

# Full Stack Dev with Firebase

Rapid prototyping with Firebase + React

**Nitya Narasimhan**  
@nitya

#reactfire  
#gdg

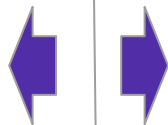
GDG Capital Region  
January 2017



# About Me



PhD  
Distributed Systems



**MOTOROLA**  
Distinguished Member  
of Tech Staff  
Ubiquitous Computing



**Principal**  
Mobile Web  
Startups



**Industry Advisor**  
Internet Of  
Things



**EDUCATION**

Competency Training  
Robotics/STEM (K-8)



**ADVOCACY**

Community Organizer & Mentor (GDG)  
Tech Speaker (Conferences)

# About You

FIREBASE



REACT

Material  
Design

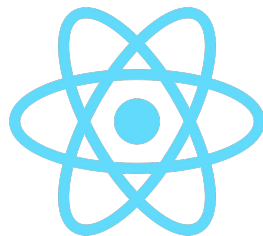
# About This Talk



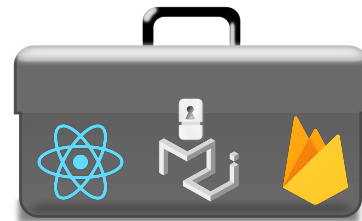
Real-Time App  
**Table Stakes**



Intro to  
Firebase



Intro to  
React

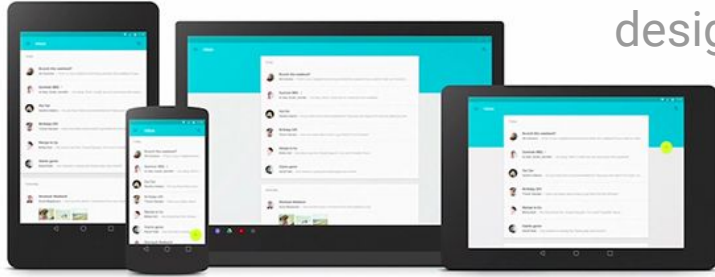


Real-Time App  
**Toolkit**

**So.... you want to build a web app?**

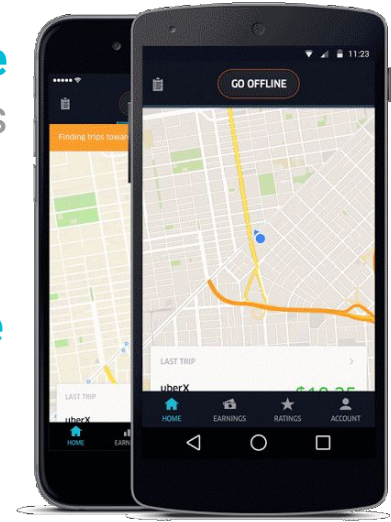
# User Expectations: UI/UX Table Stakes changing ..

Responsive  
design

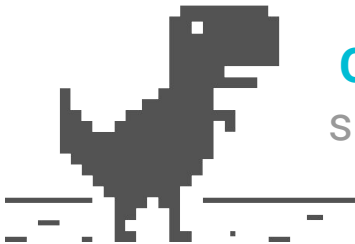


Real-Time  
updates

Reactive  
UX







Offline  
support



Fast  
interaction



 Response <100ms	 Animation 16ms render	 Idle	 Load <1000ms
---	--	---	--



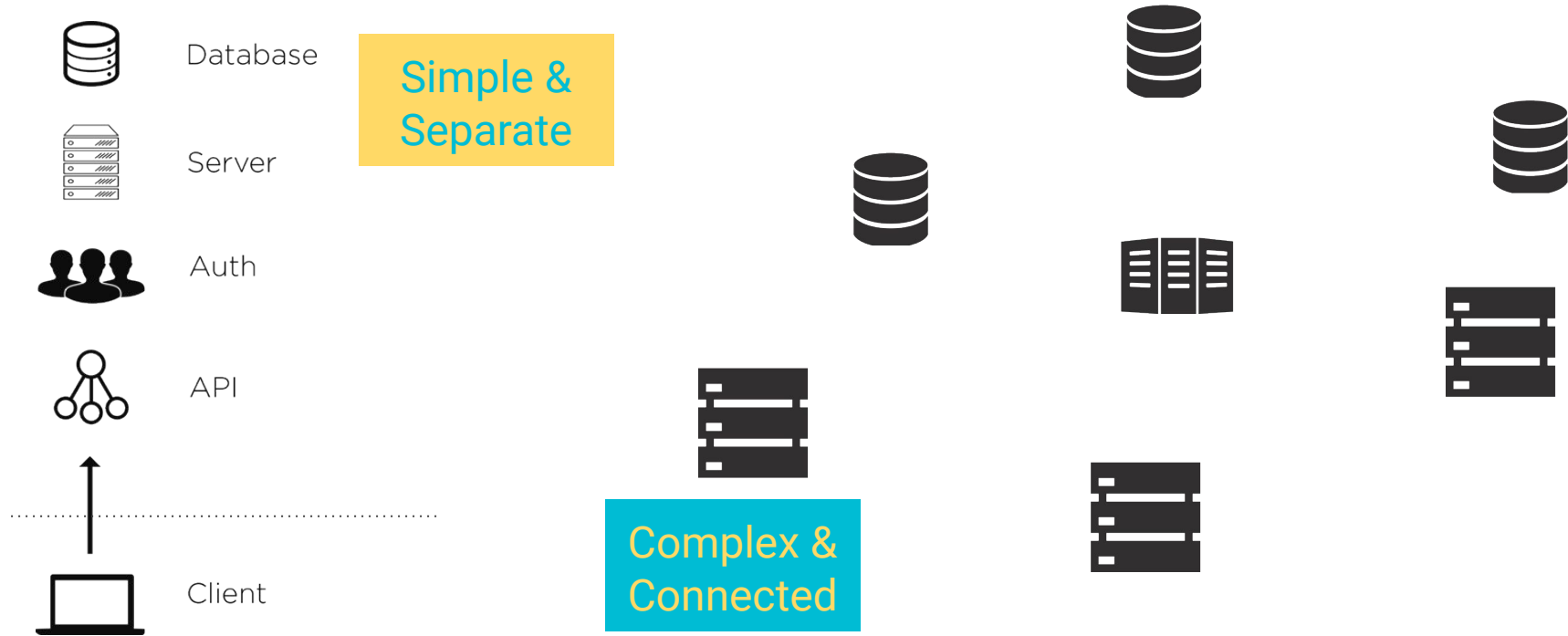
# Developer Understanding: No silver bullets..



Source: Giphy



# Architecture: Simple Vision vs. Harsh Reality





# Planning : Many questions. Decision overload.



Database

What **database** should we use?

What **back-end** tech should I use?



Server

What about user **analytics**?



Auth

What **authentication** provider do we use?

Where should I **host** my app?

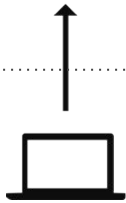


API

Does it **scale**?

How about **testing**?

Can I support **cross-platform**?



Client

Which **front-end** do I need?

Is it **fast**?

Does it work **offline**?

Is data **real-time**?

Is UX **responsive** reactive?



# Technology: Many distractions. Integration anxiety.



Database

What database should we use?

What back-end tech should I use?



Server

What about user analytics?

 **DAYS**

**SINCE LAST NEW JAVASCRIPT FRAMEWORK**



Why?

Why Not?

Why?

Why Not?



Auth

Why?

Where should I test my app?



API

Does it work scale?

How about testing?

Can it support cross-platform?



Client

**Rapidly Changing Ecosystem**

Does it work offline?

Is data real-time?

**FOMO**

Is UX responsive? reactive?



# Priorities: Users judge apps on front-end UX



Database



Server



Auth



API



Client

**Things users NEED**  
(= your app is built on this)

**Things users WANT**  
(= your app is rated on this)

What back-end tech should I use?

Where should I host my app?

What authentication provider do we use?

How about testing?

Can I support cross-platform?

Does it work scale?

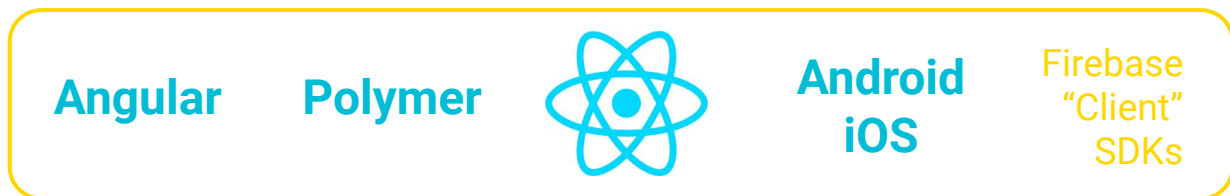
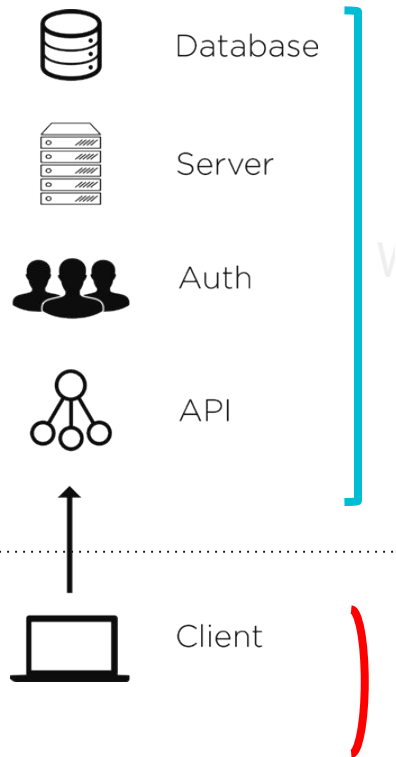
Is data real-time?

Is UX responsive?  
reactive?



**How can you balance complexity & speed?**

# Enter Firebase: Outsource your back-end ..



# Full-Stack Toolkit: Prototype with Firebase + “X”

Mobile Platform to  
**Develop, Grow & Monetize** Apps

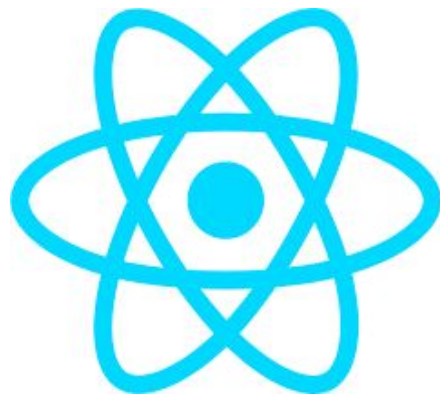
*“Just add front-end”*



**Synchronize**  
application **data**  
**across clients**

JS Library to Build Large, Complex  
**Declarative + Component-based** UI

*“Just add back-end”*



**Manage**  
application **state**  
**on client**

# Firebase

An Introduction



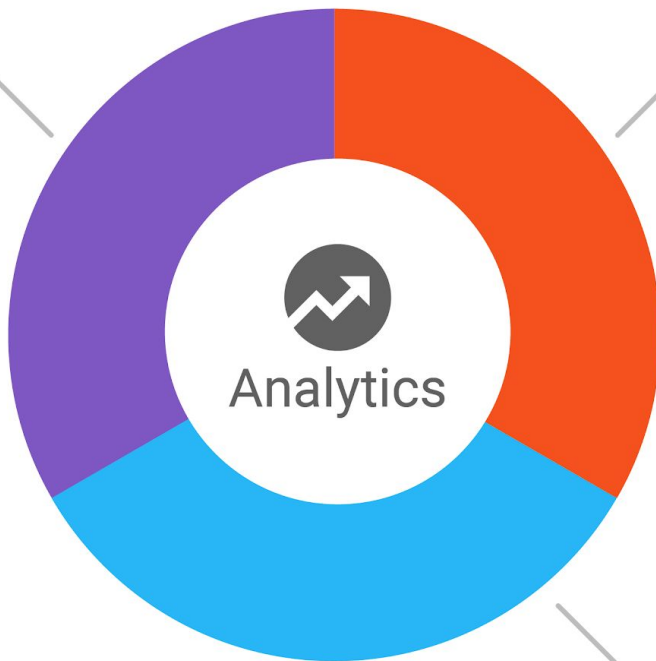
# Develop

## Backend Services

- Realtime Database
- Authentication
- Hosting
- Storage
- Cloud Messaging
- Remote Config

## App Quality Services

- Test Lab for Android
- Crash Reporting



# Grow

## Acquisition

- Dynamic Links
- Invites
- AdWords

## Re-Engagement

- Notifications
- App Indexing

# Earn

## In-app Ads

- AdMob



# 1. Create Project (Firebase Console)

### Create a project

Project name

Country/region

By default, your Firebase Analytics data will enhance other Firebase features and Google products. You can control how your Firebase Analytics data is shared in your settings at anytime. [Learn more](#)



The screenshot shows the Firebase Console interface. At the top, there's a navigation bar with a home icon, the project name 'friendlychat-react', a settings gear icon, and the word 'Overview'. Below this is a sidebar menu with categories: ANALYTICS, DEVELOP (including Auth, Database, Storage, Hosting, Test Lab, Crash), GROW (including Notifications, Remote Config, Dynamic Links), and EARN (including AdMob). The main content area has a blue header with 'Welcome to Firebase! Get started here.' Below this are three large circular buttons: 'Add Firebase to your iOS app' (blue), 'Add Firebase to your Android app' (green), and 'Add Firebase to your web app' (purple). At the bottom, there's a 'Discover Firebase' section with three cards: 'Analytics' (with an illustration of a person at a laptop), 'Auth' (with an illustration of a person with a badge), and 'Database' (with an illustration of server racks). Each card includes a brief description and a 'GET STARTED' button.

<http://firebase.console.google.com>

## 2. Configure Front-End (Firebase CLI)

Usage: firebase [options] [command]

Commands:

auth:import [options] [dataFile]	import users into your Firebase project from a data file(.csv or .json)
database:get [options] <path>	fetch and print JSON data at the specified path
database:push [options] <path> [infile]	add a new JSON object to a list of data in your Firebase
database:set [options] <path> [infile]	store JSON data at the specified path via STDIN, arg, or file
database:remove [options] <path>	remove data from your Firebase at the specified path
database:update [options] <path> [infile]	update some of the keys for the defined path in your Firebase
deploy [options]	deploy code and assets to your Firebase project
hosting:disable [options]	stop serving web traffic to your Firebase Hosting site
help [command]	display help information
init [feature]	setup a Firebase project in the current directory
list	list the Firebases to which you have access
login [options]	log the CLI into Firebase
login:ci [options]	generate an access token for use in non-interactive environments
logout	log the CLI out of Firebase
open [link]	quickly open a browser to relevant project resources
serve [options]	start a local server for your static assets
tools:migrate [options]	ensure your firebase.json format is up to date
use [options] [alias_or_project_id]	set an active Firebase project for your working directory

Options:

-h, --help	output usage information
-V, --version	output the version number
-P, --project <alias_or_project_id>	the Firebase project to use for this command
-j, --json	output JSON instead of text, also triggers non-interactive mode
--token <token>	supply an auth token for this command
--non-interactive	error out of the command instead of waiting for prompts
--interactive	force interactive shell treatment even when not detected
--debug	print verbose debug output and keep a debug log file

```
Nityas-Air:~ nitya$ firebase login
Already logged in as gdg.hudson.valley@gmail.com
Nityas-Air:~ nitya$ firebase init
```



You're about to initialize a Firebase project in this directory:

```
/Users/nitya
```

Before we get started, keep in mind:

- You are initializing your home directory as a Firebase project

? What Firebase CLI features do you want to setup for this folder? (Press <space> to select)

- Database: Deploy Firebase Realtime Database Rules
- Hosting: Configure and deploy Firebase Hosting sites

<https://firebase.google.com/docs/cli/>

# 3. Integrate Features (Firebase API/SDK)

## Develop

Build a better app and leave the operations to us. Save crucial development time and ship a high-quality, bug-free app.



### Cloud Messaging **iOS**



Deliver and receive messages across platforms reliably



### Authentication **iOS**



Reduce friction with robust authentication



### Realtime Database



Store and sync app data in realtime



### Storage **iOS**



Store files with ease



### Hosting **</>**

Deliver web content faster



### Test Lab **</>**

Test in the lab, not on your users



### Crash Reporting **iOS**



Keep your app stable

## Grow

Acquire and engage the right users at the right time. Take the guesswork out of growth.



### Notifications **iOS**

Engage with users at the right moment



### Remote Config **iOS**



Customize your app on the fly



### App Indexing **iOS**

Drive organic search traffic to your app



### Dynamic Links **iOS**



Send users to the right place inside your app



### Invites **iOS**



Empower your users to share your app



### AdWords **iOS**

Acquire users with the reach of Google

## Earn

Monetize your app through rich and engaging ads.



### AdMob **iOS**

Monetize through engaging ads









### Analytics **iOS**

Reimagine analytics for mobile

At the heart of Firebase is Firebase Analytics, a free and unlimited analytics solution. See user behavior and measure attribution from a single dashboard.

## 3. Integrate Features (Firebase API/SDK)

AngularFire	The officially supported AngularJS binding for the Firebase Realtime Database and Authentication.	</>
PolymerFire	The officially supported Polymer bindings for the Firebase platform.	</>
ReactFire	The officially supported React binding for the Firebase Realtime Database.	</>
EmberFire	The officially supported Ember binding for the Firebase Realtime Database and Authentication.	</>
BackboneFire	The officially supported Backbone binding for the Firebase Realtime Database.	</>

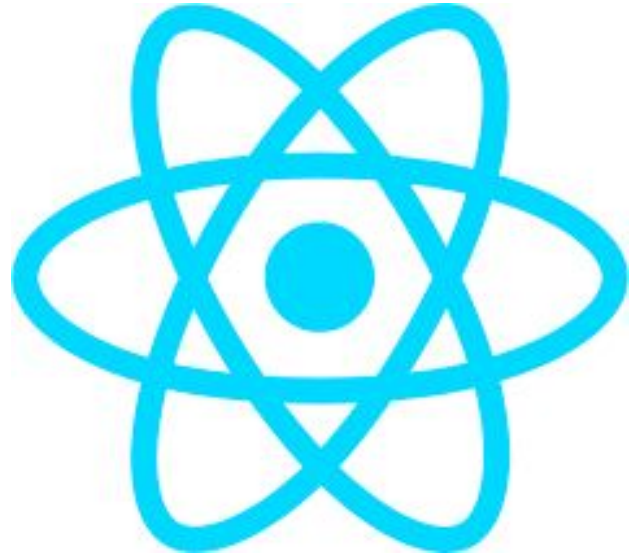
Firebase SDK	The official Firebase SDK.	  </>    
--------------	----------------------------	---

Firestore	Firestore is a light-weight React Native JS layer sitting atop the native Firebase libraries for iOS and Android.	<a href="#">GitHub</a>
-----------	---	------------------------

<https://firebase.google.com/docs/libraries/>

# React

An Introduction



# What is React?

A JS library for  
building big, fast  
user interfaces

the “V” in MV\*

(Proven) Technology from Facebook

## Declarative

Create a view for every state in your application. React will auto-render views on state changes, and update DOM with minimal diffs for efficiency

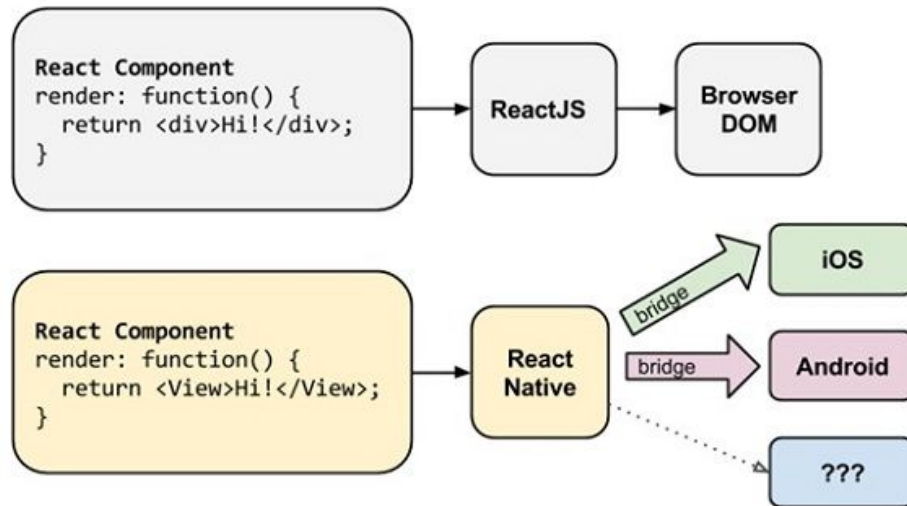
## Component Based

Encapsulate state per view (vs templates).  
Composable, to build big apps.  
Expressive. Brings full power of JS to UI

# What is React?

And it can be  
cross-platform

with React Native



[Writing Cross-Platform Apps with React-Native](#)  
(InfoQ, Bonnie Eisenmann, Feb 2016)

(Proven) Technology from Facebook

# Example: A React App

```
var Hello = React.createClass({
  render: function() {
    return <div>Hello {this.props.name}</div>;
  }
});

ReactDOM.render(
  <Hello name="World" />,
  document.getElementById('container')
);
```

1. Component Based
2. Auto-Render on state change
3. JSX Format (JS-like markup, transpiles to HTML)
4. Virtual DOM



# Scaffolding: Now much easier ...

## NPM

Manage package dependencies

## Babel

Transpiles ES6 into JS for use on older browsers

## Webpack

Collects and packs all components into one app bundle

### Create React App build passing

Create React apps with no build configuration.

- [Getting Started](#) – How to create a new app.
- [User Guide](#) – How to develop apps bootstrapped with Create React App.

#### tl;dr

```
npm install -g create-react-app

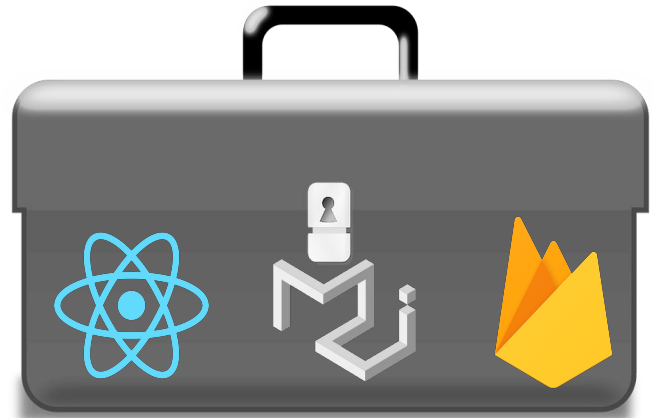
create-react-app my-app
cd my-app/
npm start
```

Then open <http://localhost:3000/> to see your app.

When you're ready to deploy to production, create a minified bundle with `npm run build`.

# React-Fire

Putting it all together



# Firecasts: “Getting Started With React + Firebase”



<https://www.youtube.com/user/Firebase/playlists>

# 4: Original Codelab: Friendly Chat App (Vanilla JS)

The screenshot shows the 'Overview' page of a Codelab titled 'Friendly Chat'. On the left is a vertical navigation menu with 12 steps: 1. Overview (selected), 2. Get the sample code, 3. Import the starter app, 4. Create a Firebase project and Setup your app, 5. Install the Firebase Command Line Interface, 6. Run the starter app, 7. User Sign-in, 8. Read messages, 9. Database Security Rules [Optional], 10. Send Messages, 11. Send Images, 12. Storage Security Rules [Optional]. The main content area shows a chat interface with a blue header 'Friendly Chat' and a user profile for 'Nicolas Garnier'. The chat history includes messages: 'Hey Ian' (Nicolas Garnier), 'Hi Nicolas! Coffee time?' (Ian Barber), 'Cool let's got' (Nicolas Garnier), and 'meet you in five!' (Ian Barber). A message with a coffee cup image is partially visible. At the bottom is a 'Message...' input field and a 'SEND' button.

## 2. Get the sample code

Clone the [GitHub repository](#) from the command line:

```
$ git clone https://github.com/firebase/friendlychat
```

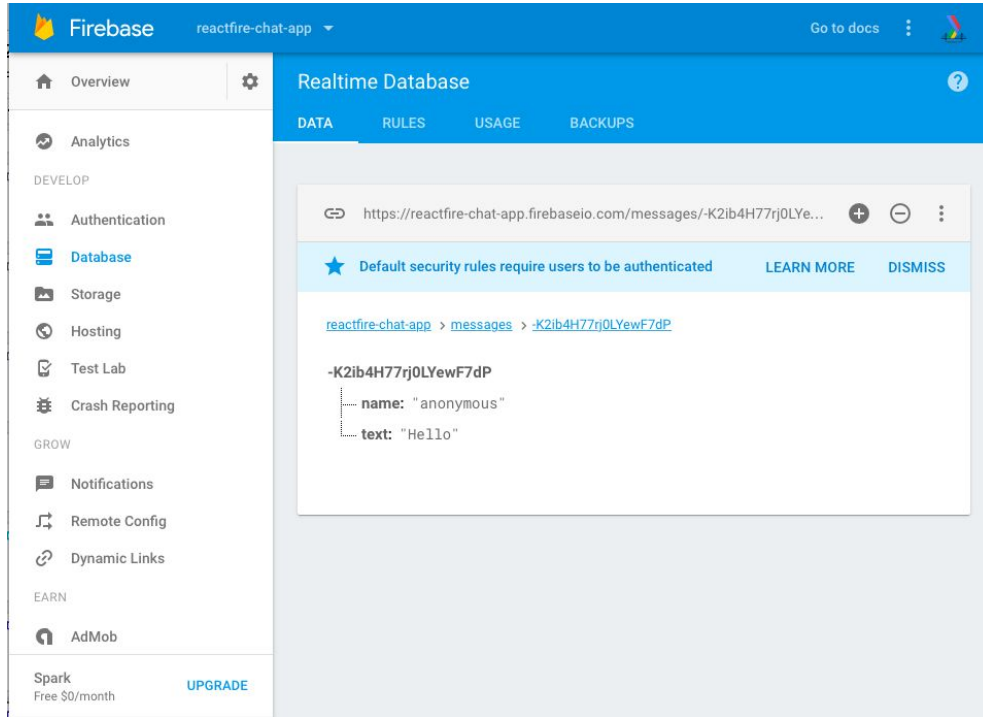
The firebase-codelabs repository contains sample projects for multiple platforms. This codelab only uses these two repositories:

- **web-start**—The starting code that you'll build upon in this codelab.
- **web**—The complete code for the finished sample app.

**Note:** If you would like to run the finished app, you will still have to create a project in the Firebase console, see the [Create a Firebase project and Setup your app](#) section for instructions.

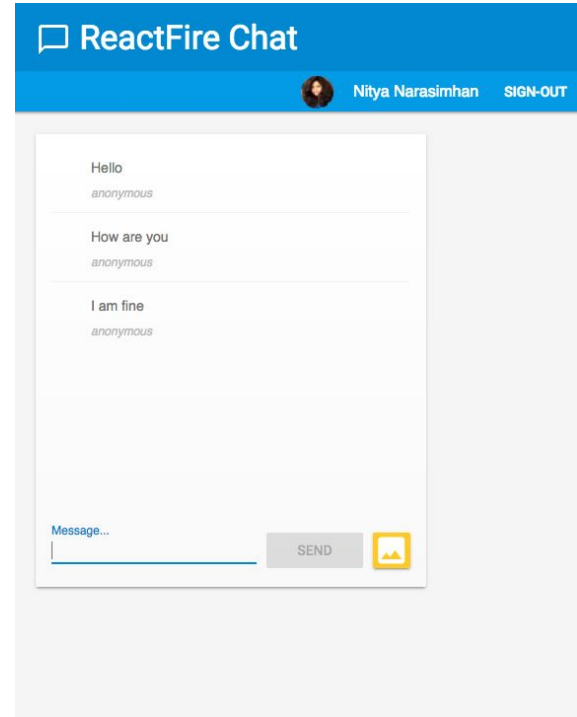
<https://codelabs.developers.google.com/codelabs/firebase-web/index.html>

# 4. Refactored Codelab: **ReactFire Chat App**



The screenshot shows the Firebase Realtime Database console for a project named 'reactfire-chat-app'. The left sidebar contains navigation options: Overview, Analytics, Authentication, Database (selected), Storage, Hosting, Test Lab, Crash Reporting, Notifications, Remote Config, Dynamic Links, and AdMob. The main area displays the 'Realtime Database' interface with tabs for DATA, RULES, USAGE, and BACKUPS. A message object is shown with the following structure:

```
reactfire-chat-app > messages > -K2ib4H77rj0LYewF7dP
{
  name: "anonymous",
  text: "Hello"
}
```



The screenshot shows the 'ReactFire Chat' application interface. The header includes the app name and a user profile for 'Nitya Narasimhan' with a 'SIGN-OUT' button. The chat area displays three messages from an 'anonymous' user:

- Hello
- How are you
- I am fine

At the bottom, there is a text input field labeled 'Message...', a 'SEND' button, and an image upload icon.

<https://reactfire-chat-app.firebaseio.com>

**Let's scan the code..**

<https://github.com/gdg-hudson-valley/firebase-projects/tree/master/reactfire-chat>

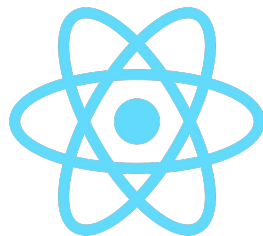
# Recap



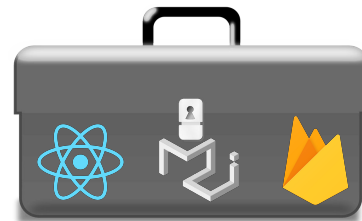
Real-Time App  
**Table Stakes**



Intro to  
Firebase



Intro to  
React



Real-Time App  
**Toolkit**

# Thank You!!

Questions?

**SLIDES:**

<http://bit.ly/gdgcr-jan2017-reactfire>

**CONTACT INFO:**

*@nitya*

*+NityaNarasimhan*

*#in/nityan*