International Women's Day - IWD 18

GDG Medellin
Google Developer Group

@UserTwoGG
Catherin Cruz

What is Firebase?

Started out as a Backend-as-a-Service (BaaS) and grew into a next-generation app-development platform on Google Cloud Platform.

You don't need to manage servers or write APIs anymore. Allows you to develop the whole application on the FE without any BE code.

Firebase can be your server, your API and your datastore, all written so generically that you can modify it to suit most of your needs.

Written in Scala, uses Netty to handle web sockets.

Benefits from using Firebase

★ The authentication. Firebase auth includes a built-in email/password authentication system. It also supports OAuth2 for Facebook, Google, Twitter and GitHub.

★ The hosting. Firebase comes with an easy-to-use hosting service for all your static files. It serves all of it from a global CDN with HTTP/2.

★ Real-time syncing of data across all clients, like Android, iOS or Web.

★ Via automatic update notifications, it keeps two systems synchronized, without manual messaging, WebSockets, etc.

★ You don't require a server for a simple UI, plus you have free hosting.

Real-time Database
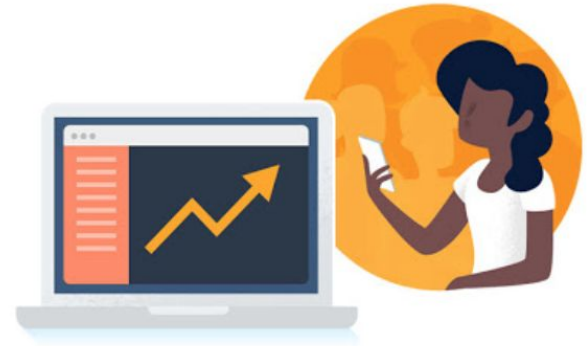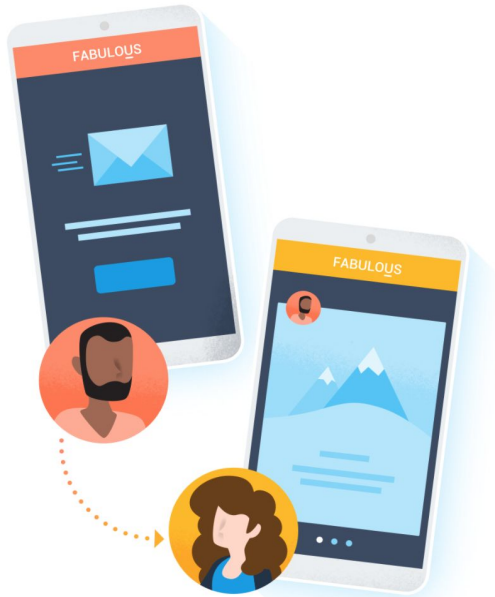
Firebase real-time NoSQL cloud database, is a service that validates the moment you save your JSON data, then these changes are immediately sent to all clients web and mobile who are connected. With minimal code, you can chat boxes, live news feed, notifying users of new posts and more.

When you connect your app to it, you're not connecting through normal HTTP. You're connecting through a WebSocket. WebSockets are a lot faster than HTTP.

All of your data syncs automatically through the single WebSocket as fast as your client's network can carry it and Firebase sends you new data as and when it's updated.

Notifications

Is a free service enabling targeted user notifications for mobile apps.

Firebase Cloud Messaging SDK (FCM), provides an option for developers seeking a flexible notification platform which requires minimal coding effort to get started, and a graphical console for sending messages.

With this, you can send notifications to all supported message targets. FCM handles the routing and delivery to targeted devices.

Services Offered

Developer Tools

Realtime Database

Pay to scale iOS 🤖 </> C++ ◁

Cloud Firestore BETA

Pay to scale iOS 🤖 </>

Cloud Functions BETA

Pay to scale iOS 🤖 </> C++ ◁

Authentication

iOS 🤖 </> C++ ◁

Business Growing

Remote Config

iOS 🤖 C++ ◁

App Indexing

iOS 🤖

Google Analytics

iOS 🤖 C++ ◁

Cloud Messaging

iOS 🤖 </>

Use Cases

Roll20

https://roll20.net/

It is a tabletop gaming built on Firebase for gaming.

It connects people to build games for themselves and also to play with others players.

Builded by a very small team, it was able to leverage technology vs developers resources.

Real-time update across web, iOS, Android, you have sort of cross-platform approach.

Twitch TV

https://www.twitch.tv/

Live streaming video platform owned by Twitch Interactive, a subsidiary of Amazon.

By Feb 2018, has 2 million broadcasters monthly, and 15 million daily active users.

Twitch still remains the leading live streaming video service for video games in the US.
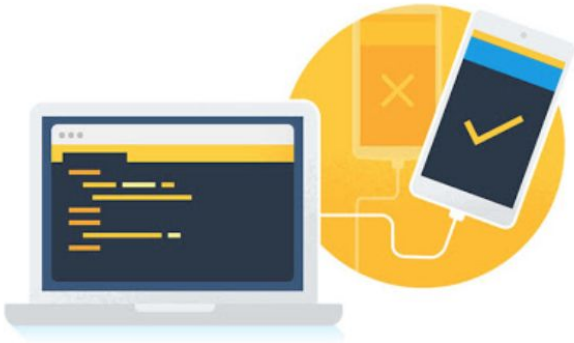
Create a new DB

➔ Add a new project, you could select a specific region

Project name

🤖 + **iOS** + **</>**

My awesome project ▾

Tip: Projects span apps across platforms ❓

➔ Go to Databases and start adding info!

iwd18-test ✕

Name key_id **Value** this is a value! ✕

CANCEL **ADD**

➔ Access the database by REST services

https://iwd18-test.firebaseio.com/webpage_title.json

Testing real-time changes

➔ Go to "Project Overview" then select your app version

➔ Listen for value events

To read and listen for changes, use on() or once() methods of firebase.database.Reference to observe events.

➔ Web app version .on() function example

```
var valueRealtime = document.getElementById('valueRealtime');
var dbRef = firebase.database().ref().child('realtime');
dbRef.on('value', snap => valueRealtime.innerText = snap.val());
```

Example: https://github.com/twogg-git/firebase-intro

Hosting a Web Page

➜ Install NodeJs Firebase tools

```
$ npm install -g firebase-tools
```

➜ Login and allow access

```
$ firebase login
```

➜ Init Firebase console in your project's folder

```
$ firebase init
```

➜ Deploy your page and get the URL

```
$ firebase deploy
```

Example: https://iwd18-test.firebaseapp.com/

Links and References

➤ Code samples
https://firebase.google.com/docs/samples/

➤ Guides for apps and games
https://firebase.google.com/docs/guides/

➤ Chat client using Firebase
https://github.com/firebase/friendlychat-web

➤ Firebase integration for React Native
https://rnfirebase.io/