



What is Firebase ?

Firebase is a mobile and web application development platform developed by firebase, Inc. in 2011, then acquired by Google in 2014.

Firebase is a platform that will allow you to develop cross-platform apps quickly. It offers a number of different services built-in, including some basic analytics.



Develop

Realtime Database

Authentication

Cloud Messaging

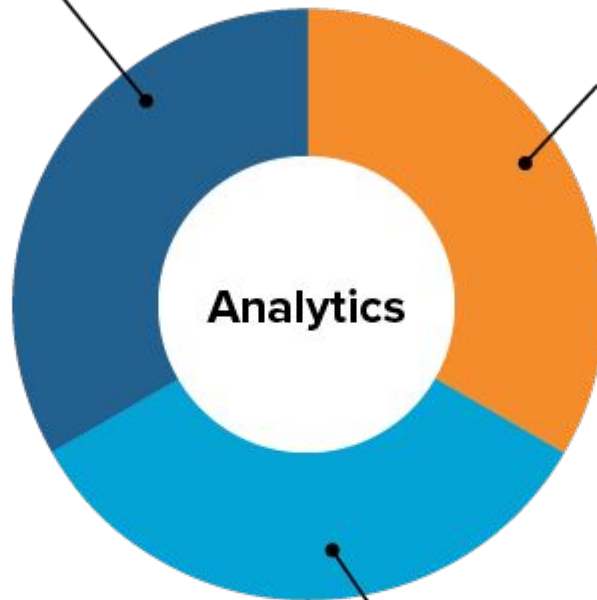
Storage

Hosting

Remote Config

Test Lab

Crash Reporting



Grow

Notifications

App Indexing

Dynamic Links

Invites

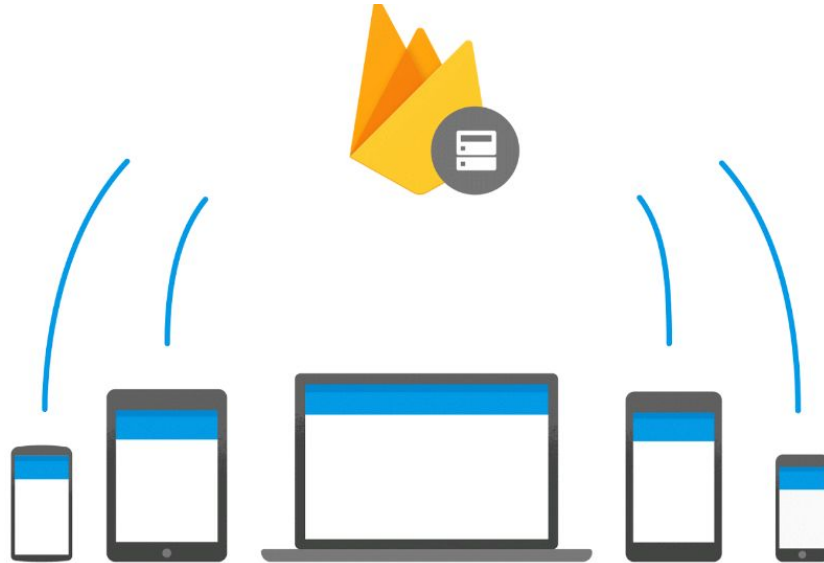
AdWords

Earn

AdMob

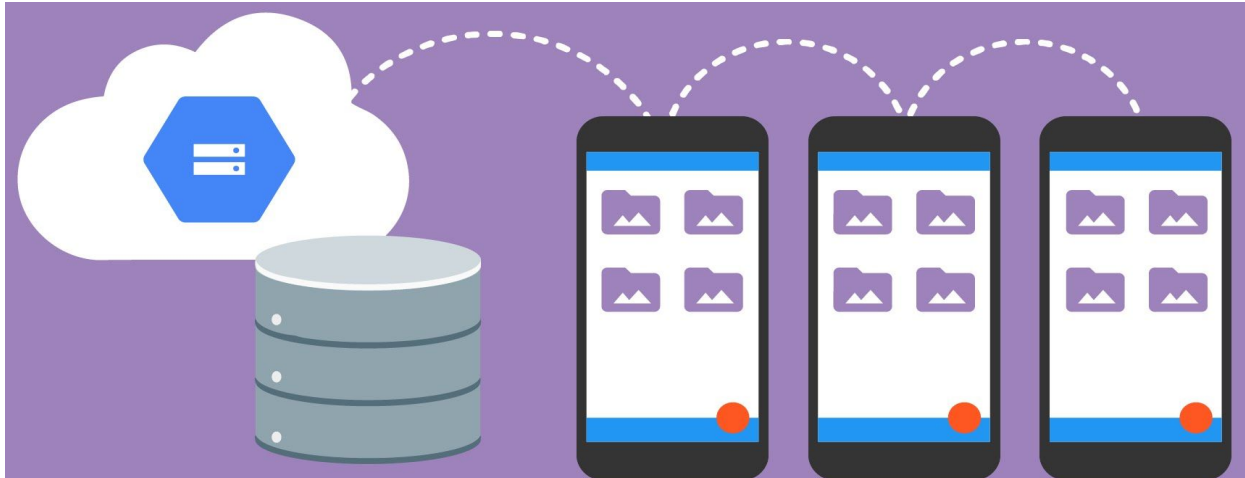
Realtime Database

Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud.[\(Ref\)](#)



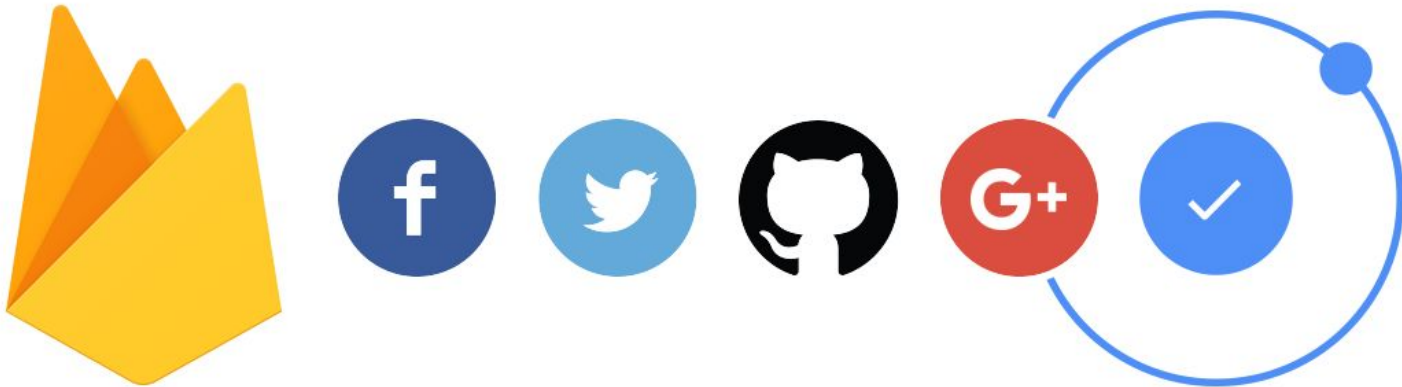
Firebase Storage

Firebase Storage provides secure file uploads and downloads for Firebase apps, regardless of network quality. The developer can use it to store images, audio, video, or other user-generated content. Firebase Storage is backed by Google Cloud Storage.[\(Ref\)](#)



Firebase Auth

Firebase Auth is a service that can authenticate users using only client-side code. It supports social login providers Facebook, GitHub, Twitter and Google (and Google Play Games). Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase.[\(Ref\)](#)



Hosting

Firebase Hosting is a static and dynamic web hosting service that launched on May 13, 2014. It supports hosting static files such as CSS, HTML, JavaScript and other files, as well as dynamic Node.js support through Cloud Functions.

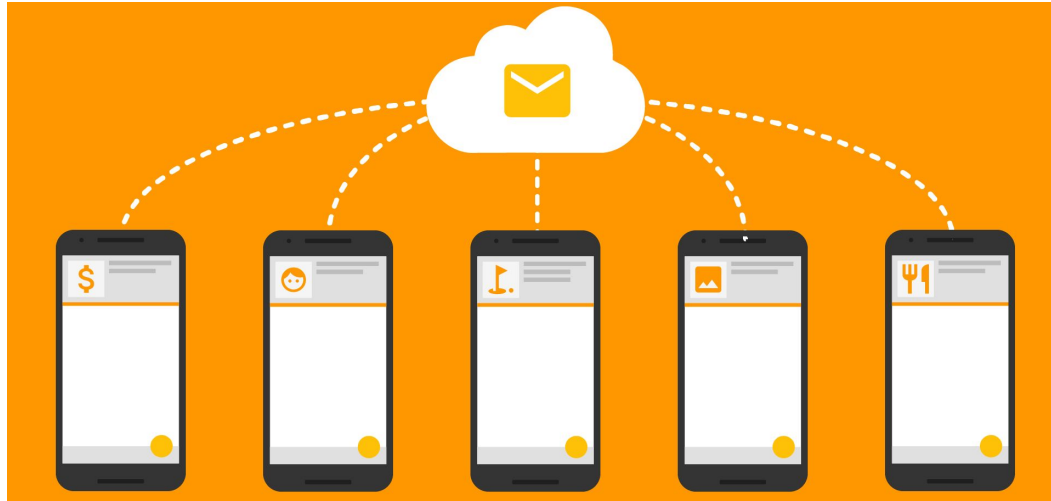


Firebase

Hosting

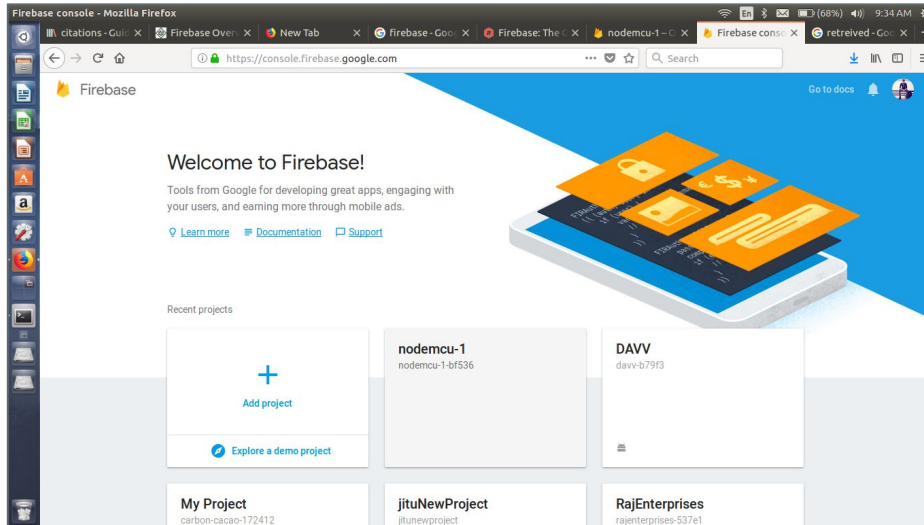
Cloud Messaging

Formerly known as Google Cloud Messaging (GCM), Firebase Cloud Messaging (FCM) is a cross-platform solution for messages and notifications for Android, iOS, and web applications, which currently can be used at no cost.

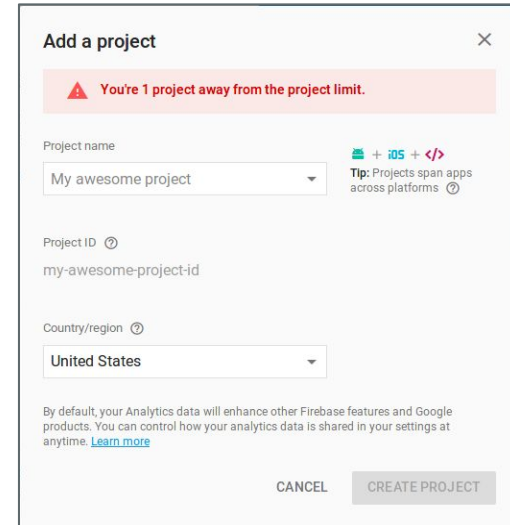


Project Creation on Firebase

1. Goto website: <https://console.firebase.google.com/>
2. Create project by clicking on 'Add Project' button, provide the project name and country name in popup window, and click on create project.

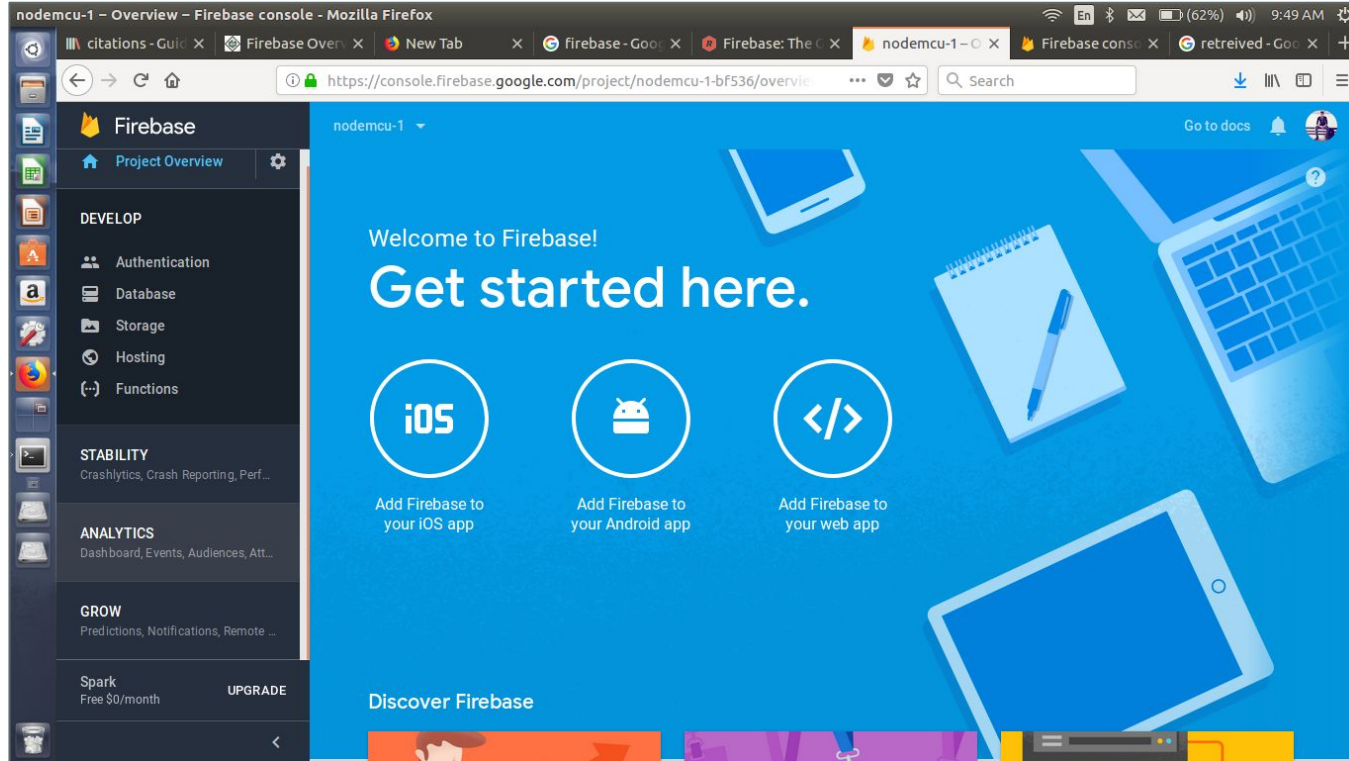


Firebase Console



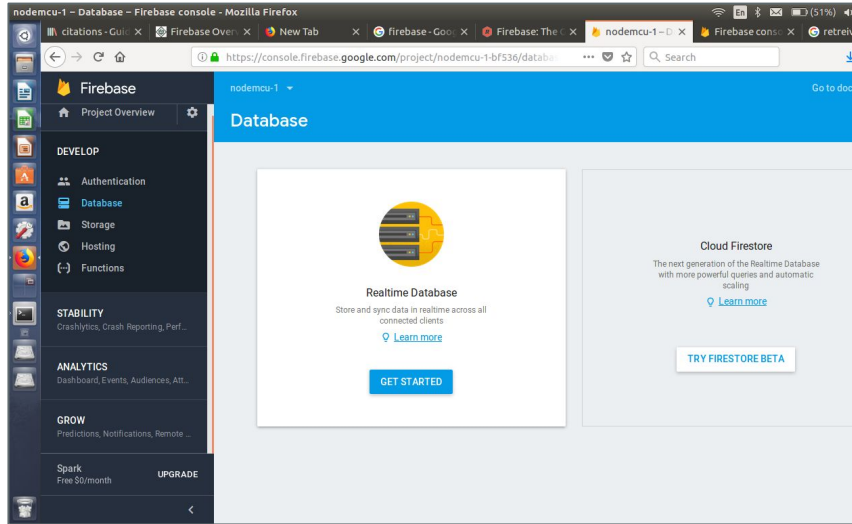
New Project Creation

Project Console Description

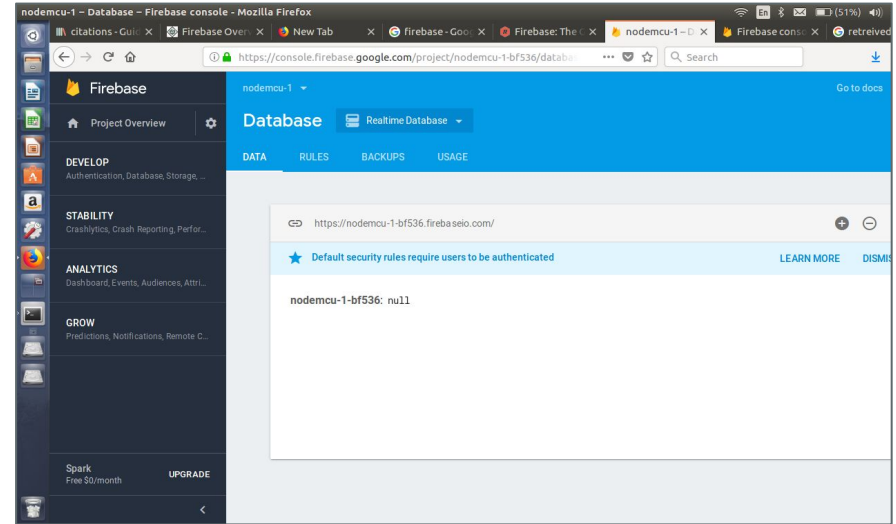


Project Console

Realtime Database Panel

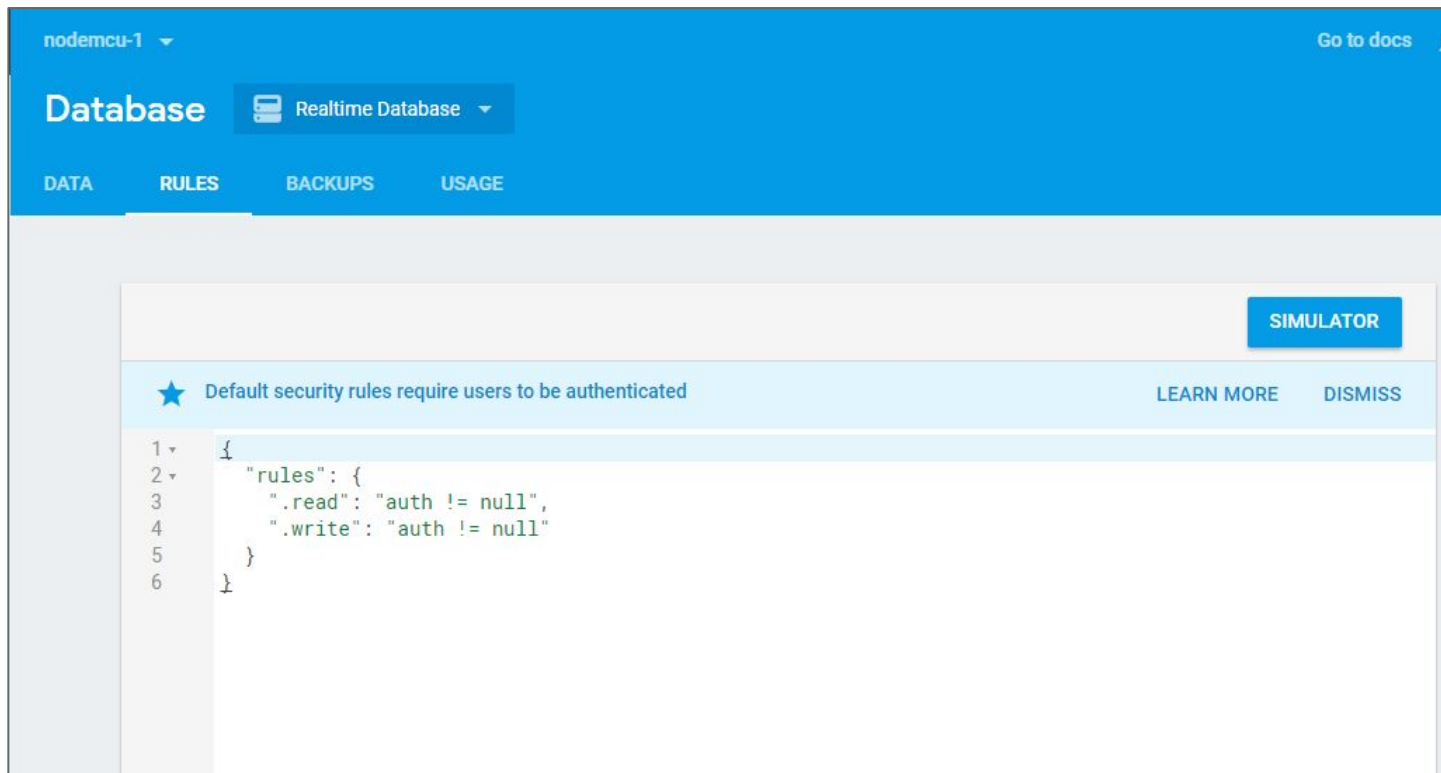


Realtime Database and Firestore



Realtime Database

Rules

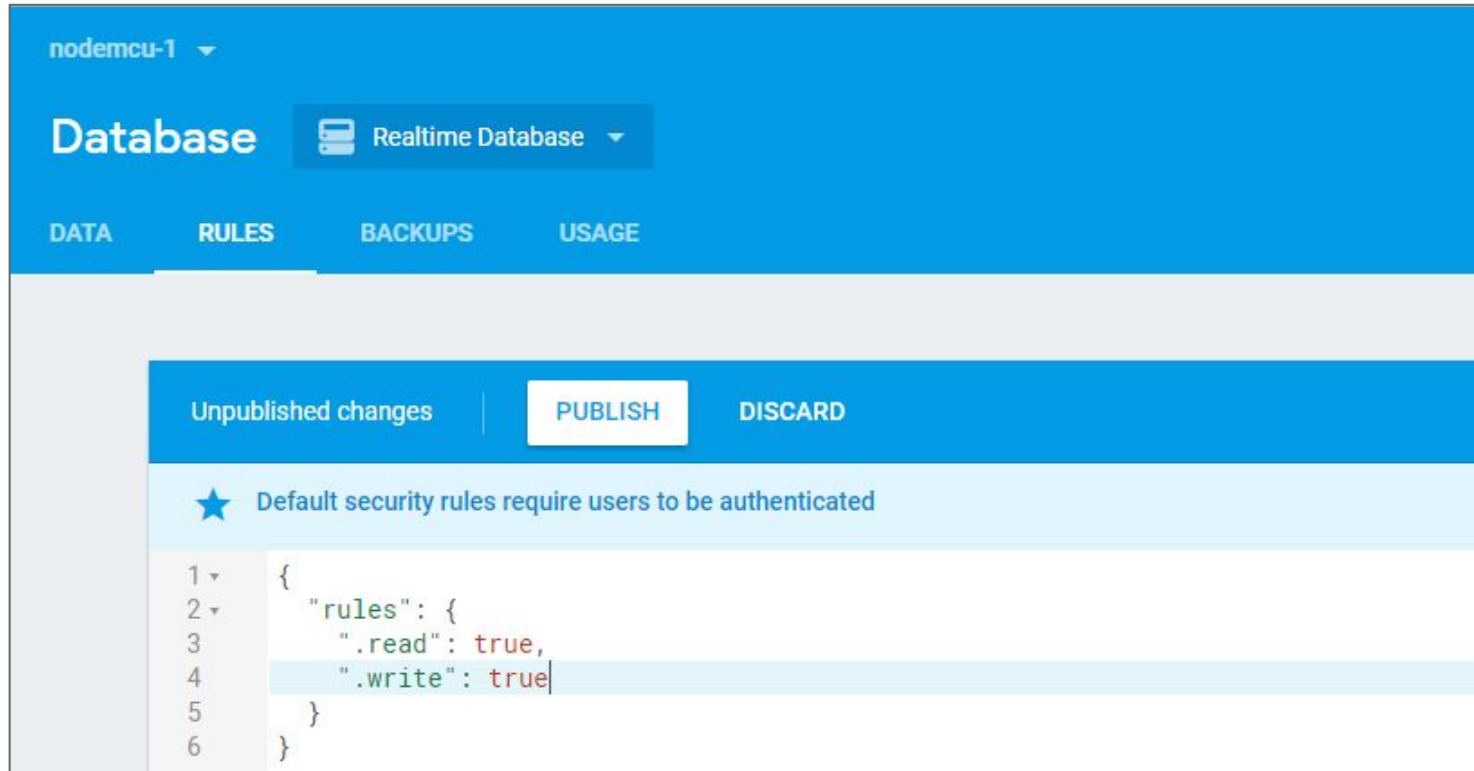


The screenshot shows the Firebase Realtime Database Rules editor. At the top, there's a blue header with 'nodemcu-1' on the left and 'Go to docs' on the right. Below the header, the word 'Database' is displayed, followed by a 'Realtime Database' dropdown menu. A navigation bar contains four tabs: 'DATA', 'RULES' (which is selected), 'BACKUPS', and 'USAGE'. The main content area has a 'SIMULATOR' button in the top right corner. Below this, a light blue banner states '★ Default security rules require users to be authenticated' with 'LEARN MORE' and 'DISMISS' links. The primary area is a code editor showing the default security rules in JSON format, with line numbers 1 through 6 on the left margin.

```
1 {  
2   "rules": {  
3     ".read": "auth != null",  
4     ".write": "auth != null"  
5   }  
6 }
```


Realtime Database Rules

Rules



The screenshot shows the Firebase Realtime Database Rules editor for a database named "nodemcu-1". The interface has a blue header with the "Database" label and a "Realtime Database" dropdown. Below the header is a navigation bar with tabs for "DATA", "RULES", "BACKUPS", and "USAGE". The "RULES" tab is selected. In the center, there's a blue bar with "Unpublished changes" and two buttons: "PUBLISH" and "DISCARD". Below this, a light blue banner states: "★ Default security rules require users to be authenticated". The main area displays a JSON rule configuration with line numbers 1 through 6 on the left. The rule is a JSON object with a "rules" property containing "read" and "write" permissions, both set to "true".

nodemcu-1 ▾

Database  Realtime Database ▾

DATA RULES BACKUPS USAGE

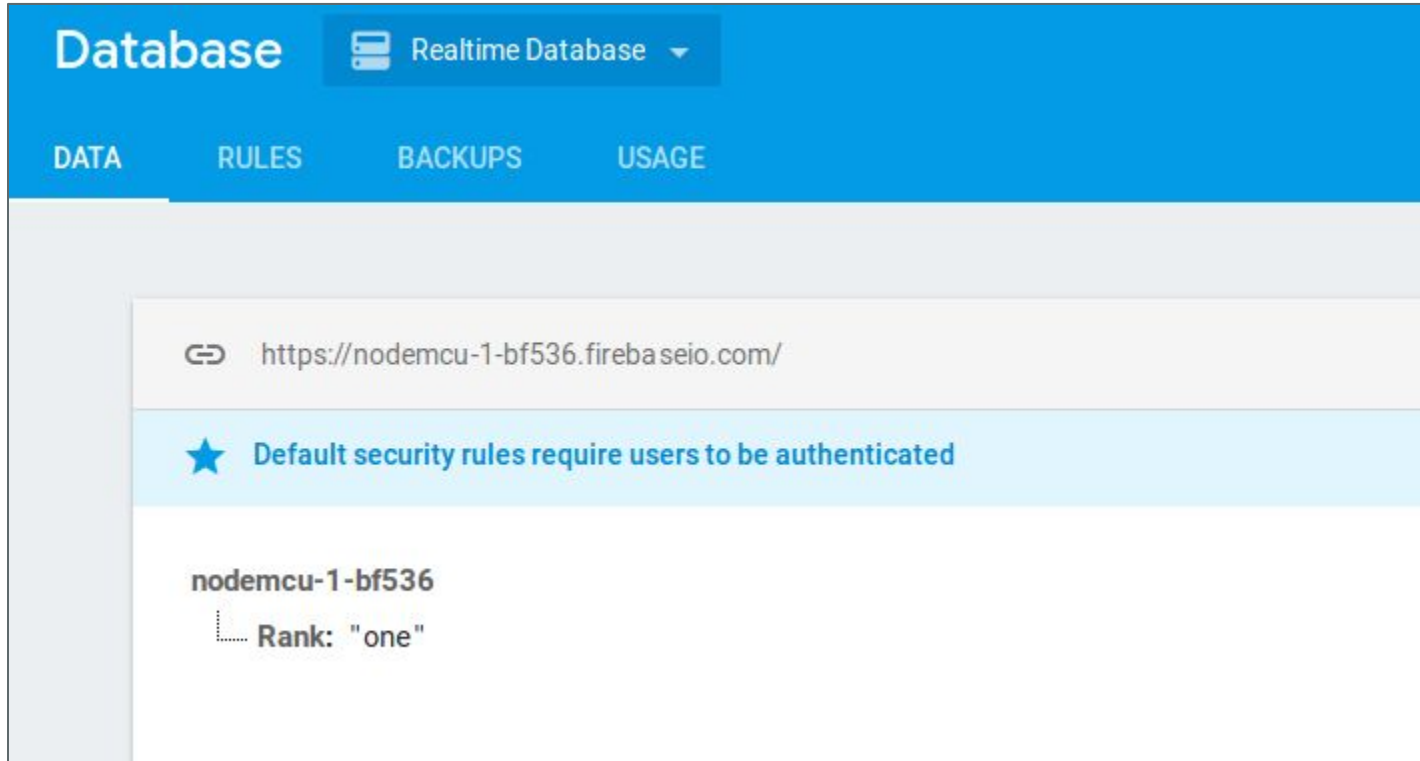
Unpublished changes | PUBLISH DISCARD

★ Default security rules require users to be authenticated

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

[Realtime Database Rules](#)

Key and Value



Key value in Realtime Database

Add ESP8266 core Library

Step-1: Open the Arduino IDE and press ctrl + comma.

Step-2: Enter http://arduino.esp8266.com/stable/package_esp8266com_index.json into *Additional Board Manager URLs* field.

Step-3: Goto Tools → Boards → Boards Manager...

Step-4: Type 'esp' in search field and select 'esp8266 by ESP8266 community' and install it.

Step-5: Now check in Tools → Boards, there is a list of esp based boards, select NodeMCU 1.0 (ESP-12E module)

Add Firebase/arduino Library

Step-1: Goto <https://github.com/firebase/firebase-arduino> , and download the ZIP file by clicking on 'Clone or download'.

Step-2: Open Arduino IDE, goto Sketch → Add Library → .ZIP library, then provide the path of .zip file of library and click on okay.

NodeMCU as a Event Listener

```
#include <FirebaseArduino.h>
#include <ESP8266WiFi.h>

// Set these to run example.
#define WIFI_SSID "RobuByte"
#define WIFI_PASSWORD "esp@8265"

void setup() {
  Serial.begin(9600);

  // connect to wifi.
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());

  Firebase.begin("davv-b79f3.firebaseio.com");
  Firebase.stream("/");
}
```

```
void loop() {
  if (Firebase.failed()) {
    Serial.println("streaming error");
    Serial.println(Firebase.error());
  }

  if (Firebase.available()) {
    FirebaseObject event = Firebase.readEvent();
    String eventType = event.getString("type");
    eventType.toLowerCase();

    Serial.print("event: ");
    Serial.println(eventType);
    if (eventType == "put") {
      Serial.print("data: ");
      /// Serial.println(event.getString("data"));
      String path = event.getString("path");
      String data = event.getString("data");
      Serial.println(path + ":" + data);
    }
  }
}
```

Conclusion

- Firebase is a google's API for development of mobile applications.
- It reduces the complexity of server design and security.
- Provides NoSQL realtime database server for real time applications.
- Firebase supports REST API, which helps to use firebase in any language with help of HTTP request response.

References

- [Getting to know NodeMCU and its DEVKIT board](#)
- [NodeMCU](#)
- [Firebase Realtime Database REST API](#)
- [ESP8266 core for Arduino IDE](#)
- [Firebase samples for Arduino](#)
- [Arduino Examples](#)