

Firestore API for IOS

Shashank Kumar Singh

ME Civil & Lecturer

Master of Computer Application

MET's ICS, Bandra (W), Mumbai, University of Mumbai

Abstract— The purpose of this study is to introduce Google firebase API and its features for IOS Application. Firestore is an API by Google for storage, remote config, analytics, remote notification and syncing into our IOS, Android and web application. A real-time database is one which stores data and fetch data continuously and update quickly real-time basis. This article will cover how to use firestore as a database backend for our IOS application development and many features like database, analytics, notification, storage, hosting etc. This study includes using many of these features of firestore into our IOS application.

Key words: FCM, Firestore, Firestore Analytics, Firestore Storage, Firestore Real-Time Database, Firestore Authentication

I. INTRODUCTION

A Database is a collection of data that store data on our computer or stored in server of server providers computers. All app (IOS, Android or Web app) need database for storing data. In IOS app, we can save data by using SQLite database, Shared Preference or cloud-based storage and in order to maintain our app we need complex code to implement even common feature, reliable network, testing tools which is not possible to every small organisation to acquire such infrastructure and maintenance tools.

Firestore provides such facilities and many features to maintain our app in long terms with minimum efforts, and provide integration of all products in one platform.

II. FEATURES OF FIRESTORE

A. Analytics:

Firestore Analytics provide details of various aspects like how much users are using our application, live streaming of users, how much user engagement on time basis and per session time used, user using app in different geographical user, gender of users and much more.

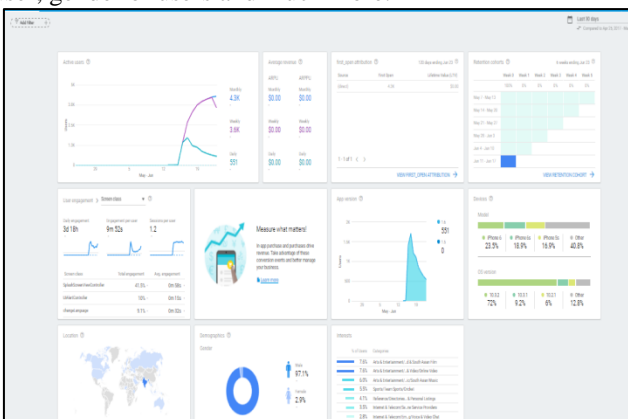


Fig. 1: Firestore Analytics

B. Remote Config:

Firestore Remote Config provide to change our app some parameter, images, colours or anything without upgrading our app. Remote Config is a cloud service that give the app owner to change some parameter remotely and user need not to update app to see those changes.

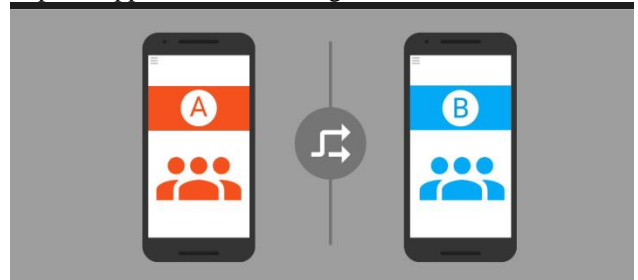


Fig. 2: Firestore Remote Config

C. Database:

Firestore Database is a Real-Time Database that provide store data and continuously observe that data to change Real-Time data to our app. Firestore Database store data in Jason Format which is form in a dictionary format(Key-value pair). Firestore Database is highly reliable and data is updated and synchronize in Real-Time.

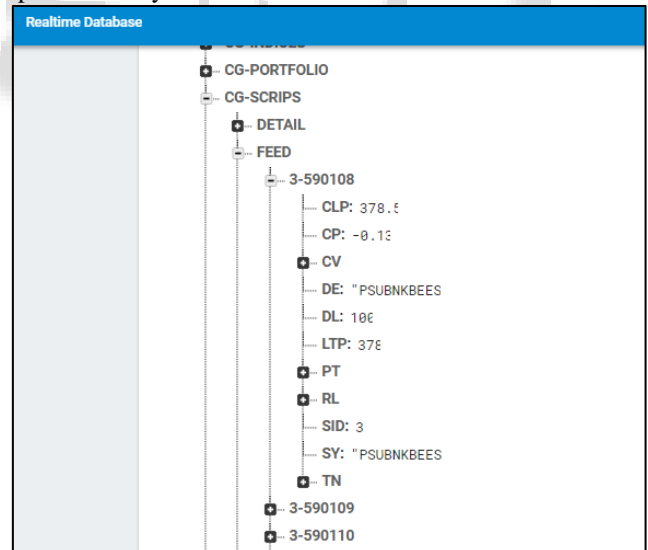


Fig. 3: Firestore Real-Time Database

D. Storage:

Firestore Storage is cloud based data storage by Google to provide store the data like text, image, video etc. Firestore storage automatically resume or pause of download as app loss or gain connectivity. Firestore Storage also provide storage with authentication basis, like allowed user only get their data.

E. Crash Reporting:

Firestore Crash Reporting provides ability to app owner to get crash or bug detail when user facing during app uses and get bug or crash on app. A detail report is generated and app owner able to know which line of code or method cause to bug and able to fix them quickly.

Crash Reporting also give detail of tools that get bug and app owner also get filtered data like which app version user using or get crashes.

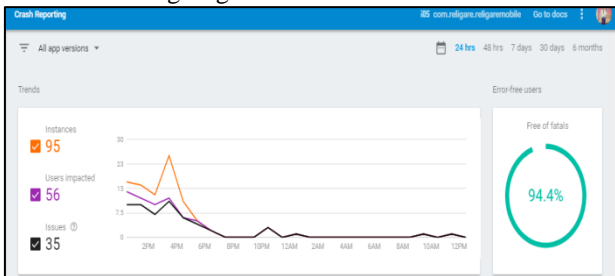


Fig. 4: Firestore Crash Reporting

F. Remote Notification:

Firestore Remote notification is cloud based service that is used for send notification to app users at any time and customized notification.

Firestore notification uses FCM (Firestore Cloud Messaging) that provide connection between server and device and allow to send notification or messages on IOS, Android or Web at free of cost.

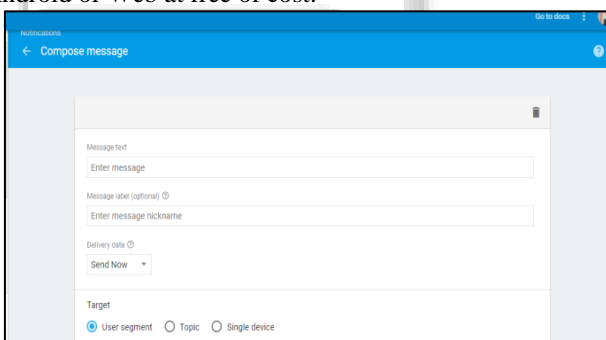


Fig. 5: Firestore Cloud Messaging

III. ADDING FIREBASE TO IOS

For adding Firestore to our IOS app we have to integrate our IOS app with firestore. Different steps are followed as:

We have to go to firestore console by following address – www.console.firestore.google.com, and add a project in firestore console by typing project name and our location.

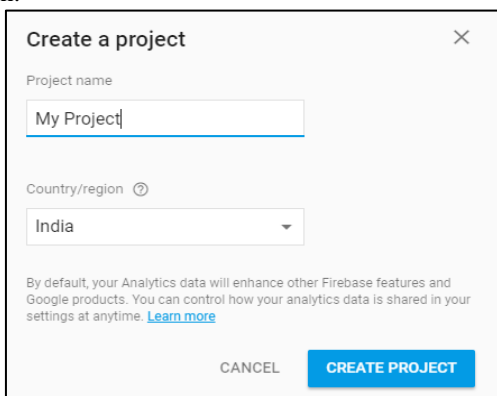


Fig. 6: Adding Firestore to our app

- 1) Click on create project.
- 2) Then we have to add our IOS bundle id which is unique for every project we create in xcode, then click on register app.

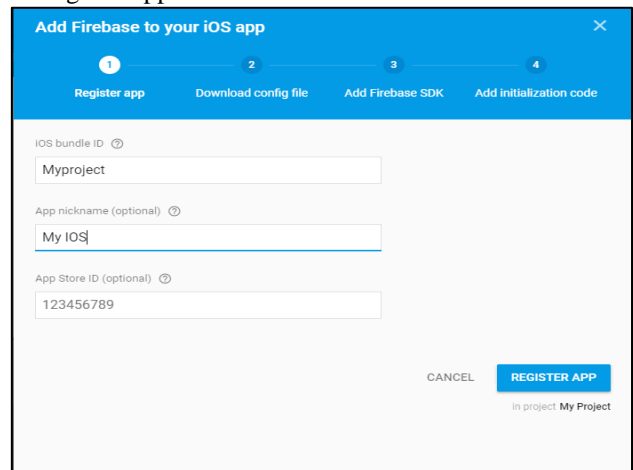


Fig. 7: Steps for adding firestore

- 3) Now we have to Download GoogleService-Info.plist file that contain information about our IOS app and we have to add this file to our xcode project.
- 4) Now we have to add pod file to our IOS project. In pod file we have to add which SDK we need from firestore and install that pod file.

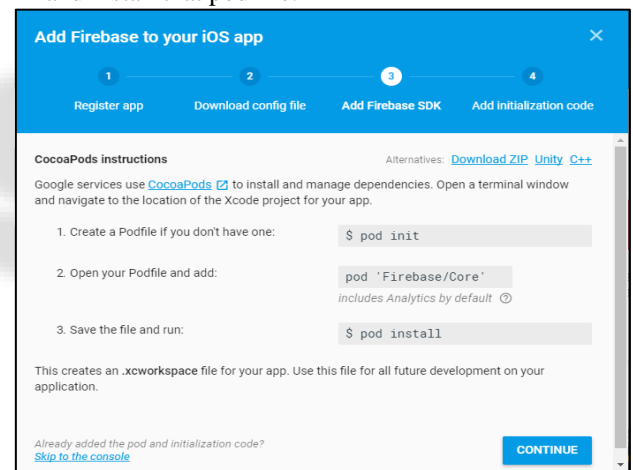


Fig. 8: Adding pod file.

- 5) After that we have to add some code to activate Firestore services in our app.

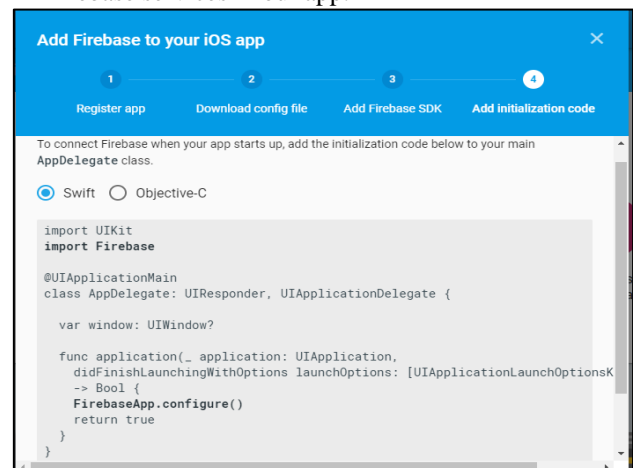


Fig. 9: Adding code to our appDelegate

IV. CONCLUSION AND FUTURE SCOPE

In this paper, I try to give quick reference and system which would make the work of developer easy by using firestore API to their IOS app and different features of firestore according to our need. Google Continuously adding new feature to firestore so we can study and implement then in our IOS app in future.

The point cover in this article are not just referenced but used in real application too. So this paper will help them who want to switch to firestore API.

REFERENCE

- [1] Firestore Products :- <https://firebase.google.com/products/>
- [2] Firestore features :-<https://www.cmarix.com/14-Benefits-of-using-firebase>
- [3] Firestore Pricing :-<https://firebase.google.com/pricing/>
- [4] Firestore Release and Services :- <https://en.wikipedia.org/wiki/Firebase>

