



Traditional Infrastructure vs. Firebase Infrastructure

Mayuri Chandakant Wadkar, Priyanka Patangrao Patil

M.C.A. (IMCOST), Mumbai University C-4, Wagle Industrial Estate, Near Mulund (West) Check Naka,
Thane (West), Maharashtra, India

ABSTRACT

The purpose of this study is to introduce everyone with the amazing features of Firebase that can be useful to rocket boost an Application.

Traditionally to get best out of an app we need a huge, expensive, bulky methods or code to be used or implemented, long process of testing which consist of a server and large amount of complex codes, third party API's and even dedicated teams. This takes customer and developer quite far from their destiny of having an application which would be reliable, bug-free and sustainable in long run.

Google's Firebase Provides such an extraordinary aid to developers/ customers which can't be possible to think of, from traditional infrastructure.

This article focuses on digging into an Amazing product named Google's Firebase and its various features such as Cloud Messaging, Authentication, Storage, Realtime Database and few more.

Keywords: *Firestore, FCM, Firebase Real-Time Database, Firebase Authentication, Firebase Analytics, Firebase Storage*

I. INTRODUCTION

To build and high-end responsive application which would work on mobile as well as the web we need certain components such as a server where our code deploy storage space for storing our files, DBMS to store data, complex code to implement even common features, reporting systems, reliable network, testing

tools, etc. It is not possible for every org to acquire such infrastructure in the first go or to bear the maintenance cost in the short run. So the best solution is to go with the firebase infrastructure which is lower in cost as compared to traditional infrastructure and a product where you can find most modern methods and equipment under one roof. Firebase is a combination of many products with Google's infrastructure and developer friendly environment. Firebase provides end to end facilities that a developer would need throughout his course of development. The main and best feature that no other provider provide is the integration of all product in one platform.

II. FIREBASE FEATURES OVER TRADITION SYSTEMS

- A. Messaging
- B. Storage
- C. Analytics
- D. Crash Reporting Database
- E. Authentication
- F. Remote Config
- G. Test Lab
- H. Database

A. Messaging: One of the cool ways to communicate or stay touch with your app user is through timely notification and messages. Notification helps you to guide, aware and inform the user.

There are many services that can be integrated into your application to enable the features, some of the services where Apple's APNs (Apple Push Notification Service) only for IOS, Adobe Mobile Service, Google's GCM (Google Cloud Messaging

Service – Replaced with FCM), PushApp, Amazon SNS (Simple Notification Service) and many more. The advantage of FCM over GCM and APNS is that it has the very high success rate as compared to any other product. So the chance of missing a notification or message by a user is less. Firebase gives a choice to developers those who are looking for an adaptable notification platform which requires minimal coding effort and a graphical console for sending messages.

B. Storage: For Storing application data such as picture, audio, documents, and videos we need storage facilities on a cloud which would enable to access data instantly, cloud storage is also important for backup and recovery.

Amazon Simple Storage Service (Amazon S3) is one of the best option available today in the market for Web Storage their storage pricing varies by region which makes it quite usually for a worldwide application.

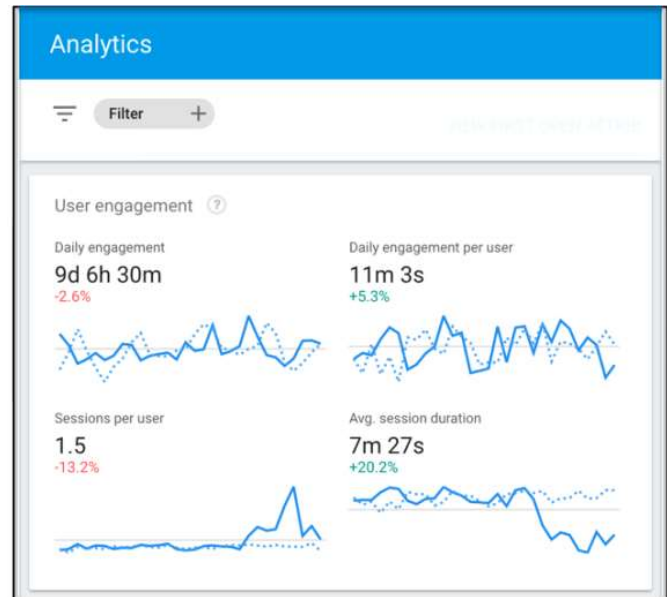
Where on the other hand Firebase pricing is same for all regions throughout the globe. On an average both charges quite same which are approx \$0.0245 per GB.

C. Analytics: Just creating a fabulous and fully functional application is not enough and it is also not possible to collect all the user information at one go. Many times user preferences can change depending upon trends, age, income etc. Analytics started back by simply counting a number of people visited on a web page.

So to tackle all such cases Analytics came into the picture, analytics helps to understand not only app rejections but also user preference, taste, likes, and choices. Analytics able to create reports based on downloads, gender, income, geographic area, age, time, trends, season.

There are very few products in the market which enable analytics integration in your application. Adobe Marketing Cloud is the best in this class, it consists many parameters based on which the organization can make their future decisions.

Firebase Analytics can compete with Adobe Marketing Cloud but in some space, Adobe does a great job, but a combination of many features makes firebase to overcome this point.



D. Crash reporting: No application is bug-free even testing it for hundreds of time, some bugs lead to the bad design and some leads to bad user experience. It requires huge funds and time to make an application crash and bug-free but investing in testing also not guarantees crash-free application. Many a time some crashes occurs at testing environment but never comes in live senior such cases leads to heavy wastage of fund and time in solving those problems.

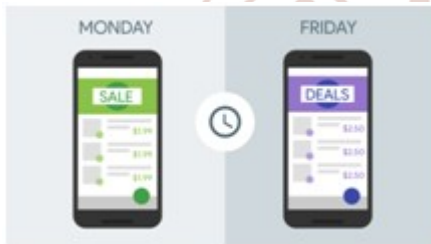
So it is an excellent idea to have a crash reporting system in our application. There are few products present in the market which were able to generate crash reports. Crashlytics is one of the few crash reporting systems which is owned by Twitter it is free to use. Firebase Crash Reporting helps to get crash reports based on the severity and also provides detail information about crash occur.

E. Authentication: Authentication is the main module in any modern application because without authentication we would not able to identify the user, and if the system is incapable of that then the application will not sustain in the long run. Traditionally for authentication user has to fill a form which consists of his information, and to become a user at multiple places or to get registered with multiple application the user has to repeatedly fill the form. This leads to bad user experience and many a most of the time user leaves the application. Apart from the user side for authentication, we have to deploy complex code with multiple API's calls on our servers which leads degradation of server

performance. So to overcome this lengthy procedure of authentication one must use Firebase Authentication, which provides authentication service along with UI. It also enables authentication using Facebook, Google, Twitter, Github credentials.

F. Remote Config: It is a cloud service that gives you a chance to change the values of variables and appearance of your application without requiring users to download an application update. This means we can change the theme of the application with respect to the season without forcing the user to update the app. This is a very handy feature by Firebase that will help to improve the user experience to a great extent.

Now we can show offers, deals, discount whenever we want just by simply changing the values at Firebase Console.



G. Test Lab: For every small and medium size organization it is not possible to conduct all kinds of test with all kinds of devices, testing plays and critical part in identifying bugs which help to reduce crashes present in the application so no developer should ignore testing of his application. Testing an iOS app is far more easy than to test an Android app because the availability of large variety of screen size for Android devices. Firebase test lab provides gadgets hosted on the Google data-center.

H. Database: There is a great demand for Real-Time Databases nowadays from customers. A real-time database is a database system which uses real-time processing to handle workloads whose state is constantly changing. Nearly most of the databases uses synchronous calls to perform database operations which means unless the first query calls end next call cannot be done, this may lead to deadlock in certain situation E.g if we have an application which needs 10 database operations on the same table, if 1st call fails due to some reason at that scenario the rest 9 calls will not be executed. In every database the application has to call the data after a certain time interval to maintain

consistency this increases the network usage. Many times the database calls are wasted because there is no guarantee that such calls will fetch unique data (Fig 1.0).

Firebase gives the Real-time database, in Firebase the data is stored as JSON and synchronized continuously to each associated client this means that the client need not has to make a call to fetch data changes, Firebase takes the initiative and notify the application every time the data is changed this means unless the data at server get changed no call or response will be done which leads to optimum utilization of bandwidth (Fig 1.1).

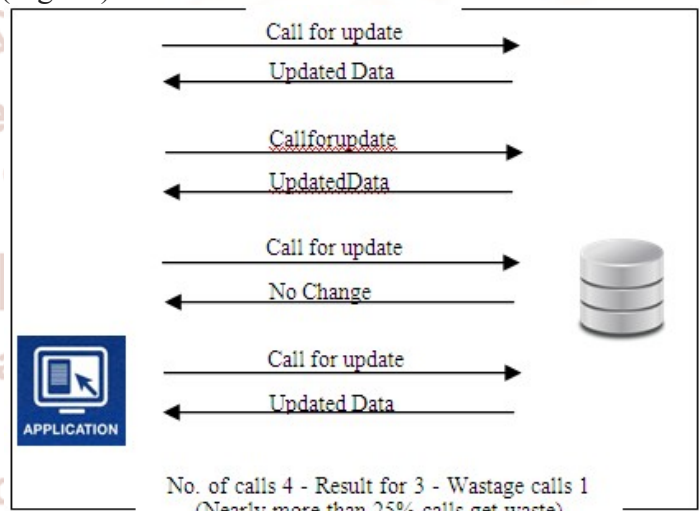
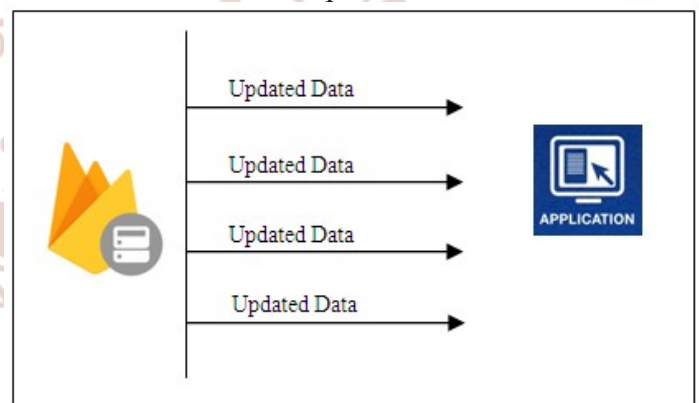


Fig 1.0 Represents Tradition Database calls and response.



No application calls needed as soon as the data changes firebase notify the application
Fig 1.0 Represents Firebase Realtime Database.

III. FIREBASE FEATURES

All point covered above speaks about the Firebase Features over Traditional system let dive a little bit in overall benefits of Firebase infrastructure.

Because it uses Google’s Infrastructure the chances of server failure are negligible and the speed and support

they provide are world class. As its official release in April 2012 (by Google) the number of developers is increasing day by day which makes a huge community for support and advancement.

There are many other services present and few of them are even better than some of the Firebase features but the integration of all products under one roof gives Firebase a competitive advantage over other.

The pricing is also an attractive feature of Firebase they provide the services at the lowest cost and more reliable than others too.

CONCLUSIONS

In this paper, I tried to undergo all possible product and systems which would make the work of developer quite easy. Mostly firebase is covered because it has rich GUI and huge developer community and also cost efficient. The Point covered in the paper are not just referenced but used in the real application too. So, I hope this paper will help out there to all those developers who are just get started development and to them also who want to switch from another platform to Google's Firebase.

ACKNOWLEDGMENT

I thank my colleagues from IMCOST Ms. Priyanka

Patil and Mr. Pranay Gaikwad who provided insight and expertise that greatly assisted the research, I thank Mrs. Sheeba James for her assistance by providing proper formats and all faculties for guiding. I would also like to show our gratitude to the librarian staff Mr. Milind Dubal for sharing references during the course of this research.

REFERENCE

1. Firebase Pricing: <https://firebase.google.com/pricing/>
2. Firebase Features :- <http://www.cmarix.com/14-Benefits-of-using-Firebase>
3. Firebase Products:-<https://firebase.google.com/products/>
4. Apples APNS:-
https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/APNSOverview.html#//apple_ref/doc/uid/TP40008194-CH8-SW1
5. Navdeep Singh, Study of Google Firebase API for Android, IJRCCE (An ISO 3297: 2007 Certified Organization) Vol. 4, Issue 9, September 2016
6. Firebase Release and Services :-<https://en.wikipedia.org/wiki/Firebase>