

## GETTING FAMILIAR WITH FIREBASE

Nandini Surana, Usman Yousuf

### ABSTRACT

In this study we will introduce everyone with Google firebase API and its related features. Firebase is a Google provided API for database storage, authentication and many more useful features which we can sync into your android, IOS or web application. A real-time database is one feature which stores data to database and fetches data from it very quickly but Firebase is not just a real-time database, it is much more than that. Firebase provides us with many SDKs for the different platforms such as node.js and python for server side and android, IOS, web for client side. This article covers an overview on how to use firebase as a backend for your Web and Android application development. Google Firebase has many features like Authentication, database, storage, hosting, notification, analytics, dynamic links, remote configuration. Analytics feature enables the application developer to understand how users are using his application. Authentication allows developers to store users data and integrate with authentication providers such as Facebook, Google etc. Firebase Supports Cloud Messaging allowing developers to send notifications to its users.

Keywords- Authentication, Firebase, Real-time database, Android

### I. INTRODUCTION

Firebase is a backend platform for building Web, Android and IOS applications. It offers real time database, different APIs, multiple authentication types, hosting platform. Firebase allows developers to focus on crafting fantastic user experiences. They don't need to manage servers or to write APIs. Firebase is your server, your API and your data store, all written so generically that you can modify it as per your need. Firebase can power your app's backend, including data storage, user authentication, static hosting and more. Focus on creating extraordinary user experiences Firebase will take care of the rest. Build cross platform native mobile and web apps with our Android, IOS and JavaScript SDK's. You can also connect Firebase to your existing backend using server-side libraries or it's REST API.

### II. HEADINGS

1. Introduction
2. Characteristics of Firebase
3. Services of Firebase
  - 3.1 Analytics
  - 3.2 Development 46 | Page
    - 3.2.1 Authentication
    - 3.2.2 Real-Time Database
    - 3.2.3 Cloud Messaging
    - 3.2.4 Hosting
    - 3.2.5 Storage
  - 3.3 Grow
    - 3.3.1 Notifications
    - 3.3.2 Remote Config
    - 3.3.3 Dynamic Links
  - 3.4 Earn
    - 3.4.1 Ad Mob
4. Adding Firebase to the Web Project

5. Using Firebase Real-Time Database for Web Application
6. Firebase over Parse Server Applications
7. Conclusion and Future Scope
8. Reference

## 2.Characteristics of Firebase

### a. Free to start, scale with ease

Firebase is absolutely free to start; all its services are provided to the developer without-any charges. But as soon as the application grows there are charges involved in it.

### b. Work across platforms

Firebase provides support for many platforms such as Web, Android and IOS. There is scope for C++ and Unity as well.

### c. Make smart, data-driven decisions

During the development of the application, the developer is been supported by many of the services provided by Firebase. So that developer doesn't have to think about storage of data and use the data as per the requirement.

## 3. SERVICES OF FIREBASE

### 3.1 Analytics :

This feature enables the application developer to understand how users are using the application. The SDK Capture events and properties on its own and allows to get custom data. The dashboard provides details Like the most active user or what feature of the application is been used most. It also provides you with summarized data.

- ❖ Designed for apps
- ❖ Event and user centric
- ❖ Connects across Firebase
- ❖ Free and Unlimited

### 3.2 Development :

**i. Firebase Authentication:** -This feature in firebase will let only authorized users access the application. Firebase provides login through Gmail, GitHub, Twitter, Facebook and also let the developer create custom authentication.

- ❖ Authentication & account management
- ❖ Supports:
  - Email & password
  - Social providers
  - Existing authentication systems

**ii. Firebase Real-Time Database:** -Database in firebase is a cloud-based database and doesn't requires SQL-based queries to store and fetch data. It is highly reliable and superfast which means that data is updated and synchronized in no time and it is maintained even when user loose the internet connection.

- ❖ Cloud-hosted No-SQL database
- ❖ Synchronization & conflict resolution
- ❖ Access directly from your app

**iii. Firebase Cloud Messaging:** - It lets you to deliver messages at different platforms with no cost.

Messaging is also used for notifications purposes.

- ❖ Firebase Cloud Messaging
- ❖ Enable Push Notifications
- ❖ Based on GCM(Google Cloud Messaging)

**iv. Firebase Hosting:** -Firebase is also used for hosting purposes. Firebase delivers web content very fast and the content is always delivered securely.

- ❖ Serve static assets (images)

- ❖ SSL (Security) by default

v. **Firestore Storage:** - Firestore also provides storage facility. It can store and retrieve content like images, videos and audio directly from client SDK. Uploading and downloading is been done in the background. Data stored is safe and it can be access by the authorized user/s only.

- ❖ Easy file storage
- ❖ Handles poor connectivity

### 3.3 Grow

i. **Firestore Notifications:** - This feature lets you create attractive push notifications. You can send this to particular user/s .It is a fast and easy way to create and send notifications.

- ❖ Let you re-engage users quickly without any hassle.
- ❖ No additional coding required
- ❖ Messages can be sent to particular devices or all devices
- ❖ Integrates with analytics

ii. **Firestore Remote Configuration:** - This feature is basically for the testing of the application prior to apply any updates to it.

- ❖ Control custom key-value pairs from the Console
- ❖ Changes propagate instantly

iii. **Firestore Dynamic Links:** - This feature is to check whether your application is responsive amongst all the platforms or not.

- ❖ Customize different user experiences via a single URL
- ❖ Works across platforms
- ❖ Preserves URL state, even through app installation flow

### 3.4 Earn

i. **Ad Mob:** - This feature allows developer to earn through application. It is supported across many platforms and is easy to implement .

- ❖ Earning source by the application
- ❖ Easy to linkup with the application
- ❖ Supports cross platform (IOS as well as Android)

\*\*\*\*\*ad diagram\*\*\*\*\*

## 4. ADDING FIREBASE TO THE WEB-PROJECT

To add Firebase to the app, you'll need a Firebase project, the Firebase SDK, and a short snippet of initialization code which will have few details about the project.

- i. Create a Firebase project in the Firebase Console, if you don't already have one.
  - ii. If you already have an existing Google project associated with your app, click **Import Google Project**. Otherwise, click **Create New Project**.
  - iii. If you already have a Firebase project, click **Add App** from the project overview page.
  - iv. Click **Add Firebase to your web app**.
  - v. Note the initialization code snippet, which you will use in your application.
- o The snippet contains initialization information to configure the Firebase JavaScript SDK to use Authentication, Cloud Storage, the Real time Database, and Cloud Firestore.
  - . The individually installable components are:
    - firebase-app - The core firebase client (required).
    - firebase-auth - Firebase Authentication (optional).
    - firebase-database - The Firebase Real-time Database (optional)
    - firebase-firestore - Cloud Firestore (optional).
    - firebase-storage - Cloud Storage (optional).
    - firebase-messaging - Firebase Cloud Messaging (optional).

## 5. USING FIREBASE REAL-TIME DATABASE FOR WEB APPLICATION

The Firebase Real time Database is a cloud-hosted database. Data is been stored as JSON and synchronized in real time to every connected client. When you build cross-platform apps with IOS, Android, and JavaScript SDKs, all of your clients share one Real time Database instance and automatically receive updates with the Recently updated data.

### Basic write operations:

For basic write operations, you can use `set()` to save data to a specified reference, replacing any existing data at that path. For example a social blogging application might add a user with `set()` as follows:

```
function writeUserData(userId, name, email, imageUrl) {
  firebase.database().ref('users/' + userId).set({
    username: name,
    email: email,
    profile_picture : imageUrl
  });
}
```

### Read data once

In some cases you may want a snapshot of your data without looking for changes, such as when initializing a UI element .You can use the `once()` method to simplify this scenario: it triggers once and then does not trigger again .

```
var userId = firebase.auth().currentUser.uid;
return firebase.database().ref('/users/' +
userId).once('value').then(function(snapshot) {
  var username =(snapshot.val() && snapshot.val().username) || 'Anonymous';
  //...
});
```

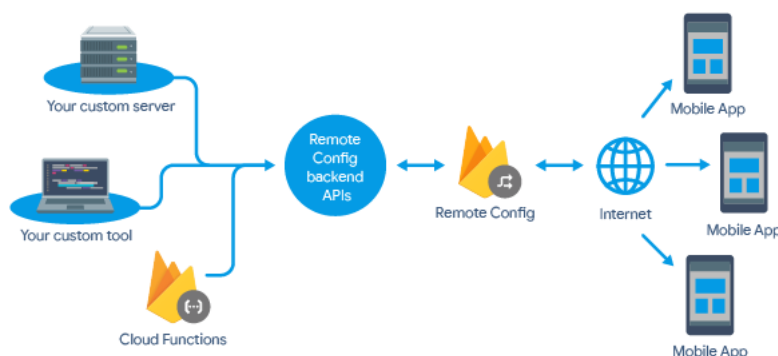
## 6. FIREBASE OVER PARSE SERVER APPLICATIONS

- Firebase is provided by Google while Parse Server is provided by Facebook.
- The purpose of Firebase is for Fast Real Time Applications, And that of Parse Server is for Static backend services.
- Setup for Firebase is fast, easy and automatic while for Parse server it is manual Setup.
- Firebase stores data in JSON format and uses Google Cloud Storage, while Parse Server usually uses

MongoDB.

- In Firebase, Notifications is supported for cross platforms while in Parse Server it is only for Android and IOS.

- Firebase provides Google Hosting which is free up to 100 simultaneous connections, Parse Server provides self-hosting and Parse hosting without any limits.



## 7. CONCLUSION AND FUTURE SCOPE

In this paper, it has been discussed about Google provided firebase API and its unique features. It has been conveyed how to add firebase to our web project and how to use different features in our android application according to our requirements. Google is adding new features to firebase continuously; Ad Sense for firebase is in beta phase .

So in future, we can study those new features and how to implement them in android application and

why we need to keep using firebase than any other database service out there. Developers can save time on developing a server for their applications and just use firebase instead.

Firebase is coming with a new feature called as cloud. Firestore which can be used for the storage of large data. Future research can be focused on authentication service provided by Firebase, how to implement authentication in your applications, authentication analysis for cross platform development.

Research can also be focused on firebase cloud functions and how to host our own servers on Firebase Platform.

## 8.Reference

Read Write Data on Web – Firebase

a. (<https://firebase.google.com/docs/database/web/read-and-write>)