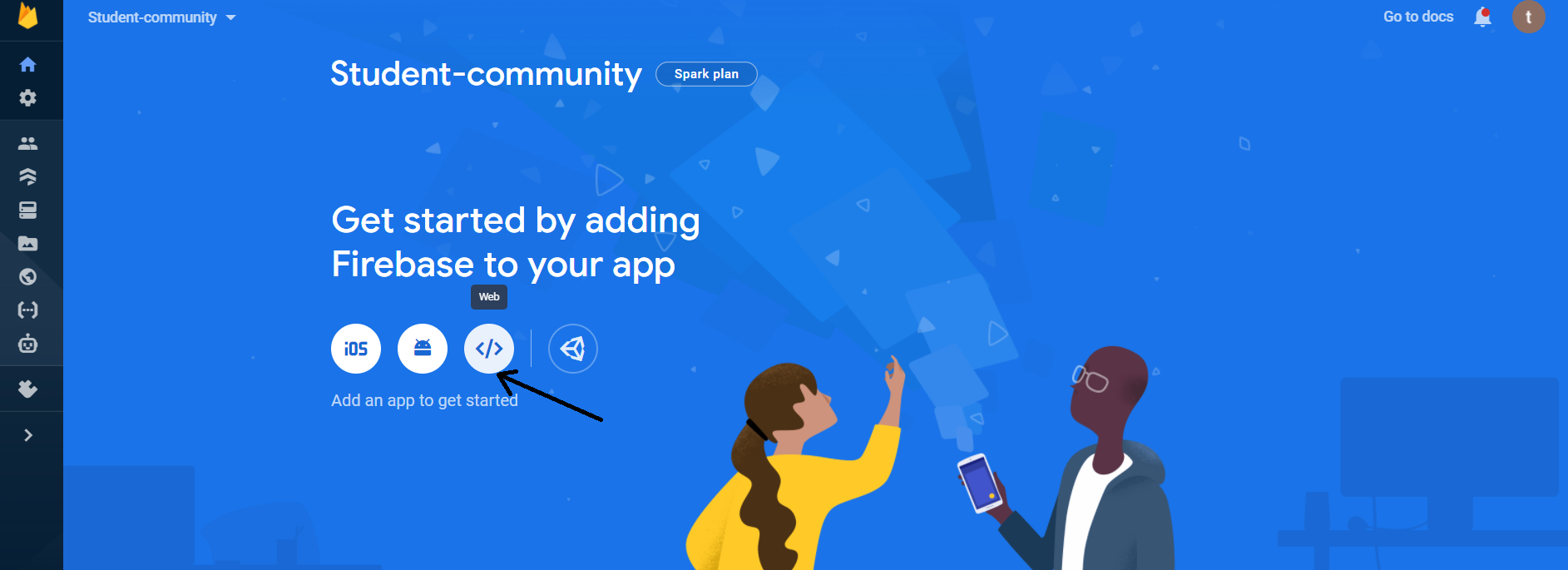
**Disable** the switch, and click on **Create Project**.



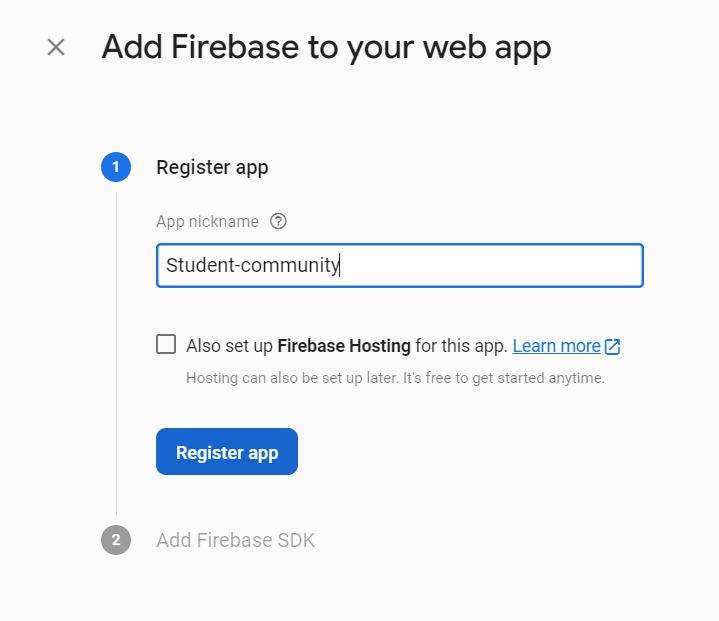
It will take some time to create the project.



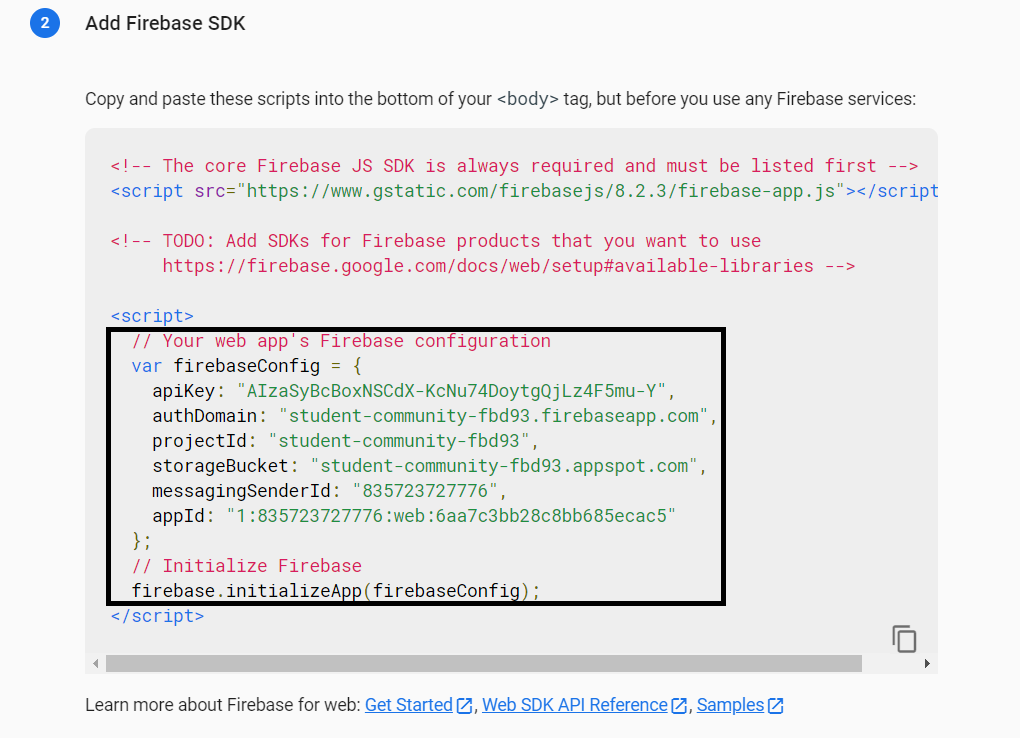
Once your project is ready, click on **Continue**.



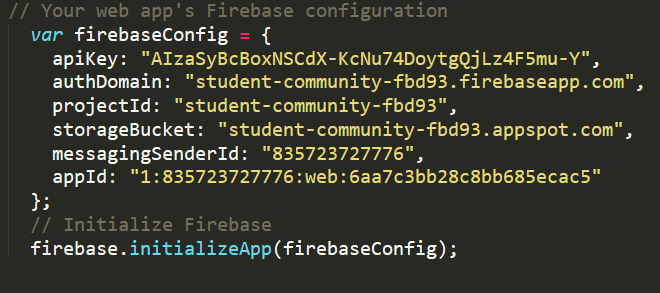
Click on the Web icon.



Now, here, we have to register our app. Add a nickname to the app and then click on **Register App**.



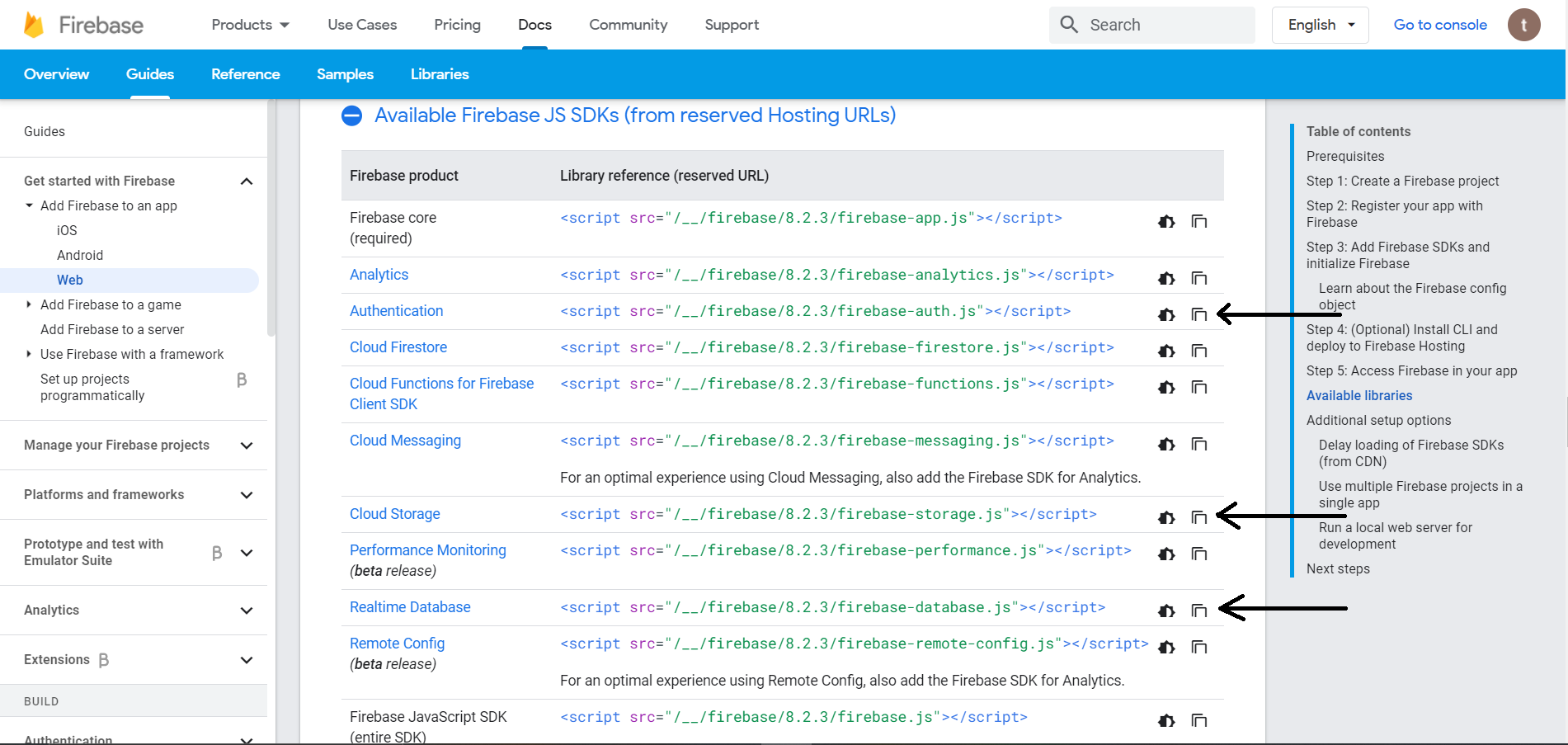
Then you will see the above code snippet, **copy** the firebaseConfig code and **paste** it into your javascript file app.js



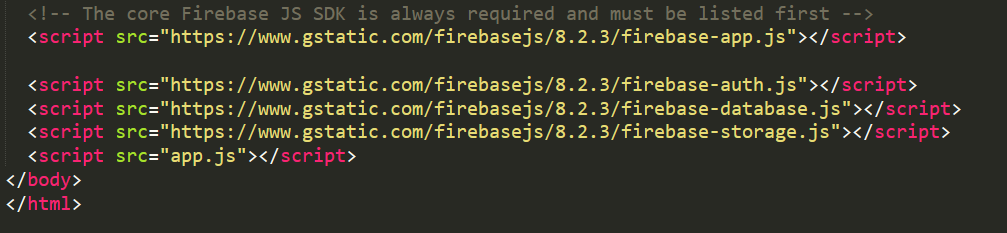
Paste the code in app.js



Then, go back to your firebase and **copy the first script tag and it’s contents** and paste it into all the html pages.



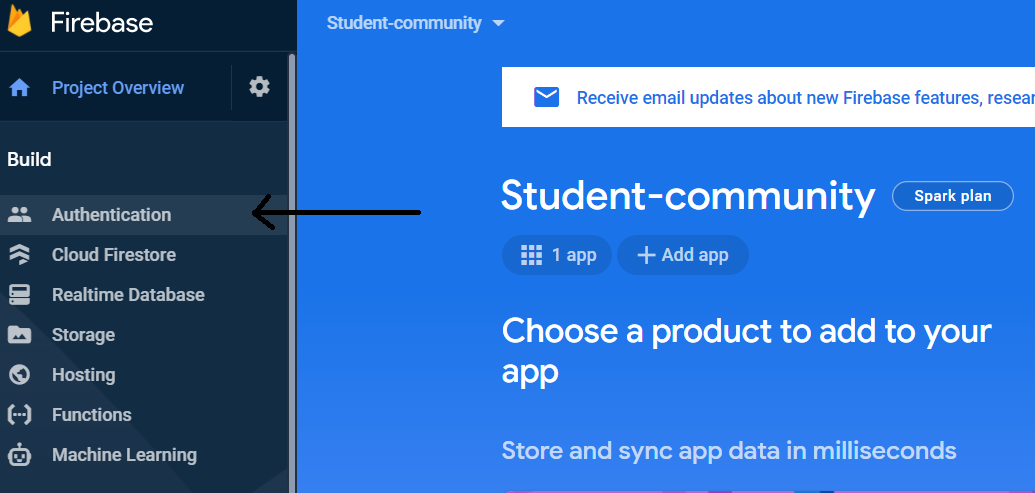
We also have to copy these **3** script tags for **firebase-auth**, **firebase-database**, **firebase-storage**.



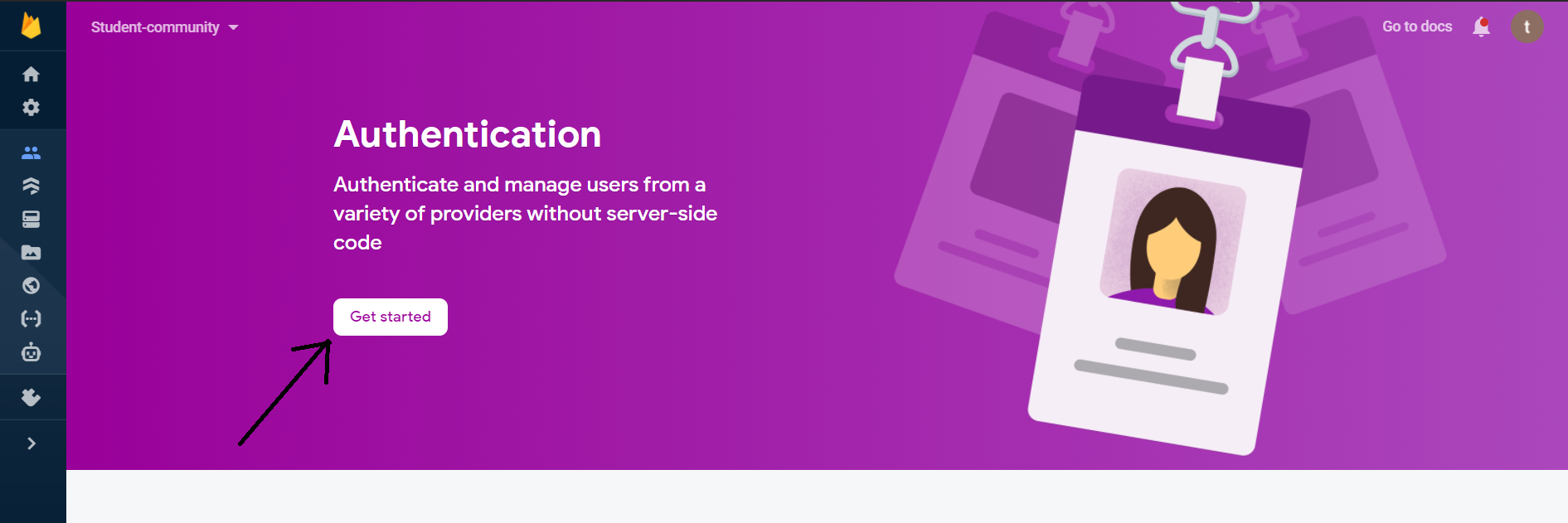
And **paste** it into the html pages.



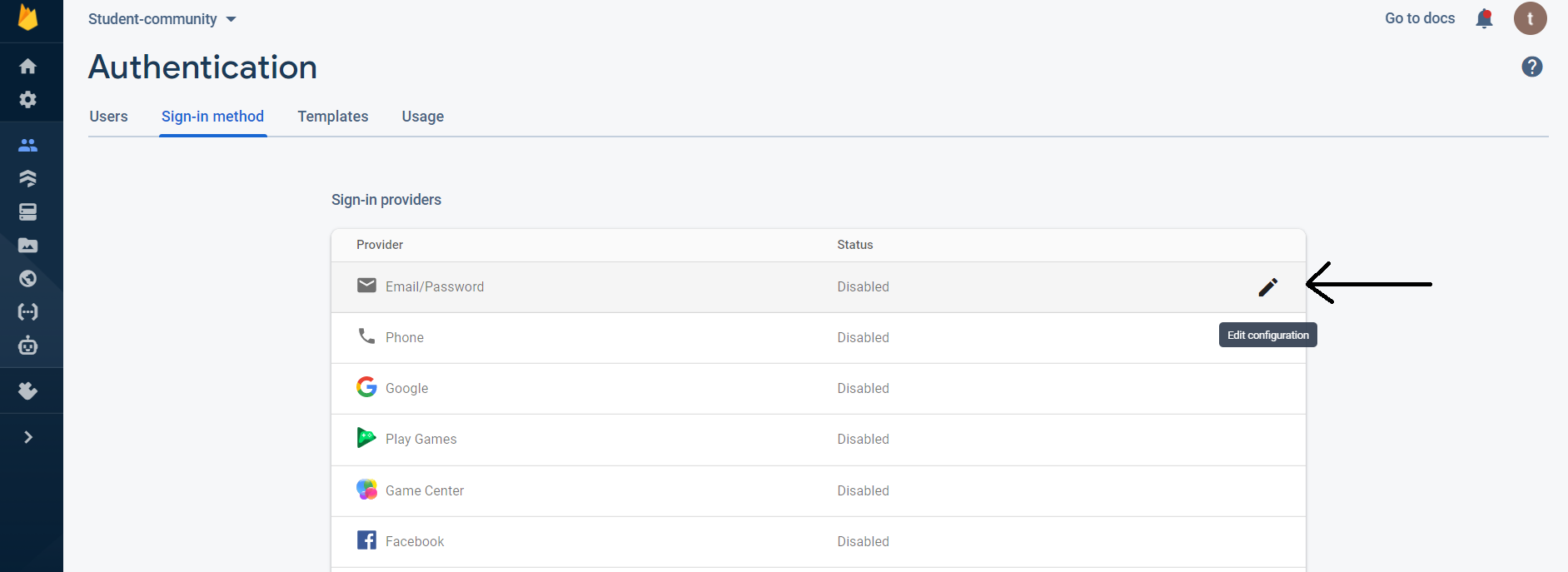
Then click on **Continue to console**.



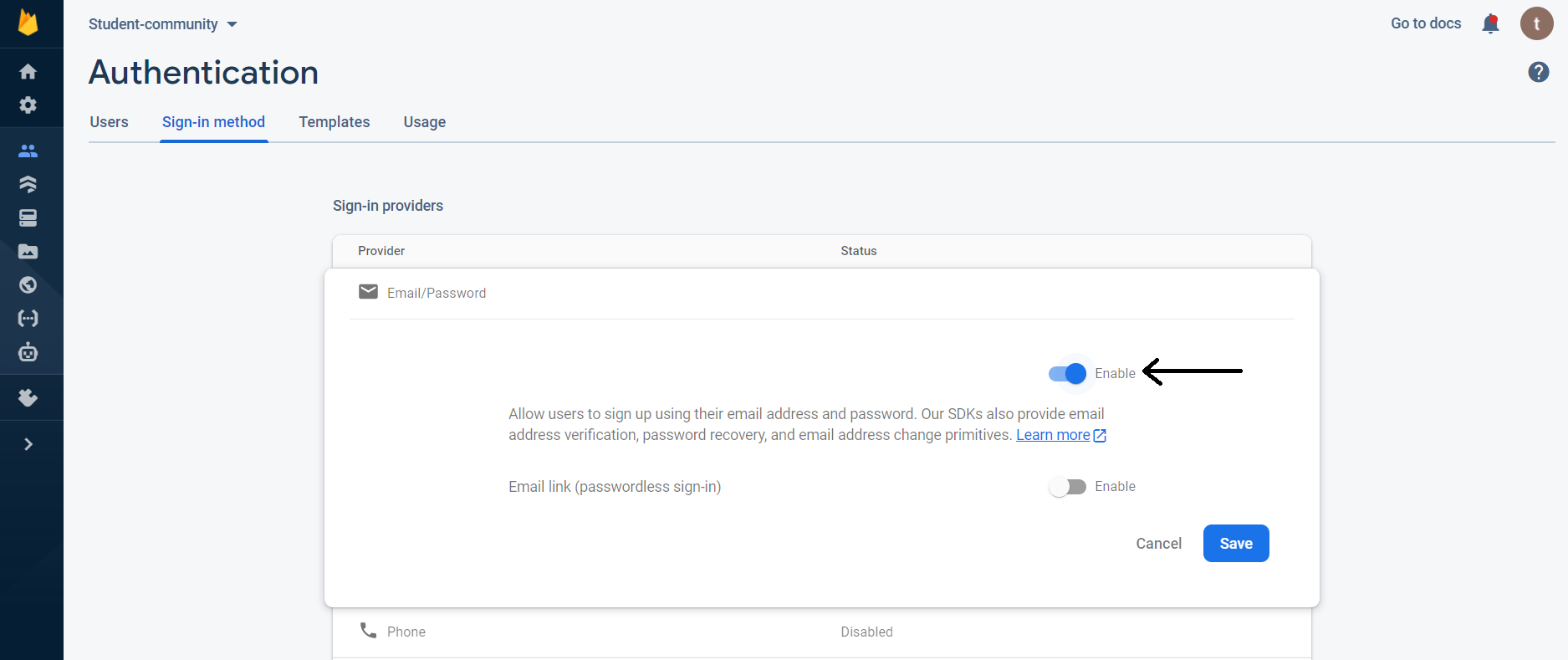
Then click on **Authentication**, to go to the authentication page.

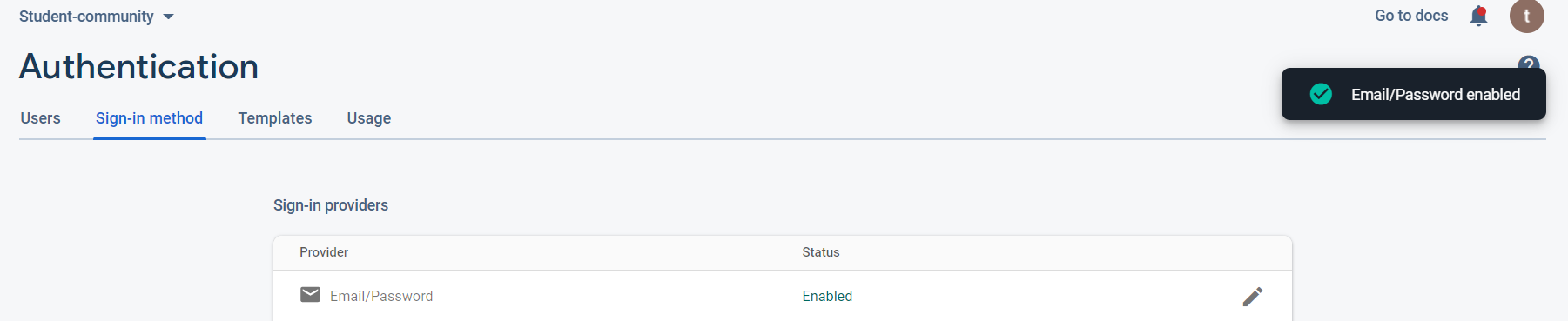


Click on **Get started**.

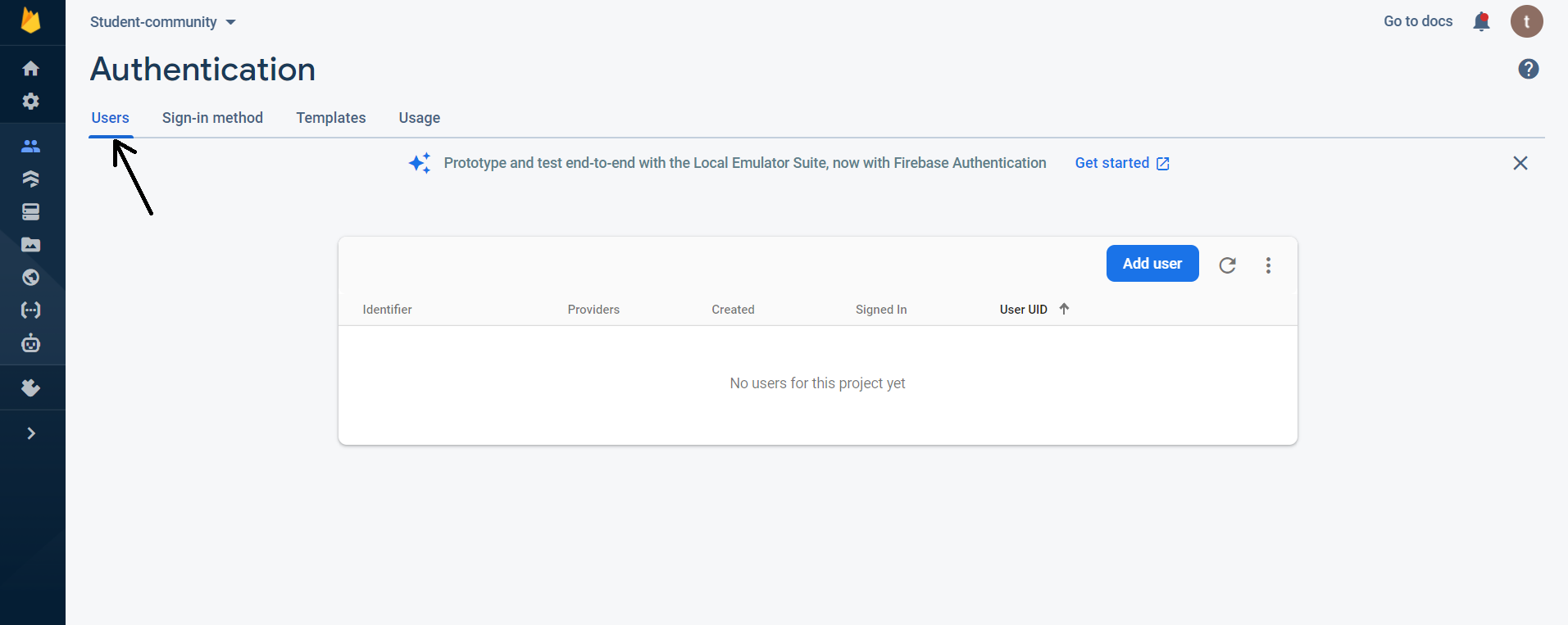


You will see this Sign-in method page, here click on the **Edit configuration** **icon**.

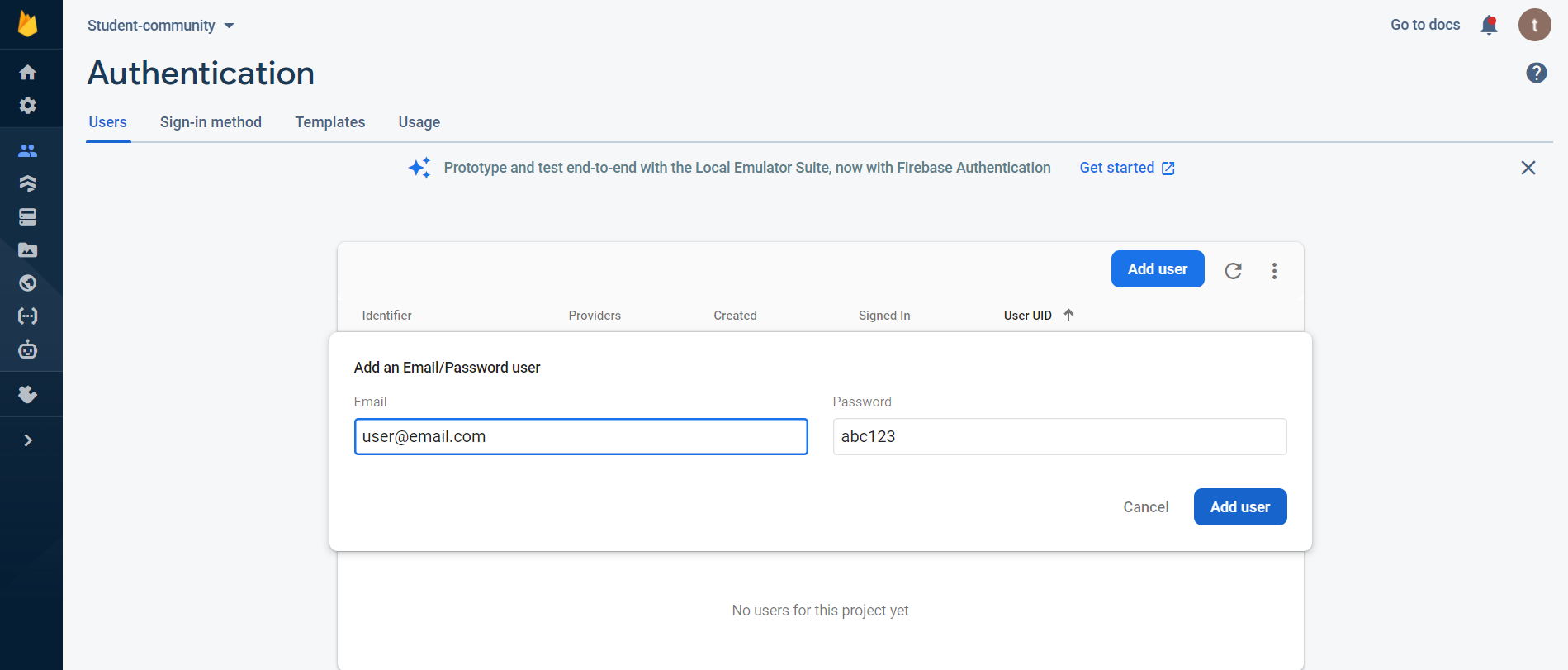
Here, switch the first button to **Enable**, to enable sign-up using email and password. And then click on **Save**.



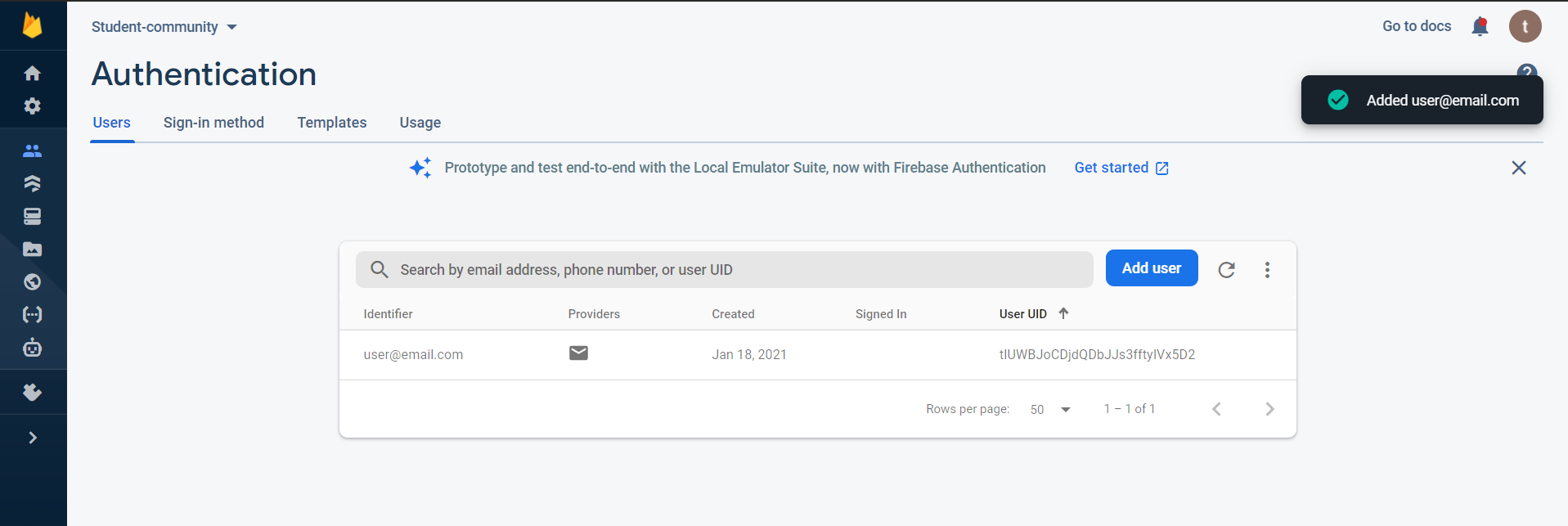
If it’s successfully done, you will see that it is Enabled.



Then go to **Users**, and click on **Add user**.



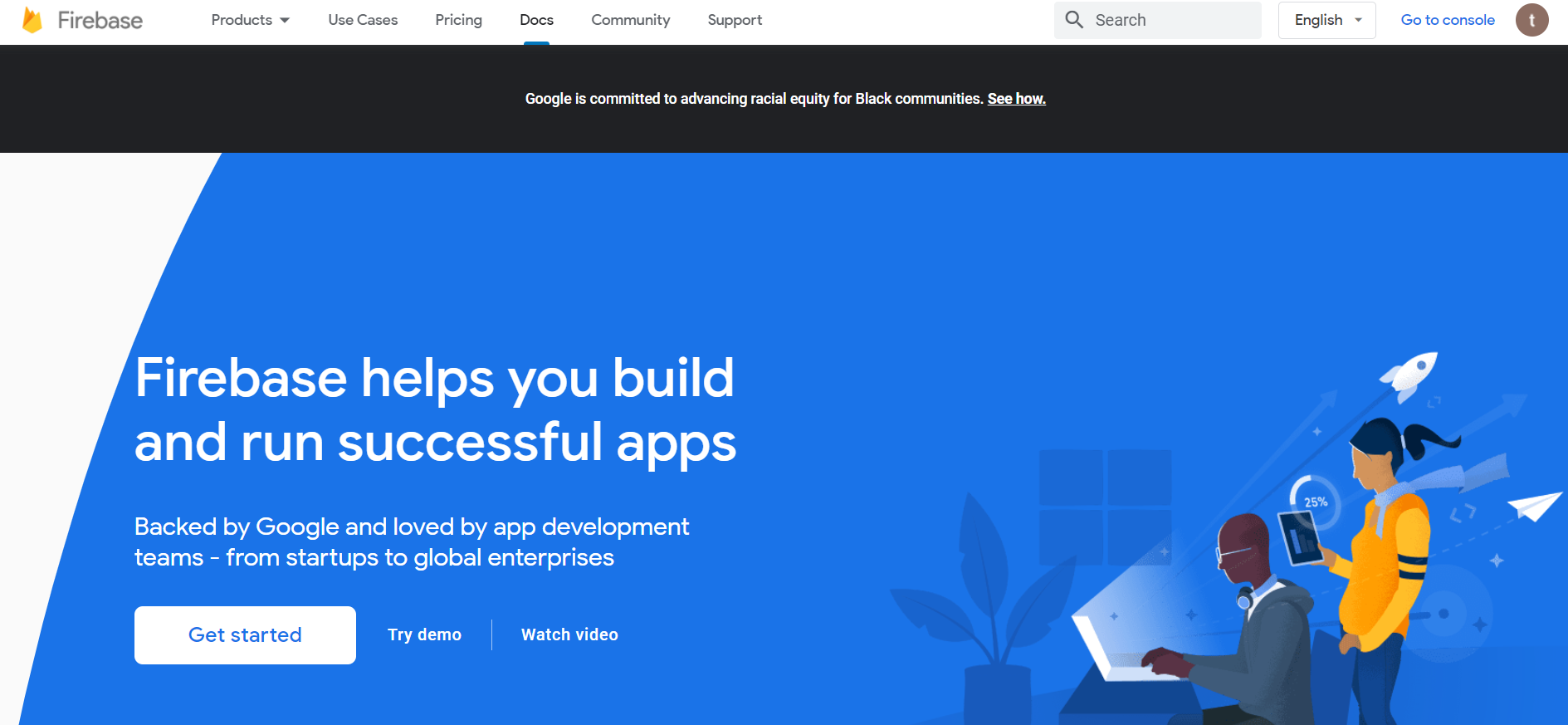
Add a user **email** and **password**, and click on Add user.



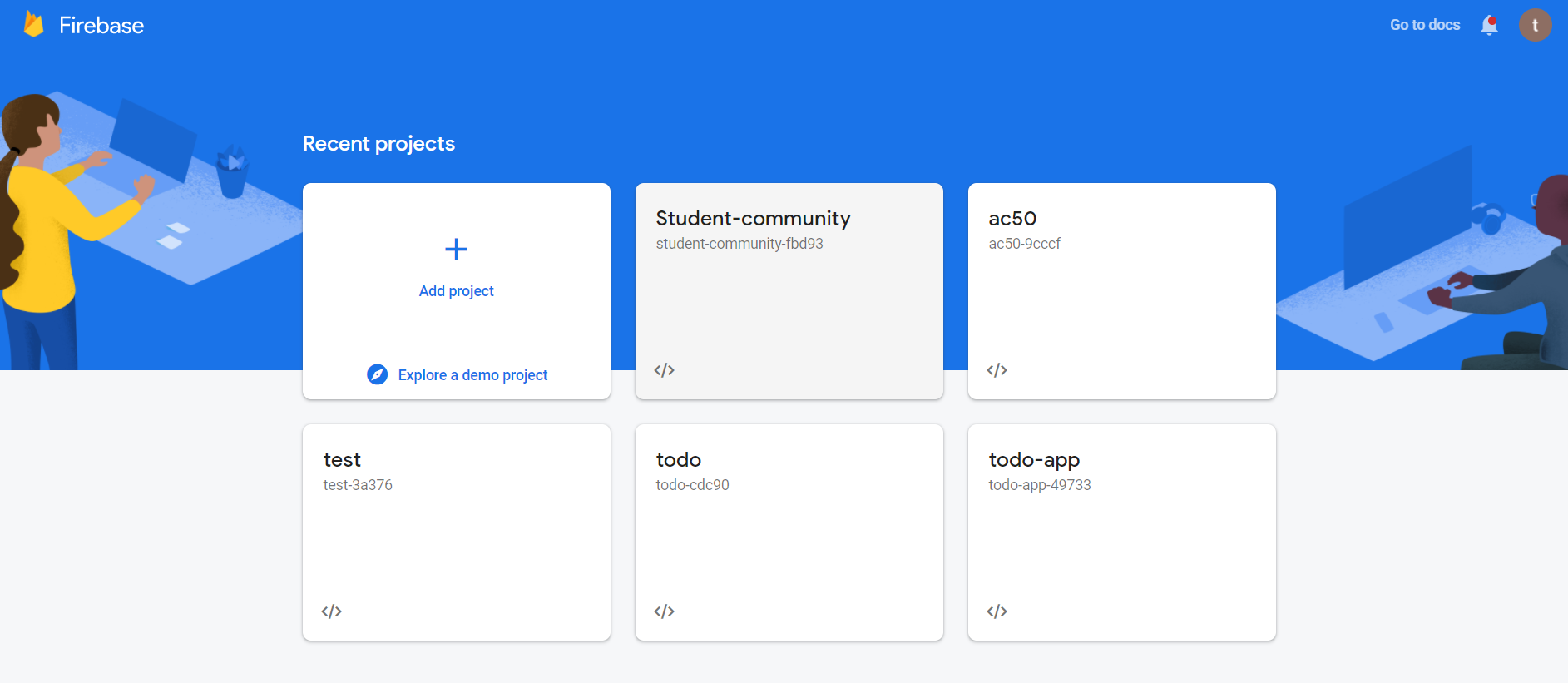
If a user has been successfully added, you will see that it has been added in the Users list.

Now, in the app.js we already have our firebase configurations so now we have to make a **login function**, then we can test its functionality on the login page with the user that we already added in the firebase database. If the login was successful, then we will proceed further and create the **signup function**, so that we can add users directly from our signup page. Then create a **logout function**.

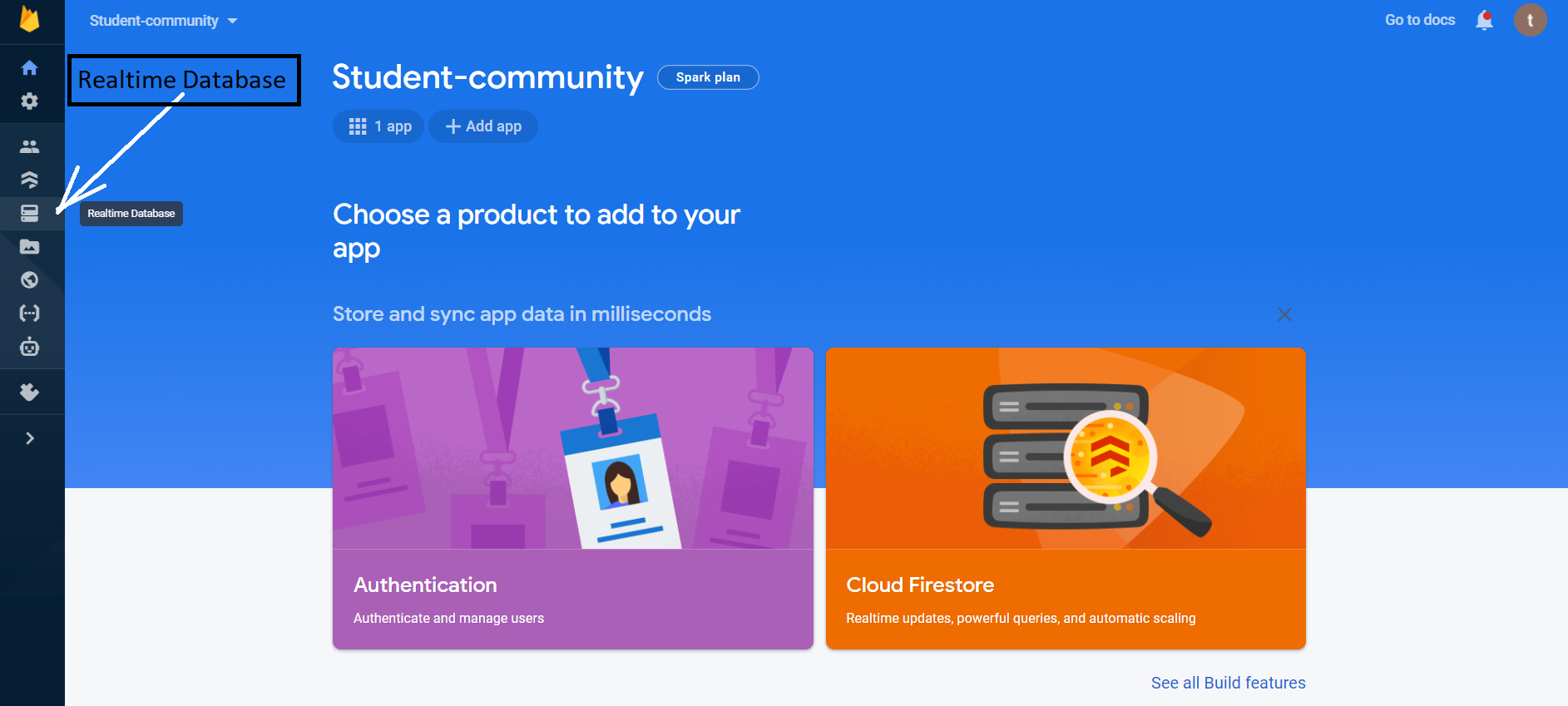
**Firebase Create Database for Creating Posts**



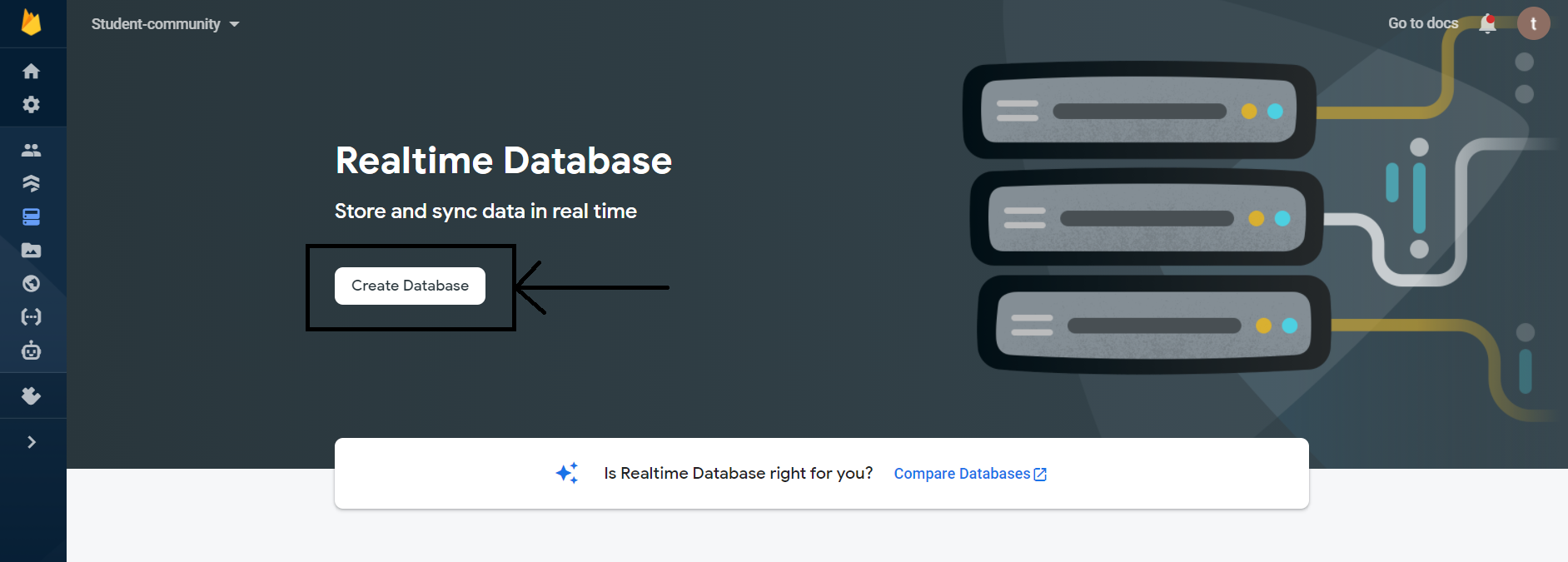
Go to Firebase and then click on **Get started**.



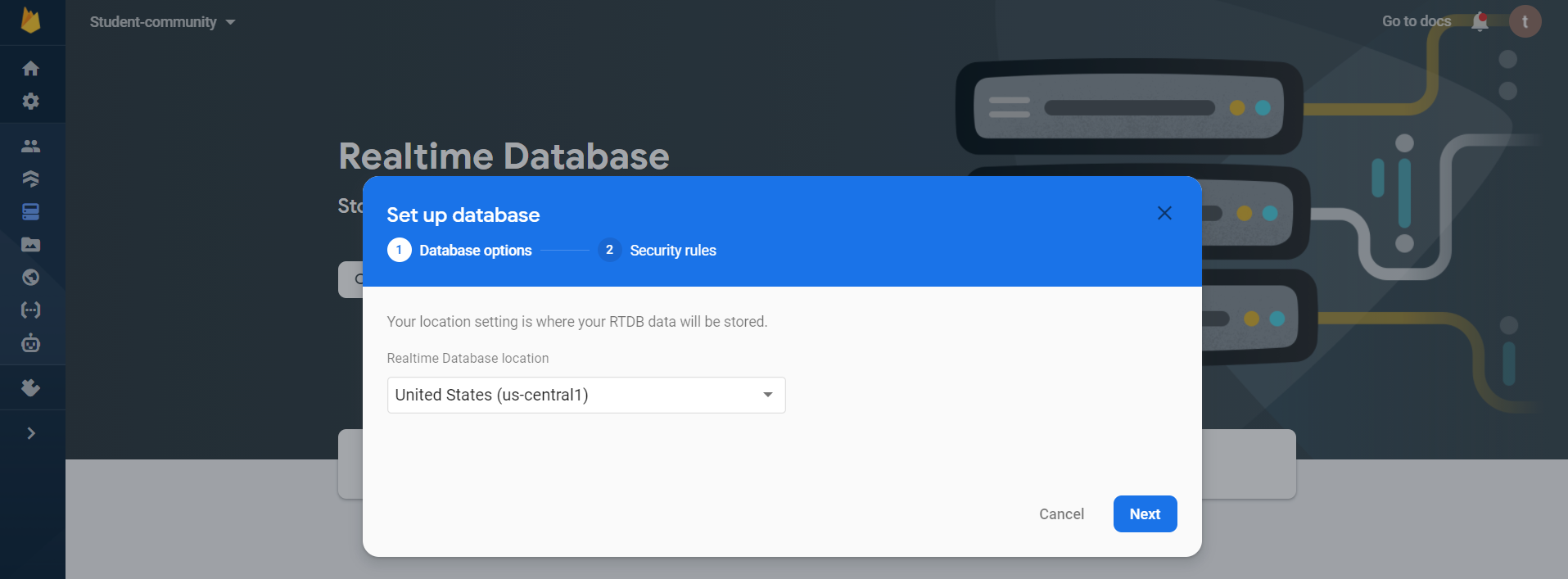
Then **click** on the project-name you had given to the project, here, it is **Student-community**.



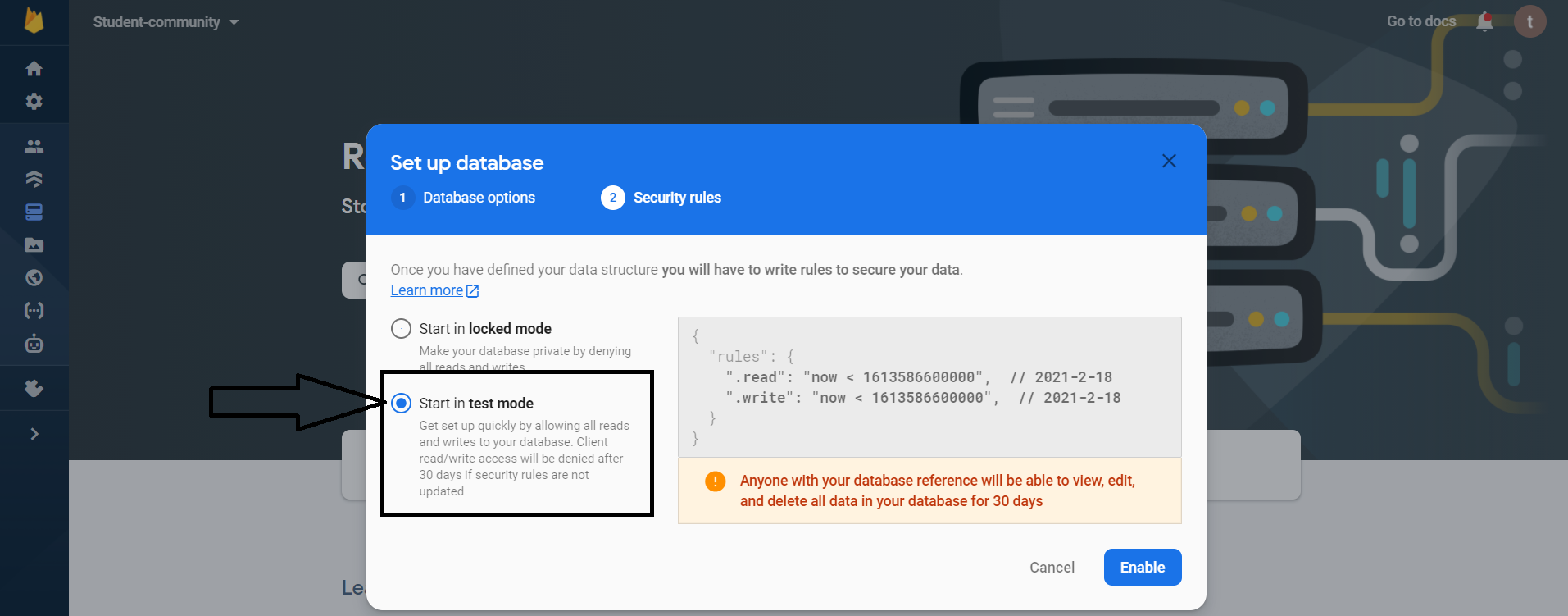
Then, from the left-side dashboard click on the **Realtime Database** Icon.



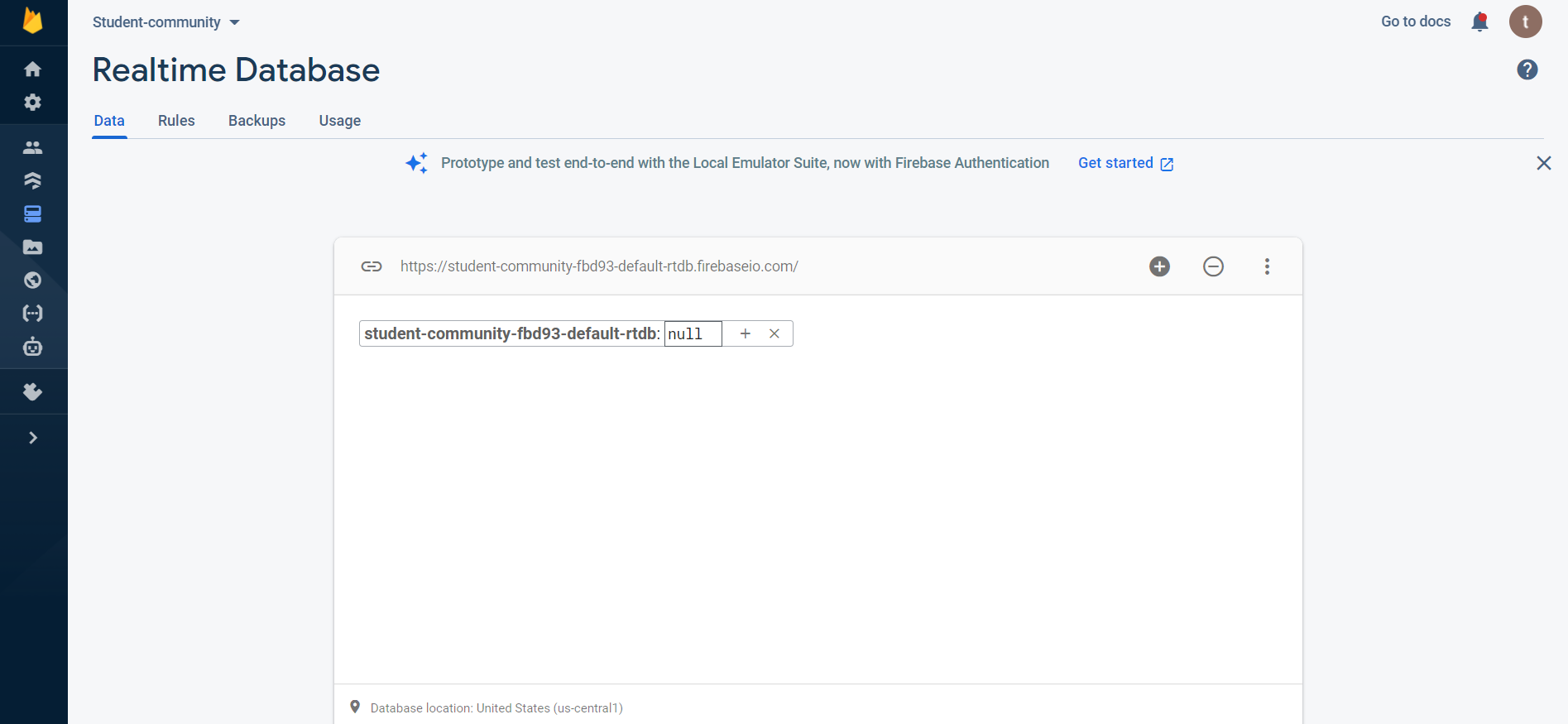
Click on **Create Database**.



Then you will see this pop-up, **keep the default settings** i.e keep the realtime database location to the United States and click on **Next**.



Then click on Start in **test mode**, so that anyone can access i.e signup/login to your pages. And then click on **Enable**.



If the database was successfully created, then you will see that an empty **database** with a name that is your project name followed by a unique ID will be **created**.

After this, we will make our **addPostBtnClicked function** which will push our entered Post Data from our post.html page into the firebase.

Then, to render all the posts on the home-page create **readPostData function**.

Now we have to make the **edit post** and **delete post** functionality.

For that, in Edit&DeletePost.js we will have to make the **editButtonClicked** function which will display the edit box and pre-populate the edit textbox with the previous post content and post title.

Then we will make **savePostBtnClicked** function which will get data from the input fields and update the firebase database.

For deleting the post, we will make a deleteButtonClicked function which will remove that post from the firebase database.

And in createPost.js we will have to add onclick addEventListener for editButtonClicked and deleteButtonClicked.

Then for adding **comments/answers**, in comments.js we will make **addCommentsButtonClicked** function, and then the **saveCommentBtnClicked** function to update the firebase database with the new comment/answer.

For **likes**, in likes.js we will make the **like** function.

Then for our **Profile page**, in profilePage.js we will make **viewYourPosts** function, which will display posts of only the logged in user that is the posts created by that user.