

# SERVERLESS WITH FIREBASE

**JERRY JALAVA**

Senior System Architect, Google  
Developer Expert, Authorised Trainer

[JERRY@QVIK.FI](mailto:JERRY@QVIK.FI) | @W\_I



**QVIK**



**LETS TAKE A STEP BACK**

**“SERVERLESS” OPTIONS IN**

# **GOOGLE CLOUD PLATFORM**

# OPTIONS

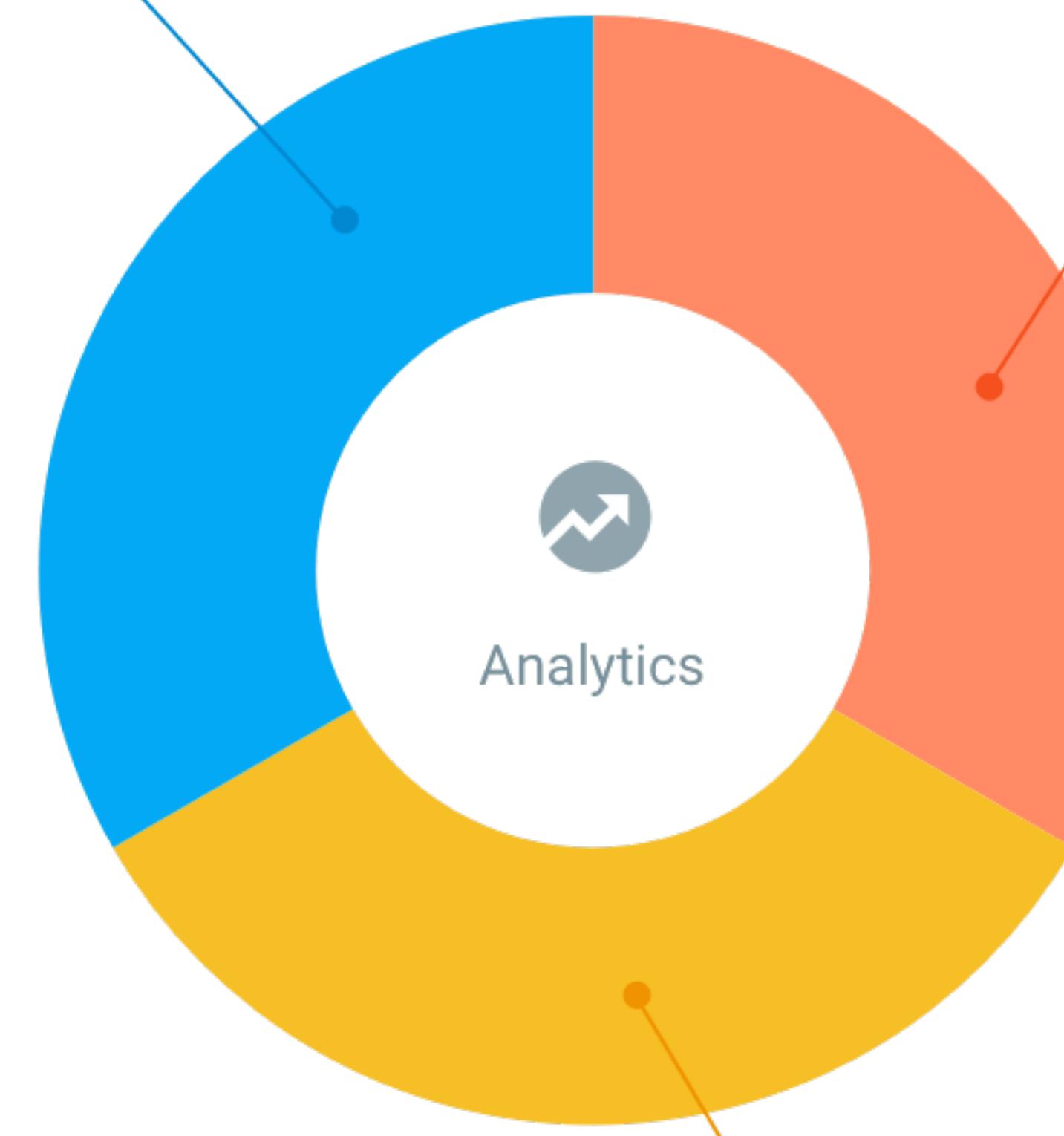
- Google Cloud Storage (static web pages)
- Google Cloud Functions (Alpha)
- AppEngine (Standard Environment)
- Firebase (Hosting, Realtime Database, Authentication, Storage)



WHAT IS  
**FIREBASE?**

## DEVELOP

-  Realtime Database
-  Authentication
-  Cloud Messaging
-  Storage
-  Hosting
-  Remote Config
-  Test Lab
-  Crash Reporting



## GROW

-  Notifications
-  App Indexing
-  Dynamic Links
-  Invites
-  AdWords

## EARN



<https://firebase.google.com>

# WHAT AM I DEMOING

- Firebase Hosting
- Firebase Realtime Database
- Firebase Auth



## CLIENT SIDE CODE

```
1 // Initialize the Firebase library with connection config
2 firebase.initializeApp(config);

4 // Show login dialog with Google Auth provider
5 // $(after login button has been clicked)
6 firebase.auth().signInWithPopup(
7     new firebase.auth.GoogleAuthProvider()
8 );

10 // Listen for changes in the auth state and store the user info to DB
11 firebase.auth().onAuthStateChanged(user) => {
12     if (user) {
13         firebase.database().ref(`users/${userId}`).set({
14             username: user.displayName,
15             email: user.email
16         });
17     }
18 });
```

## CLIENT SIDE CODE

```
20 // Listen for newest 30 Post datas
21 firebase.database().ref('posts').limitToLast(30).on('child_added', (data) => {
22   console.log('Post title:', data.val().title);
23 });
24
25 // Listen for users own Posts
26 const currentUserId = firebase.auth().currentUser.uid;
27 firebase.database().ref(`user-posts/${currentUserId}`)
28 .on('child_added', (data) => {
29   console.log('Own Post title:', data.val().title);
30 });
31
32 // Store new Post to all posts and to users own posts
33 const newPostKey = firebase.database().ref().child('posts').push().key;
34 let updates = {};
35 updates[`/posts/${newPostKey}`] = postData;
36 updates[`/user-posts/${uid}/${newPostKey}`] = postData;
37 firebase.database().ref().update(updates);
38
39 // To remove a single Post
40 firebase.database().ref(`/posts/${postId}`).remove();
41 firebase.database().ref(`/user-posts/${userId}/${postId}`).remove();
```

**DEPLOY THE CODE WITH SINGLE COMMAND**

**\$ FIREBASE DEPLOY**



LIVE DEMO

[HTTPS://MICRO.TOTALLYON.ME/](https://micro.totallyon.me/)

[HTTPS://GITHUB.COM/JERRYJJ/SERVERLESS-MICROBLOG](https://github.com/jerryjj/serverless-microblog)

# QVIK

THANK YOU

[www.qvik.fi](http://www.qvik.fi)

