iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** I **Month of publication:** January 2022

DOI: https://doi.org/10.22214/ijraset.2022.39958

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue I Jan 2022- Available at www.ijraset.com

A Review on Firebase (Backend as A Service) for Mobile Application Development

[1]Prachi R. Saraf, [2]Sakshi M. Jadhao, [3]Saurabh J. Wanjari, [4]Shital G. Kolwate, [5]Prof. Ankush D. Patil
[1, 2, 3, 4, 5]*Department Of Computer Engineering*
[1, 2, 3, 4]*Government College Of Engineering, Yavatmal, Maharashtra, India,* [5]*Assistant Professor, Government College Of Engineering, Yavatmal, Maharashtra, India*
[1, 2, 3, 4]*Dr. Babasaheb Ambedkar Technological University, Lonere, India*

Abstract: ***If we see current scenario mobile users are rapidly increasing in a very large scale. Better performance and good visuals are needed while developing any mobile applications. In the current mobile market, there are various operating system which are supported by many mobile devices. . Since these platforms have different SDKs and tools for developing mobile applications, SDK tools will include a range of things, including libraries, documentation, code samples, processes, and different guides that developers can use and integrate/combined into their own apps. Cross platform development offers the flexibility for building applications using universal language like Javascript which can then be exported to numerous smartphone platforms. Firebase is a platform for mobile and web application with tools and infrastructure which allows developers to build high quality applications. Firebase consists of features that developers can use together to meet their needs. Earlier Firebase was used as a real-time database, which provides developers an API to sync and store data across various clients. Over time, it's product has been expanded to become a full suite for an app development. Day by day new platform approaches are evolved. Firebase is also powerful API i.e Application Program Interface to sync and store data in realtime helping to build realtime apps for the web. It also takes care of most of the backend concerns and is dubbed a Backend as a Service (BaaS) and also pushes state to your application utilizing web sockets.***

Keywords: Android, iOS, Cross Platform Development, Flutter, Dart, Widgets, Firebase.

I. INTRODUCTION

In the era of rapid prototyping, we can get bright ideas, but sometimes they are not applicable if they take too much work. Frequently, the back-end is the limiting factor - many considerations never apply to server-side coding due to lack of knowledge or time. Firebase is BaaS [Backend-as-a-Service], and it is a real-time database which is basically designed for mobile applications. Google Firebase is Google-backed application development software which allows developers to develop Android, IOS, and Web apps**.** For reporting and fixing app crashes, tracking analytics, creating marketing and product experiments, firebase provides several tools. The Firebase Realtime Database is a cloud-hosted database in which data is stored as JSON. In real-time data is synchronized to every connected client. All of our clients share one Realtime Database instances and automatically receive updates with the newest data, when we build cross-platform applications with our iOS, and JavaScript SDKs. The Realtime database helps our users collaborate with one another. Cloud Firestore is newer, but it's not replacing the Firebase Real-time Database. Cloud Firestore is a flexible as well as scalable NoSQL cloud database. It is used to store and sync data for client and server-side development. It is used for mobile, web, and server development from Google Cloud Platform and Firebase.

II. LITERATURE REVIEW

A. Backend As A Service(BaaS)

In the area of mobile application development, there are two major terms on which the developers has to work i.e. for frontend and backend. Backend is the most crucial part of mobile application development which responsible for storing the data, securing data, etc. The backend of the application is like a server for mobile apps, as it stores and sorts the data properly and the end user can only see the necessary information. BaaS i.e Backend-as-a-Service is a cloud service model in which developers outsource all the behind-the-scenes aspects of a web applications or mobile application so that they only have to write and maintain the frontend part. BaaS vendors provide pre-written software for activities that take place on servers, such as user authentication, database ,remote updating, and push notifications (for mobile apps), as well as cloud storage and hosting.

©IJRASET: All Rights are Reserved

B. Firebase

1) *Introduction to Firebase:* Firebase is a real-time database and also acts as a Backend-as-a-Service(BaaS). It allows to store a list of objects. Google Firebase is Google-backed application development software which allows developers to develop applications for **Android, iOS,** and **Web apps**. Firebase is a grouping of Google's many services in the cloud, including instant messaging, user authentication, real-time database, storage, hosting, etc.[1] Firebase offer real time database, authentication, cloud storage, cloud functions, etc. Firebase accomplishes real-time data in the database. Firebase makes easy to exchanges the data to and from the database. Firebase provides backend for- iOS, Android, and Web applications. Firebase applications can be arranged over a protected connection to the firebase server. Firebase offers a dashboard simple control. It provides number of useful services. It is highly secure and minimal setup. Cloud Functions feature is one of the new feature of firebase cloud which allows developers to write programs in JavaScript language and placed them on the Firebase cloud platform.[1] Using Firebase, each unit can directly access the database.[2] Firebase is a Google-owned multi-service cloud-computing solution for mobile and web developers [3]. The feature grouping in Firebase rushes the cloud database integration automatically in both web and mobile app.[4]

2) *Ease of using Firebase:* Firebase is a web application development platform created by Google. It lets you to developed the whole application on the front-end without any server-side code. At the same time, it does let you set up some server-side logic through Firebase Functions if you need to react to certain events (Creation of data or files, login, https requests) so that you can send emails or push notifications or process the data after it is written.It's easy to start a project with Firebase or add a Firebase to your project. It allows real-time database connection, which means multiple users can see the changes in the data when the data gets created or edited. Data transmission is handled with web sockets so you don't have to send requests to get new data, you only need to subscribe once.The same applies for file storage. Quick setup authentication through the major providers (Google, Twitter, Facebook, GitHub). https by default - secure http traffic without setting up certificates. Any static html/javascript content can be hosted.

3) *Services of Firebase*

a) *Analytics:* This feature are also enables the application developer to understand how users are using his application. The SDK capture events and properties on its own and also allows you to get custom data. The dashboardalso provides details like your most active user or what feature of your application is used most. It also provides you with summarized data.

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

b) *Development*

- *Firebase Authentication:* In the current world, authentication of user is most crucial requirements for Android apps. Authenticating users is very important and writing the code for authentication is a major task, but with the help of firebase we can do it very easily and effectively. Some of the apps recognize the identity of a user. By knowing the identity of user, the application saves the data securely over the cloud. Firebase Authentication supports authentication using email, Google, facebook,mobile phone number etc. In order to authenticate your app, first you have to get the authentication credential from the user. The credential may be phone number, password, etc. After getting the credentials, it is transferred to the Firebase Authentication SDK. Firebase backend will verify the credentials entered by the user and gives some response to the user. We can verify the identity of user by providing authentication token If the user is successfully signed in by into the application then the data stored in the database can be accessed by the user. If the user is authenticated then users can read and write data to the Firebase Realtime Database and Cloud Storage. But we can modify the rules in firebase realtime database and cloud storage and control the access of users.
- *Firebase Realtime Database:* The Firebase Realtime Database is a cloud-hosted and NoSQL database that also gives you feature to store and sync data between your users in realtime.

Key features of firebase include:

- Documents and collections with powerful querying
- iOS, Android, and SDKs with offline data access
- Real-time data synchronization
- Automatic data replication with strong consistency

- Synchronizes data between two devices in real-time. Our different Android, iOS, and Javascript SDKs sync your app data almost instantly. This makes it easy to build reactive apps, automatically sync data across devices, and build powerful collaborative features.
- Enables offline data access through a powerful, on-device database. This local database meaning your app will function smoothly, even when your users lose connectivity. This offline mode is available on Web, iOS and Android.
- Enables serverless development. Cloud Firestore's client-side SDKs take care of the complex authentication and networking code you'd normally need to write yourself. Then, on the backend part, we provide a powerful set of security rules so you can control access to your data. Security rules let you can also control which users can access which documents, and let you apply complex validation logic to your data as well. Combined, these features allow your mobile app to connect directly to your database.
- Combine with the rest of the Firebase platform. You can easily configure Cloud Functions to run custom code whenever data is written, and also our SDKs i.e Software Development kit automatically integrate with Firebase Authentication, to help you get started quickly.

c) Firebase Storage: It is a stand-alone solution for uploading content like images and videos from an iOS and Android device and also from the web. No server required for the typical firebase fashion.

Firebase storage is specially designed for scale, security and network resiliency.

- *Scale:* Every file backed by Google cloud storage when it uploaded, which scales is peta bytes.
- *Security:* Files can be secured by using Storage Security Rules.
- *Network Resiliency:* There is no worry to uploading and downloading in a case of poor network connections it done automatically , doesn't need to track them yourself.

C. Grow

1) Firebase Notifications: This feature allows to create attractive push notifications.You can send to particular or to all the users.

a) Fast and Easy to create and send a notification.

b) No additional coding required.

c) Messages can be sent to particular devices or all devices

2) Firebase Remote Config: This feature is very helpful for your application to test your application before applying any updates to it. You can test a user's behavior by inspecting its use with your application.

a) Custom key-value pairs is controlled from the Console.

b) Changes propagate instantly .

3) Firebase Dynamic Links: By using this feature you can check whether your application is responsive amongst all the platforms or not, its properly work or not, It display similar in Android, IOS and web or not.

1. Through a single URL, it customize different user experiences.
2. Works across platforms.

D. Earn

1) Ad Mob: This feature allows you to earn or gain through your application. It is supported across many platforms and easy to implement in your application.

a) Earn by your application.

b) It Support Android and IOS.

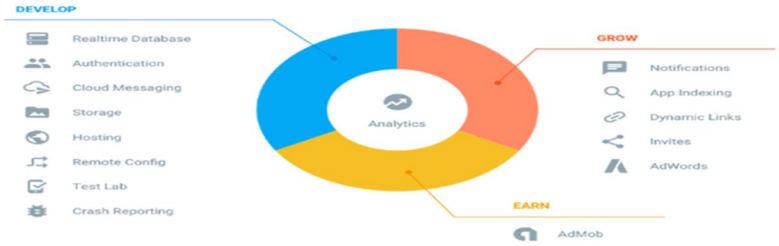


Figure 1: Services Of Firebase[11]

III. AWS AMPLIFY

A. Introduction to AWS Amplify

AWS Amplify is a set of tools and services that can be used together to help front-end web applications and mobile developers build scalable full stack applications, powered by AWS. Amplify let you configure applications backends and connect your app in minutes, deployed static web apps in a few clicks, and easily manage app content outside the AWS console. Amplify supports popular web frameworks including JavaScript, React, Angular, Next.js, and mobile platforms including Android, iOS, React Native, Flutter.

AWS Amplify main aims to provide a fantastic end-to-end platform that will let you create and keep cloud-enabled applications that can effectively deliver native-like experiences. When the backend is serverless, developers will eventually get more time to focus on their applications core functionalities and user experience.

B. Key features of AWS Amplify

a) *Authentication:* Its powered by Amazon Cognito, Amplify Auth offers pre-built workflows for sign-in, sign-up, forgotten passwords, and multi-factor authentication within the managed user directory.

b) *DataStore:* Cover real-time and offline capabilities for data synchronization across all platforms leveraging the on-device storage engine. Being powered by AWS AppSync and Amazon DynamoDB, it considerably simplifies dealing with distributed data.

c) *Analytics:* Teams favor data-driven approaches to learn their users' behaviors, increase retention and customer engagement. It provides auto-tracking, custom metrics to gather and analyze useful insights.

d) *API:* This vital feature ensures local data access, synchronization, and real-time updates from different sources. It includes your custom data sources as well as Amazon Aurora Serverless or Amazon DynamoDB.

e) *Storage:* It protects content storage and management by leveraging the simplified mechanism of the AWS Amplify module. It takes just a few steps to get your app from prototype to production while using cloud-scale storage.

IV. COMPARATIVE STUDY

Table 1: Comaprison between Firebase and AWS Amplify

Features	Description	Firebase	AWS Amplify
Parent Company	Powered by	Backed by Google	Backed by Amazon
Cloud	Integrate With	GCP	AWS
Platform Support	Runs on which platform i.e. provides an SDK	iOS, android, web	iOS, android, web
Storage	Data stored on	Google Cloud Storage	Directly on S3(Simple Storage Service)
API	Make secure HTTP request to REST and GrapgQL APIs	REST	REST and GraphQL
Cost	Maintenance and hosting fees	Lower	Higher
Project Size	Suitable for	Small, medium, enterprise	medium, enterprise and not for samll
Extensions	Adding features and functions	Yes(Firebase Extensions)	No
Analytics	Collecting analytics data for the app	Yes(Integration with Google Analytics)	Yes(Integration with AWS Pinpoint)
Authentication	Secure your service with authentication	Yes(Firebase Authentication)	Yes(Integration with Amazon Cognito)
Database	Database Supported	Yes(Firestore & Realtime)	Yes(Powered by AWS AppSync and Amazon DynamoDB)
A/B Testing	Comparing two versions and measuring the difference between	Yes	No

	the performance		
Push Notifications	Short messages that pop up on the user's mobile or desktop, pushing them to take some action.	Yes	Yes
Machine Learning	Adding machine learning feature to app	Yes	Yes
Dynamic Links	Deliver dynamic links to deliver customized user experience	Yes	No
Crashalytics	Crash and issues report for an app you reduce the trouble shooting time	Yes	No
Data Sync	Sync real time and offline data with apps	Yes(Yes via Firestore or Realtime database)	Yes(Yes via PubSub)

C. Advantages of Firebase

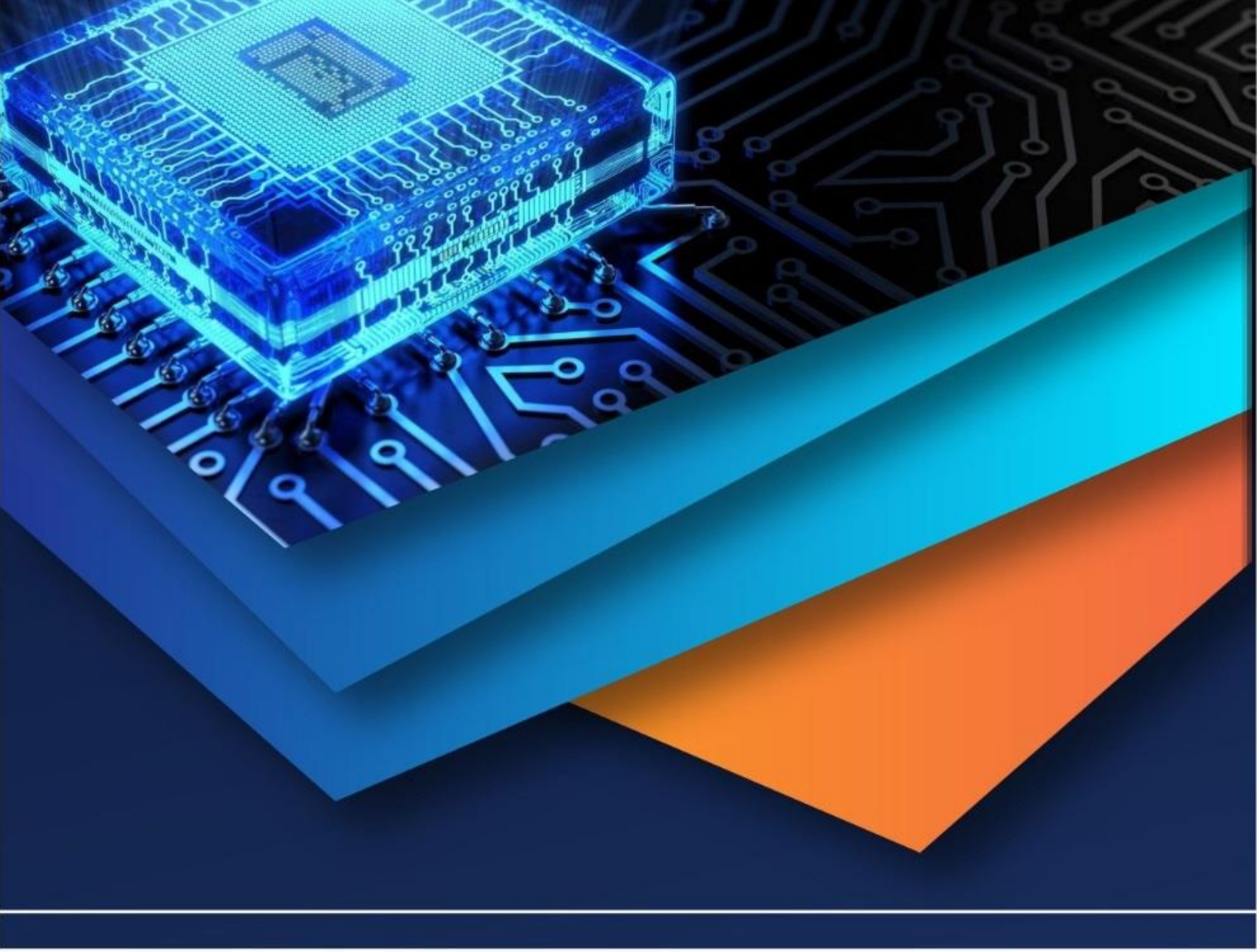
1) *Fast & Safe Hosting:* Another important advantage of Firebase is its secure & fast hosting services. Firebase hosting supports all content types, including web applications, dynamic and static contents.
2) *Real-time Database Helps to Store and Synchronize Data:* The Firebase Realtime Database is a cloud-hosted database in which data is stored as JSON. The data is synchronized in real-time to every connected client
3) *Cloud Firestore:* Other benefit of Firebase is its Cloud Firestore. This NoSQL database makes it relaxed for programmers to transfer and store data for front and backend development.
4) *Firebase Testing Services to Improve App Quality:* Testing services is another Firebase benefit due to whic developers prefer to use this software development platform. Firebase doesn't only offer its users a single testing service but offers a range of testing services.
5) *Firebase Authentication:* Nowadays, most of the apps have the login facility and the developer aims to simplify and secure it better. Therefore, the support of Firebase Authentication is there to do that task with an easy sign-in process.

V. CONCLUSION

Firebase is a BaaS service from Google, it provides NoSQL databases fully integrated with a REST API, its very easy to use, fully managed, and ascendable. AWS Amplify is an open-source framework completely integrated with AWS, works with GraphQL and multiple databases, and allows mobile and web development. But in order to develop application with low cost and for small and medium enterprise, firebase is the most suitable backend as it provides ush notifications, machine learning, dynamic links, authentication, etc. Real-time data synchronization makes data management work easy and faster. And we have aimed for the simplicity and ease-of-use that is always top priority for Firebase, while still making sure that firebase can scale to power even the largest apps.it is optimize for offline use so the realtime database use local cache on the device to store changes and data will be automatically synchronized when user comes online. Hence the firebase is so reliable and convenient to use than the other database.

REFERENCES

[1] Li WJ, Yen C, Lin YS, Tung SC, Huang S. JustIoT Internet of Things based on the Firebase real-time database. In2018 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE) 2018 Feb 8 (pp. 43-47). IEEE.
[2] Alsalemi A, Al Homsi Y, Al Disi M, Ahmed I, Bensaali F, Amira A, Alinier G. Real-time communication network using firebase cloud IoT platform for ECMO simulation. In2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) 2017 Jun 21 (pp. 178-182). IEEE.
[3] "Firebase | App success made simple," Firebase. [Online]. https://firebase.google.com
[4] Tram, Mai. "Firebase." (2019).
[5] Nikhil M. Dongre, Nikhil M. Dongre, Journal of Computer Engineering (IOSR-JCE), Volume 19, Issue 2, Ver. I (Mar.-Apr. 2017), PP 65-77
[6] Arb GI, Al-Majdi K. A Freights Status Management System Based on Dart and Flutter Programming Language. InJournal of Physics: Conference Series 2020 May 1 (Vol. 1530, No. 1, p. 012020). IOP Publishing.
[7] Wu, Wenhao. "React Native vs Flutter, Cross-platforms mobile application frameworks." (2018).
[8] Dahl, O., 2019. Exploring end user's perception of flutter mobile apps.
[9] HTML Canvas Gauges, https://github.com/Mikhus/canvas-gauges.
[10] https://www.ijirset.com/upload/2017/cotii/3_CS_COTII_2017_Firebase_cloud.pdf
[11] https://miro.medium.com/max/1200/1*G6Wfom8Zibpt35YxJgWLVA.png



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)