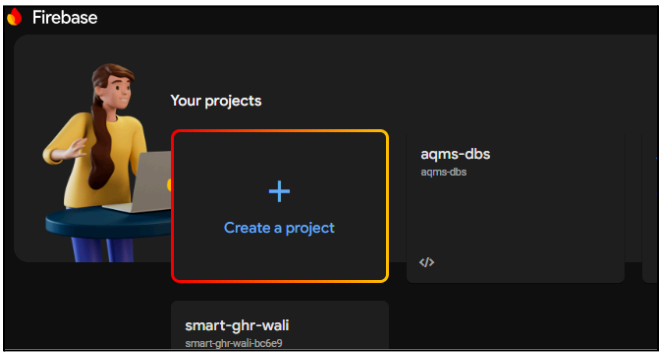
Getting started with ESP32 and Firebase

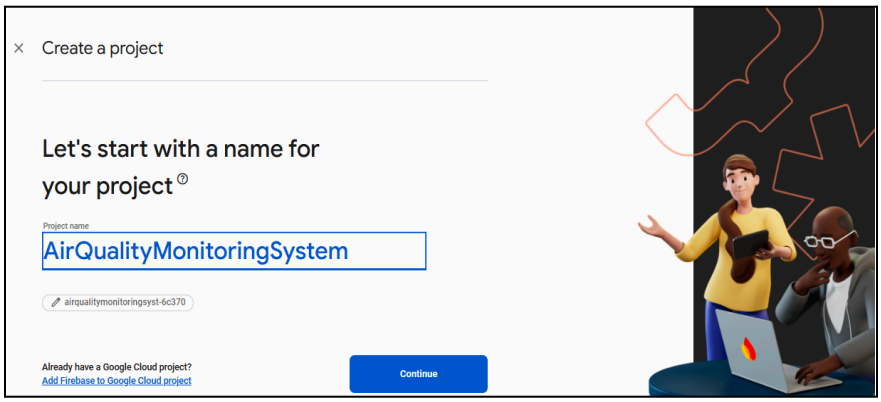
Backend Setup

Here are the steps to set up Firebase for your backend:

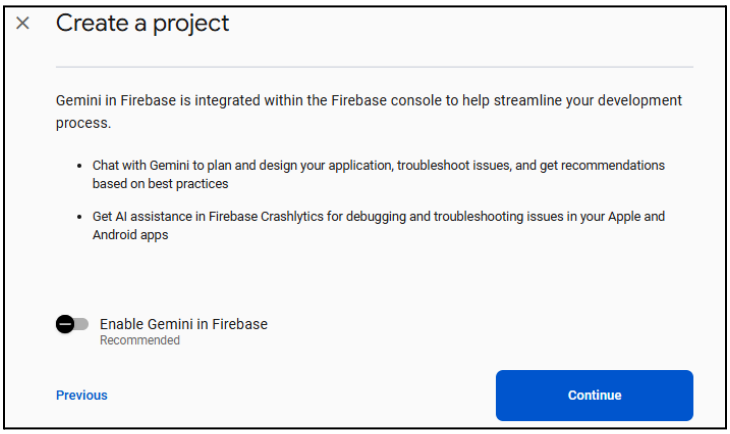
1. Go to **Firebase** and sign in with your **Google account**.
2. Click on **Add Project** to create a new project.

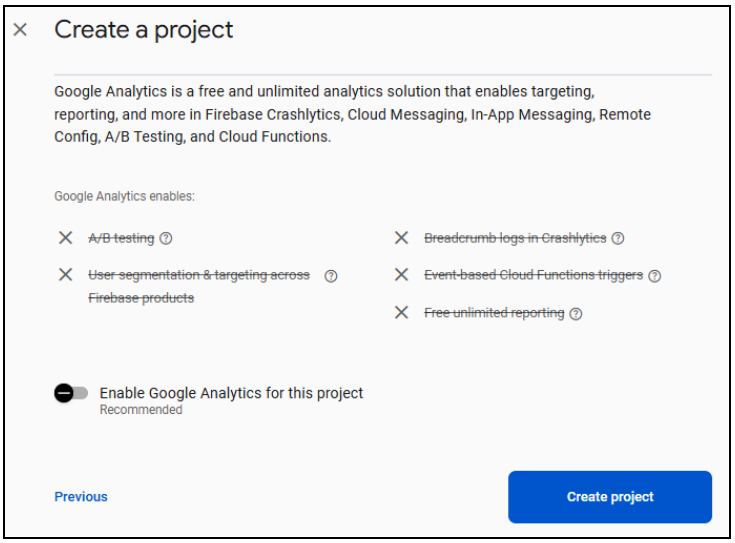


3. Enter a **name** for your project.
4. Click the **Continue** button.

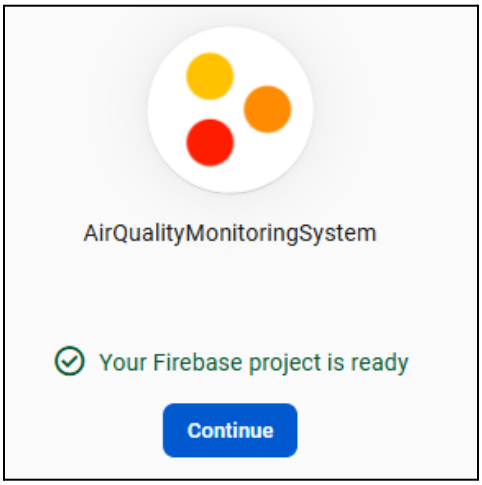


5. Disable the *Enable Google Analytics for this project* option — this will not be needed for this project. Finally, select *Create project*.

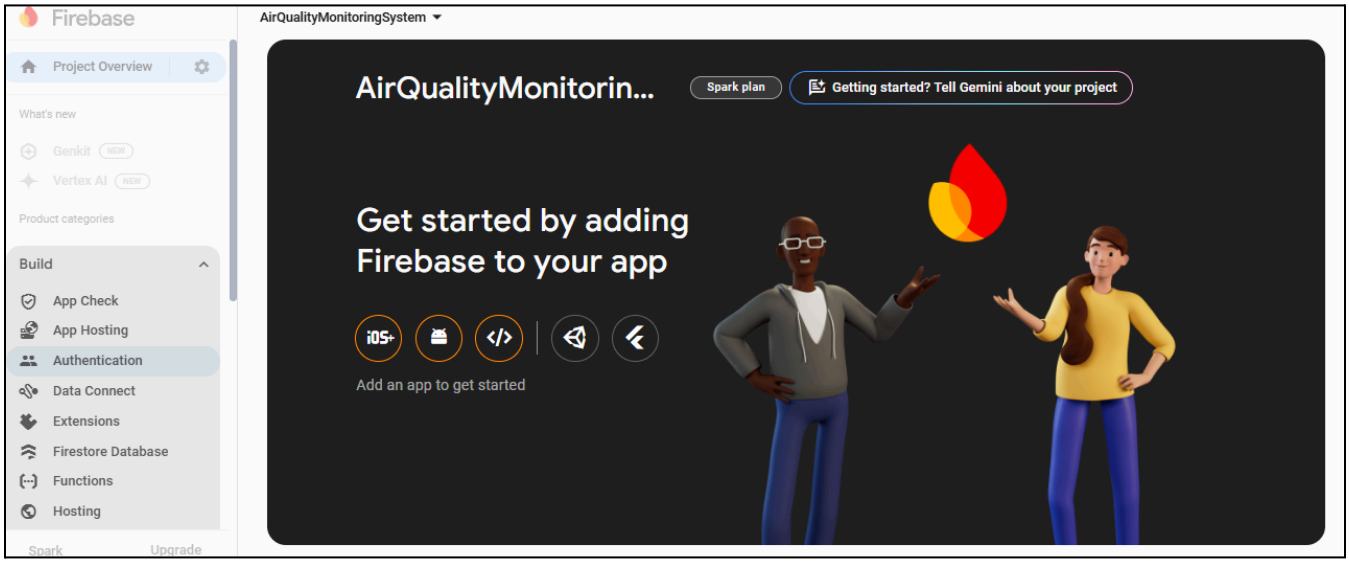




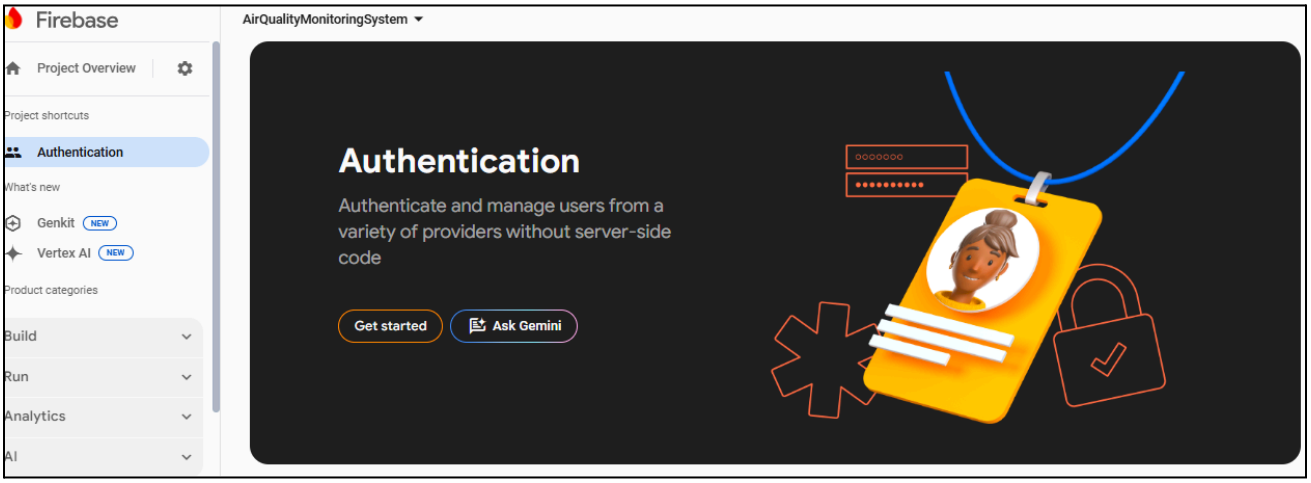
6. Firebase will begin setting up your project. Once completed select *Continue,* and you will be taken to your projects overview page on the Firebase console.



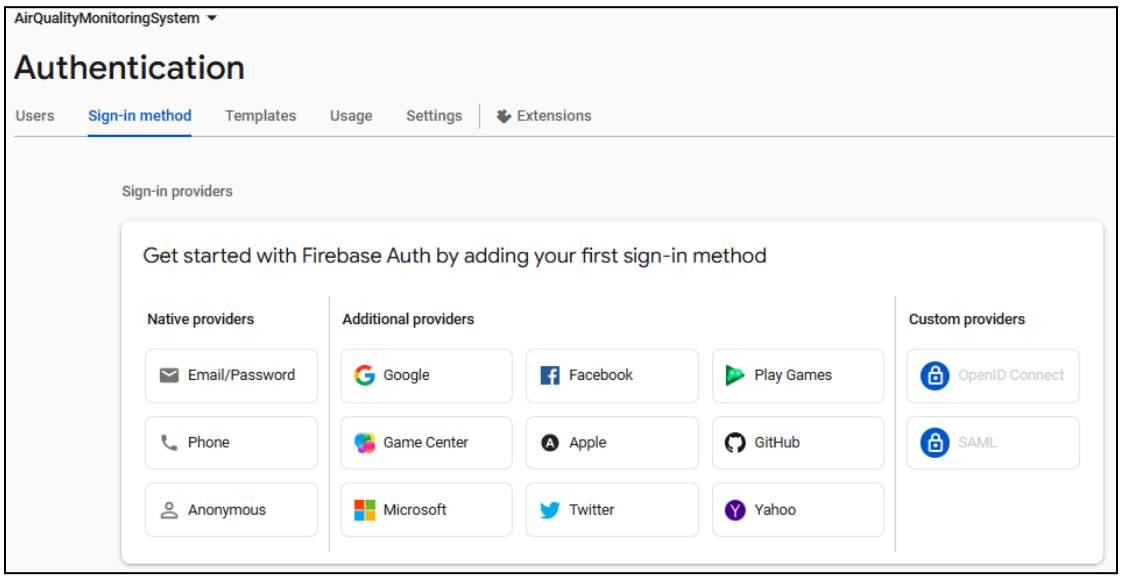
7. The first thing we want to do is setup authentication options for our project. Select the *Authentication* menu option on the top left, and you will be taken to the authentication page.

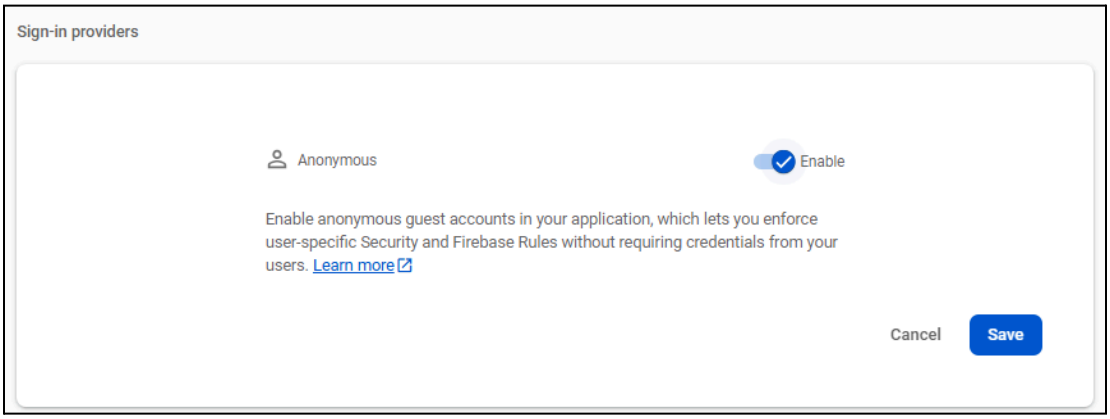


8. Select the *Get Started* button.

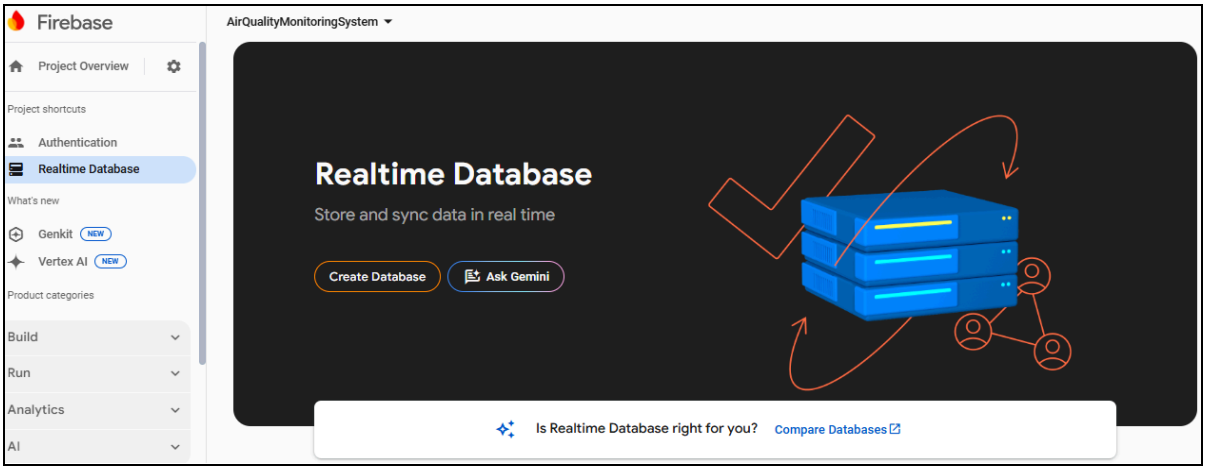


9. We're going to start by using the *Anonymous* sign-in for our ESP32 devices. Enable *Anonymous Sign In* as shown below and select the *Save* option.

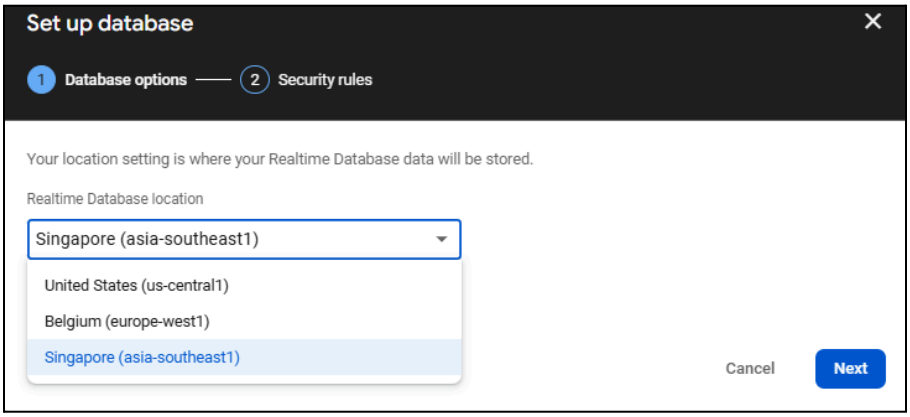




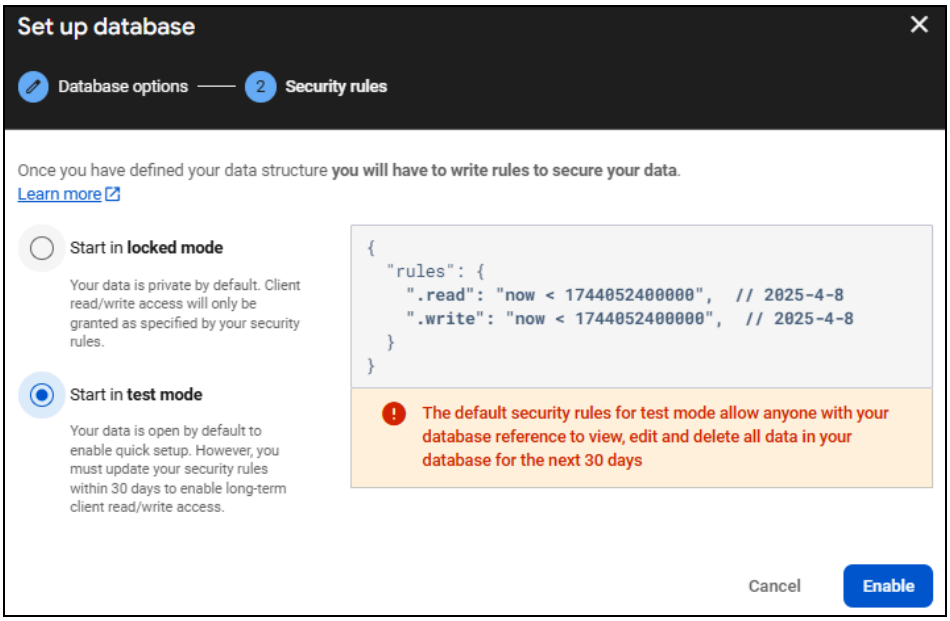
10. Next, we need to create a database that will hold all our sensor data. To do this, select the *Realtime Database* menu option on the top left, and you'll be taken to the Realtime Database page. Select the *Create Database* button to initialize the database creation menu.



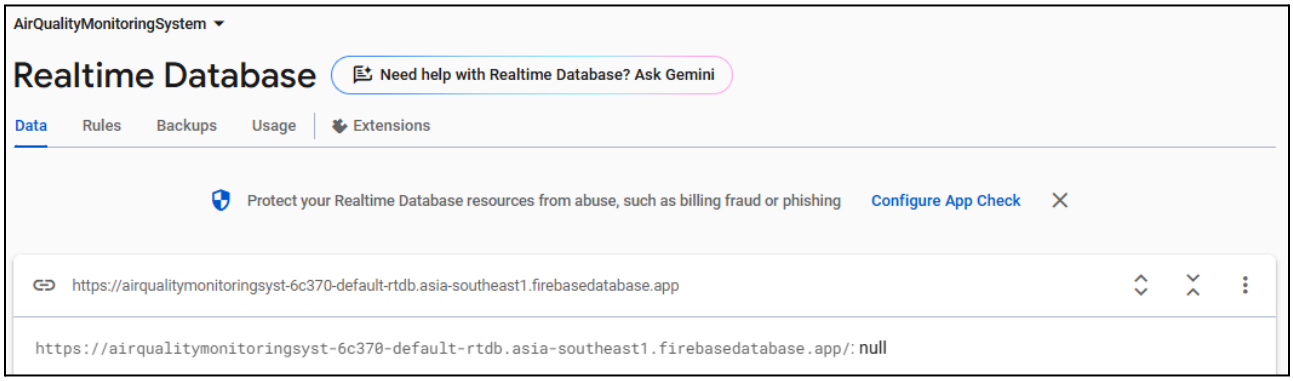
11. In the database creation menu, you can choose a location that is closest to you, and select *Next*.



12. You will be presented with the option to initialize your database in *locked mode* or *test mode*. Select *test mode* for now. The main difference is that in test mode a database access rule is placed allowing unauthorized access to your database for thirty days. If you plan to take your project into production, you will need to disable this in the future. Select the *Enable* button.

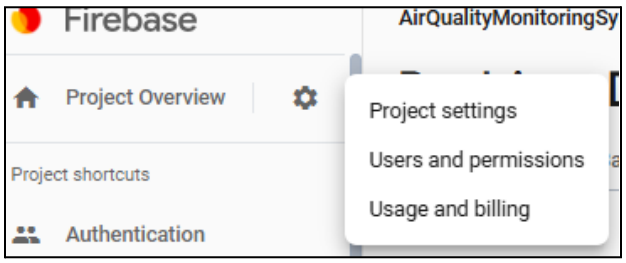


13. Lastly, you should see a new page with your new empty database, and we are all set!

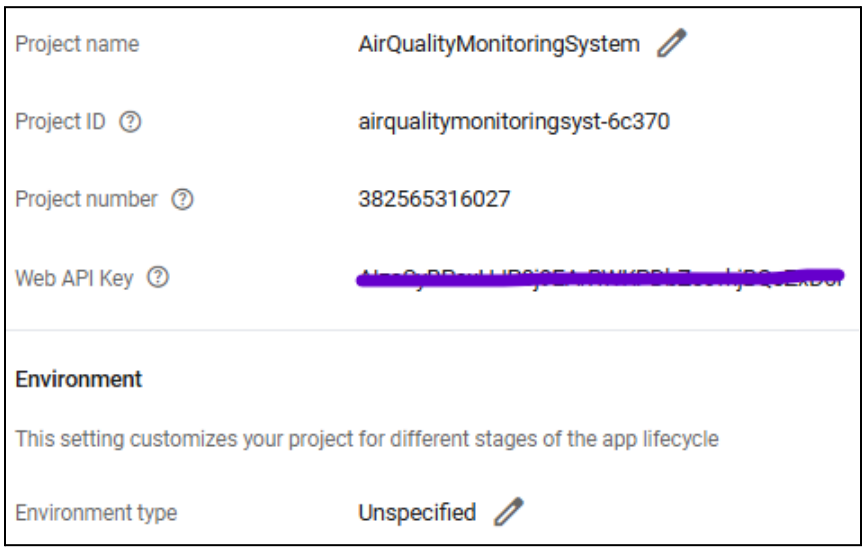


14. Before we continue there are two items you need to copy and store for future use in our embedded application. The first item we need is our real-time database URL, which you can find by copying the URL of the real-time database page, in our case it will be https://airqualitymonitoringsyst-6c370-default-rtdb.asia-southeast1.firebasedatabase.app/

15. The second item we need to store is the project's API key. To get your API key navigation to the project settings page select the settings icon on the top right, and then select the *Project settings* menu item as shown below:



16. In the project settings page, you should see your *Web API Key*, copy this, and store it.



We are all set to get started on our embedded application.

Conclusion:

Setting up Firebase for ESP32 enables real-time data monitoring and storage with seamless cloud integration. By configuring authentication, initializing the Realtime Database, and using FirebaseESP32, efficient communication is established. This setup provides a scalable foundation for various IoT applications, allowing further enhancements in security and data management.