

Serverless



"Serverless is a cloud computing execution model where the cloud provider dynamically manages the allocation and provisioning of servers. A serverless application runs in stateless compute containers that are event-triggered, ephemeral (may last for one invocation), and fully managed by the cloud provider."

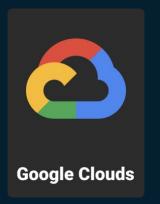


Benefits

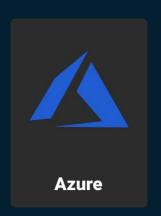
- Reduced time to market and quicker software release.
- Reduces the complexity of software.
- Simplifies packaging and deployment and requires no system administration.
- Lower operational and development costs. (to some point)
- Scalability you are very flexible with traffic handling
- Serverless is managed by a cloud provider Your servers are small, the cloud provider dynamically creates or destroys servers when demand grows or shrinks
- Fits with microservices, which can be implemented as functions.



Providers













https://firebase.google.com/











Build better apps







Authentication Authenticate users simply and securely

Hosting Deliver web app assets with speed and security

Cloud Storage Store and serve files at Google scale

Realtime Database Store and sync app data in milliseconds

Improve app quality

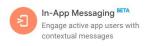
Crashlytics Prioritize and fix issues with powerful, realtime crash reporting





App Distribution BETA Distribute pre-release versions of your app to your trusted testers

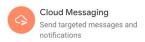
Grow your business









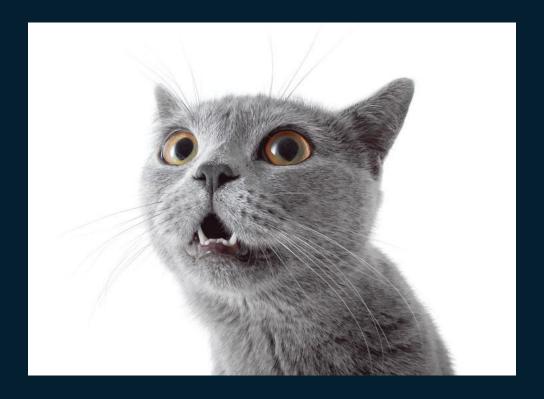








End of backenders era?





Setup

- 'npm i -g firebase-tools'
- 2. `firebase init PROJECT_TYPE` to initialise a project
 - a. **Project types:** database, firestore, functions, hosting, storage, or emulators
- 3. 'firebase init functions'



```
• • •
 1 const functions = require("firebase-functions");
 2 const express = require("express");
 4 const app = express();
 6 app.get("/", (req, res) \Rightarrow {
     res.send("Hello Friend!");
 8 });
10 exports.app = functions.https.onRequest(app);
11
```



firebase.json

```
• • •
 1 {
     "hosting": {
       "rewrites": [
           "source": "**",
           "function": "app"
12 }
```



`firebase serve --only functions,hosting`

To run emulation locally



Enable cache

```
• • •
 1 app.get("/user", (req, res) \Rightarrow {
     res.set("Cache-Control", "public, max-age=300, s-maxage=600");
     res.send(`${Date.now()}`);
 8 });
```



Connect firebase DB

```
• • •
 1 const firebase = require("firebase-admin");
 3 const firebaseApp = firebase.initializeApp(functions.config.firebase);
 5 const getFriends = async () \Rightarrow {
    const ref = firebaseApp.database().ref("friends");
     return await ref.once("value").then((snapshot) ⇒ snapshot.val());
 8 };
```



Firebase DB docs

Setup

https://firebase.google.com/docs/database/admin/start#node.js

Get data

https://firebase.google.com/docs/database/admin/retrieve-data

Save data

https://firebase.google.com/docs/database/admin/save-data



Firebase Web code lab

https://codelabs.developers.google.com/codelabs/firebase-web/



AWS

https://aws.amazon.com/sdk-for-node-js/

